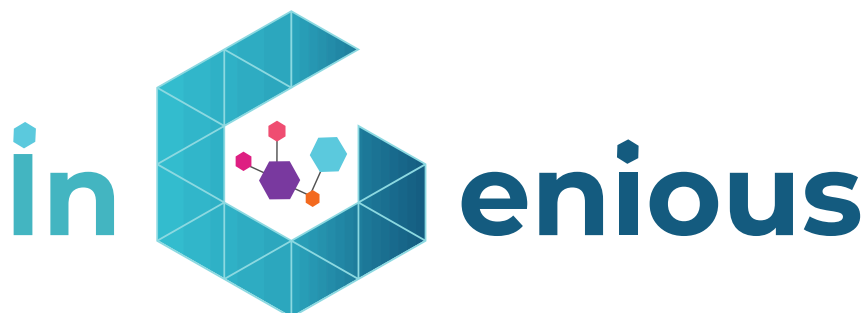




Grant Agreement No.: 957216
Call: H2020-ICT-2018-2020

Topic: ICT-56-2020
Type of action: RIA



D6.3 Final Evaluation and Validation

Revision: v.1.0

Work package	WP6
Task	Task 6.3 Trials and validation
Due date	31/03/2023
Submission date	31/03/2023
Deliverable lead	COSSP
Version	1.0
Authors	Carla San Miguel (COSSP), Chiara Iorfida(COSSP), Jose Boix (COSSP), Pablo Ferrer (COSSP), Pedro Pérez (COSSP), Christos Politis (SES), Alexandr Tardo (CNIT), Ivo Bizon Franco de Almeida (TUD), José Luis Cárcel (FV), Joan Meseguer (FV), Giacomo Bernini (NXW), Pietro Piscione (NXW), Erin E. Seder (NXW), Miguel Cantero (5CMM), Manuel Fuentes (5CMM), Miriam Ortiz (5CMM), Héctor Donat (5CMM) Nuria Molner (UPV), Francisco Javier Curieses (UPV), Raúl Lozano (UPV), Iván Viciado (UPV), David Gomez-Barquero (UPV), Nuria Oyaga de Frutos (NOK), Clemens Saur (NCG), Carsten Weinhold (BI), Laura Gonzalez Estebanez (ASTI), Rodrigo Martinez (ASTI), Joe Cahill (iDR), Shane Bunyan (iDR), Eddy Higgin(iDR), Jose Costa-Requena (CMC)
Reviewers	Alexandr Tardo (CNIT), Nuria Molner (UPV), David Gomez-Barquero (UPV), Francisco Javier Curieses (UPV), Raúl Lozano (UPV), Iván Viciado (UPV), Carsten Weinhold (BI), Christos Politis (SES), Efstathios Katranaras (SEQ), Pablo Ferrer

	(COSSP), Chiara Iorfida (COSSP), Carla San Miguel (COSSP)
--	---

Abstract	This deliverable provides the achieved results in the deployment of PoCs and Demos for the different trials, including the technical validation of iNGENIOUS use cases. It follows the plan defined in D6.1 – Initial Planning for Testbeds and the set-up and development and integration activities defined in D6.2 PoC Development, Platform and Test-bed Integration.
Keywords	PoCs, Demos, Trials, Test Cases, Set-up, Execution, KPIs, Impact Assessment, Lessons Learned and Potential improvement

Disclaimer

The information, documentation and figures available in this deliverable are written by the "Next-Generation IoT solutions for the universal supply chain" (iNGENIOUS) project’s consortium under EC grant agreement 957216 and do not necessarily reflect the views of the European Commission.

The European Commission is not liable for any use that may be made of the information contained herein.

Copyright notice

© 2020 - 2023 iNGENIOUS Consortium

Project co-funded by the European Commission in the H2020 Programme		
Nature of the deliverable:		DEM
Dissemination Level		
PU	Public, fully open, e.g. web	✓
CL	Classified, information as referred to in Commission Decision 2001/844/EC	
CO	Confidential to iNGENIOUS project and Commission Services	

** R: Document, report (excluding the periodic and final reports)*

DEM: Demonstrator, pilot, prototype, plan designs

DEC: Websites, patents filing, press & media actions, videos, etc.

OTHER: Software, technical diagram, etc.



Executive Summary

The present document is the output of Task 6.3 Trials and Validation and describes the measurement campaigns and trials developed in iNGENIOUS Proof of Concept (PoCs) and Demonstrations (Demos).

Once detailed the objectives of the PoCs and Demos, the setup and execution activities for the validation and demonstration are presented, as well as possible issues occurred during the execution and mitigation actions adopted.

Then, the validation and results presentation are described following the test case verification and KPI calculation.

Finally, the deliverable summarizes the impact assessment, lessons learned and potential improvements on a technical level for trials and testbeds.



Table of Contents

1	Introduction	17
2	PoC - Automated Robots with Heterogeneous Networks	19
3	PoC - Transportation Platforms Health Monitoring.....	33
4	Demo – Situational Understanding in Smart Logistics Scenario	45
5	Demo – Improved Drivers’ Safety with MR and Haptic Solutions....	66
6	Demo – Intermodal Asset Tracking via IoT and Satellite	76
7	PoC - Supply Chain Ecosystem Integration	98
8	Additional Research Activities – Satellite Direct Access.....	113
9	Conclusion	120
	References	123
	Annex I: Factory UC - Automated Robots with Heterogeneous Networks	124
	Annex II: Transport UC - Transportation Platforms Health Monitoring.....	138
	Annex III: Port Entrance UC - Situational Understanding in Smart Logistics Scenario	148
	Annex IV: AGV’s UC - Improved Drivers’ Safety with MR and Haptic Solutions	175
	Annex V: Ship UC - Intermodal Asset Tracking via IoT and Satellite	182
	Annex VI: DLT’s UC - Supply Chain Ecosystem Integration.....	195



List of figures

Figure 1: gNB (left), EasyBot (top-middle), eBot (bottom-middle) and Tribot (right) 20

Figure 2: Architecture of factory UC demo in Burgos 21

Figure 3: Execution of the demo in Burgos 22

Figure 4: Setup illustration at TUD’s testbed used to demonstrate the integration between Flexible PHY/MAC, 5G core, and MANO. 23

Figure 5: Web GUI showing two end-to-end network slices provisioned... 23

Figure 6: Grafana KPI dashboard visualization of UE to UE with video streaming running. 1) Top: Downlink throughput – iperf3 test (Pink). 2) Middle: Uplink throughput – iperf3 test (Pink) and video streaming (Purple). 3) Bottom: RTT uplink – influxDB connection (Yellow) and iperf3 test (Purple). 27

Figure 7: Grafana KPI dashboard visualization of UE to core without video streaming running. 1) Top: Downlink throughput – iperf3 test (Orange). 2) Middle: Uplink throughput – iperf3 test (Orange). 3) Bottom: RTT uplink – influxDB connection (Yellow) and iperf3 test (Purple). 28

Figure 8: Grafana KPI dashboard visualization of UE to core with video streaming running. 1) Top: Downlink throughput – iperf3 test (Blue). 2) Middle: Uplink throughput – iperf3 test (Blue) and video streaming (Purple). 3) Bottom: RTT uplink – influxDB connection (Yellow) and iperf3 test (Purple). 28

Figure 9: Illustration of the components used in the measurement setup.29

Figure 10: Sensing and GW Modules for Rail-Health Data Logging 33

Figure 11: Rail-Health Flatspot Harshness and Bearing Defect Demonstrator (Concept & Design) 35

Figure 12: Rail-Health Simulated Fault & Sinusoidal Fault Injector Set-up ... 36

Figure 13: BI’s M³ hardware/software co-design platform realized on FPGA development board, with external Ethernet extension board 36

Figure 14: Raspberry Pi 4B single-board computer with a Trusted Platform Module (TPM) attached to the GPIO pin header 37

Figure 15: Simulated satellite infrastructure with ingress and egress routers, enabling the smart edge sensor (BI IoT device) to connect to the cloud server (BI cloud) 37

Figure 16: Main dashboard table showing RATLS connection establishment and touch controls for enabling and disabling remote attestation 38

Figure 17: Secondary display showing smart sensor state and defect classification. 38

Figure 18: Data Integration and ML-Based Algorithm Approach 47

Figure 19: Data Integration and ML Pipeline Division Approach 47

Figure 20: Data sources and model components developed for the demonstration 49

Figure 21: Autoregressive + ML method to predict TTT 51



Figure 22: Final TTT data frame..... 51

Figure 23: Gate In time series analysis 52

Figure 24: Gate In SARIMA model instantiation 52

Figure 25: Random Forest Regressor hyperparameter tuning for the TTT model 53

Figure 26: TTT Random Forest model instantiation and fitting..... 53

Figure 27: Port Call and Gate Access online data ingestion setup.....54

Figure 28: Overview of the cloud service architecture used in the demonstration..... 55

Figure 29: Autoregressive + ML based TTT prediction setup..... 55

Figure 30: Overview of predicted vessels arriving to port of Valencia in the Awake.AI web application. 57

Figure 31: Predicted route and arrival time to port of Valencia for a selected vessel. 57

Figure 32: Port Entrance UC demonstration custom web interface 58

Figure 33: True TTT vs prediction TTT using the gate-in/out validation service. 59

Figure 34: Port Entrance UC IoT Tracking dashboard for one week testing 59

Figure 35: Diagram of Services..... 61

Figure 36: Main setup and components integrated in the trial..... 66

Figure 37: Testing area in the Port of Valencia 67

Figure 38: AGVs A, B and C for the AGV’s UC 68

Figure 39: Nokia’s (left) and Fivecom’s (right) cockpits. 69

Figure 40: Unity application developed for the digital twin..... 71

Figure 41: Real scenario (left) and digital twin with the AGV included (right). 71

Figure 42: Remote cockpit including the SensGlove haptic gloves, VR glasses and Digital Twin..... 72

Figure 43: End-to-end architecture of the final demo..... 79

Figure 44: i) RF Uplink ground Station: ATF #33 Antenna, Diameter: 9m, Vertex, Tx/Rx, Ku-band, ii) RF Downlink Ground Station: MBA#102 Antenna, Diameter: 4.5m, Multi-Beam Antenna, Rx only, Ku-band, and iii) SES GEO Satellite ASTRA 2F (28.2oE) - Europe Ku-band beam 80

Figure 45: i) Satcube transportable satellite terminal and ii) Smart IoT Gateway 81

Figure 46: Front and back of the iDirect’s 5G-enabled Velocity™ Intelligent Gateway hub..... 81

Figure 47: iQ200, iQ Desktop and 9350 Modem..... 81

Figure 48: i) 22G1 purchased container and ii) iNGENIOUS Container..... 82

Figure 49: Final demo device installation..... 83

Figure 50: End-to-end architecture of the final demo (part II) 84



Figure 51: iNGENIOUS container starting rail transport from Valencia to Madrid.....84

Figure 52: SatCube, Smart IoT GW and iNGENIOUS container at the Port of Valencia 86

Figure 53: Temperature of the iNGENIOUS container during the trip from Valencia to Piraeus and vice versa.86

Figure 54: Humidity of the iNGENIOUS container during the trip from Valencia to Piraeus and vice versa..... 86

Figure 55: Overview screenshot of the Cloud-side dashboard, giving a general impression of the received data during the real-time measurements at the Port of Valencia on 21 November 2022 87

Figure 56: GPS location of the IoT devices during the real-time measurements at the Port of Valencia on 21..... 87

Figure 57: Temperature, measured in real-time from the IoT devices, in the Port of Valencia on 21 November 202288

Figure 58: Humidity, measured in real-time from the IoT devices, in the Port of Valencia on 21 November 2022.....88

Figure 59: Battery state of charge of the IoT devices, measured in real-time in the Port of Valencia on 21 November 2022 89

Figure 60: Door state of the iNGENIOUS container, measured in real-time in the Port of Valencia on 21 November 2022 89

Figure 61: Accelerometer measured in real-time in the Port of Valencia on 21 November 2022.....90

Figure 62: End-to-end latency for transmitting the measured data from the IoT devices to the SES Cloud through satellite at the port of Valencia on 21 November 2022.....90

Figure 63: Ship UC Demo part B – IoT message received.91

Figure 64: GPS location reported by the sensor in Part B trip 92

Figure 65: Temperature, humidity and accelerometer values by the sensor in Part II trip 92

Figure 66: DLT Events Visualizer representing the DigitalAsset and the associated Trustpoint for the VesselArrival event in Livorno seaport....102

Figure 67: IoT device used for sealRemoved event.....103

Figure 68: DLT Events Visualizer representing the DigitalAsset and the associated Trustpoint for the sealRemoved event in Valencia seaport. 104

Figure 69: Service vehicle in the Port of Livorno with the IoT tracking device installed on board.105

Figure 70: Tracking Application - Livorno Dashboard..... 106

Figure 71: Heat sensors installed in iDR lab in Killarney. 114

Figure 72: Transmission of IoT data over satellite lab and live testbed setups115

Figure 73: Microsoft Azure IoT cloud dashboard showing IoT information. 116



Figure 74: Example of in-house IoT cloud dashboard, based on Grafana, showing IoT information. 116

Figure 75: Example of in-house IoT cloud dashboard, based on Grafana, showing IoT information. 118

Figure 76: Tribot architecture124

Figure 77: EasyBot architecture.....125

Figure 78: Ebot architecture.....126

Figure 79: Tribot AGV126

Figure 80: EasyBot AGV.....126

Figure 81: Ebot AGV.....126

Figure 82: 5G base station127

Figure 83: IRSRP values obtained through the walk test around the industrial unit. 135

Figure 84: End-to-end architecture iperf3 test UE to UE.136

Figure 85: End-to-end architecture iperf3 test UE to core.136

Figure 86: End-to-end architecture used for the KPI measurement setup with 5Probe..... 137

Figure 87: Overview of prediction model components, required features, and source datasets..... 148

Figure 88: Kernel density estimates of the empirical distributions of actual and simulated total container dwell times in the port of Valencia. 149

Figure 89: Vessel ETA prediction model pipeline150

Figure 90: Simulated vs. actual weekly numbers of containers leaving port of Valencia by truck151

Figure 91: MLOps pipeline overview.151

Figure 92: Gate-in dataset resampled.....152

Figure 93: Port Call Dataset152

Figure 94: Final TTT data frame.....153

Figure 95: Gate In time series analysis153

Figure 96: SARIMA hyperparameter tuning for the Gate In model 154

Figure 97: Port Entrance UC database structure155

Figure 98: IoT tracking service deployment infrastructure.156

Figure 99: Final vessel ETA prediction model accuracy statistics*172

Figure 100: 5G Network connection setup175

Figure 101: Relation between modems and devices.....175

Figure 102: SenseGlove haptic gloves.....176

Figure 103: Example of RTT during Valencia Port tests.....179

Figure 104: Example of the decoded frames during Valencia Port tests. 180

Figure 105: Example of the downlink data rate for AGV-B during Valencia Port tests. 180



Figure 106: Indoor bluetooth range for Neurodigital (left) and SenseGlove (right) haptic gloves..... 181

Figure 107: Outdoor bluetooth range for Neurodigital (left) and SenseGlove (right) haptic gloves..... 181

Figure 108: iDR lab testbed system overview183

Figure 109: i) iDR Lab testbed including an iQ200, iQ Desktop, 9350 modems, IoT GW & Satellite Channel Emulators x2, ii) iDR Lab Testbed generic sensor used to measure temperature and humidity of the lab and iii) iDR Lab Testbed iDirect’s 5G-enabled Velocity™ IGW hub.....183

Figure 110: Scenario 1 architecture for the demonstration of the use case...196

Figure 111: Scenario 2 architecture for the demonstration of the use case..197

Figure 112: Scenario 3 architecture for the demonstration of the use case. 198

Figure 113: Scenario 4 architecture for the demonstration of the use case..199

Figure 114: sequence diagram for the demonstration of the Scenario 1. 200

Figure 115: DigitalAsset for the VesselArrival event in Livorno seaport. 200

Figure 116: DigitalAsset for the VesselDeparture event in Livorno seaport..201

Figure 117: DigitalAsset for the GateIn event in Livorno seaport.....201

Figure 118: DigitalAsset for the GateOut event in Livorno seaport.202

Figure 119: Trustpoint of the VesselArrival event in Livorno seaport.202

Figure 120: Trustpoint of the VesselDeparture event in Livorno seaport.203

Figure 121: Trustpoint of the GateIn event in Livorno seaport.203

Figure 122: Trustpoint of the GateOut event in Livorno seaport..... 204

Figure 123: Sequence diagram for the demonstration of the Scenario 2. 204

Figure 124: sealRemoved event data at DVL level.205

Figure 125: Sequence diagram for the demonstration of the Scenario 3.206

Figure 126: IoT Tracking Sensor message format.....206

Figure 127: IoT Tracking Sensor GPS message.207

Figure 128: GPS data coming from the Symphony M2M Platform and aggregated at DVL level.....207

Figure 129: Sequence diagram for the demonstration of the Scenario 4.....208

Figure 130: The main interactions between the DVL and Pseudonymized Module.208



List of tables

Table 1.	Mapping of use-case names to test-case identifiers	18
Table 2.	Data flows in Burgos' demo.....	22
Table 3.	Factory UC issues on execution	24
Table 4.	Factory UC Test case verification	25
Table 5.	Factory UC KPIs.....	26
Table 6.	Flexible PHY/MAC throughput measurements with <i>UDPtest</i>	30
Table 7.	Transport UC issues on execution.....	39
Table 8.	Transport UC Test case verification	40
Table 9.	iDR Lab Testbed Usage.....	41
Table 10.	Transport UC KPIs	42
Table 11.	IoT Tracking based TTT measurement tests.....	60
Table 12.	Port Entrance UC Issues on execution	61
Table 13.	Port Entrance UC Test case verification.....	62
Table 14.	Port Entrance UC KPIs Results	64
Table 15.	AGV UC Issues on execution.....	72
Table 16.	AGV's UC Test case verification	73
Table 17.	AGV UC KPIs	74
Table 18.	AGV UC KPIs. Comparision Neurodigital vs GloveSense haptic gloves.	74
Table 19.	SES's ASTRA 2F Space Segment.....	80
Table 20.	Ship UC Issues on execution	85
Table 21.	ICMP RTT of Satellite Link.....	91
Table 22.	Ship UC Test case verification.....	93
Table 23.	Ship UC KPIs.....	94
Table 24.	Scenarios used for the demonstration of the DVL/DLT UC	99
Table 25.	DVL/DLT UC Issue on execution.....	108
Table 26.	DVL/DLT UC Test case verification.....	109
Table 27.	DVL/DLT UC KPIs.	110
Table 28.	Satellite channel characterization SNR values	118
Table 29.	Information flows for Tribot AGV.....	124
Table 30.	Information flows for EasyBot AGV	125
Table 31.	Information flows for Ebot AGV	126
Table 32.	AGVs employed demonstration in Burgos.....	126
Table 33.	Main parameters and configuration of 5G network.....	127
Table 34.	Equipment for factory UC demonstration in Burgos.....	128



Table 35.	UC1_TC_01 verification.....	129
Table 36.	UC1_TC_02 verification.....	129
Table 37.	UC1_TC_03 verification.....	129
Table 38.	UC1_TC_04 verification.....	130
Table 39.	UC1_TC_05 verification.....	130
Table 40.	UC1_TC_06 verification.....	131
Table 41.	UC1_TC_07 verification.....	131
Table 42.	UC1_TC_08 verification.....	131
Table 43.	UC1_TC_09 verification.....	132
Table 44.	UC1_TC_10 verification.....	132
Table 45.	UC1_TC_11 verification.....	133
Table 46.	UC1_TC_12 verification.....	133
Table 47.	UC1_TC_13 verification.....	133
Table 48.	UC1_TC_14 verification.....	134
Table 49.	UC1_TC_15 verification.....	134
Table 50.	UC3_TC_01 verification.....	138
Table 51.	UC3_TC_02 verification.....	138
Table 52.	UC3_TC_03 verification.....	139
Table 53.	UC3_TC_04 verification.....	139
Table 54.	UC3_TC_05 verification.....	139
Table 55.	UC3_TC_06 verification.....	140
Table 56.	UC3_TC_07 verification.....	140
Table 57.	UC3_TC_08 verification.....	141
Table 58.	UC3_TC_09 verification.....	141
Table 59.	UC3_TC_10 verification.....	142
Table 60.	UC3_TC_11 verification.....	142
Table 61.	UC3_TC_12 verification.....	142
Table 62.	UC3_TC_13 verification.....	143
Table 63.	UC3_TC_14 verification.....	143
Table 64.	UC3_TC_15 verification.....	144
Table 65.	UC3_TC_16 verification.....	144
Table 66.	UC3_TC_17 verification.....	145
Table 67.	UC3_TC_18 verification.....	145
Table 68.	UC3_TC_19 verification.....	145
Table 69.	UC3_TC_20 verification.....	146
Table 70.	UC3_TC_21 verification.....	146
Table 71.	UC3_TC_22 verification.....	147



Table 72.	UC3_TC_23 verification.....	147
Table 73.	UC5_TC_13 description	158
Table 74.	UC5_TC_13 description	159
Table 75.	UC5_TC_15 description	159
Table 76.	UC5_TC_16 description	160
Table 77.	UC5_TC_17 description	161
Table 78.	UC5_TC_18 description	162
Table 79.	UC5_TC_19 description	163
Table 80.	UC5_TC_20 description	163
Table 81.	UC5_TC_21 description	164
Table 82.	UC5_TC_01 verification.....	165
Table 83.	UC5_TC_02 verification	166
Table 84.	UC5_TC_03 verification	167
Table 85.	UC5_TC_04 verification.....	167
Table 86.	UC5_TC_05 verification	168
Table 87.	UC5_TC_06 verification	168
Table 88.	UC5_TC_07 verification	169
Table 89.	UC5_TC_08 verification.....	169
Table 90.	UC5_TC_09 verification	169
Table 91.	UC5_TC_10 verification	170
Table 92.	UC5_TC_11 verification	170
Table 93.	UC5_TC_12 verification	170
Table 94.	UC5_TC_13 verification	171
Table 95.	UC5_TC_14 verification	171
Table 96.	UC5_TC_15 verification	172
Table 97.	UC5_TC_16 verification	172
Table 98.	UC5_TC_17 verification	173
Table 99.	UC5_TC_18 verification	173
Table 100.	UC5_TC_19 verification	174
Table 101.	UC5_TC_20 verification	174
Table 102.	UC5_TC_21 verification	174
Table 103.	UC2_TC_01 verification	177
Table 104.	UC2_TC_02 verification	177
Table 105.	UC2_TC_03 verification	177
Table 106.	UC2_TC_04 verification.....	177
Table 107.	UC2_TC_05 verification	178
Table 108.	UC2_TC_06 verification.....	178



Table 109. UC2_TC_07 verification178

Table 110. Overview of iDR lab testbed activities..... 184

Table 111. UC4_TC_01 verification..... 184

Table 112. UC4_TC_02 verification.....185

Table 113. UC4_TC_03 verification.....185

Table 114. UC4_TC_04 verification185

Table 115. UC4_TC_05 verification..... 186

Table 116. UC4_TC_06 verification..... 186

Table 117. UC4_TC_07 verification.....187

Table 118. UC4_TC_08 verification.....187

Table 119. UC4_TC_09 verification..... 188

Table 120. UC4_TC_10 verification..... 188

Table 121. UC4_TC_11 verification..... 188

Table 122. UC4_TC_12 verification 189

Table 123. UC4_TC_13 verification 189

Table 124. UC4_TC_14 verification..... 189

Table 125. UC4_TC_15 verification 190

Table 126. UC4_TC_16 verification..... 190

Table 127. UC4_TC_17 verification..... 190

Table 128. UC4_TC_18 verification..... 191

Table 129. UC4_TC_19 verification..... 191

Table 130. UC4_TC_20 verification..... 191

Table 131. sealRemoved event data model.....205

Table 132. UC6_TC_01 verification.....209

Table 133. UC6_TC_02 verification.....209

Table 134. UC6_TC_03 verification.....210

Table 135. UC6_TC_04 verification.....210

Table 136. UC6_TC_05 verification.....211

Table 137. UC6_TC_06 verification.....212

Table 138. UC6_TC_07 verification.....213

Table 139. UC6_TC_08 verification.....214

Table 140. UC6_TC_09 verification.....215

Table 141. UC6_TC_10 verification.....216

Table 142. UC6_TC_11 verification.....217

Table 143. UC6_TC_12 verification.....218



Abbreviations

3GPP	3rd Generation Partnership Project
A2A	Authority to Authority
ACK	Acknowledge
AES	Advanced Encryption Standard
AI	Artificial Intelligence
AIDA	Automazione Integrata Dogane Accise (Integrated Automation Customs Excise)
AIS	Automatic Identification System
AGV	Automatic Guided Vehicle
API	Application Programming Interface
B2A	Business to Authority
B2B	Business to Business
BBU	Baseband Unit
BT	Bluetooth
BTC	Bitcoin Native Token
CIoT	Consumer Internet of Things
CPU	Central Processing Unit
CSE	Common Service Entity
CSV	Comma Separated Values
DL	Downlink
DLT	Distributed Ledger Technology
DVL	Data Virtualization Layer
E2E	End to End
ECDSA	Elliptic Curve Digital Signature Algorithm
EDA	Exploratory Data Analysis
ETA	Expected Time of Arrival
ETD	Expected Time of Departure
ETSI	European Telecommunications Standards Institute
FER	Frame Error Rate
FMEDA	Failure Modes, Effects and Diagnostics Analysis
FPGA	Field Programmable Gate Array
GAD	Geographic Anomaly Detection
GEO	Geostationary
GPS	Global Positioning System
GSM	Global System for Mobile Communications
GSMA	Global System for Mobile Communications
GUI	Graphic User Interface



GW	Gateway
HTTP	HyperText Transfer Protocol
ICT	Information and Communications Technology
ICMP	Internet Control Message Protocol
IEC	International Electrotechnical Commission
IMSI	International Mobile Subscriber Identity
IoT	Internet of Things
IP	Internet Protocol
ISO	International Organization for Standardization
IT	Information Technology
LAN	Local Area Network
LO-LO	Lift On – Lift Off
LoRa	Long Range
LTE	Long Term Evolution
M2M	Machine to Machine
MAC	Media Access Control
MEC	Mobile Edge Computing
ML	Machine Learning
MR	Mixed Reality
NAT	Network Address Translation
NB-IoT	Narrowband-IoT
NDA	Non Disclosure Agreement
NEF	Network Exposure Function
NFV	Network Function Virtualization
NSA	Non Standalone
NSD	Network Service Descriptor
NSMF	Network slice management function
NSSMFs	Network slice subnet management functions
NTN	Non Terrestrial Networks
NWDAF	Network Data Analytics Function
ODU	Outdoor Unit
OS	Operating System
OU	Occasional Use
PC	Personal Computer
PCS	Port Community System
PMIS	Port Management Information System
PoC	Proof of Concept
PSU	Power Supply Unit



QoE	Quality of Experience
QoS	Quality of Service
R&D	Research and Development
RAN	Radio Access Network
RF	Radio Frequency
RO-RO	Roll On – Roll Off
ROS	Robot Operating System
RoT	Root of Trust
RPI	Raspberry Pi
RRH	Remote Radio Head
SA	Standalone
SARIMA	Seasonal Autoregressive Integrated Moving Average
SCADA	Supervisory Control And Data Acquisition
SDR	Software Defined Radio
SHA	Secure Hash Algorithm
SNR	Signal to Noise Ratio
SOAP	Simple Objects Access Protocol
SR	System Requirement
TC	Test Case
TCP	Transmission Control Protocol
TLS	Transport Layer Security
ToD	Tele-operated Driving
TPCS	Tuscan Port Community System
TTT	Truck Turnaround Time
UC	Use Case
UDP	User Data Protocol
UE	User Equipment
UL	Uplink
UPF	User Plane Function
UR	User Requirement
URLLC	Ultra-Reliable Low-Latency Communications
USRp	Universal Software Radio Peripheral
VPN	Virtual Private Network
VSAT	Very Small Aperture Terminal
VSMF	Vertical Service Management Function
WAN	Wide Area Network
WiFi	Wireless Fidelity
WP	Work Package



1 Introduction

In this chapter, the deliverable's objective and structure are described as well as useful information for the reader is provided.

1.1 Objective of the Document

The main objective of the deliverable is to present the validation of results of iNGENIOUS PoCs and Demos, by describing the measurement campaigns and trials developed.

Following the methodology defined deliverable in *D6.1 Initial planning for testbeds* [1], where specific test cases have been identified, and the work developed in *D6.2 PoC development, platform and test-bed integration* [2], where specific test cases have been identified, and the work developed in the D6.2 [2], where set-up and configuration activities have been defined, it details the achieved results in the deployment of the PoCs and Demos and includes their technical validation against the requirements defined in WP2.

The deliverable first presents the main objectives of the demonstrations, highlighting the technologies and solutions tested to improve logistics activities along complex supply chains.

Once detailed the objectives, the setup and execution of the demonstrations are provided by describing the configuration of the solutions used and the activities carried out for the execution of the PoCs and Demos. Issues occurred during the execution are identified and the mitigation actions adopted are also detailed.

In order to ensure the validation of results achieved, per each PoC and Demo, the verification of the test cases is described, by detailing the results achieved. Then, the calculation of KPIs is presented, providing reference to the test cases, target defined and reached. Any deviations from the defined target are described and justified. To complete the validation of the results, an impact assessment is presented, describing main achievements and impact reached.

Finally, D6.3 [3] provides a set of lessons learned during the PoCs and Demos execution and validation. Additionally, the document offers potential improvements on a technical level that could be further developed in the existing demonstrations.

The following sections present the results of each PoC and Demos, which are *Automated Robots with Heterogeneous Network*, *Improved Drivers' Safety with MR and Haptic Solutions*, *Transportation Platforms Health Monitoring*, *Intermodal Asset Tracking via IoT and Satellite*, *Situational Understanding in Smart Logistics Scenario* and *Supply Chain Ecosystem Integration*.

In addition to the presentation of results and validation, the deliverable also provides a description of additional research activities carried out during the project and focuses on demonstrating satellite direct access to transmit IoT data.

1.2 Structure of the Document

The deliverable follows the following structure:

- Section 2 focuses on the use of automatic robot control for industrial automation (Factory UC).
- Section 3 focuses on the transportation platform to show how asset health tracking can lead to lower operational costs and higher asset availability (Transport UC).
- Section 4 focuses on enhancing the situational understanding of events in maritime ports and terminals (Port Entrance UC).
- Section 5 focuses on improving the driver’s safety by combining the use of mixed reality and haptic solutions for controlling AGVs in a real scenario (AGV UC).
- Section 6 focuses on providing End-to-End (E2E) asset tracking using various connection and backhaul technologies (Ship UC).
- Section 7 focuses on providing two different interoperable layers in order to abstract the complexity of the underlying machine-to-machine (M2M) platforms and DLT solutions (DVL/DLT UC).
- Section 8 focuses on additional research carried out during the project which was outside the scope of the selected use cases.
- Annexes include additional information on the UCs, mainly on execution and test case verification. The annexes follow the same structure of the deliverable to allow readers to easily find information for each specific section. The main sections reported are filled only in case there was additional information to report.

In all the sections the execution and results obtained in each demo and PoC are described.

1.3 Navigating this document

The deliverable provides an overview of the activities related to trials and measurement campaigns performed in the PoCs and Demos and their validation against requirements and KPIs defined in WP2. In this deliverable to help readers to map the UCs to the test case validation, we use the identifiers such as “UC1_TC_01”, which refers to test case #01 of the Factory UC. Therefore, the following table is provided:

UC name	UC short name	Test case identifier
Automated Robots with Heterogeneous Networks	Factory UC	UC1_TC_X
Improved Drivers’ Safety with MR and Haptic Solutions	AGV UC	UC2_TC_X
Transportation Platforms Health Monitoring	Transport UC	UC3_TC_X
Intermodal Asset Tracking via IoT and Satellite	Ship UC	UC4_TC_X
Situational Understanding in Smart Logistics Scenario	Port Entrance UC	UC5_TC_X
Supply Chain Ecosystem Integration	DVL/DLT UC	UC6_TC_X

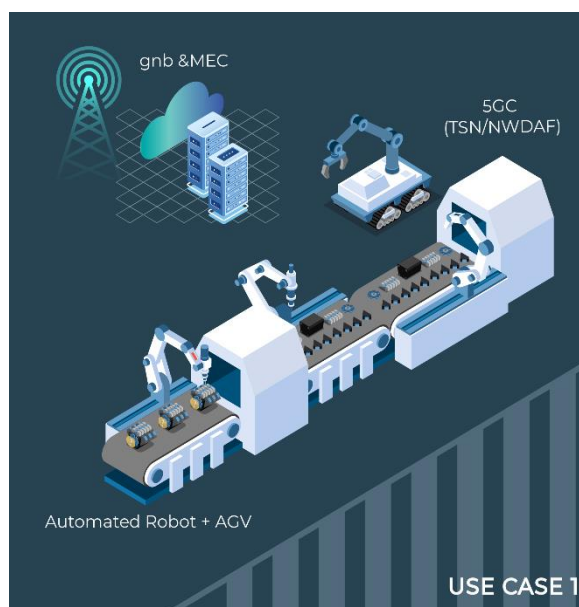
Table 1. Mapping of use-case names to test-case identifiers



2 PoC - Automated Robots with Heterogeneous Networks

2.1 Objective and Description

The final evaluation of the technological integration in the Factory UC is divided into two parts. The first part, hereafter named **full-5G-RAN** PoC, shows the successful integration of a fully 5G compliant industrial communications network. This setup consists of 5G modems that were developed by 5CMM, which are responsible for exchanging information among industrial end-devices and the 5G base station unit (gNB). The data received over the air at the gNB is then routed through the 5G core provided by CMC. The end-devices that are used in this PoC are AGVs provided by ASTI. The main objective of this demonstration is to validate the communication performance among these industrial devices through a lightweight 5G-compliant modem. Thus, it represents an important step towards the employment of 5G wireless technology in industrial scenarios.



The second part, named **flexible-RAN** PoC, showcases the integration of non-3GPP physical layer (PHY) and medium access control (MAC) techniques with a 5G core network, which is in turn managed by the end-to-end network slice orchestration framework (i.e., the MANO), which is composed by an end-to-end network slice management function (NSMF) integrated with two dedicated network slice subnet management functions (NSSMFs) for the 5G core and the flexible-RAN. The non-3GPP radio access technology is referred as Flexible PHY/MAC in previous deliverables throughout the project timeline.

For each PoC one testbed has been developed. The full-5G-RAN setup is located in the University of Burgos, Spain, where the final integration of the components as well as the performance evaluation have been carried out in February 2023. The flexible-RAN PoC setup is assembled in the laboratories of the Technische Universität Dresden, Germany. The final integration and performance evaluation have also been carried out in February 2023.

2.2 Setup and Execution

The following subsections provide description of the setup and execution of Factory UC. Additional information can be found in Annex I – Setup and Execution.

2.2.1 Part I

The demonstration takes place in Burgos, Spain, more concretely in an industrial space in the University of Burgos, where the Joint Research Unit between ASTI and the University of Burgos is located. The setup components are:

- **gNB (RRH + BBU + GPS) n40:** The gNB consists of Nokia outdoors mini-Macro Airscale model. The BBU is connected to RRH through Single Mode Fiber and 10 Gbps network capacity. The BBU gets the time synchronization through GPS signal. The base station was configured with a bandwidth of 20 MHz in the range 2370 – 2790 MHz.
- **5G Core Standalone:** The 5G core is installed in E900-4E Supermicro with 2 SFP+ interfaces of 10 Gbps where one of them is connected to the gNB BBU. The 5G Core is installed in bare metal where all the 5G Core network functions are running as individual processes over Linux Ubuntu 20.04LTS. The Supermicro server has additional 1 Gbps and 10 Gbps network interfaces if needed for connecting the 5G Core to a Data Network (DN).
- **3 AGVs:**
 - eBot: 5G modem + raspi + RealSense.
 - Tribot: 5G modem + raspi + controller.
 - EasyBot: 5G modem + raspi + humidity and temperature sensors.
- **5G modem for connecting the PCs to the network as another UE.**

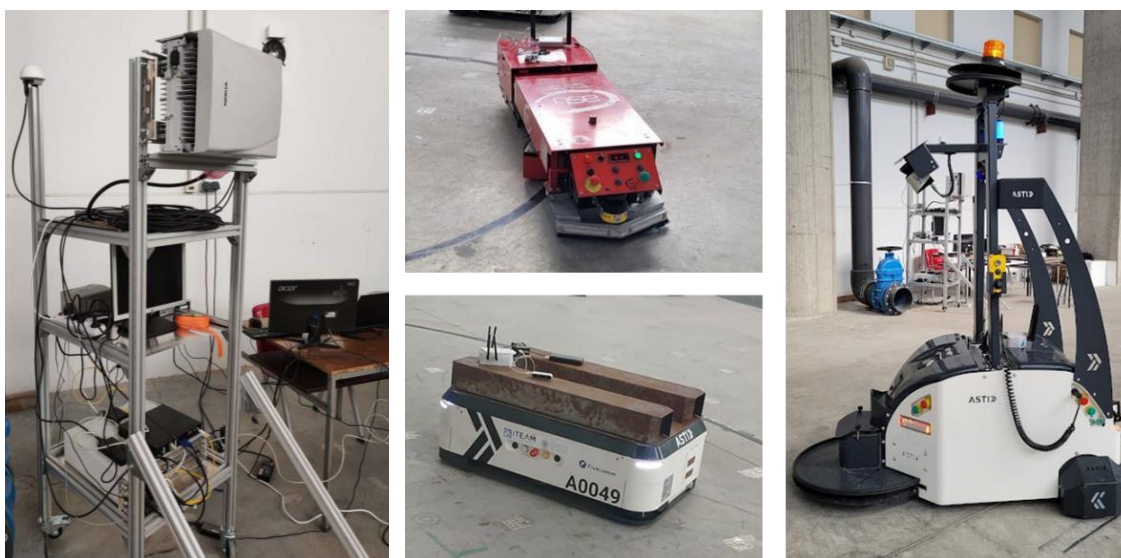


Figure 1: gNB (left), EasyBot (top-middle), eBot (bottom-middle) and Tribot (right)

In Figure 2 the architecture of the setup and how all the components are interconnected is illustrated.

The **eBot** has a 5G modem connected to the 5G LAN and to one Raspberry Pi with CAN bus via ethernet. This Raspberry Pi receives the control commands sent by a laptop through the 5G network and translates them to the data that the AGV understands. In addition, there is also an Intel RealSense camera connected to the Raspberry Pi that sends the real-time video back to the laptop.

The **Tribot** also has a 5G modem connected and one Raspberry Pi with CAN bus via ethernet. The controller is connected directly to the 5G Core, and it sends the control commands to the Tribot. The Raspberry Pi receives the data and sends it to the AGV. The AGV is sending information about its internal variables (linear speed, rotation speed, level battery, errors states and others) to the core.

The **EasyBot** is moved automatically following a black magnetic band on the floor of the facility. It is connected to a 5G modem and provides the core with information about the temperature and humidity of the environment by using sensors installed in the AGV with a Raspberry Pi. In this case, these values are collected by the sensor DHT11 with Arduino and sent to the Raspberry.

The specific architecture of each AGV and more details about the AGVs and setup deployments can be found in Annex I – Setup and Execution.

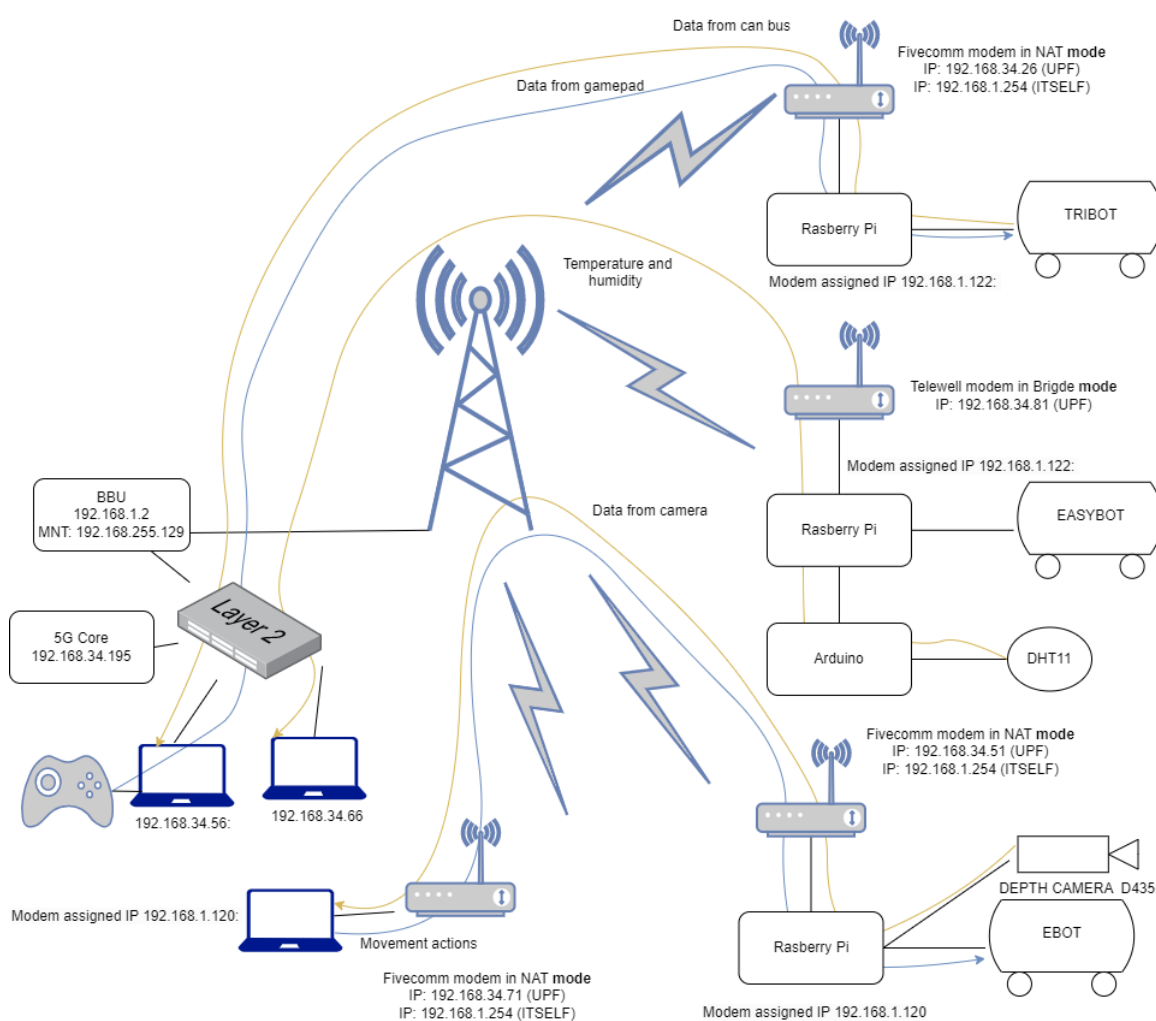


Figure 2: Architecture of factory UC demo in Burgos



Figure 3: Execution of the demo in Burgos

All data flows in the demo are summarized in Table 2.

AGV	Sender	Receiver	Data
eBot	PC 192.168.34.71	AGV 192.168.34.51	Control commands
eBot	AGV 192.168.34.51	PC 192.168.34.71	Real-time camera flow
Tribot	Core 192.168.34.195	AGV 192.168.34.26	Controller motion actions
Tribot	AGV 192.168.34.26	Core 192.168.34.195	Internal variables from AGV
EasyBot	AGV 192.168.34.81	Core 192.168.34.195	Humidity and temperature measurements

Table 2. Data flows in Burgos' demo

2.2.2 Part II

This PoC focuses on showcasing the integration between the flexible PHY/MAC, 5G core and MANO. The deployed setup diagram is shown in Figure 4, which illustrates how the MANO software components, such as the end-to-end NSSMF, Core NSSMF and RAN NSSMF, are connected to TUD's testbed equipment. The information exchange among the Flexible PHY/MAC and the RAN NSSMF is accomplished through the Tactile API developed within WP5. This API is based on JavaScript Object Notation (JSON) format and the configuration files are exchanged via the user datagram protocol (UDP).

The implemented PHY protocol running at the Flexible PHY/MAC base station (BS) listens continuously and waits to get the resource allocation from the RAN NSSMF. This latter component sends a JSON file via UDP containing the desired resource allocation for each application (UE). Then, the Flexible PHY/MAC BS extracts this information and analyses it (total of allocations must be $\leq 100\%$), and distributes the resources among the UEs accordingly sending

acknowledgement message to the MANO to let it know about the current status. Whenever an error occurs, e.g., if the total of allocations is more than 100%, or the allocation was not successful, the BS informs the MANO about the error and waits to receive a new resource distribution.

The Flexible PHY/MAC BS forwards the data traffic from each application UE using tunnel interfaces provided by a gNB emulator named UERANSIM. Hence, the traffic of the Flexible PHY/MAC is encapsulated in the 5G compliant format before being routed through the 5G core.

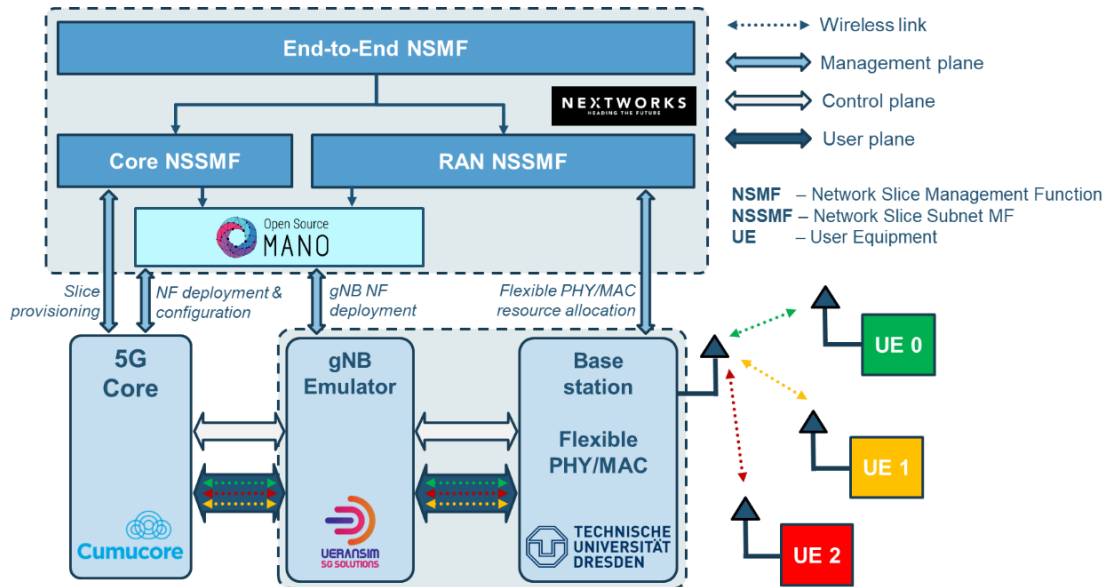


Figure 4: Setup illustration at TUD's testbed used to demonstrate the integration between Flexible PHY/MAC, 5G core, and MANO.

From a software deployment perspective, the NSMF, the two NSSMFs, and the web graphical user interface have been deployed as docker containers in the TUD testbed. The NSMF realizes the high-level logic for end-to-end network slice management and orchestration, the Core NSSMF interacts with the 5G core to configure and provision tailored slices in the 5G core network, while the RAN NSSMF manages the resources at the Flexible PHY/MAC RAN level.

Figure 5 depicts the outcome of two end-to-end network slices provisioned into the TUD testbed, each of them composed by a 5G core subnet slice and a Flexible PHY/MAC RAN subnet slice.

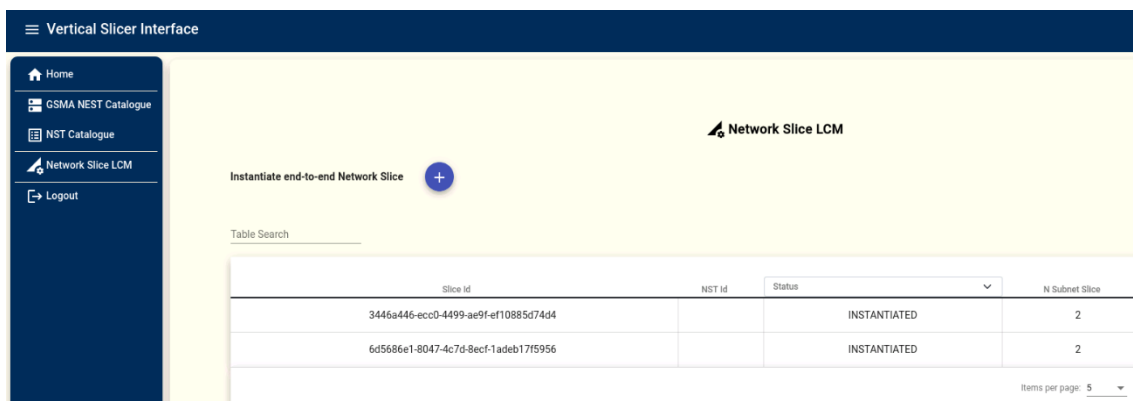


Figure 5: Web GUI showing two end-to-end network slices provisioned.

2.2.3 ISSUES ON EXECUTION

The following subsections provide description of issues encountered during the demonstration and mitigation actions to solve them.

Description of the issue	Mitigation measures
Some radio links between UEs and BS in the setup of part II have lower throughput than expected due to the processing capacity of the host PCs.	Instead of employing a video transmission to demonstrate the setup, emulated devices are used to send data streams from each UE.
Laser of eBot AGV in part I was detecting itself as an obstacle	Recalibration of the laser following ASTI instructions.
5G Core deployment in bare virtualized environment due to lack of Internet access	The 5G Core was deployed as Non-Public Network (NPN) without Internet access which made the installation in a virtualized environment based on containers not possible, since the installation required downloading SW packages from Linux repositories. To overcome this limitation the 5G Core was installed in bare metal after downloading SW binaries into the server.

Table 3. Factory UC issues on execution

2.3 Validation and Results

This section provides a detailed description of the validation results obtained after the execution of the demonstration. Impact is analyzed after explaining the result validation, verification of test cases, KPIs and assessment of the UC.

2.3.1 TEST CASES VERIFICATION

In this section, the results of each test case, identified in D6.1 [1] for the Factory UC are presented. When writing these contributions, it is planned to have the final review in Valencia. To this end, the UPV testbed is taken into consideration instead of the ASTI testbed, because of the connectivity problems encountered. Specifically, the ASTI testbed is isolated from the Internet, thus it was not possible to remotely connect to it, and the installation of the necessary software from the different remote locations was not possible.

For some test cases, beyond the result of the test case itself, is reported also the testbed/lab where it has been executed.

Test Case ID	Result
UC1_TC_01 – Hardware and software implementation	Passed*
UC1_TC_02 – Core network integration testing	Passed*
UC1_TC_03 – Gateway test	Passed
UC1_TC_04 – Onboard industrial IoT network slice templates and NF descriptors	Passed
UC1_TC_05 – Automated deployment of industrial IoT network slice instance	Passed
UC1_TC_06 – Automated termination of industrial IoT network slice instance	Passed



UC1_TC_07 – Manual scaling of an industrial IoT network slice instance	Passed
UC1_TC_08 – Automatic slice configuration through 5GC NSM	Passed
UC1_TC_09 – Automated deployment of industrial IoT network slice instance and of an edge robot control application as part of network slice instance	Passed
UC1_TC_10 – Automated termination of industrial IoT network slice instance and of edge robot control application as part of network slice instance	Passed
UC1_TC_11 – Subscription to either NWDAAF or NEF for collecting monitoring and analytics information related to the network slices, NFs and UEs	Passed
UC1_TC_12 – Deletion of either NWDAAF or NEF active subscription	Passed
UC1_TC_13 – Automated slice scaling triggered by AI\ML platform using NWDAAF data	Passed
UC1_TC_14 – Robot interface connectivity	Passed
UC1_TC_15 – Test of API	Passed

Table 4. Factory UC Test case verification

A couple of the tests listed above have been partially achieved. In particular, **UC1_TC_01** validated the implementation of the Flexible PHY/MAC, and as it is explained in the next subsection, the target latency was achieved. However, the target throughput wasn't achieved due to the spectrum bandwidth available. Similarly, the same reasoning for the partial of the target KPIs of **UC1_TC_01** can be applied to **UC1_TC_02** since the PHY throughput is the network bottleneck.

In summary, the main aim of the test cases execution and verification was to validate the implemented heterogeneous hardware and software network technologies in support of the industrial IoT scenario with automated robots and AGVs (namely the CMC 5GC, the 5CMM modems, the ASTI AGVs, the TUD Flexible PHY/MAC, the NXW end-to-end network slice orchestration framework). Specifically, the tests covered the integration and validation of the CMC 5GC, the 5CMM modems, the ASTI AGVs, the TUD Flexible PHY/MAC, the NXW end-to-end network slice orchestration framework, which have been demonstrated to fulfil the planned functionalities and achieve the defined tests. In summary, according to the table above, this verification covered the following aspects:

- The Nokia commercial gNB supporting the band N40 was installed and configured to operate with the assigned frequency license and bandwidth i.e. 20MHz.
- The CMC 5GC was installed in Supermicro server and configured with operator and network code assigned for Non Public Networks (NPN) i.e. MCC=999 and MNC=99. The integration between the CMC 5GC and the NXW end-to-end network slice orchestration for 5G network slices automated deployment and operation.
- The integration between the TUD Flexible PHY/MAC and the NXW end-to-end network slice orchestration for flexible RAN automated control and management and transparent interworking the legacy 5G networks.
- The automation capabilities of the NXW end-to-end network slice orchestration for 5G network slices lifecycle management (including onboarding, instantiation, scaling operations), assisted by AI/ML.
- The implementation of FPGA-based PHY based on generalized frequency division multiplexing.

A detailed description of the test cases execution and verification is provided in the Annex I – Validation and Results.

2.3.2 KPIS

The following table shows the KPIs that were measured and considered relevant during the use case validation. A detailed explanation on how the measurements of the KPIs were taken can be found in the KPIs.

KPI	Test Case Reference	Target	Actual
Coverage	UC1_TC_14	0,01 km ²	0,001 km ²
Mobility	UC1_TC_14	< 30 km/h	8 km/h
Security	UC1_TC_14 UC1_TC_02	High	High
Data rate per camera (uplink)	UC1_TC_14	6 – 24 Mbps	6 – 6,5 Mbps
Data rate per robot UE-UE (without camera)	UC1_TC_14	10 Mbps	14,6 Mbps (UL/DL)
Data rate per robot UE-core (without camera)	UC1_TC_14 UC1_TC_02	10 Mbps	46,1 Mbps (DL) 14,6 Mbps (UL)
Data rate per robot UE-UE (with camera)	UC1_TC_14 UC1_TC_02	10 Mbps	8,95 Mbps (UL/DL)
Data rate per robot UE-core (with camera)	UC1_TC_14 UC1_TC_02	10 Mbps	42 Mbps (DL) 9,01 Mbps (UL)
Datagram transmission reliability (uplink)	UC1_TC_14 UC1_TC_02	-	100% up to 20 Mbps
Connection density per robot	UC1_TC_14	10k/Km ²	13.6k/Km ²
E2E latency for remote control (command from application to remote device)	UC1_TC_14 UC1_TC_02	10-50 ms	12-50 ms
Reliability for remote control	UC1_TC_14	99,999%	99,999% (no disconnections during tests)
Throughput (Flexible-RAN)	UC1_TC_01	Max: 10 Mbps Min: 0.1 Mbps	Max: 2.94 Mbps Min: 0.34 Mbps
E2E Latency (Flexible-RAN)	UC1_TC_01	Max: 10-50 ms Min: 1-5 ms	Max: 2.9 ms Min: 1.6 ms

Table 5. Factory UC KPIs

Regarding the coverage KPI, the walk test (to measure the quality of the radio signal) was performed in the interior of the industrial unit, considering an area of 0,001 km². This area was considered sufficient for the use case execution, thus not measuring the outside of the industrial unit. The setup includes the 5GLAN feature available in the 5G Core that allows to create private group of devices that can only connect with each other. The 5G Core creates a virtual interface for interconnecting all the devices part of the same 5GLAN group. Other devices outside the 5GLAN group will not be able to connect to the ones in the group.



The 5GLAN allows to isolate and secure the communications within a virtual private group created by the 5GLAN.

The camera was set to send a video streaming with 1280x720 pixels resolution. With that configuration, the uplink traffic sent by the camera was oscillating between 6 and 6,5 Mbps. In Figure 6, in the middle graph, the video streaming throughput is shown in purple.

As it is further explained in the KPIs, the throughput of the network was measured considering different setups: a setup in which the tests were being performed from the UE to the core (see Figure 85) and another with the tests being performed between 2 UEs (see Figure 84). Also, different conditions were considered: with and without video streaming running.

The first test performed was the communication between two modems connected to the 5G network with the core 5GLAN feature. The iperf3 test showed a throughput of 14,6 Mbps UL/DL without video streaming running and 8,95 Mbps with the video streaming running. The average RTT (Round-Trip Time) during the test was about 200 ms with video streaming running in Figure 6. In these tests, the RTT between command from the applications to a remote device can be analyzed, where the values (with video streaming running) range from 50ms to 100ms, therefore it can be concluded that the approximate E2E latency is between 25ms to 50ms.

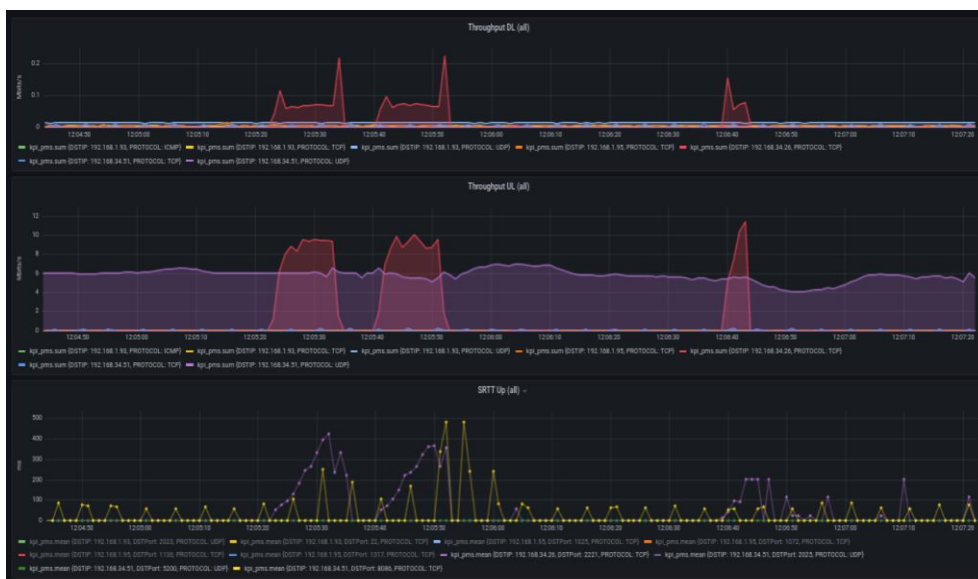


Figure 6: Grafana KPI dashboard visualization of UE to UE with video streaming running. 1) Top: Downlink throughput – iperf3 test (Pink). 2) Middle: Uplink throughput – iperf3 test (Pink) and video streaming (Purple). 3) Bottom: RTT uplink – influxDB connection (Yellow) and iperf3 test (Purple).

The second test was the communication between a UE and a laptop connected directly to the 5G core. Without video streaming running the result of the throughput measured with iperf3 was of 46,1 Mbps DL and 14,6 Mbps UL and the average RTT obtained was about 120 ms (see Figure 7). With the video streaming running 42 Mbps DL and 9,01 Mbps UL were obtained as throughput and 190 ms as the average RTT (see Figure 8) We can observe that the throughput is lower and the RTT is higher, since the real-time video consumes bandwidth and resources.





Figure 7: Grafana KPI dashboard visualization of UE to core without video streaming running. 1) Top: Downlink throughput – iperf3 test (Orange). 2) Middle: Uplink throughput – iperf3 test (Orange). 3) Bottom: RTT uplink – influxDB connection (Yellow) and iperf3 test (Purple).



Figure 8: Grafana KPI dashboard visualization of UE to core with video streaming running. 1) Top: Downlink throughput – iperf3 test (Blue). 2) Middle: Uplink throughput – iperf3 test (Blue) and video streaming (Purple). 3) Bottom: RTT uplink – influxDB connection (Yellow) and iperf3 test (Purple).

It was considered relevant for the UC to measure the maximum bandwidth that the 5G network could support in the uplink. This measure can help to figure out the maximum data transmission from the UE to the core without loss of packets. To perform the measure, iperf3 test was used on UDP mode, analysing the bandwidth limit where UDP datagrams started being lost, this limit was established in 20Mbits. We can determine that it is possible to send up to 20



Mbits per second of data without loss datagrams. In the use case validation, it was only 6-7Mbits per second of video streaming transmission, thus implying we have about 13 Mbits to transmit more information from the UE to the 5G network.

To determine the performance of the Flexible PHY implementation with different numerologies, end-to-end measurements were conducted using the tool *UDPtest*, which was developed at the Vodafone Chair Mobile Communication Systems (TUD). This software tool generates UDP packets, transmits them to an IP address and port and receives them from another port. The size and the time between the UDP packets can be set, hence varying the data throughput. The tool can measure the throughput, the latency and the frame error rate (FER). The measurement setup is visualized in Figure 6. The tool *UDPtest* is running on Device 1, which in an NI USRP-2974. The UDP packets are transmitted over an Ethernet connection to the Device 2, an NI PXIe-1082 with and NI USRP-2944R. This device runs the PHY transmitter and sends the signal over the wireless channel to the Device 3. This device is an NI USRP-2974 which runs the PHY receiver. The received UDP packets are forwarded over Ethernet to the Device 1, which measures the throughput, the frame error ratio (FER) and the latency.

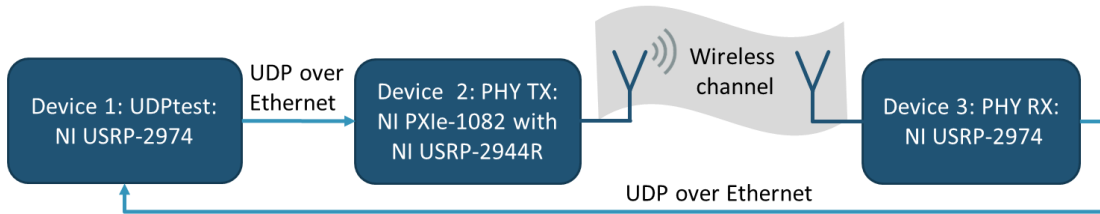


Figure 9: Illustration of the components used in the measurement setup.

A wireless line-of-sight (LoS) channel was set with a distance of 4 meters in a controlled and static laboratory environment. A transmit power of 0 dBm was defined and the sample rate was fixed to $B = 20$ MHz. 3.75 GHz was used for the carrier frequency. For $N \geq 1024$, the cyclic redundancy check (CRC) with 16 bits was applied, and CRC with 8 bits otherwise. Eight (N_{pilots}) pilot symbols are employed at all payload configurations, and N represents the number of samples of each multicarrier symbol. However, a smaller number of pilots, $N_{pilots} = 4$, are used for the control channel. The time between the UDP packets was defined to be 200 μ s for $N \geq 2048$ and 100 μ s otherwise.

Without considering the host processing limitations, the throughput of the PHY is calculated based on the PHY parameters and the occupied bandwidth:

$$\text{throughput PHY} = \frac{B}{N_{\text{frame}}} N_{\text{bitsFrame}},$$

where $N_{\text{frame}} = N_{\text{preamble}} + N_{\text{payload}}$ defines the number of samples in a frame. The number of samples in the preamble is defined as $N_{\text{preamble}} = 2N_{\text{chirp}} + N_{\text{CP}} + N_{\text{CS}}$ and for the payload as $N_{\text{payload}} = N + N_{\text{CP}} + N_{\text{CS}}$. The length of the chirp, the cyclic prefix (CP) and the cyclic suffix (CS) are defined as $N_{\text{chirp}} = 64$, $N_{\text{CP}} = 32$ and $N_{\text{CS}} = 15$, respectively. The overhead ratio, describes how many samples of the frame are used for the preamble, and is given as:

$$\text{overhead ratio} = \frac{N_{\text{preamble}}}{N_{\text{frame}}} = \frac{N_{\text{preamble}}}{N_{\text{preamble}} + N_{\text{payload}}}$$

We propose 11 different configurations of the PHY in order to meet different application requirements. The configurations are divided into four groups, namely, H, M, L and C. Specifically, H, M and L represent the high, medium and low throughput configurations, respectively, while C is reserved for the control information. The calculated and the measured throughput for the different configurations are depicted in Table 6 where the FER = 0 was achieved for all configurations, meaning that the wireless link was reliable during these tests.

Config	N	N_{on}	K	M	QAM Order	Bytes per block	Throughput PHY (Mbps)	Over head ratio	Throughput measured (Mbps)
H0	2048	1792	2048	1	64	666	5.87	0.08	2.94
H1	2048	1680	128	16	64	624	5.50	0.08	2.76
M0	2048	1792	2048	1	16	443	3.90	0.08	1.96
M1	2048	1680	128	16	16	415	3.66	0.08	1.84
M2	1024	896	1024	1	16	219	3.52	0.14	1.75
M3	1024	810	64	16	16	205	3.17	0.14	1.64
L0	1024	896	1024	1	4	108	1.73	0.14	0.86
L1	1024	810	64	16	4	101	1.56	0.14	0.81
L2	512	448	512	1	4	53	1.44	0.24	0.42
L3	512	360	32	16	4	42	1.14	0.24	0.34
C0	64	52	64	1	4	4	0.21	0.61	Too few bytes for <i>UDPtest</i>

Table 6. Flexible PHY/MAC throughput measurements with *UDPtest*

The measured throughput is computed with the tool *UDPtest* using the whole PHY module as seen in Table 6. The difference between the measured throughput and the throughput of the PHY arises due to the limited processing speed of the host, which is not able to handle a higher UDP throughput without dropping packets.

The measured end-to-end latency is similar for all configurations ranging between 1.6 ms and 2.9 ms. In the PHY, a smaller frame size leads to a smaller latency. However, due to no significant smaller latency in the measurements, it can be concluded that the major latency comes from the host UDP processing and the UDP communications over the network.

The throughput of the configuration C0 was not measured, since the tool *UDPtest* cannot generate smaller UDP blocks than 30 bytes. However, the transmission of the control information should be robust, which is achieved with the configuration C0. This was verified by several measurements in both LoS and non-LoS wireless channels. It was observed that when the received signal is synchronized, the control information is reliably detected.

2.3.3 IMPACT ASSESSMENT

This use case focused on cooperative automated robots for future smart factory production lines or warehouses, which are enabled by the integration of a heterogeneous network that interconnects end-devices from different



technologies and dynamically adapts itself to the application requirements. With the deployment of edge computing, the industrial network will be regarded as a distributed computation platform that enables the programming and scheduling of robots and other resources for multiple tasks. The PoCs of this use case demonstrate how 3GPP-compliant wireless communications systems are able to provide services for industrial scenarios.

The allocation of the robot's controller into the MEC provides important cost-saving benefits and new functionalities. This strategy allows simplifying and reducing the hardware within the robots, as hardware to compute the control strategies is deployed in the MEC and can be shared by all robots. The advantages are not only cost-saving related but to all benefits of virtualization: easy deployment, flexibility, replicability, and redundancy, among others, are extensible to the robot sector. This allows to improve the efficiency, flexibility and quality of the supply chain and production processes handled by robots.

In industrial facilities, AGVs, robots, transport vehicles and people circulate. All of them could be equipped with devices capable of sensing the environment. In this way, they could capture information on temperature, humidity, noise, presence of particles in the air, etc. of the points where they are passing through. All this information could be monitored in real time and it can be possible to detect events and anomalies in the processes, allowing decisions to be made about the processes based on the data collected by the sensors. Additionally, this information could be used to train predictive maintenance systems to react to anomalies in the production chain before they occur.

The validation of the end-to-end network slice management capabilities on top of the flexible RAN and PHY-MAC technologies represents a highly impacting result, as it demonstrates the integration of non-standard RAN technologies with legacy 5G networks, specifically in private 5G scenarios for smart factories. Indeed, while the PHY-MAC developed in the project, and deployed and demonstrated in the TUD testbed is a non 3GPP standard technology, the integration performed with the 3GPP compliant CumuCore 5GC validates the feasibility of its deployment, and use in legacy 5G private networks. This allows to avoid the deployment of multiple ad-hoc non-standard networks, and thus enable the integration of heterogeneous technologies under the same 5G private network.

Moreover, the use of end-to-end network slice orchestration capabilities enables full network automation by provisioning network and computing resources for operation in private 5G networks for industrial IoT scenarios. This allows to drastically reduce the complexity of private 5G networks management and control, especially in industrial contexts and environments where networking expertise might be limited. In addition, the validated end-to-end orchestration functionalities enable the delivery of tailored slices in support of diverse concurrent vertical services, including AGV and robot control, AR/XR video streams, IoT sensing/actuation traffic.

The deployment of 5G as Non-Private Network (NPN) with 5GLAN and TSN functionality can address the needs of industrial use cases that require secure mobile communications. NPN can bring secure infrastructure connected only to local data network and optimize specific industrial scenarios resources.



2.4 Lessons Learned and Potential Improvements

A possible improvement in the protocol for sending sensor data to the base station is using an event-based sampling approach. Currently, sensor data is captured periodically with a constant sampling period. Each time a sensor sample is taken it is sent to the station. The value of the sensed variable may not have changed in value from the previous sample, but it is sent anyway. In contrast, in event-based sampling, the sensed variable information is only sent when an event is detected, i.e. a relevant change in the variable. In this way, by using event-based sampling, communication bandwidth could be saved.

The 5GLAN brings some added value when creating private groups of devices to be connected within the industrial network. However, networking and IP planning has to be done differently than public networks which connect from mobile devices to public Internet and require Firewall and NAT. Instead, the 5GLAN connects mobile devices to fixed devices in the same Data network which requires flat IP addressing and proper routing policies to ensure device to device communication between wired and wireless devices.

It has been observed during the measurements with the Flexible-RAN setup that significant contribution to the observed end-to-end latency comes from the networking protocol on top of the MAC and PHY layers. This means that for obtaining end-to-end latencies close to 1 ms, more attention should be paid to the upper layers of the communications protocol stack. A potential improvement for the Flexible PHY/MAC is the ability to support runtime reconfigurability of the PHY in a frame-by-frame fashion.

The integration, testing and demonstration activities which involved the end-to-end network slice orchestration framework have shown the importance of the availability of well-defined and accurate management and control APIs for the support of full automation in service and slice deployment and operation. Specifically, the early availability of the CMC 5GC APIs, as well as those exposed by the PHY-MAC control, allowed to implement in software proper network slice management logics, and also prepare mock-ups to carry out standalone early integration and validation activities. This is a crucial aspect and lesson learned especially when software and hardware components are provided by different vendors or institutions in general. However, it is equal (if not more) important to have standardized APIs and operational workflows. While this is true for the 5GC, where API exposure is at the hearth of the 3GPP specifications (with the Network Exposure Function – NEF - functionalities), still for the 5G RAN this is an open issue. Different vendors still expose custom and tailored (often non-open) APIs to control and manage their RAN network functions. In this direction, the wide adoption of standard (or de-facto standard) solutions like the one from the Open RAN (O-RAN) architecture would allow to further improve the multi-vendor interoperability for end-to-end 5G networks, as well as make introduction of full automation in slice and services management and operation.



3 PoC - Transportation Platforms Health Monitoring

3.1 Objective and Description

The Transport UC demonstrates safe and secure micro-edge sensors for monitoring and detecting wear and tear of wheels and axles of cargo train carriages. The micro-edge sensors are attached to each axle and pooled via edge-gateways capable of data fusion. These gateways in turn are connected via terrestrial and non-terrestrial (e.g., satellite) access networks to cloud servers for trend analysis and defect-based maintenance alert management. The overall communication is encrypted for security purposes with an added layer of remote attestation to ensure identity and software integrity of the communication endpoints.

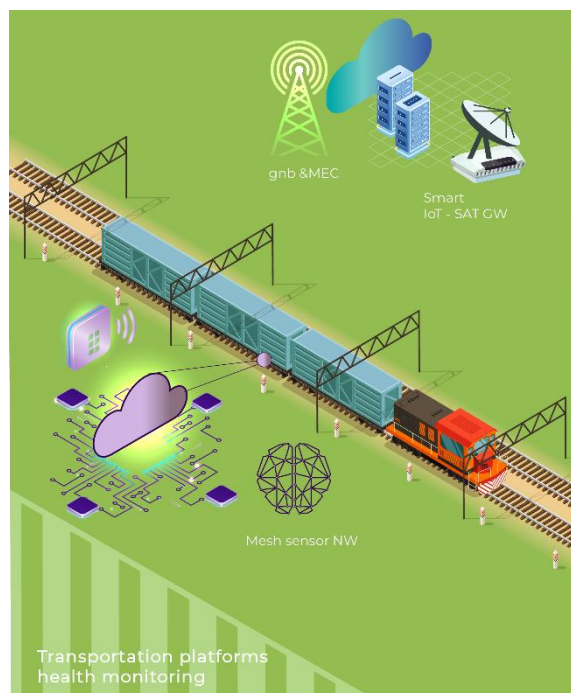


Figure 10: Sensing and GW Modules for Rail-Health Data Logging

Smart Edge Sensors: During the project, two real-world test series with a total of seven train carriages hosting 75 injection faults were conducted. This resulted in 12,630 datasets. These were analysed via machine learning algorithms. Eventually, the machine learning algorithms were refined with physical models. The refined algorithm was tested and retested with empirical



and physical fault model simulation and simulator signals. The resulting re-refined algorithms were then reapplied to the initial real-world data.

The PoC demonstration for final review covers how smart edge sensors can detect and quantify various defects and communicate to cloud platforms for long trend analysis and maintenance alert management.

The edge sensor demonstration includes a physical fault simulator, the sensors for signal pick-up, the near real-time computation engine for feature generation, classification, and meta signal generation, as well as the interface to the end-to-end secure communication platform.

End-to-end Secure Communication: In addition to demonstrate the vibroacoustic sensors and how they work, the use case also highlights how smart edge sensors can report their measurements using novel infrastructure for secure communication. In the Transport UC, safety depends not only on the accuracy of the smart edge sensors, but also on the security of the communication. To ensure safety of the train equipment, defects must be reported (i.e., not redirected or suppressed) before an accident can happen. Therefore, the sensor must be able to transmit securely the information about the defect to a control centre of the train company. It should be able to use whatever connectivity option is available at a certain time and location, but the security of the communication must be ensured.

To this end, the iNGENIOUS project innovates in the area of secure embedded computers and end-to-end secure communication over networks. The PoC demonstrates a computer architecture targeted at IoT devices, along with the operating system M³, which is co-developed with this architecture. The M³ hardware/software co-design follows an isolation-by-default approach to make building secure IoT devices easier. Currently, it is realized as a system-on-chip architecture on a Field-Programmable Gate Array (FPGA). Details about this architecture, its capabilities and security properties are described in Chapter 3 of D3.3 [3].

The FPGA/M³ component aims to demonstrate end-to-end secure encryption and integrity protection of the sensor information, which is transmitted from an IoT device to a cloud server using industry-standard Transport Layer Security (TLS). However, the main purpose of this part of the PoC demonstrator is to enable even stronger security guarantees by enhancing TLS with remote attestation. BI integrated remote attestation with TLS to create a combined protocol called RATLS (Remote Attestation with Transport Layer Security). In addition to establish cryptographic protection of the communication, this protocol also enables the secure exchange of information about the identity and integrity of software running on both the IoT device and the cloud server. RATLS is demonstrated with mutual attestation of both the IoT device and cloud endpoints.

The RATLS connection between the IoT device and the cloud is routed through a simulated satellite link to demonstrate feasibility and also the ability to ensure ubiquitous connection between the sensor and cloud.

The PoC demonstration of the Transport UC has taken place in labs at BI and NCG in March 2023 (M30 of the project) and makes use of the satellite testbed provided by iDR.



3.2 Setup and Execution

The PoC demonstration is a lab-based setup showing how rail-health monitoring could be implemented. It consists of six main components: 1) a physical train axle fault-simulator, 2) vibration sensors for signal pick-up, 3) a FPGA-based computing engine for signal pre-processing and fault classification, 4) the M³ FPGA for IP communication and remote attestation, 5) a simulated satellite link for reporting detected defects from IoT sensor to the cloud, and 6) an interactive demonstration dashboard.

Train axle fault-simulator and vibration sensor: While the Transport UC is a real-world application, it is not possible to demonstrate evolving commercial transport lorry defects in a practical manner in real-time. Therefore, the PoC demonstration consists of a live demo via a physical rail-health fault-simulator. The physical fault-simulator can simulate defect-free operation or flat spots with or without additional bearing defects. The concept design of defect-simulator, as well as its physical realization is shown Figure 11. The rail axle runs on a rolling stand simulating the rail track. Surface defects on the rolling stand simulate real-world equivalent flat-spots of 3 to 8 cm in width. As the rolling stand is moved parallel to the wheel axle, various combinations of single and multiple flat spots can be simulated and fault-injected into the sensing system. Bearings supporting the axle can be chosen with or without bearing defects (inner, outer, roller, or combination). Both speed and load parameters can be adjusted. This allows the testing and data collection of millions of simple and complex fault combinations, far exceeding the available real-world data.

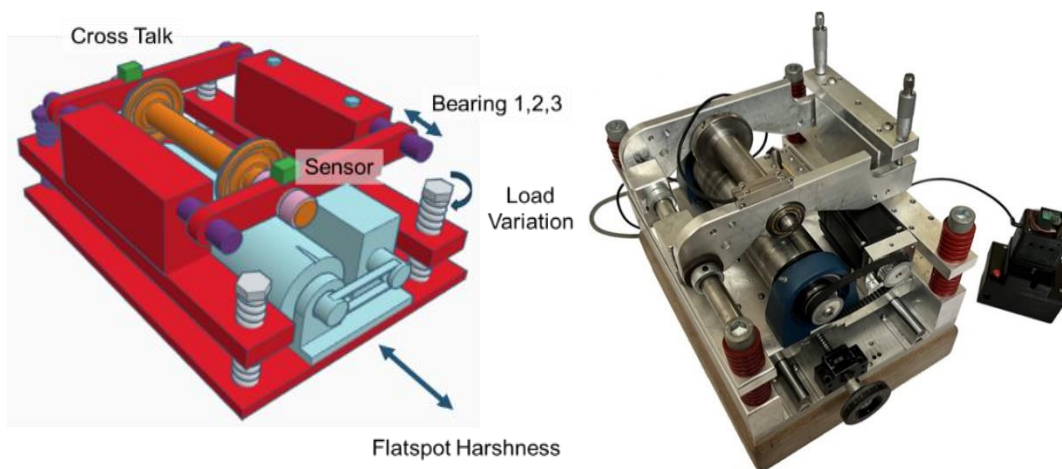


Figure 11: Rail-Health Flatspot Harshness and Bearing Defect Demonstrator (Concept & Design)

This physical fault simulator was correlated with known real-world data and then used to validate theoretical simulation fault models. Theoretical simulation fault models were very important in this development to determine maximum sensing resolution requirements and in understand sensing limits. Theoretical data was used for boarder testing of the embedded hardware algorithm implementation. Embedded hardware typically uses lots of approximations, to stay performance and energy efficient. To make sure that there are no undesirable effects, boarder testing is a very effective validation technique. Figure 12 shows the test setup for simulated fault injection and physical sinusoidal fault injection testing.





Figure 12: Rail-Health Simulated Fault & Sinusoidal Fault Injector Set-up

The combination of real-world, simulation and physical-fault model testing ensures that the integrity of the algorithms developed survives real world variation and allows recognizing even previously unseen real-world events.

FPGA-based IoT computer and cloud server: The third main component is an FPGA-based prototype of a secure-by-default embedded computer. The FPGA is shown Figure 13, together with an Ethernet extension board that enables network connectivity. The FPGA is connected to the microcontroller of the NCG sensor via a UART link. The system-on-chip synthesized onto the FPGA is the hardware part of a hardware/software co-design that supports the M³ operating system. The M³ OS hosts an application that has access to the UART interface. This application obtains both the raw sensor data and the defect classification from the NCG smart sensor and sends the data to a server application running on a Raspberry Pi 4B single-board computer (fourth component, see Figure 14). This Raspberry Pi represents a cloud server of the hypothetical train company, which monitors their assets and would react to reported defects by sending affected carriages to maintenance.



Figure 13: BI's M³ hardware/software co-design platform realized on FPGA development board, with external Ethernet extension board

The Raspberry Pi 4B uses a Trusted Platform Module (TPM) as the root-of-trust for generating an attestation report about the server software running on the single-board computer. On the IoT device side, a signature service running on a dedicated processor tile simulates the root-of-trust that BI designed for the M3 hardware/software co-design platform. On both the IoT device and the cloud endpoint, RATLS obtains a software attestation report from the respective root-of-trust that is part of the system. Using these attestation reports, both endpoints validate the identity and integrity of the software running their respective peer.

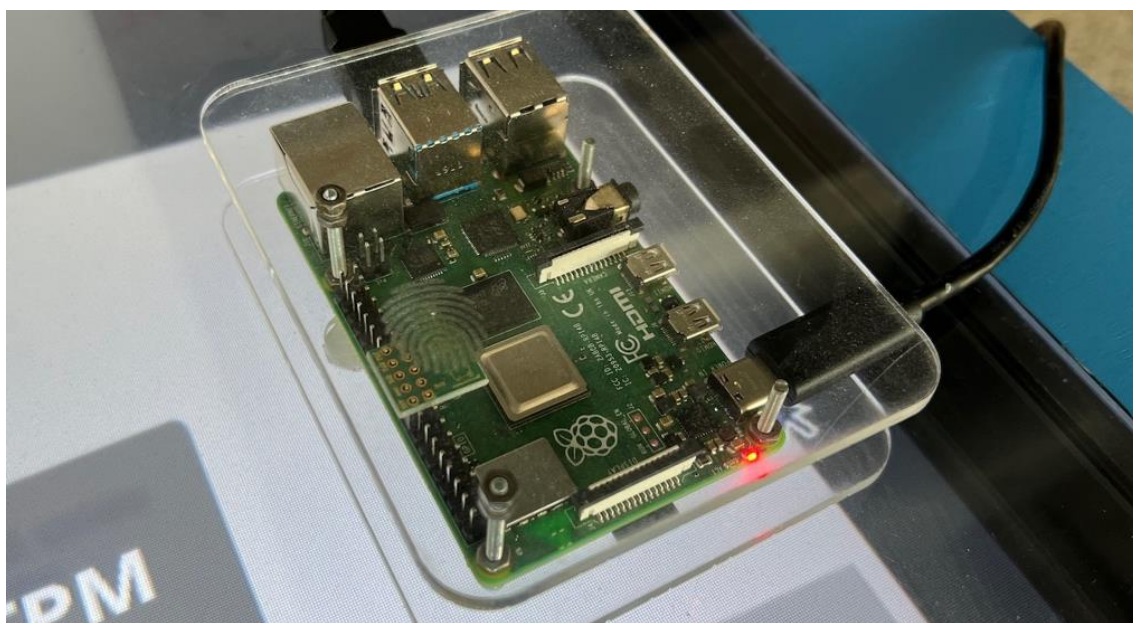


Figure 14: Raspberry Pi 4B single-board computer with a Trusted Platform Module (TPM) attached to the GPIO pin header

Simulated satellite link: To simulate the satellite network for this testbed, iDR provided access to a simulated satellite emulator that connected to real satellite remotes and hub terminals, which are accessible via ingress and egress routers at iDR and SES premises. As shown in Figure 15, BI’s FPGA and the Raspberry Pi connect via the internet to the simulated satellite network endpoints to establish a RATLS connection over the simulated satellite. For this use case, the simulated satellite network characterised the timing behaviour of a real satellite in geo-stationary (GEO) orbit.

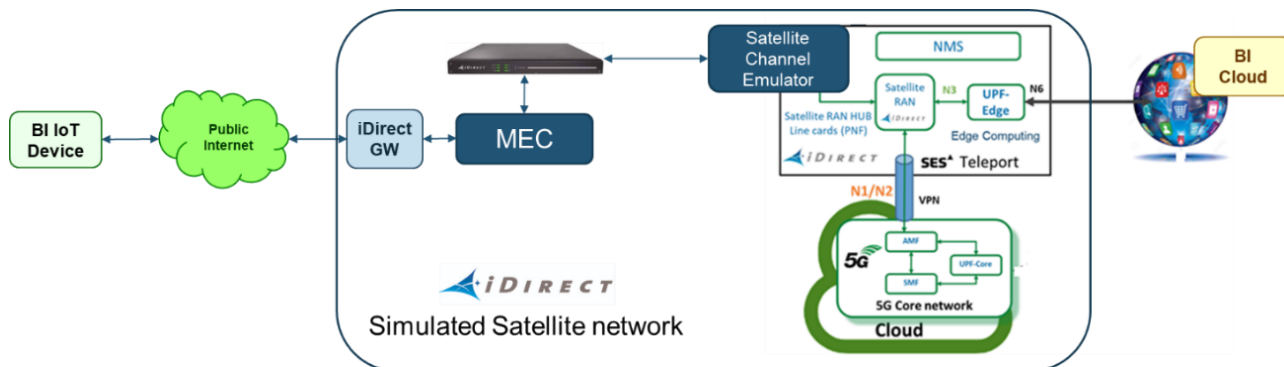


Figure 15: Simulated satellite infrastructure with ingress and egress routers, enabling the smart edge sensor (BI IoT device) to connect to the cloud server (BI cloud)

Demonstration Dashboard: There is sixth component of the PoC demonstration for the Transport UC that is an interactive dashboard. The dashboard was built by BI and consists of a table with a large touchscreen (see Figure 16) that visualizes the state of the RATLS connection establishment and data transmission. Remote attestation can be switched on and off to demonstrate the security enhancements.

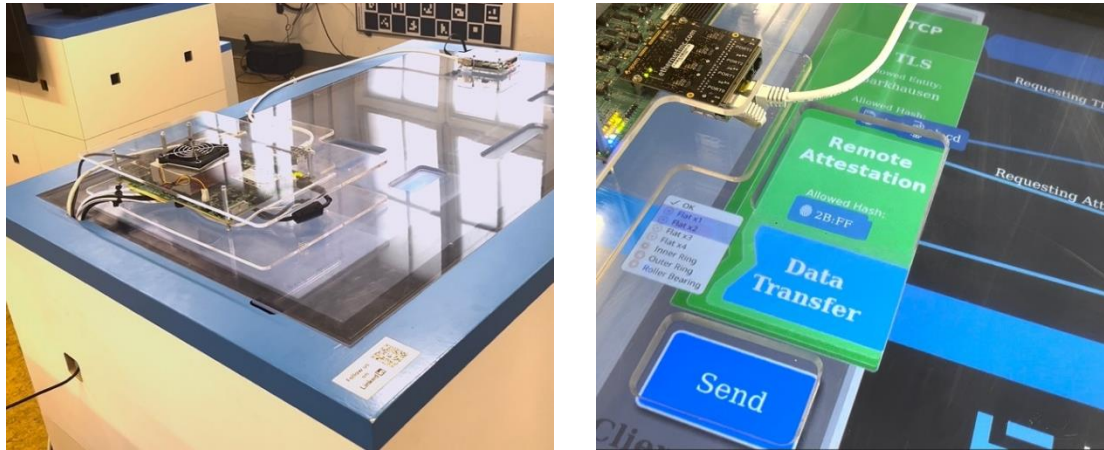


Figure 16: Main dashboard table showing RATLS connection establishment and touch controls for enabling and disabling remote attestation

A secondary display (Figure 17) shows the internal state of the smart sensor and the raw data measured by the vibroacoustic sensor, as well as the classification of the vibration patterns (“OK” and various types of “flat spots” and “bearing defects”).

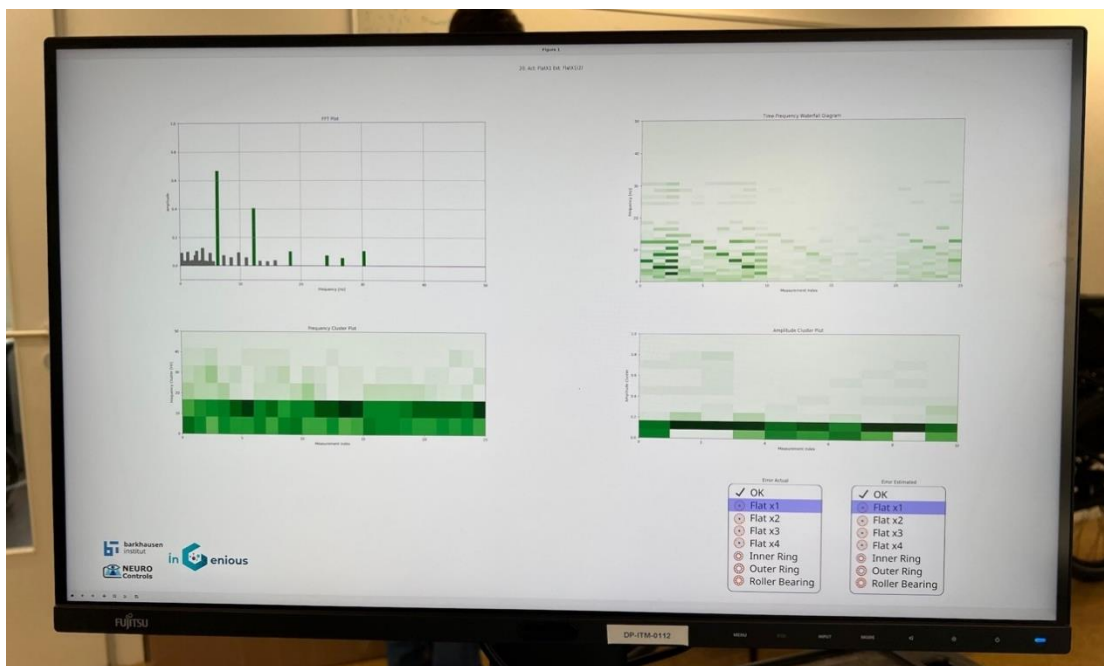


Figure 17: Secondary display showing smart sensor state and defect classification.

The PoC demonstration included the following steps:

1. **NCG train axle simulator:** Configuration of “no defect” as well as various types of defects (“flat spots”, etc.) that are measured using the vibroacoustic sensor. The measured data and classification are shown on the secondary screen of the dashboard and explained.

2. **BI cloud server:** Configuration of the cloud server to run either “correct” or “manipulated” software, which will be checked by the FPGA/M³-based IoT device.
3. **BI FPGA/M³-based IoT device:** Reporting of detected defect type to the Raspberry Pi cloud server by initiating, via the touchscreen, the transmission using both standard TLS (state of the art) and RATLS (with stronger guarantees about connection).
4. **BI touchscreen table:** Discussion of the results and security properties of the transmission for “correct” / “manipulated” software and “standard TLS” / “RATLS” configurations.
5. **IDR data logs:** Review of telemetry captured for RATLS transmission over the simulated satellite link to verify data encryption.

3.2.1 ISSUES ON EXECUTION

The following subsections provide description of issues encountered during the demonstration and mitigation actions to solve them.

Description of the issue	Mitigation measures
Early during execution of the project, BI identified the risk that an FPGA-based hardware implementation of a root-of-trust for the M ³ platform would not be ready to use for demonstration.	According to the mitigation plan for the identified risk, the software-based parts of the root-of-trust were run on a dedicated processor tile to simulate the behaviour of a fully integrated root-of-trust in the hardware. The risk and mitigation measures have been documented in D6.1 [1], D6.2 [2], and D3.3 [3].
A stability issue caused certain application scenarios to crash consistently, including the remote-attestation scenario that is required for demonstrating the Transport UC. A bug in the FPGA-based hardware design of the M ³ platform is suspected to be the cause for this problem. Due to limited (hardware) debugging capabilities of the experimental M ³ platform, the bug has not been fixed at the time of writing. Development, testing, and integration activities were slowed down.	As a mitigation, the M ³ hardware design and operating system have been downgraded to the base version used for the mid-term review. This version was known to be sufficiently stable for the remote attestation scenario. System services and integration for client-side measured application launch and root-of-trust signature service have been backported to finalize demonstration activities.

Table 7. Transport UC issues on execution

3.3 Validation and Results

In this section, the results the test cases identified in D6.1 [1] are summarized.

3.3.1 TEST CASES VERIFICATION

In this section, the test case verification is reported. More additional information on test case results can be found in Test Cases Verification.

Test Case ID	Result
UC3_TC_01 – Lifetime Operation – Battery Life	Passed*
UC3_TC_02 – Connectivity Frequency	Passed*

UC3_TC_03 – Connectivity Coverage	Passed
UC3_TC_04 – Edge Storage	Passed
UC3_TC_05 – Multimodal Connectivity	Passed
UC3_TC_06 – Monitoring Resolution	Passed*
UC3_TC_07 – Monitoring Capability	Passed*
UC3_TC_08 – Cloud Defect Validation (Optional)	Passed
UC3_TC_09 – Gateway Defect Validation	Passed
UC3_TC_10 – Security (Phishing)	Passed
UC3_TC_11 – Security (Listening)	Passed
UC3_TC_12 – Security (Flash)	Passed*
UC3_TC_13 – Security (Commanding)	Passed*
UC3_TC_14 – Data Encryption	N/A
UC3_TC_15 – Functional Safety	Passed
UC3_TC_16 – Fire/Explosion Safety	Passed
UC3_TC_17 – The radio access should be able to run local application processing when user selects low latency for selected applications	N/A
UC3_TC_18 – Extended Satellite Coverage – Confidentiality of satellite backhauled sensor data	Passed
UC3_TC_19 – Communication Load Optimization	N/A
UC3_TC_20 – OTA upgradeability (Optional)	N/A
UC3_TC_21 – Extended Satellite Coverage – Satellite Multi-Protocol Support	Passed
UC3_TC_22 – Extended Satellite Coverage – IP Connectivity	Passed
UC3_TC_23 – Extended Satellite Coverage – Satellite backhaul latency	Passed

Table 8. Transport UC Test case verification

The Transport use case is evaluated based on three groups of test cases.

Smart edge sensors: The first set of test cases (UC3_TC01 through UC3_TC09, UC3_TC15, UC3_TC16) covers the smart edge sensors developed by NCG. In summary, the results for these tests show that commercial rail health monitoring is technically and economically possible, even when considering KPI changes not known at the beginning of this project. Remaining, or shall we call them new problems such as 30 instead of 12 years of autonomous operation can be addressed with emerging new technologies such as triboelectric energy harvesters. The research proved that commercial rail-health monitoring can be achieved with very high defect resolution with low-cost Bill-of-Material designs. It also showed defect validation at the edge is far more cost and energy efficient than in the cloud, and that edge classification can be easily validated via simple statistical classifiers to achieve high levels of functional safety integrity. The edge sensing concepts developed for rail-health can be easily ported to other domains – such as industrial condition monitoring of rotary equipment (pumps and motors), giving us a platform for economic growth.

IoT device security and communication with the cloud: The second set of tests, performed by BI, aims at demonstrating that the state of the art in IoT communication security and IoT device security has been advanced. Both



aspects were considered together and cover the enhancements made to the M³ hardware/software co-design platform and the Transport Layer Security Protocol (TLS).

Test cases UC3_TC_11, UC3_TC_12, and UC3_TC_13 cover the extension of TLS with the concept of remote attestation, which resulted in the combined RATLS protocol. They also cover the integration of RATLS with two roots-of-trust: Industry-standard TPM 2.0 (used in the Raspberry Pi that represents the cloud server) and the root-of-trust that has been designed and partially implemented for the M³ platform. This part of the Transport use-case PoC demonstration shows that security guarantees for cooperation and communication between an IoT device and a cloud server are stronger than with standard TLS, as both endpoints mutually attested each other. This attestation performs a verification of the identity and integrity of the software on either device, resulting in end-to-end secure communication between a trustworthy device and cloud server.

Satellite connectivity: The final set of test cases covers the simulated satellite link, aimed at demonstrating feasibility and the ability to ensure ubiquitous connection between the sensor and cloud. The simulated satellite link mimics the timing behaviour of a real satellite in geo-stationary orbit.

The iDR simulated satellite network was used for staging and testing configurations on an ongoing basis and to validate test cases UC3_TC_18, UC3_TC_21, UC3_TC_22, and UC3_TC_23. Figure 15 provides an overview of the simulated satellite network setup and how it was used to connect the IoT device to the BI cloud.

Table 9 provides a summary of the dates the end-to-end testing was performed along with a brief description of activities carried out using iDR’s simulated satellite network.

Lab testing date	Use case testing activities	Lab testing	Status
25/02/22	Design & configuration for Midterm demo	Designed testbed	Passed
05/04/22	Initial testbed testing over simulated satellite network	Verified initial test setup including satellite lab system, GEO satellite simulator and routing. Introduced satellite delay of 560ms and test end-to-end.	Passed
25/10/22	Transport UC Satellite testing	BI successfully routed experimental TLS variant without FPGA over simulated satellite network	Passed
09/12/22	Transport UC Satellite testing	BI successfully routed experimental TLS variant using FPGA over simulated satellite network	Passed
08/03/23	Final Demo Preparation	Verified testbed and routing was in configured correctly. Tested & validated end-to-end IP connectivity over simulated satellite network	Passed
15/03/23	Final Demo	Performed end-to-end demonstration between BI endpoints over simulated satellite network and gathered relevant captures	Passed

Table 9. iDR Lab Testbed Usage

A detailed description of the test case execution and verification is provided in Annex II – Validation and Results.



3.3.2 KPIS

In this section, KPIs defined for Transport UC are listed and the results are briefly summarized.

KPI	Test Case Reference	Target	Actual
Autonomous Operability	UC3_TC_01	12-year operability without maintenance. 5+ data points per day.	Achieved Low power MCU OK Low energy COM OK
Critical Event Monitoring	UC3_TC_02 UC3_TC_03 UC3_TC_04 UC3_TC_05 UC3_TC_06 UC3_TC_07 UC3_TC_08 UC3_TC_09	Wake up on event 30 fault intensity classes FS 1 fault intensity classes FS Defect validation No additional load sensor No additional speed sensor	Achieved Flat spot 5+ OK Bearing defects 3 OK Statistic defect val. OK No load sensor OK No speed sensor OK
Cost Effectiveness	UC3_TC_01 UC3_TC_09 UC3_TC_15 UC3_TC_16	Less than €25 per Sensor	Achievable 2 sensors per axle OK 1 sensor per axle 90% OK
Functional Safety	UC3_TC_15	SIL Level 2 PFH/PFD Metrics	Argumentation possible
Fire & Explosion Safety	UC3_TC_16	ATEX Compliance	Achievable ATEX battery OK Low energy reserve OK
Security Attack Robustness	UC3_TC_10 UC3_TC_11 UC3_TC_12 UC3_TC_13 UC3_TC_14 UC3_TC_20	Gateway–Cloud Security + Sensor–Gateway Security	Achieved: End-to-end secure communication + endpoint identity and integrity verification
Connectivity Coverage	UC3_TC_03 UC3_TC_18 UC3_TC_21 UC3_TC_22 UC3_TC_23	Concept BLE Mesh Concept Lora Mesh Concept NB-IoT + Satellite Backhaul	Multimodal connectivity

Table 10. Transport UC KPIs

The original edge sensor KPIs were fully achieved. The additional targets defined after the start of the project were not achieved. For economic reasons, the lifetime expectancy was increased from 12 to 30 years. This makes battery operation unachievable. Larger batteries are also not ATEX compliant, so the energy concept has to shift from battery operated to harvester operated. Fortunately, there is enough vibrational energy to in the range between 100-1000Hz to make vibrational energy harvesting possible. Tribo-electric energy harvesting is cheap and capable, the question is if such an extensive lifetime can be achieved with such technology. The research has been started, outside the scope of the iNGENIOUS project.

The KPI on *Security Attack Robustness* is non-quantitative, as it relates to improved security. This KPI has been met, as additional security guarantees (end-to-end secure communication with verifiable endpoint identity and integrity) are enabled by integration of remote attestation with TLS.



3.3.3 IMPACT ASSESSMENT

The iNGENIOUS Transport UC focuses on eight improvement areas when compared with conventional IoT sensors:

- Energy Harvesting – 25+ years of operation.
- Always On – Incident Location Determination.
- Micro Edge Sensing – Low Power Computing.
- Mesh – Healthiest Node Communication.
- Lora – Pay per Uses Data Transmission.
- Cloud – Feature based Fault Verification.
- Secure Authorized – TLS + Remote Attestation.
- Novelty Detection – Sensor Swarm spiced with Novelty Loggers.
- Satellite Connectivity – coverage extension and satellite IoT payload optimization.

Part of the work was purely conceptual, while another part achieved full function maturity. The combination of Edge-Computing, Edge-Node Attestation and Multi-Modal communication enables further projects.

The ideal edge sensor requires no physical wiring and no battery maintenance. It is always-on when needed; and can be used in mobile and stationary applications. This requires smart low-power designs.

Sensors collect inherently sensitive information. It can be personal and/or it can be commercial information. To ensure unauthorized access TLS and data encryption are not enough. Remote attestation stops unauthorized access and prevents brute force attacks. It is an ideal security extension for financial and medical IoT applications.

Information gaps due to communication outage can be costly, Multi-modal connectivity reduces communication outage while optimizing bandwidth usage and minimizing communication cost. In this particular use-case, the optimization of communication energy was the key driving factor. Edge-Sensors and Edge-Gateways running on batteries or harvesters are always energy starved. Using the most energy efficient communication paths for all sensors as a swarm is the best way to stretch limited energy reserves. Adding satellite connectivity extends the possible use-cases to far out of reach regions, warzones, and shipping applications.

The work invested in building both physical and theoretic fault models for train carriage axles has a wonderful side-effect for future-based research. Vibrational tribo-electric energy harvesting is an essential area for future development. The physical fault-generator can be used to quantify available energy levels in defect-free and defect-invested situations. Neuromorphic computing is very good at pattern recognition. In a very noise contaminated environment, pattern recognition is not very effective. However, the simulated fault model can be used to define signature points to be recognized in signal patterns. This will allow us to develop extremely efficient neuromorphic classifiers in the future.

As the applied technologies span so many use-cases, further cooperation between the cooperation partners involved is almost ensured. Currently there



is discussion on a joint medical edge application using remote attestation to ensure security and confidentiality.

NCG will exploit the research performed in the Transport UC by partnering with Rail-Tech Software as a Service (SaaS) providers and by providing accelerated AI development support. The SaaS provider manages domain specific knowhow and data commercialization. The Edge IoT Sensor developed for this UC will be developed further and then licenced to the Rail-Tech SaaS Partners.

Remote attestation is not a new concept, but to this date, it is hard to deploy in practice in distributed systems like IoT networks. By integrating remote attestation with industry-standard TLS, the Transport UC removes a barrier that system designers face when developing new IoT solutions with strong security requirements. Furthermore, the isolation-by-default approach of the M3 platform with a root-of-trust integrated will make it easier to build embedded computers for IoT devices. Overall, this work has the potential to build a more secure and therefore more trustworthy Internet of Things. But the principles and building blocks can be applied to any distributed system beyond IoT.

3.4 Lessons Learned and Potential Improvements

Rail-Health condition monitoring involves many stakeholders. Each of them wants to reap benefits from this innovative technology, but in the end only one stakeholder will pay for the IoT investment. In this particular use case, it is the rail-carriage leasing operator, which wants to minimize maintenance cycles. If possible, regular 6-year maintenance intervals shall be shifted to 12 years or longer. Accident prevention, reduction of rail track damage from poorly maintained assets, and better planning cycles, are not considered in the overall business case. And twelve years amortization duration is fairly long. Without legislation requirements on accident prevention, or penalties on infrastructure damage due to poor managed assets, it takes really gutsy Chief Finance Officers to accept such a long Return on Investment period. But without proven in use track records and field data, it is difficult for technology innovations to influence regulatory policy. Automotive airbags were first equipped in the early 1970s, while mandated legislation did not follow until 1999. So, the challenge is to find an early adopter and/or expand the use case benefits to get more traction.

4 Demo – Situational Understanding in Smart Logistics Scenario

4.1 Objective and Description

The main objective of the use case and its demonstration was to enhance the situational understanding of events in maritime ports and terminals by combining multiple data sources and, subsequently, developing different Artificial Intelligence models able to predict and optimize the time spent by trucks inside the port facilities, i.e., truck turnaround times (TTT).

To achieve this main objective, the present use case demonstration proved the following aspects:

- Ingestion and integration of online data sources (PCS data, Gate In/Out events, etc.) in two different scenarios: the Port of Valencia and the Port of Livorno.
- Vessel schedules, cargo flow and truck traffic level calculation and predictions for both the Port of Valencia and Livorno scenarios by exploiting different ML-based prediction models.
- TTT calculation and prediction at both ports Valencia and Livorno by exploiting different ML-based prediction models.
- Development of online API able to provide vessel schedule, cargo flow, truck traffic and TTT predictions.
- Development of visualization interface showing historical and real-time predictions for the vessel schedules, cargo flow, truck traffic and TTT parameters.
- Visualization of the past and ongoing accuracy of predictions for each parameter.
- Testing real-time positioning of trucks inside the Port of Valencia and Port of Livorno by using IoT tracking devices.
- Development of a Geographic Anomaly Detection module to validate the positioning data obtained from IoT tracking devices.
- Development of a graphical interface for visualizing the IoT tracking measurements with maps in the Port of Valencia and Livorno.
- TTT validation by comparing the data obtained through ML-based prediction models and the information retrieved from IoT tracking tests.



The main part of the Port Entrance UC was performed through the statistical validation of Artificial Intelligence models developed using large historical datasets collected from the ports, and by demonstrating the operation of models with continuous data integrations in an online cloud environment. Dedicated dashboards were designed to show charts for the predicted parameters. These charts visualize the predictions for main target metrics including vessel arrival times, container traffic rates, and truck turnaround times. The predictions were performed by exploiting a combination of ML-based prediction models. Statistical analysis of the accuracy of these models was implemented using data science best practices, e.g., by applying nested cross-validation to estimate model performance in an unbiased manner.

Complementing this demonstration, real-time positioning data was visualized and represented in a graphic interface based on HERE maps API [4]. For case of the port of Valencia, this data was collected by the IoT tracking devices installed in trucks performing regular import/export operations at the port. For the port of Livorno, this positioning data is provided by the IoT tracking devices installed in the cars owned by the port's staff, which were used to simulate a truck.

The different steps of the demonstration were executed in different time periods at the ports of Valencia and Livorno. AWA and FV carried out the deployment of ML-based models for predicting TTT in a software-based environment during the last week of January and the first weeks of February. At the port of Valencia, real IoT tracking tests were performed between 15th and 22nd of January. At the port of Livorno, these tests were performed between 6th and 19th of February.

Through this demonstration, the project managed to prove that the situational understanding of events in maritime ports and terminals can be enhanced by combining multiple data sources and exploiting Machine Learning and IoT technologies.

ML-based models were developed to optimize and predict TTT by exploiting data from different data sources such as Port Community Systems (PCS), summary declarations and Gate Access Systems. Additionally, IoT technology was exploited to validate the results obtained from ML-models through the data obtained in real-time positioning tests performed over trucks in the port of Valencia and Livorno.

By enhancing situational understanding and optimizing truck turnaround times, maritime ports and terminals could reduce truck traffic congestion in peak traffic times. An efficient optimization of the flow of trucks contributes to reduce congestions and queues, thus leading to higher port and terminal performance. At the same time, the reduction of queues and congestion leads to lower CO2 emissions.

4.2 Setup and Execution

The process of setup and execution of this demonstration was split in two main parts: one related to the integration of data sources and the development and deployment of ML-based algorithms, and another related to the execution of IoT tracking tests.

4.2.1 PART I

In this activity, the setup and execution were designed according to the traditional data science approach where data analysis, data ingestion, data preparation, model development, model deployment and data visualization phases are typically executed.

Since the demonstration applies mainly machine learning models designed to be trained using supervised learning and statistical models based on distribution fitting, this activity has been conducted by following a two-step strategy, first exploiting historical or offline data sources in model development and training, and then deploying the models for inference using real-time or online data (see Figure 18).

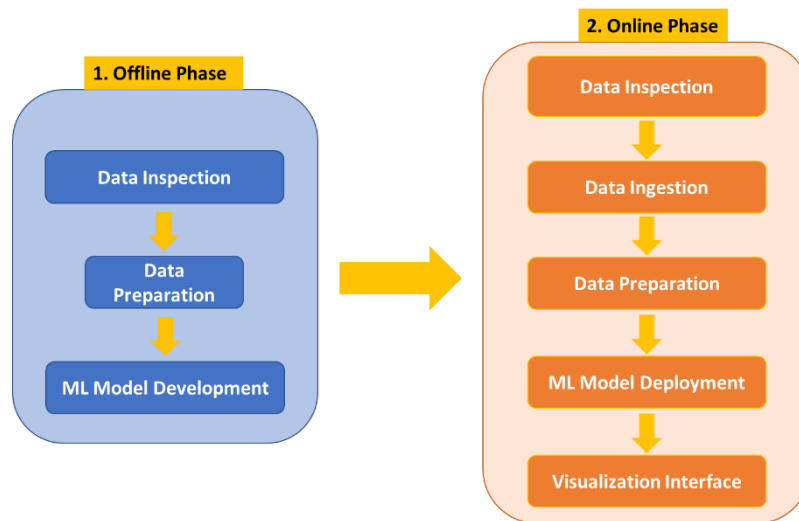


Figure 18: Data Integration and ML-Based Algorithm Approach

Within both stages, the phase related to ML model development and deployment was performed through a combined strategy where two different ML pipelines were designed to approach the problem from two possible angles:

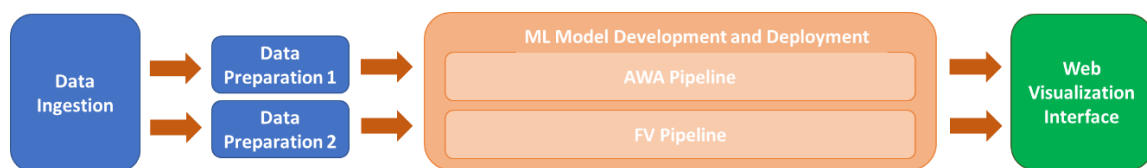


Figure 19: Data Integration and ML Pipeline Division Approach

On the one hand, AWA designed a ML pipeline based on the combination of different AI models for vessel Estimated Times of Arrival (ETA), cargo exchange volumes per vessel port call and container dwell times, which were used to generate estimates of future port gate exit event rates, and in turn were used as inputs in predicting the variation of truck turnaround times at the port. This approach assumes a worst-case scenario where only port call data is available for the service in the online phase. This can be assumed to be a common scenario in many commercial deployment cases, as e.g. cargo exchange and vehicle gate event data is not readily available for all organizations in most ports. Furthermore, the combination of models and data features used in this pipeline is designed to provide as accurate predictions as possible for long-term

operations, e.g. several weeks ahead, primarily on a daily or weekly level of temporal granularity.

On the other hand, FV designed an ML pipeline relying on the use of autoregressive models (i.e., SARIMA) and typical regression models for calculating truck turnaround times. By exploiting use of autoregressive models, this approach relies on past gate in events to predict future gate in events in the short term. These future gate events are used as input for the TTT regressive model that predicts the final truck turnaround times. This approach relies on a best-case scenario approach where port call and gate in data is available. Consequently, and because of the nature of autoregressive models, this pipeline is able to provide accurate results in the short-term.

According to the proposed approach, the execution procedure of the different phases for both offline and online phases is described as follows:

Offline Stage

a) **Data Inspection:** Multiple offline data sources were first leveraged for developing and training ML-based algorithms. In particular, the set of data sources used in this first stage were:

- **Vessel Port calls:** Historical dataset extracted from ValenciaportPCS API that contains information related to the arrival and departure of vessels, i.e., port calls, at the port of Valencia for 2019 and 2020 time periods. The dataset was provided in CSV format and is composed of 11818 registers and 15 columns, including information related to the vessel name, Estimated Time of Arrival (ETA), Estimated Time of Departure (ETD), Actual Time of Arrival (ATA), Actual Time of Departure (ATD), terminal of operation, etc.
- **Vessel Master:** Historical dataset extracted from VESSL system (owned by FV), which includes detailed information of the characteristics of a large set of vessels performing container shipping activities. The dataset is available in CSV format and is composed of 3280 rows and 9 columns, where each row refers to a vessel and columns include information like the IMO and vessel dimensions (length, breadth, draught, Gross Tonnage, TEU, etc).
- **Summary Declarations (COARRI):** Historical dataset obtained from ValenciaportPCS that contains information related to the containers loaded and discharged from vessels arriving and departing from the port of Valencia in 2019 and 2020 period. The dataset was provided in CSV format and is composed of 145703 rows and 35 columns where each row refers to a container discharge event and columns provides information related to the vessel name, port call identifier, container plate, operation type, container full or empty, timestamp for the loading or discharge from the vessel, etc.
- **Gate Access Data (Gate In):** Historical dataset obtained from Gate Access Systems that contains information related to vehicle ingress to the port of Valencia between 2019 and 2020. The dataset was provided in CSV format and is composed of 3.946.842 rows and 15 columns where each row refers to a vehicle entry to the port. Columns provide information related to the gate in event such as the gate id, truck plate number, vehicle country, gate in timestamp, etc.

- **Gate Access Data (Gate In – Gate Out events):** Historical dataset obtained from Gate Access Systems that contains information related to the ingress and departure of trucks to the port of Valencia in 2019 and 2020. The dataset was provided in CSV format and is composed of 178378 rows and 5 columns, where each row refers to the combination of gate in and gate out event for one vehicle, and columns provide information related to the truck plate, container plate and the timestamp for the ingress and the exit of the truck.
 - **Global Automatic Identification System (AIS) Data:** Global maritime regulations require commercial vessels to transmit their locations and other vessel information through the global VHF-based AIS system. An extensive set of global AIS data was collected during the project to enable development of prediction models for vessel traffic schedules, and implementation of online predictions based on streaming data. This data was ingested at a rate with order of magnitude 5-10 million messages per hour, resulting in a raw dataset of global AIS data starting from 2021 with total magnitude of order 100-200 billion AIS messages. This data was filtered and processed to produce labelled vessel voyages to target ports, which were used in machine learning model development.
- b) **AWA Pipeline:** As previously described, AWA's pipeline focused on the development of predictive models for vessel ETA, cargo exchange volumes per vessel port call, and the distribution of dwell times of containers in the port before exiting by truck. These models were used to generate estimates of future port gate exit event rates, which in turn were used as inputs in predicting the variation of truck turnaround times at the port. This modelling was limited to the subset of data where timing information was available for all steps of the cargo flow in the port, including vessel arrival, container discharge, and exit through the gate by truck.

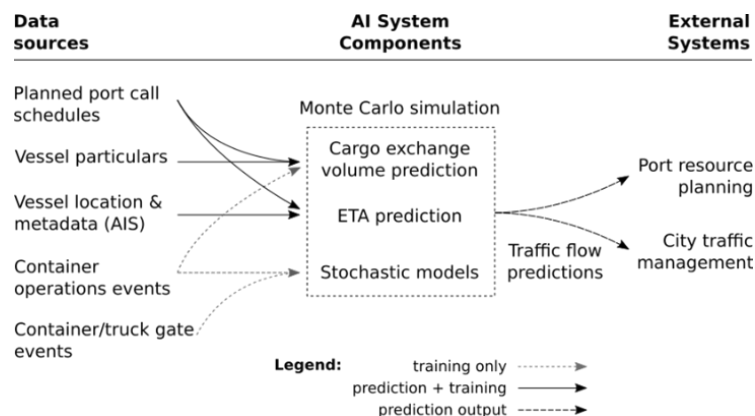


Figure 20: Data sources and model components developed for the demonstration.

b.1) Offline Data Preparation:

On a high level, the datasets used in the development can be divided into vessel traffic (AIS) data, port call data, cargo operations data, and gate event data. The main objective in offline data preparation in the AWA pipeline was to combine the available datasets in a way that enabled tracking the times of related events surrounding the port operations. For example, a vessel is tracked over its voyage to the port, the voyage can be associated with port call information (such as start and end times, locations, cargo information,



etc.), cargo operations during the port call can be associated with timestamps for individual cargo item movements (e.g. a container discharged from the vessel), and hinterland events (such as a truck exiting the port) can be associated with individual cargo items. On a higher level, creating these connections requires each dataset to include identification information also present in other associated data. Once data was prepared for analysing the individual steps in container flow through the port, models were developed to describe and predict related time durations. Developed prediction model components include vessel arrival time prediction, cargo exchange volume prediction, container dwell time prediction, and truck turnaround time prediction.

b.2) Model Development:

The fundamental idea in the developed prediction pipeline is that individual vessel arrival times can be estimated using existing information sources and machine learning (ML) models, and while it is difficult to predict the dwell time of an individual container in the port, the total flow rates of containers related to the vessel port calls can be approximated using stochastic models.

The model pipeline simulates the number of containers transported by trucks out of the port during a selected time range. This is implemented as a Monte Carlo (MC) simulation, which models the rate of containers exiting the port by adding a predicted number of randomly sampled container dwell times to predicted vessel arrival times.

For training the prediction models, the vessel arrival times are obtained from the actual times of arrival in the Valencia port call dataset, and the numbers of outbound containers by truck are estimated from the container operations and gate events datasets. The distributions for container dwell times are estimated empirically by distribution fitting.

In addition to the discharge and port dwell time distributions, which are here assumed to be stationary, the main dynamic inputs needed for the model described above are the vessel arrival times and numbers of containers to be discharged per vessel to be carried out of the port by trucks. Of these, the container discharge numbers can be expected to be more difficult to obtain from external sources, as terminal operators do typically not share such cargo exchange information publicly. To enable predictions without receiving this information, dedicated regression models are applied to predict the cargo discharge volumes per port call. These were implemented as extreme gradient boosting (XGBoost) models, with model selection and hyperparameter tuning performed using 3-fold cross-validation and testing of the model selection and acquisition of test data for performance evaluation obtained using additional 10-fold nested cross-validation.

To apply the above-described traffic prediction models over as long-time frames and as accurately as possible, future vessel arrival schedules were predicted using global Automatic Identification System (AIS) data. A separate prediction model pipeline was developed to predict for a given vessel its current destination, the geographical voyage trajectory to this destination, and the duration of the voyage along this trajectory. These

component models applied various machine learning techniques trained using extensive historical vessel traffic data.

The truck traffic rate variation obtained as output of the predictive simulation system described in 149 of Annex III was used as an input feature in predicting truck turnaround times at the port. This was determined to be the most important input feature in feature importance analysis of regression models for truck turnaround time. These regression models were implemented using the XGBoost framework and the cross-validation procedures described above for cargo volume prediction.

- c) **FV Pipeline:** There is an alternative method to predict the TTT of the port, especially for short-term time periods (i.e. 24-48 hours). This method combines two artificial intelligence models. The first one consists of an autoregressive method (i.e., SARIMA) to predict the number of trucks that will enter the port (i.e., Gate-IN events). The second one uses a Machine Learning algorithm (i.e., Random Forest Regressor) to predict the TTT using the output of the previous model and other variables such as vessel traffic, hour, and the day of the week (see Figure 21).

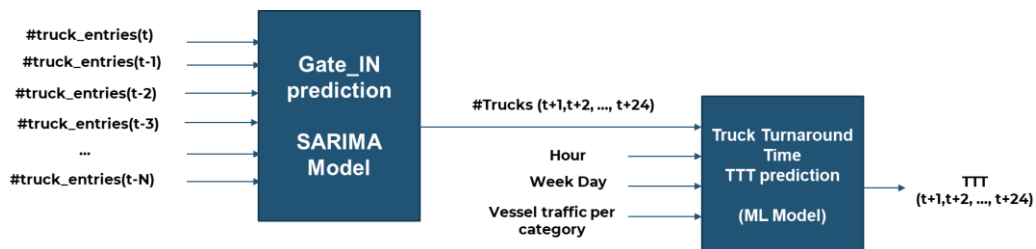


Figure 21: Autoregressive + ML method to predict TTT

c.1) **Offline Data Preparation:**

After identifying the data sources required to approach the TTT prediction, data needed to be prepared and merged to create the final datasets to be injected as input to the ML-models. In this case, since TTT can be directly influenced by maritime and terrestrial events, the data sets exploited to feed TTT models were port calls dataset and gate access data (including gate in and gate out events). The dataset used to train the TTT prediction model contains the following variables (see Figure 22):

	Fecha_Entrada	Hour	WeekDay	Num_trucks	TTT	catA	catB	catC	catD	catE	catF	total
0	2020-01-02 06:00:00+01:00	6	3	138	101.212500	1	0	0	1	0	1	3
1	2020-01-02 07:00:00+01:00	7	3	533	118.761696	1	0	0	1	0	1	3
2	2020-01-02 08:00:00+01:00	8	3	654	142.066942	1	1	0	1	0	1	4
3	2020-01-02 09:00:00+01:00	9	3	375	135.889007	1	1	0	1	0	1	4

Figure 22: Final TTT data frame

On one side, gate access data was first processed with the following data preparation by: i) importing the datasets containing gate-in and gate-out events to Jupyter Notebook; ii) merging these datasets using the truck-plate parameter for the matching; iii) dropping the truck plate parameter; iv) calculating the TTT for each gate-in and gate-out pair by subtracting the timestamp of the former to the later; and, v) resampling the resulting dataset to calculate the average TTT on an hourly basis.



Moreover, the vessel port call dataset the vessel master dataset was processed together by using the International Maritime Organization (IMO) parameter, classifying the vessels into 6 groups depending on the vessel's size, and resampling the resulting dataset on an hourly basis.

In parallel to the above data preparation tasks, to train the gate-in prediction model, the gate-in dataset has been resampled to calculate the number of trucks per hour from the first to the last timestamps appearing in the datasets.

The final step consisted of merging the three prepared sub-datasets to obtain the one shown in Figure 22 For more detailed information about the data preparation tasks, the reader is referred to the Annex III.

c.2) **Model Development:**

Gate-IN forecast model

After representing the Gate-in dataset in a chart – by executing the `seasonal_decompose()` function from the python's `statsmodels` library – it can be seen that the data has a strong seasonal component (see Figure 23). For this chart plot, the weekends have extracted.

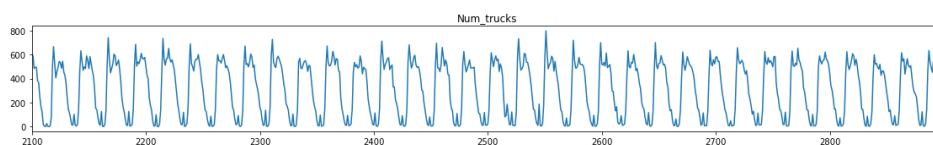


Figure 23: Gate In time series analysis

In time series forecasting, autoregressive models (a.k.a ARMA models) are used to give good results. For the gate-in prediction, the SARIMA [5] (Seasonal Autoregressive Integrated Moving Averages) model was used. The model development phase consists of finding out the $(p,d,q) \times (P,D,Q)m$ parameters [6] that allow us to generate the model and train it with the dataset prepared. To do so, an accurate analysis of the time series was performed using various functions of the `statsmodels`. Using the results of this analysis, the `auto_arima()` method of the `pmdarima` library [7] was run to get the best parameters of SARIMA model (i.e. p,q,P and Q). The best model parameters' combination is $(2,0,2)(1,0,0)24$.

The model was generated using the `SARIMAX()` class of `statsmodels` library and fitted with the dataset using its `fit()` function.

```
model_auto = SARIMAX(port_entries_week_sub, order=(2,0,2), seasonal_order=(1,0,0,24))
results_auto = model_auto.fit()
```

Figure 24: Gate In SARIMA model instantiation

TTT prediction model

With the TTT_train dataset obtained after the data preparation phase, values were first normalized to $[-1,1]$. To find out the best configuration parameters (a.k.a hyperparameters) that control the learning process of our models, the whole set of observations was split into a train set and a test set. The former was used to initially fit the model while the latter was used to evaluate the predictions done by the fitted model with the true values from this partition. A good rule of thumb in ML is to split the dataset in



80/20. In this, a random split of our observations (see line 8 in Function 2) was made using the Panda's *sample* function of our dataset class.

The next step consisted of finding the best hyperparameters for the ML algorithm used to generate the model. In this case, a Random Forest Regressor [8] using the *RandomForestRegressor* class of ScikitLearn library was leveraged. The most common parameters to fit were the number of decision trees (i.e. the n-value) and the depth of the trees (d-value). A *for* loop (see Figure 25) was developed to consecutively train a new RF model changing the values for these hyperparameters, calculate the Mean Absolute Error for each iteration, and save the result in a separated dataset. The combination of hyperparameters that generated the model with the lowest MAE is 10 number of decision trees and 7 as the maximum depth of the trees.

```
Metrics_df=pd.DataFrame(columns=["MAE","n_depth"])
for depth in range (2,30):
    RF_Regressor = RandomForestRegressor(n_estimators=10, max_depth=depth, random_state=0)
    RF_Regressor.fit(X_dataset_train, Y_dataset_train)
    predictions = RF_Regressor.predict(X_dataset_test)
    mae_n=np.array(mean_absolute_error(Y_dataset_test, predictions),dtype='float32')
    Metrics_df = Metrics_df.append(pd.Series([float(mae_n), str(depth)], index=Metrics_df.columns ), ignore_index=True)
```

Figure 25: Random Forest Regressor hyperparameter tuning for the TTT model

Finally, the model was generated by instantiating the *RandomForestRegressor* class with the selected hyperparameters and fitting the model with the training dataset.

```
RF_Regressor = RandomForestRegressor( max_depth=7, random_state=0)
RF_Regressor.fit(X_dataset_train, Y_dataset_train)
```

Figure 26: TTT Random Forest model instantiation and fitting

Online Stage

a) Data Inspection:

- **Port calls:** Data extracted in real-time from ValenciaportPCS API that contained information related to the arrival and departure of vessels, i.e., port calls, at the port of Valencia. The data was provided in JSON format and includes information related to the vessel name, Estimated Time of Arrival (ETA), Estimated Time of Departure (ETD), Actual Time of Arrival (ATA), Actual Time of Departure (ATD), terminal of operation, etc.
- **Gate Access Data (Gate In and Gate Out events):** Data extracted in real-time from PI System OSIsoft (M2M platform) that contains information related to the ingress and departure of trucks to the port of Valencia. The data was provided in JSON format and includes information related to the truck plate, container plate and the timestamp for the ingress and the exit.

b) Data Ingestion:

To deploy the model, its input data that was used to compute the prediction needs to be available and accessible online through an API. The Gate-in, port call and gate-in/out data were made accessible through the implementation described in the architecture of Figure 27.

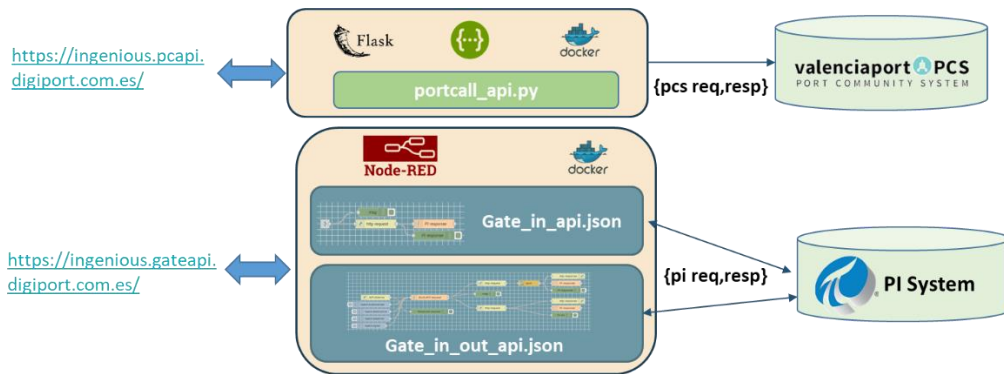


Figure 27: Port Call and Gate Access online data ingestion setup

In the setup of Figure 27, there are two main components that make the models' input data available online:

- PortCall API:** Component implemented in python that provides a HTTP/REST API to access future port call schedule information through a GET request with given date parameters at the request. This API was consumed by the component running the prediction models. This component was implemented using the Flask [9] tool which provides a Swagger front-end for the API test and documentation. It leverages the *flask_restx* and *flask_httpauth* libraries. The information returned in this API was retrieved directly from the Port Community System of the Port of Valencia.
- Gate API:** Component that provides an HTTP/REST API to access the gate in and out information. It also runs a message processing mechanism that simplifies the way it is accessed by the PI System. This module was implemented using the Node-Red [10] tool in which two separated files have been generated. The first, called *Gate_in_api.json*, includes the mechanisms to calculate the number of vehicles that entered the port within the time interval provided with the GET request to this API. The second one, called *Gate_in_out_api.json*, provides the code to calculate the average TTT for the given time interval at the received request. This API call was used for the online validation of the TTT predictions.

The components above are executed on two different Docker components in a Linux virtual machine running in the datacenter of Fundaci3n Valenciaport.

- Online Data Preparation:** The data preparation followed the same approach explained in the offline stage.
- AWA Model Deployment:** The developed models were deployed as microservices in a Kubernetes cluster managed by Awake.AI in the Amazon Web Services (AWS) cloud platform, as illustrated in Figure 28. The system consists of multiple microservices for data ingestion and processing, continuous monitoring of events, and applying prediction models using latest available input data. The microservices communicate streaming data through a Kafka messaging backend and on-demand requests through REST APIs. Various databases in the cloud environment are used to enable stateful service operation as necessary. To demonstrate the prediction models developed for the use case, the vessel ETA models are integrated to the commercial Awake.AI Smart Port web application enabling interactive

visualization and testing of the models. In addition, all developed models are included in a custom service which implements the entire developed prediction pipeline and provides a custom web interface for testing and demonstration.

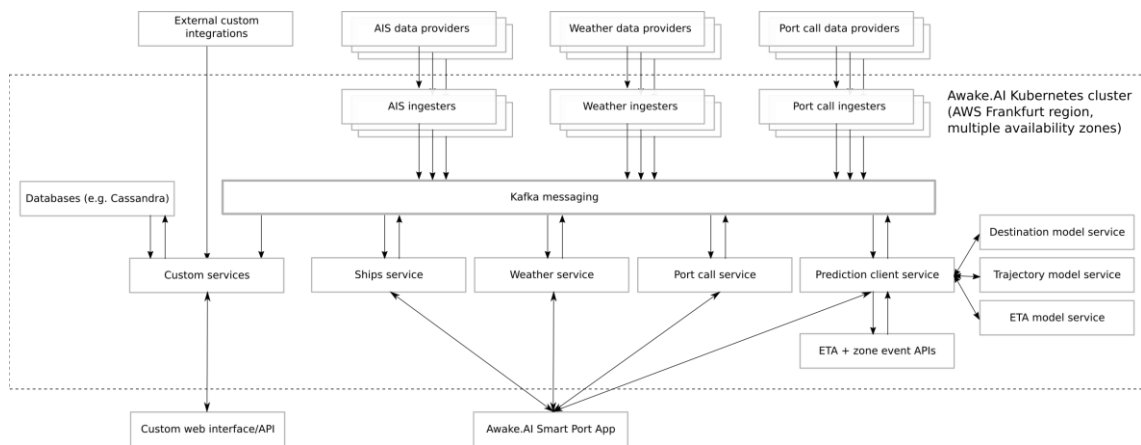


Figure 28: Overview of the cloud service architecture used in the demonstration.

e) **FV Model Deployment:**

As for the autoregressive-based method to predict TTT, the models' execution setup used is shown in the diagram of Figure 29. The whole process to get the TTT prediction was triggered by calling the <http://ingenious.ttt.digiport.com.es> using a Http GET request.

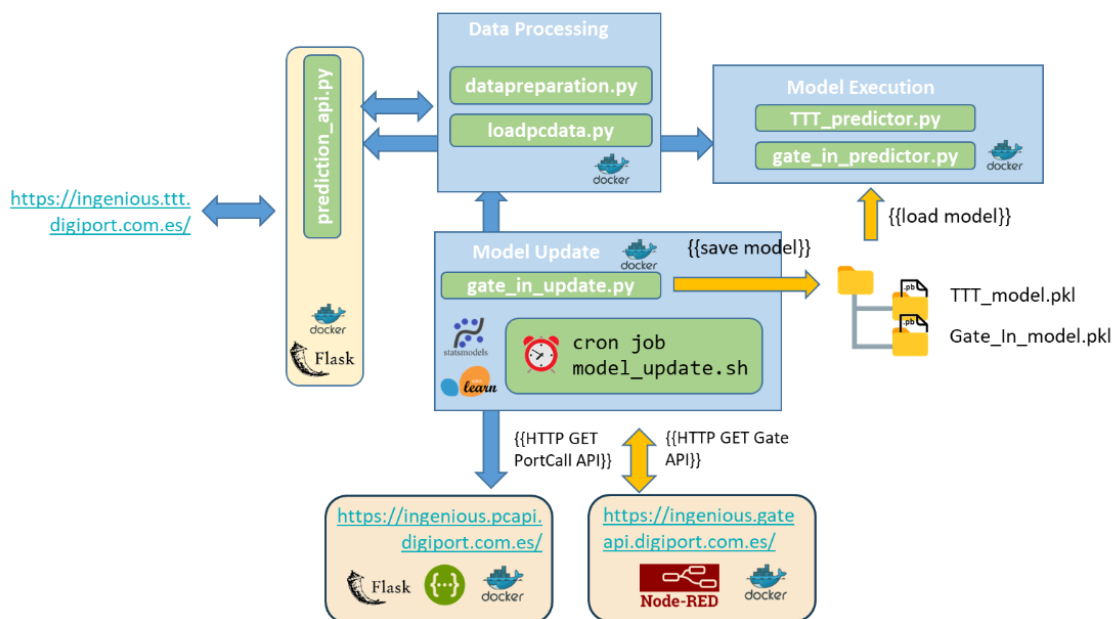


Figure 29: Autoregressive + ML based TTT prediction setup

In Figure 29 we can observe four main modules which were deployed in Docker containers:

- **API** – This module provides an HTTP/REST API to request for a TTT prediction for the next 24 hours. It is implemented in Python with the prediction_api.py file. In this file, the GET requests are received which triggers the coordination of prediction's calculation. It first calls the prepare_data() function of datapreparation.py file to get the input

parameters for the TTT prediction. Once the input parameters are retrieved, the `make_prediction()` function of `TTT_predictor.py` file is executed to compute the prediction. The result is enveloped in a GET JSON response with the 24 forecasts of TTT, one for each consecutive hour from the moment of the API call.

- **Data Processing** – In this module the process to get ready the input parameters for the TTT predictive model is executed. In the `datapreparation.py` file the `loadpcdata.py` is invoked to get the port call schedule of next 24 hours from the Por Call API. The raw data gotten from this API call is processed to calculate the vessel traffic per vessel category in the next 24 hours. In this file, the `gate_in_predictor.py` file from the Model Execution module is also called to compute the Gate_In prediction that provides the number of trucks that will enter the port in the next 24 hours. Finally, the `datapreparation.py` file creates the `weekday` and the `hour` variables, joins them with the port call and gate in variables' values and returns the data to the API module to trigger the prediction of TTT.
 - **Model Execution** – This component executes the TTT and Gate In predictions. To do so, it provides the `TTT_predictor.py` and `gate_in_predictor.py` python files which implements the `make_prediction()` functions to compute the prediction based on the input parameters given in the call. In the `gate_in_predictor.py` the `Gate_In_model.pkl` file – which holds the model produced in the offline phase – is first loaded and then, the `get_forecast(steps=24)` function of the SARIMA model is executed to get the output array with the prediction values. The TTT prediction is similarly computed inside the `TTT_predictor.py`. In this case, the `TTT_model.pkl` file is loaded to extract the Random Forest model and the `predict()` built-in function of the `RandomForestRegressor` class is executed.
 - **Model Update** – In this module the SARIMA model that predicts the number of trucks is updated with a dedicated linux cron job that executes the `model_update.sh` file daily. The `model_update.sh` file calls the `model_update()` function of `gate_in_update.py` python file which runs and trains a new SARIMA model with an updated array of past Gate In events as input data. The new array of port entries is retrieved with the Gate API available under the URL <https://ingenious.gateapi.digiport.com.es/>. The generated model is then saved (using the `dump()` function of `pickle` Python library) in the Docker volume within the file system of the virtual machine of running the execution setup.
- f) **Visualization Web Interface:** The visualization web interface has been designed to represent the main outputs obtained from both ML pipelines.

Figure 30 and Figure 31 below show the visualization of vessel ETA predictions implemented in the Awake.AI Smart Port web application. These have been developed into a stand-alone commercial service during the iNGENIOUS project and are included as the first step in the multimodal traffic prediction pipeline in the demonstration. In Figure 30, global vessel data is filtered to show vessels currently predicted to be arriving to port of Valencia. Hovering over a vessel shows an info box with basic vessel and voyage data and the predicted arrival time. In Figure 31, a single vessel has

been selected, which provides the predicted route and ETA, along with additional vessel details.

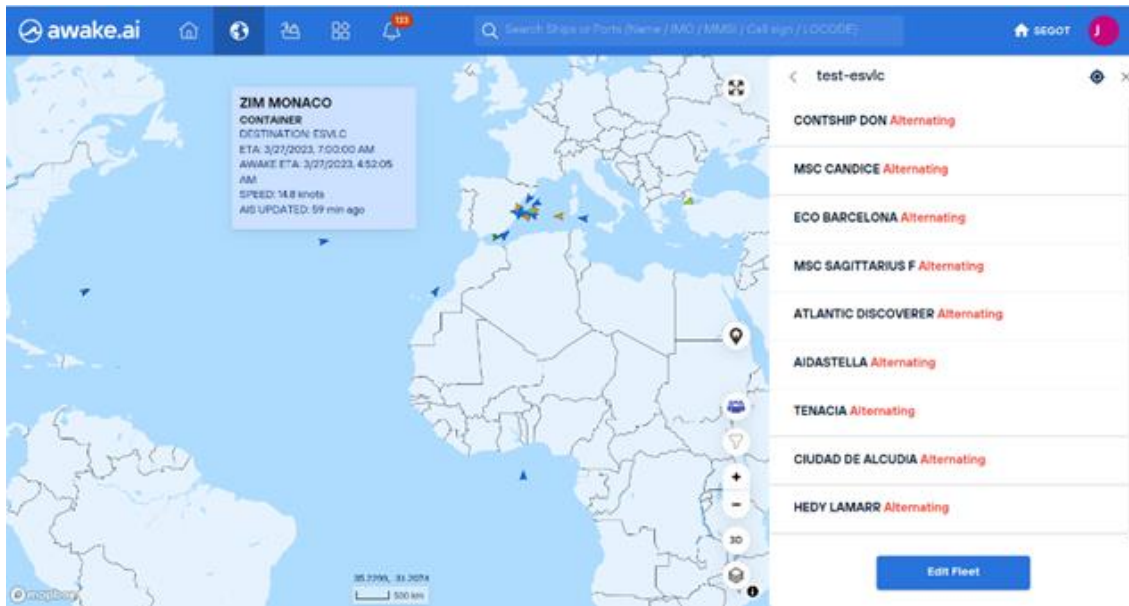


Figure 30: Overview of predicted vessels arriving to port of Valencia in the Awake.AI web application.

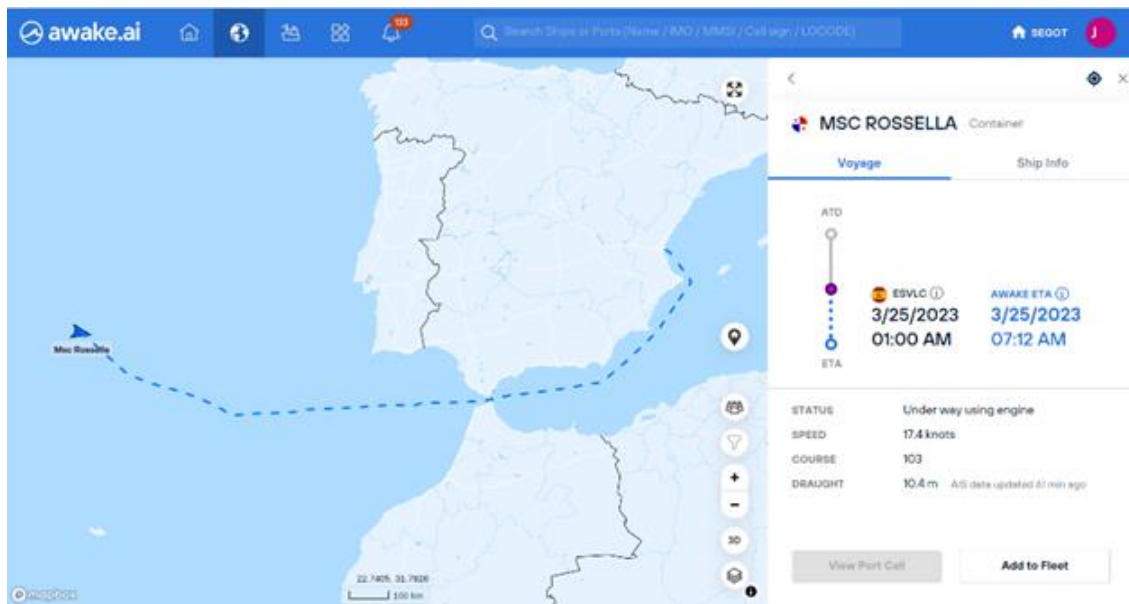


Figure 31: Predicted route and arrival time to port of Valencia for a selected vessel.

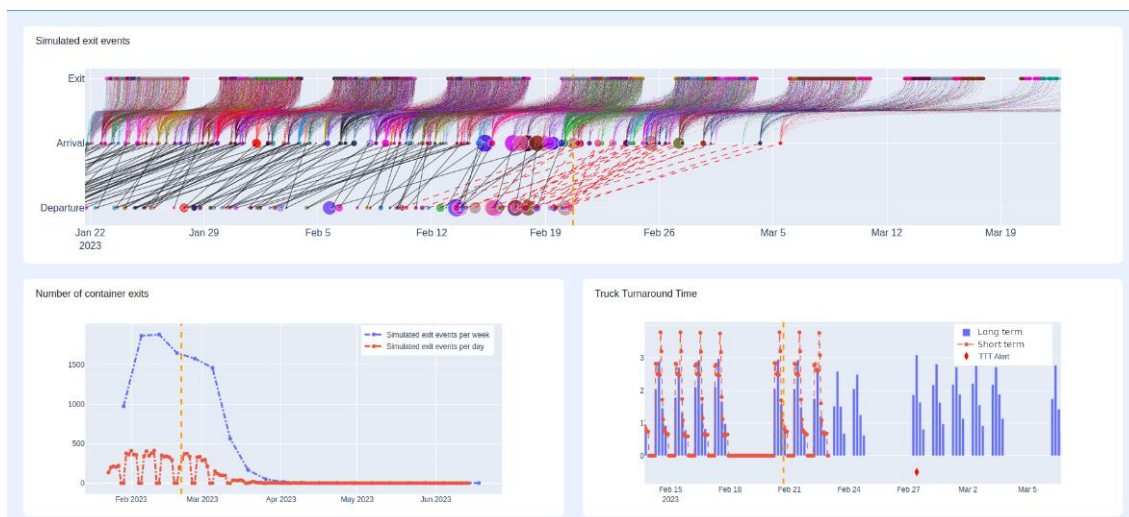


Figure 32: Port Entrance UC demonstration custom web interface

The other parts of the developed prediction pipeline were demonstrated using a custom service developed specifically for this purpose. This integrates necessary port call information from APIs provided by the ports, vessel data from global commercial providers, vessel schedule predictions from other Awake.AI microservices as outlined above and applies the developed ML models and predictive simulations to estimate future traffic rates. Figure 32 shows a screenshot of the custom web interface developed for the demonstration.

Truck Turnaround Time Validation

The predictions obtained after the execution of the demonstration can be observed in the different graphs represented in Awake’s visualization framework, shown in Figure 30, Figure 31 and Figure 32.

Gate-in/out Validation Service

To validate predictions for a specific time frame, FV developed a new service that extracts information related to past real gate in and gate out, enabling the calculation of real truck turnaround times and the validation of predictions a posteriori. The service, which can be accessed through the following API: https://ingenious.ttt.digiport.com.es/ingenious_ttt/test, provides an array of the real truck turnaround time values observed for the time frame when the prediction was obtained.

After running the validation service, real results and predictions are represented in the TTT visualization framework:

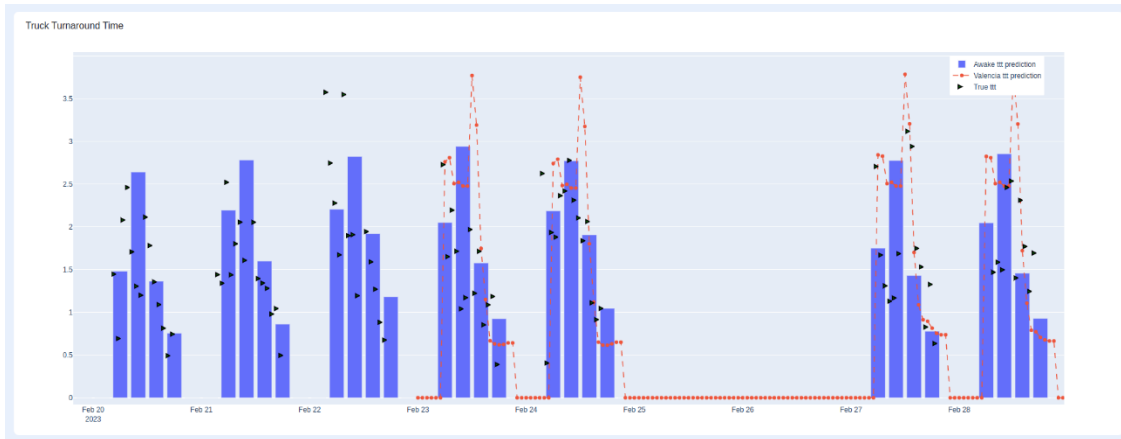


Figure 33: True TTT vs prediction TTT using the gate-in/out validation service.

As explained in the Annex III, on test case descriptions, main performance tests for the developed models are performed using historical datasets to provide sufficient statistical coverage for estimating the results. However, as there may be differences in the statistics of the modeled processes as seen in historical data versus current inputs, smaller subsets of data are collected from the online service running the developed models, and online service performance is compared to the historical data analysis results. Obtained test results with online data for the whole prediction pipeline indicate a 13 % relative median absolute error in daily maximum turnaround time, which corresponds well with the respective results obtained with more comprehensive historical datasets.

IoT Tracking Validation

In addition to the validation procedure performed through the specific validation service developed by FV, IoT tracking test results can also be used as a reference for assessing if the magnitude of the predictions is in line with the magnitude of real events. In particular, UPV developed a dashboard where tracking results gathered with the IoT tracking devices can be visualized helping the easier analysis and understanding of a truck situation at a glance. An example of the data obtained for a one-week testing period at the port of Valencia is shown in Figure 34.

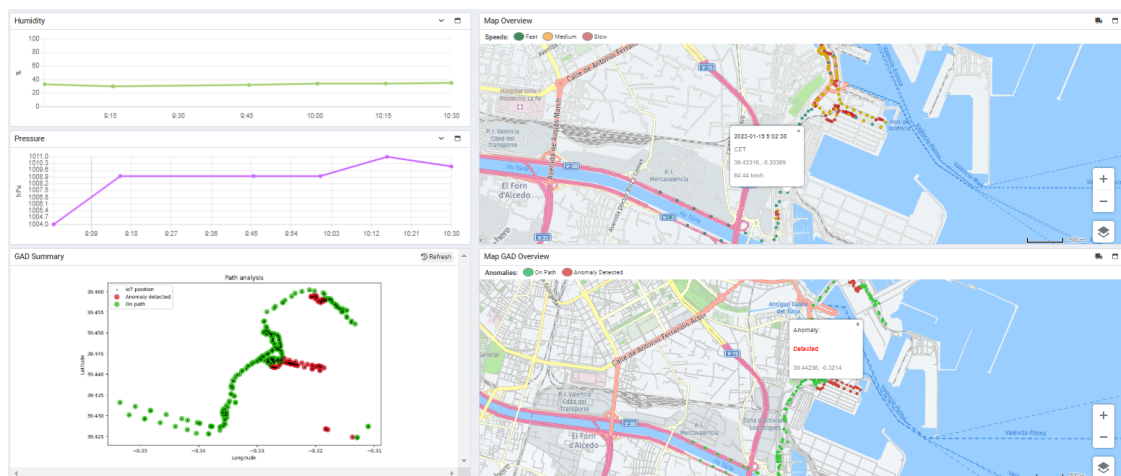


Figure 34: Port Entrance UC IoT Tracking dashboard for one week testing

As it can be observed in the figure above, many tracking points were obtained for trucks entering and exiting the port several times per day. By analyzing the



entry and exit timestamp at the port gate, TTT values can be calculated. In particular, the following values were extracted from this analysis:

Port Entry Date	Port Exit Date	TTT
3/2/2023 8:33:18	3/2/2023 9:26:41	53m:23s
3/2/2023 12:01:13	3/2/2023 13:22:05	01h:20m:52s
3/2/2023 15:57:07	3/2/2023 17:21:09	01h:24m:02s
2/2/2023 9:37:32	2/2/2023 10:09:45	32m:13s
1/2/2023 10:19:20	1/2/2023 15:57:38	05h:38m:18s
30/1/2023 15:07:17	30/1/2023 17:45:21	02h:38m:04s

Table 11. IoT Tracking based TTT measurement tests

The results shown in Table 11 demonstrate that typically TTT times obtained in the time frame of daily working hours, the values have the same order of magnitude as the ones shown in Figure 33 (i.e., from 1 to 3 hours). It is worth to note that the measurements taken by the IoT tracker do not coincide in time with the results shown in the Figure 33 and, thus, it is not possible to directly compare these measurements with predictions made by the algorithms.

4.2.2 Part II

The setup related to the execution of IoT tracking tests required both the setup of devices and a number of services for data management.

The tracker device selected for carrying out the IoT tracking was a MT821 manufactured by Mictrack (its specifications in Annex III). This device is a Mini waterproof GPS tracker that uses the latest CAT M1 & NB-IoT technology to provide low power consumption and optimized data transmission at low cost.

The tracker was configured to work with NB-IoT and specific IP. This device send data of two types: cell data and GPS data. The important data are the GPS data whose format is as follows:

MT;6;867035047588320;R0;10+20230105182049+39.454987+-0.328259+22.14+69+2+3744+113

Services

The set of services developed to obtain data from the tracker as well as its representation on the dashboard are:

1. Uc_5 database – PostgreSQL Database for persist data.
2. *comm_protocol* – Server UDP used to wait data from any device that send this data format type and saves GPS coordinates.
3. dashboard – Flask Application (HTTP Server) in Python.
4. Geographic Anomaly Detection (GAD) – Script in Python whose purpose is to detect possible anomalies in the tracker tracks.

These services are represented in the following diagram:



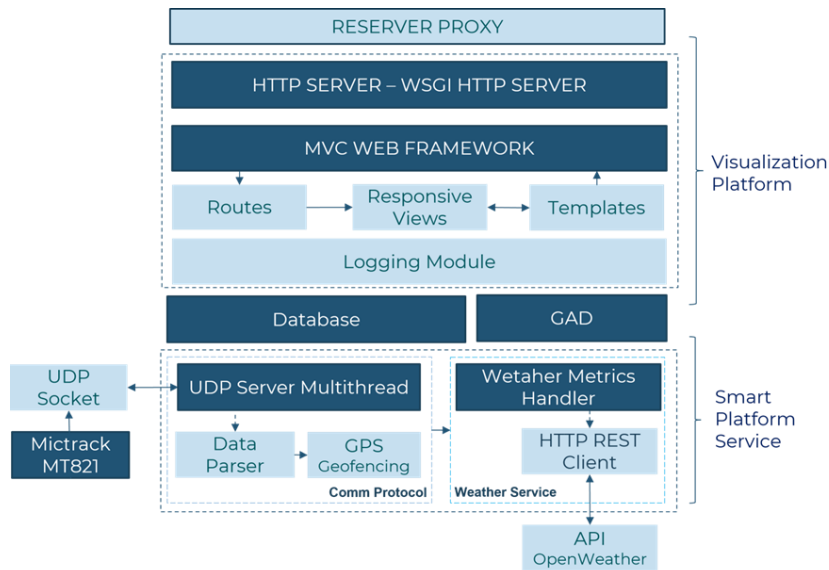


Figure 35: Diagram of Services

Looking at this diagram, we can see which services *comm_protocol*, dashboard and GAD need to have a connection to the database. In this case, *comm_protocol* is responsible for storing the data in the database. On the other hand, the dashboard and GAD services use the database to get data. These two-services fetch data for:

- The GAD service reads data from *temp_gps_tracker_gps_data* to generate anomaly results via images and CSV files.
- The dashboard reads data from different tables to get GPS data, weather data and port entries for a specific day. It also fetches resources generated by GAD services to display them in the view.

For more information and details about the installation process or execution, the reader is referred to the Annex III – Setup and Execution.

4.2.3 Issues on Execution

The following subsections provide description of issues encountered during the demonstration and mitigation actions to solve them.

Description of the issue	Mitigation measures
Lack of comprehensive data for modelling	Simplification of applied models, estimation of the performance effects of missing data.
Lack of data availability for real-time exchange (online)	Developing additional models to estimate missing features (results in performance degradation, which is evaluated separately). Additional performance evaluation using available historical datasets.
Lack of APIs to access critical data	Implementing parts of the validation using offline datasets.
Lack of coverage of commercial LTE/NB-IoT/LTE-M networks	Field tests were performed with IoT tracking devices to verify the coverage at the Port of Valencia and Livorno
Authorization of hauliers for installing IoT tracking devices	Tracking was performed when trucks are inside the port facilities. UPV and FV configured tracking devices

Table 12. Port Entrance UC Issues on execution



4.3 Validation and Results

This section provides a detailed description of the validation results obtained after the execution of the demonstration. Impact is analyzed after explaining the result validation, verification of test cases, KPIs and UC assessment.

4.3.1 TEST CASES VERIFICATION

In this section, the results of each test case, identified in D6.1 [1] for the Port Entrance UC are presented.

Test Case ID	Result
UC5_TC_01: Quality of historical datasets	Passed
UC5_TC_02: Integration of different data sources	Passed
UC5_TC_03: Prediction model accuracy in training	Passed
UC5_TC_04: Performance evaluation in production	Passed
UC5_TC_05: Reception of trucks' geoposition	Passed
UC5_TC_06: Onboard supply chain slice templates and NF descriptors	N/A
UC5_TC_07: Automated deployment of network slice	N/A
UC5_TC_08: Automated termination the slice instance	N/A
UC5_TC_09: Manual scaling of a running slice instance	N/A
UC5_TC_10: MANO interaction with DVL for collecting data	N/A
UC5_TC_11: MANO interaction with DVL to stop collecting	N/A
UC5_TC_12: Automated slice scaling with DVL collected data	N/A
UC5_TC_13: Correctness of datasets time event information	Passed
UC5_TC_14: Correctness of datasets resource ID information	Passed
UC5_TC_15: Vessels' ETA prediction model performance	Passed
UC5_TC_16: Web application for visualizing analytics	Passed
UC5_TC_17: Web application alert on truck traffic levels	Passed
UC5_TC_18: Web application authentication	Passed
UC5_TC_19: communication interfaces DVL, TPCS and M2M Platforms	Passed
UC5_TC_20: communication interface between the AI-based Platform and the DVL	N/A
UC5_TC_21: Dashboard visualization with real time and historical data	Passed

Table 13. Port Entrance UC Test case verification

For what concerns the test case verification, the 62% of test cases were executed successfully. The other 38% of test cases were not executed as the development involved was discarded for non-applicable (N/A). The original goal of test cases 06-09 was to develop an AI/ML algorithm for closed-loop slice optimization based on the combination of application/M2M (from DVL) collected and processed data with network related data (from the 5GC).

However, as explained in D6.2 [2] for the related development activities, at the Port Entrance UC the cross-layer MANO does not control nor manage any 5G network, and the DVL deployed on the field cannot provide insightful data for the network slice optimization purposes. For these reasons, it was agreed with the FV and CNIT to not provide such AI/ML driven network slice optimization capabilities. Nonetheless, some of these test cases (test cases 6-9) can be mapped into the test cases of AGVs UC.

The UC5_TC_10, UC5_TC_11 and UC5_TC_12 have not been executed because no relevant data for ML training have been identified from DVL.

More details on the execution of Port Entrance UC test cases as the description and the results obtained can be found in Annex III.

4.3.2 KPIS

The following table shows the KPIs that were measured and considered relevant during the use case validation.

KPI	Test Case Reference	Target	Actual
Truck Turnaround Times (TTT) Idling Times	UC5_TC_21	10% reduction	7% reduction of TTT predictions/simulations if the truck traffic (input parameter) variance is reduced to a rolling mean with window length of 5 days. 25% reduction of TTT predictions/simulations if the truck traffic used is the mean of the gate events' dataset.
Time Prediction Accuracy	UC5_TC_03	≤ 10 % mean error* <i>*Note: median values are considered instead of mean to handle outliers and skew present in the error statistics.</i>	ETA predictions Relative median absolute error 5 %, averaged over voyage durations. Traffic rate distributions Relative median error of predicted container exit volumes: daily 13 %, weekly 4 %. Turnaround times Long term model: Relative median error of predicted daily maximum turnaround time 10 %.
Data Availability	UC5_TC_04 UC5_TC_16	≥ 99 % uptime	AWA platform availability (hosting use case service and application) 100 % over past 90 days.
Data Source Sufficiency	UC5_TC_02	≥ 3 sources	Valencia: 4 online sources (AIS data, vessel data, port call data, truck gate event data) Livorno: 2 online sources (AIS data, vessel data)
Data Quality	UC5_TC_01 UC5_TC_02	Sufficient by ISO/IEC 25012 metrics	Data accuracy, consistency, credibility, and currentness are found sufficient for the application. Data completeness should be improved for future exploitation by ensuring that truck turnaround data is fully captured.
Security	UC5_TC_18	High data confidentiality, privacy and integrity	AWA applications and services implemented in high availability, fault tolerant cloud environment using multiple availability zones per service. Automated service monitoring, security scanning, and alerting implemented. Authentication used in all services.



IoT Positioning Accuracy	UC5_TC_05	≤ 5 m	MT821 tracking device used for a series of tests, has generated GPS data with an error of less than 3 meters.
Data Protection Impact Assessment (DPIA)	N/A	Data Protection Impact assessed	The impacts have been considered and analyzed for all the demonstrations with the update made to the documents D7.5 and D7.6.
Privacy User Guide Availability	N/A	Approval through the exchange of some mails and verbal conversations	The logistics company as well as the truck driver have been duly informed of the proof we aimed to do in the project, the period in which we needed to keep the tracker inside the truck and which data we were going to collect.
Confidentiality and integrity protection of personal data	N/A	100%	Any personal or identifiable data from the truck driver was collected in the IoT Tracking demo (i.e., Part II)
Logs of privacy events	N/A	100%	Truck plates are not considered as confidential nor personal data as it is not linked with any person at all. Geo-positioning of the IoT trucking device has been turned off once the truck abandons the port area.

Table 14. Port Entrance UC KPIs Results

For many prediction problems, the data source availability in the online stage is a requirement with a major impact on the accuracy of the predictions. As can be seen in the Table 14, the errors obtained in the predictions meet the established targets (average TTT error equal to 10%) with the data available. Moreover, it has been proven that the TTT reduction is achievable if the variance of the truck arrivals at the port is minimized, a quite probable scenario as it is a general goal of ports, nowadays, through the implementation of national single windows for dispatching containers in a scheduled manner.

All in all, the sufficient availability and quality of the data has allowed to obtain acceptable levels of efficiency of the models. All this enabled the service deployment in production. The prediction service is up and running in a cloud environment. As for the IoT Tracking part of the demonstration, the expected level of accuracy in the geo-positioning of Trucks was also achieved.

4.3.3 IMPACT ASSESSMENT

In large city ports, upon the arrival of large ships there is the risk of congestion building up within the city. While traffic peaks could be reduced, for example, by expanding the opening hours of the port, it would be more efficient to target the use of resources such as extended operation times only to those days or weeks where traffic peaks will occur. As stated, e.g. in the Port Authority of Valencia’s environmental and energy policy, “Modern port management and market competition have led port companies to concentrate and increase the volume of their activities, and accordingly they use ever larger amounts of resources, which makes the inclusion of eco-efficient management criteria increasingly more important.” The prediction capabilities developed in the use case are targeted to enable planning and management of operations according to such criteria. One of the key targets in the sustainability policy of



the port is prevention and minimisation of emissions, consumption, discharges, noise, and waste produced because of its activities. A key motivation of the work performed in the use case is to provide tools for better predicting the truck congestion which contributes to emission and noise in the port city. Being able to predict congestion is an important enabler for taking steps to reduce related disadvantages. Further details on potential impacts and future exploitation plans of the presented developments are described in the deliverable D7.3 of the project.

4.4 Lessons Learned and Potential Improvements

During the execution of the present demonstration, and with all the historical data and data sources made available, two main outcomes have been observed:

- Reaching the objectives, requirements, and KPIs set for the predictive analytics application is found to be feasible both regarding offline machine learning model development and online service deployment.
- Improvements could be obtained primarily by enhancing the coverage of available historical datasets and current data integrations.

Besides the lessons learned mentioned above, the following points for improvement identified too:

- The long-term prediction pipeline was demonstrated with minimal data available in service deployment. Test results with historical data indicate that having access to some additional features would improve the achievable results regarding prediction accuracy. Useful data already existing in port systems include e.g. cargo exchange volume per port call, distribution of modalities for exchanged cargo per port call (transshipment, train, truck).
- Additional data inputs would help to better capture outlier cases e.g. in truck turnaround (temporary high congestion), which is the main challenge with the current models; however, available data correlated to such scenarios was not identified during the project.
- The data available for evaluating truck turnaround times was not complete and would benefit from infrastructure (gate monitoring system) improvements to ensure that all port visits and related timestamps are fully captured.
- Further testing is needed to evaluate and optimize the developed application for production use. A primary issue to consider would be to experiment with potential mechanisms and practical changes in port traffic operations/planning with the aim to translate improved predictability into reduced congestion.

5 Demo – Improved Drivers’ Safety with MR and Haptic Solutions

5.1 Objective and Description

This use case focuses on improving the driver’s safety by combining the use of mixed reality (MR) and haptic solutions for controlling AGVs in a real scenario. In this particular case, the demonstration has been done in the port of Valencia. The particular setup used in the trial is depicted in Figure 36. Autonomous vehicles used on industrial areas may need eventually human help to accomplish with difficulties on the road. With an immersive and remote cockpit this human support can be done in an intuitive way, allowing the operator to control different AGVs from a far indoor cockpit, avoiding potential dangers of industrial areas. This is known as Teleoperated Driving.

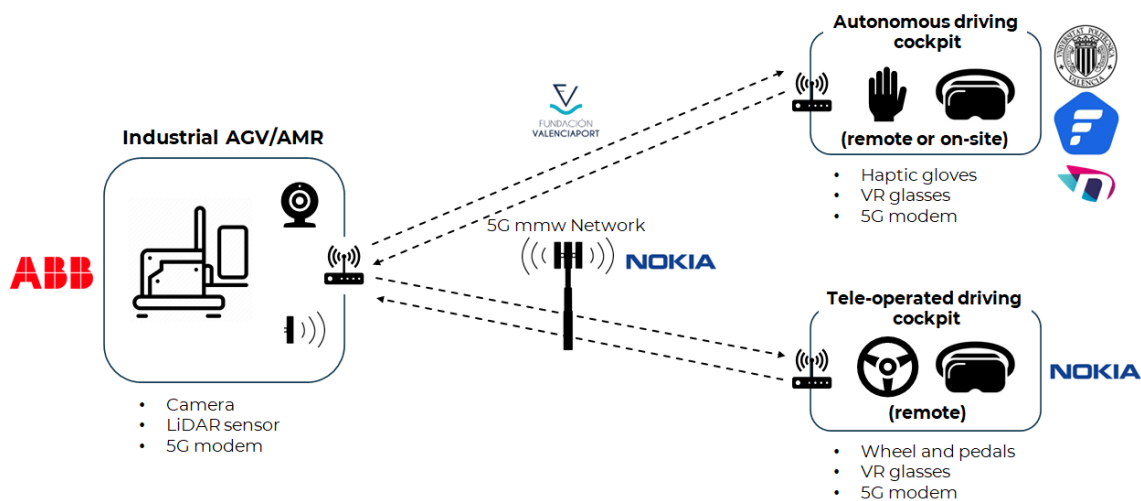


Figure 36: Main setup and components integrated in the trial.

The pillars of this use case are the telepresence of the operator to know the environment in which the vehicle moves around and the interaction with the operator to control it. The use of haptics to control AGVs is also the key of this demonstration. To support the telepresence service, several 360° video cameras with low latency have been installed in the AGV provided by ABB. In the case of the interaction for vehicle’s control, this is achieved with the cockpit and the use of haptic gloves that allow the vehicles to be managed remotely and to obtain feedback from the sensors in real time.



Through the storage and graphical representation of different KPIs in relation to the state of the network and the application, conclusions have been drawn about different aspects.

For the demonstration of this use case, we ran the AGVs modifying the load on the 5G network to analyse how each of the AGVs responds. The scheme to follow was as follows:

1. A prioritized AGV (provided by ABB) was moving inside Passengers Dock area with remotely or autonomous control.
2. A normal user represented by a smartphone started to generate different traffic density on the same 5G network for a load test:
 - a. Test 1: No load
 - b. Test 2: TCP DL 100 Mbps
 - c. Test 3: TCP DL 400 Mbps
3. Meanwhile, the different KPIs were collected for storage and graphic representation with Grafana.

Different measurements have been carried out to analyse the behaviour of the 5G network in different circumstances. These tests have been the following:

1. Average Downlink Throughput up to 132 meters.
2. Average Uplink Throughput up to 132 meters.
3. Maximum Average Latency with radio load up to 132 meters.
4. Maximum Average Latency without radio load up to 132 meters.
5. Maximum Average Latency with radio load and Latency prioritization up to 132 meters.

The final demo for this Use Case was held from November 28 to December 1 at a specific area in the passengers' dock of Valencia Port, shown in Figure 37.



Figure 37: Testing area in the Port of Valencia

The KPIs were measured to ensure that the network and the system accomplish with the system requirements to perform ToD. This requires a low latency with the maximum throughput connectivity to ensure a realistic driving environment for the operator. Three video streams, driving commands and ACK messages needed to be transmitted in real time to the cockpit.

The main objective of the demonstration is to prove an innovative ToD and ensure a proper connectivity for the AGV to perform the ToD. Also, in a second part of the demonstration the new haptic gloves from SenseGlove were validated and a digital twin was developed for improving the remote cockpit.

5.2 Setup and Execution

The following subsections provide description of the setup and execution of AGV UC. Additional information can be found in Annex IV – Setup and Execution.

5.2.1 Part I

The demo was planned with three real AGVs moving in the Passengers Dock, which images can be found in Annex IV:

- AGV-A: ABB's AGV.
- AGV-B: NOK's Robotnik AGV.
- AGV-C: 5CMM's Robotnik AGV.
- Figure 38 shows the three different AGVs used in AGV UC.



Figure 38: AGVs A, B and C for the AGV's UC

Each of the AGVs was connected to the 5G network through an Asus mobile phone via tethering which details can be found in Annex IV.

Each test scenario considers two different smartphones as 5G modems:

- Smartphone for generating traffic: used to generate different levels of traffic during the execution of each test case.

- Smartphone for AGVs connectivity with Ethernet Interface: mounted in all the AGVs for the 5G mmW n258 connectivity.

Two implementations were done to control AGVs, as shown in Figure 36:

- **NOK's Cockpit:** Used to control NOK's and ABB's AGVs remotely via ToD. It is composed of the MR glasses, in which the AGV's video is received with extra information about latency and other parameters; the steering wheel, which controls the AGV direction; the pedals, which set the AGV speed; the gear change, which allows the AGV to be moved forwards or backwards.
- **5CMM's Cockpit:** Used to control 5CMM's and ABB's AGVs remotely or on-site. The Haptic Gloves are used to control the AGVs by doing different gestures, as well as to receive haptic feedback. The Haptic Gloves can be connected to the cockpit with Bluetooth for enabling an on-site control. In the remote version, MR glasses are also used to receive the AGV's video, creating a remote immersive experience. In the on-site version, the user is directly seeing the AGV, so no glasses are needed.



Figure 39: Nokia's (left) and Fivecom's (right) cockpits.

For the measurement of KPIs, NOK has developed an infrastructure that allows them to be measured at two different levels: application and platform. At the application level, all the values that have to do with the system that allows remote driving of vehicles are obtained. Some of these values are referring to the AGV, such as speed or rotation, and others referring to the cockpit such as the FPS at which the video from the cameras is decoded, the RTT of the video stream or the throughput. On the platform side, all the measurements of KPIs that are related to the state of the 5G network and the machines that are responsible for its operation are carried out. In these values are the ping latency to modems RTTs and the values of throughput, memory usage, and CPU and GPU utilization, both in the MEC and in the 5G Core.

Three main cases were defined for the execution of the demo, the first of them with two sub-cases. Each case/sub-case was run three times.

In the first case the ABB's AGV was moving inside Passengers Dock area with autonomous control. It detected an obstacle in its path and stopped giving control to the remote driver who circled it manually and returned control to the AGV. This case has two variants that give rise to two sub-cases. In the subcase 1, the teleoperated driving is done by the remote driver using the NOK's Cockpit. In the subcase 2, the AGV start a supervision mode either remotely or on-site using the haptic gloves to control the AGV with intuitive hand movements. In

this use case, for the 5G connection, an International Mobile Subscriber Identity (IMSI) identified with the name “IMSI-AGV-A-1” which is prioritized has been assigned to the AGV. All the details about it can be found at Annex IV.

The second case consists of the remote driving of the NOK’s AGV inside Passengers Dock area using the NOK’s Cockpit. As in the previous case, a new prioritized IMSI called “IMSI-AGV-B-1”, which details can be found at Annex IV, has been assigned to the AGV.

In the last case the 5CMM’s AGV was moving inside Passengers Dock area with autonomous control and the operator supervised it either remotely or on-site. The operator used the haptic gloves to send the AGV to different predefined points to follow and when an obstacle is found, it circled it autonomously and continued its way. The operator can also stop or resume the AGV movement using the haptic gloves. For the 5G connection the “IMSI-AGV-C-1”, which was prioritized, was assigned to the AGV. All the details about the IMSI can be found at Annex IV.

While the AGV was running in each case, a normal user was connected using another IMSI which was not prioritized to generate three different load tests:

- Test 1: No load.
- Test 2: Traffic load of 100 Mbps.
- Test 3: Traffic load of 400 Mbps.

5.2.2 Part II

As an extension to the demo performed in the port of Valencia, a second part of this use case was carried out in UPV campus. The main objective was to integrate and validate the new haptic gloves from SenseGlove purchased by Fivecomm in their cockpit. The purpose was also to improve the remote cockpit by creating a digital twin to monitor the AGV in a digital environment.

The first activity consisted in the integration of the new haptic gloves for controlling the AGV. A new functionality was added in these devices called force-feedback, which stands for the resistance when trying to curve the fingers to emulate the sensation of grabbing an object. This feature was implemented for the control of the AGV as well as all the gestures recognition. The specifications of the haptic gloves are further explained in Annex IV – Setup and Execution.



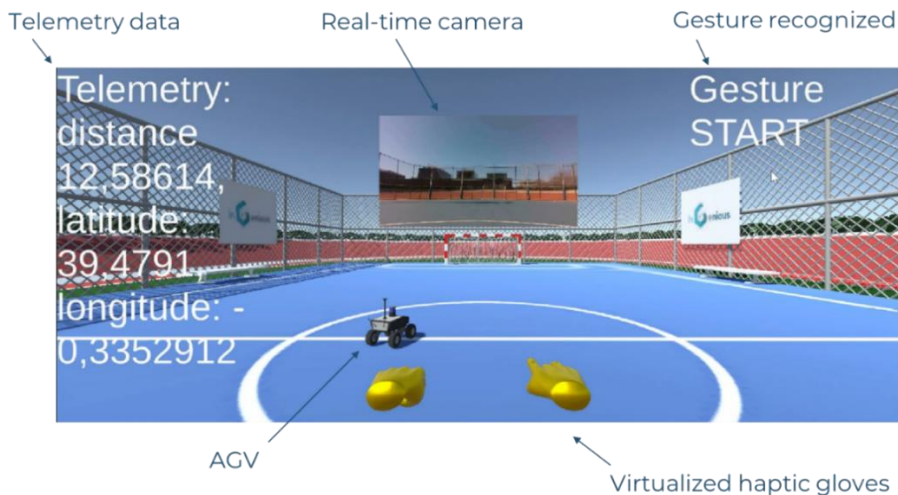


Figure 40: Unity application developed for the digital twin.

In a second step, a **digital twin** was developed. The velodrome in the UPV campus was modelled in a 3D environment and imported in the Unity application used in the cockpit. The Unity application interface is represented in Figure 40.

The position and orientation data gathered from the sensors in the AGV was sent in real-time to the cockpit for the representation in the digital twin, permitting to see the robot in the same position as in the real world. Figure 41 shows a comparison between the real scenario and the digital twin represented.



Figure 41: Real scenario (left) and digital twin with the AGV included (right).

As in the first part of the demonstration in the port of Valencia (Fivecomm’s cockpit), the gloves were used to indicate the AGV to move autonomously to a predefined point, being possible to stop and resume the movement anytime.

Apart from the position and orientation data, a video flow was sent from the robot to the cockpit with a real-time camera, as well as the GPS coordinates and the data from the proximity sensor to know how far the AGV is to the closest object. All this feedback data is represented in the cockpit application and can be seen in Figure 42. More details are provided in Annex IV.



Figure 42: Remote cockpit including the SensGlove haptic gloves, VR glasses and Digital Twin.

5.2.3 ISSUES ON EXECUTION

The following subsections provide description of issues encountered during the demonstration and mitigation actions to solve them.

Description of the issue	Mitigation measures
5G network down by saturation	Restarting the core in Madrid.
Mobiles did not connect correctly to the 5G network, so an IP was not obtained with which to establish communication	<ol style="list-style-type: none"> 1. Restarting mobile phones to get correct connection to 5G network. 2. Restarting the 5G core in Madrid.
The connection via tethering with the AGV system was lost when the mobile screen was locked	Correctly configure the mobile settings together with the "Caffeine" application to avoid screen lock and tethering disconnection.
Autonomous mode not operational when executing the demo	Remote control using the haptic gloves instead
VPN problem when connecting the AGV	Manual configuration of the routing tables and VPN
Battery of the haptic gloves not lasting more than 10 minutes	Charging all the time. If the gloves shut down restarting the application
Bluetooth range does not cover the full area. Gloves disconnecting when they are far from the server	Keeping close to the cockpit (< 20 m)

Table 15. AGV UC Issues on execution

5.3 Validation and Results

The results obtained in the demonstrations of this use case have been, in general terms, satisfactory. Most of the values established as target have been met and the deviations obtained in some of them have been reasonably justified. In addition, it has been possible to carry out remote driving of the three



considered AGVs with the two cockpits designed using the 5G network deployed in the Port of Valencia.

5.3.1 TEST CASES VERIFICATION

In this section, is summarized the results of each test case identified in D6.1 [1]

Test Case ID	Result
UC2_TC_01 - Perform measurements of 5G millimeter wave coverage in Segovia.	Passed
UC2_TC_02 - AGV teleoperation via 5G millimeter wave in Segovia.	Passed
UC2_TC_03 - Immersive cockpit.	Passed
UC2_TC_04 - Fivecomm's cockpit integration for AGV teleoperation.	Passed
UC2_TC_05 - Perform measurements of 5G millimeter wave coverage in Valencia Port.	Passed
UC2_TC_06 - ASTI AGV teleoperation via 5G millimeter wave in Burgos.	Passed
UC2_TC_07 - ASTI AGV teleoperation via 5G millimeter wave in Valencia Port.	Passed

Table 16. AGV's UC Test case verification

All the defined test cases have successfully passed, obtaining the values of the different KPIs of interest. This has allowed an analysis of the use case and the needs of the network to carry out remote driving of vehicles with different cockpits. More details about each test case, such as the description, detailed expected results and detailed actual results can be found in Annex IV – Validation and Results.

5.3.2 KPIS

In this section the KPIs for each test case are presented. To get the target value we used the values defined in the D2.1 [11]. Results are shown in Table 17:

KPI	Test Case Reference	Target	Actual
Segovia Coverage	UC2_TC_01	~ 500 m	412 m
Segovia End-to-end latency	UC2_TC_01	< 100 ms	Min: 25,06 ms Max: 38,2 ms Avg: 32,8 ms
Valencia Port Coverage	UC2_TC_02, UC2_TC_05 UC2_TC_07	~ 500 m	420 m
Valencia Port End-to-end latency	UC2_TC_05	< 100 ms	Min: 28,1 ms Max: 33,9 ms Avg: 30,3 ms
Valencia Port Availability	UC2_TC_02, UC2_TC_03 UC2_TC_04, UC2_TC_07	> 99.999 %	100 %
Valencia Port Reliability	UC2_TC_02, UC2_TC_03 UC2_TC_04, UC2_TC_07	> 99.999 %	97,93%
Valencia Port Mobility	UC2_TC_02, UC2_TC_03 UC2_TC_04, UC2_TC_07	< 30 Km/h	12,3 Km/h

Valencia Port AGV-A Data rate	UC2_TC_02	~ 10 Mbps	Min: 9,2 Mbps Max: 10,8 Mbps Avg: 9,97 Mbps
Valencia Port AGV-A End-to end latency	UC2_TC_02	< 100 ms	Min: 32,6 ms Max: 162,1 ms Avg: 53,9 ms
Valencia Port AGV-C Data rate	UC2_TC_04	~ 10 Mbps	Min: 5,41 Mbps Max: 19,2 Mbps Avg: 13,2 Mbps
Valencia Port AGV-C End-to end latency	UC2_TC_04	< 100 ms	Min: 24,3 ms Max: 212,7 ms Avg: 57,4 ms
Burgos AGV-B Data rate	UC2_TC_06	~ 10 Mbps	Min: 9,06 Mbps Max: 11,13 Mbps Avg: 10,3 Mbps
Burgos AGV-B End-to end latency	UC2_TC_06	< 100 ms	Min: 207,3 ms Max: 507,9 ms Avg: 498,1 ms
Valencia Port AGV-B Data rate	UC2_TC_07	~ 10 Mbps	Min: 9,19 Mbps Max: 10,8 Mbps Avg: 10,1 Mbps
Valencia Port AGV-B End-to end latency	UC2_TC_07	< 100 ms	Min: 30,9 ms Max: 164,8 ms Avg: 67,2 ms
Velodrome mobility	UC2_TC_04	< 30 km/h	10,8 km/h
Velodrome video throughput	UC2_TC_04	-	360p: 2,5 Mbps 720p: 8 Mbps 1080p: 16 Mbps

Table 17. AGV UC KPIs

For this use case, all the parameters that directly affect the remote driving quality of an AGV with a 5G network have been considered. Consequently, all the KPIs are related to the level of network saturation (data rate, latency), and its features (availability, coverage, mobility, reliability). As most of the measurements are carried out periodically along the demonstration duration, the statistics that most realistically represent each of them are required.

In general, there was not a big deviation from target value. In the case of coverage, the values that appear in the table are limited by the space available to make the test. For the end-to-end latency deviation in Burgos tests, this is because it was an integration job between ABB and NOK that was carried out remotely and not in person. The objective of these tests was to be able to remotely drive the ABB's AGV with the NOK's cockpit correctly.

A new KPI was added for the measurement of the performance in the velodrome demo (*Velodrome video throughput*). This KPI measures the throughput used in the demo for different video qualities. A comparison of the old and new haptic gloves was carried out, and the results appear in Table 18.

KPI	Neurodigital	SenseGlove
Bitrate (wired)	375 kbps (UL) - 250 kbps (DL)	Not possible
Bitrate (wireless)	75 kbps (UL) - 20 kbps (DL)	52 kbps (UL) - 6 kbps (DL)
Bluetooth range	32 m (outdoor) - 12 m (indoor)	9 m (outdoor) - 5 m (indoor)

Table 18. AGV UC KPIs. Comparison Neurodigital vs GloveSense haptic gloves.



5.3.3 IMPACT ASSESSMENT

Potential benefits that can be achieved in this scenario include, automatic handling of assets, human-machine iteration by working remotely in unexpected circumstances and scalability (e.g., working remotely in multiple sites governed by AGVs).

There are a lot of places where AGVs are used to facilitate human work. The main environments in which this situation occurs is in factories, ports, airports, or warehouses, places where the transport of goods is necessary. These autonomous vehicles facilitate the task of transportation in these places but have a problem in reacting to unforeseen events during their journey. Although most of them incorporate into their system a mechanism that allows the vehicle to stop after an unforeseen event, it is possible that after this stop the vehicle doesn't know how to continue. This problem is solved in this scenario in which the remote driver can take control of the AGV, solve the issue, and return control back to the AGV.

As for the immersive component of the solution, immersive vehicle driving allows the operator to feel like driving a real car from the driver's seat which makes driving more real and therefore safer.

5.4 Lessons Learned and Potential Improvements

In relation to remote driving with the use of 5G networks in an immersive environment, the main lesson learned is that safe driving is possible with the established components. However, optimal performance of the network is needed. We observed that the service is still highly dependent on the network performance. When the latency obtained is higher than the established limit, the communication between the operator and the AGV is affected, putting security at risk.

Another component with room for improvement is the haptic glove reaction capability. Both gloves tested in INGENIOUS (Neurodigital and SenseGlove) are a good alternative for providing orders, but the haptic feedback in both models does not feel completely natural yet.

The arrangement of the cameras and their overlapping in the cockpit are also essential so that the operator feels comfortable and can drive safely. As a possible improvement in this aspect, we recommend exploring the use of new cameras to improve the immersion experienced by the teleoperator and therefore security.

In terms of potential improvements in the administration of the 5G network, it has been detected that it is necessary to separate the uplink and downlink into different slices, as well as to assign different priorities depending on the needs, in order to obtain a better distribution of network traffic and, therefore, a better driving experience.

6 Demo – Intermodal Asset Tracking via IoT and Satellite

6.1 Objective and Description

Ships are equipped with legacy communication networks allowing the exchange of information with port terminals when the ship is docking, but not when it is sailing. However, no sensors are installed in containers to monitor and collect real-time data on cargo location and conditions and container safety.

Thus, the Ship Use Case aims at providing E2E asset tracking via satellite backhaul from the IoT RAN to the corresponding data/control centre, enabling real-time/periodic monitoring of predetermined parameters (temperature, humidity, GPS, movement, bumps, etc.) of shipping containers when they are sailing on the sea, while terrestrial IoT connectivity is provided when the ship approaches the port and while on land.

To enable this ubiquitous coverage, IoT tracking devices have been installed on the shipping containers transported on both maritime and inland segments. The end-to-end intermodal asset tracking would allow shipment information to be ubiquitously available across all connected platforms and interested parties in real-time.

As part of the Ship Use Case of the iNGENIOUS project, live over-the-air and lab demonstrations were conducted, paving the way towards inter-modal asset tracking via IoT and Satellite, such as:

- Live over-the-air mid-term demo in Valencia, Spain, on 27-28 April 2022.
- Live over-the-air final demo in Valencia, Spain, on 03 October-09 November 2022, 21-23 November 2022 and 01-09 March 2023.
- Ongoing lab demonstration and support using the iDR lab testbed.

The **mid-term and final live over-the-air (OTA)** demonstrations together included and demonstrated:

- Installation of sensors on iNGENIOUS container.
- Installation of the Smart IoT GW on the ship.
- Installation of the satellite terminal at the Port of Valencia.
- iNGENIOUS container transport by ship from the Port of Valencia to the Port of Piraeus and vice-versa.
- iNGENIOUS container transport by rail from the Port of Valencia to Madrid Rail Terminal.



- iNGENIOUS container transport by truck from Madrid Rail Terminal to Valencia.
- Integration of the Smart IoT GW with the sensors.
- Integration of the Smart IoT GW with the M2M platform.
- End-to-end connectivity using satellite backhaul.
- Graphic interface for representing the real-time data from the sensors.

The difference between the live over-the-air mid-term and final demo was that in the mid-term demo, the iNGENIOUS container was not transported by ship, truck or train, and also the satellite terminal was not deployed at the Port of Valencia. Instead, a VPN connection was established to simulate a direct Ethernet connection between the Smart IoT Gateway located at the Port of Valencia, Spain and the satellite terminal deployed at SES's premises in Betzdorf, Luxembourg. The "Satellite VPN" was tunnelled over this VPN to provide the satellite connection towards the IoT Cloud/Data centre. The live over-the-air mid-term demo was used to validate the configurations and connectivity of the Smart IoT Gateway with IoT sensors, the satellite terminal and the IoT Cloud /Data Centre in order to de-risk the live over-the-air final demo. Hence, in the following sections, we will describe only on the live over-the-air final demo.

The **iDR lab testbed** was designed and implemented to reflect the live over-the-air setup and was used throughout the project to prepare for and stage the live demonstrations with limited windows of live satellite capacity. This testbed proved to be a valuable asset for this use case as it ensured optimal use of the valuable live satellite capacity. More information about the iDR lab testbed can be found in the Annex V.

6.1.1 WORKFLOW OF THE FINAL DEMO

The live over-the-air final demo is split into two parts. The first part included the ship transportation of the iNGENIOUS container from Piraeus to Valencia and vice versa, while the second part included the container transportation using a truck. The workflow is described below:

Part I

1. The iNGENIOUS container was equipped with heterogeneous IoT devices able to monitor the real time location of the cargo, the status of the sensor's battery, internal environment of the container such as temperature and humidity and detect critical events such as door opening, arrived at the Container Terminal on the 3rd of October. The container was loaded on to a COSCO vessel at the Port of Valencia on the 4th of October 2022. The next day the vessel started its trip towards the Port of Piraeus.
2. During the trip, the heterogeneous IoT devices were sending status updates once per day.
3. The Smart IoT GW was installed on the bridge of the COSCO vessel to aggregate the messages from the IoT devices.
4. The COSCO vessel arrived at the Port of Piraeus on 8 October 2022 and the container was discharged. Subsequently, the container was stored at the COSCO Piraeus Terminal whilst waiting for its return trip.

5. On the 27th of October, the iNGENIOUS container was loaded on another COSCO vessel and on the same day the vessel started its trip towards the Port of Valencia.
6. During the trip, the heterogeneous IoT devices were sending status updates, again once per day.
7. The Smart IoT GW was removed from the first vessel and installed on the bridge of the second COSCO vessel to aggregate the messages from the IoT devices.
8. The vessel arrived at the Port of Valencia on 8 November and the iNGENIOUS container was discharged on the same day. It was then transported to the depot near the Port of Valencia on 9 November 2022.
9. The Smart IoT GW was also removed from the vessel. After analysing the collected data from the Smart IoT GW, the researchers noticed connectivity issues with the IoT devices during the trip. However, the sensors themselves had the capability to store the data.
10. On 21st of November, the satellite terminal was installed at the Port of Valencia, as well as the Smart IoT GW, while the iNGENIOUS container was placed nearby (<20m) (see Figure 52).
11. The communication between the Smart IoT GW and the IoT devices was established, and the satellite connection was setup.
12. The data was then sent to the IoT Cloud/Data centre via the satellite backhaul connection. The baseline space segment, which was used corresponds to SES's GEO ASTRA 2F satellite, which provided seamless connectivity between the satellite terminal at the Port of Valencia and iDirect's 5G-enabled Velocity™ Intelligent Gateway (IGW) hub located at the SES teleport in Betzdorf, Luxembourg.
13. Subsequently, the data was visualized from a graphical interface connected to the IoT cloud.

Part II

14. The iNGENIOUS container was transported from the depot to Valencia Port Rail Terminal the 1st of March 2023. The next day, March the 2nd the container was loaded in a COSCO train and transported to Madrid.
15. During the trip, the heterogeneous IoT devices were sending status updates once per hour.
16. The Smart IoT GW was not installed because during inland transport the IoT devices sent data by NB-IoT directly to the cloud.
17. The COSCO train arrived at Madrid Rail Terminal on 2nd March 2023 and the container was discharged. Subsequently, the container was stored at the Madrid Dry port whilst waiting for its return trip.
18. On the 9th of March, the iNGENIOUS container was loaded on a truck transported to Valencia.
19. During the trip, the heterogeneous IoT devices were sending status updates, again once per hour.
20. The truck arrived at the depot the same day, March 9th at 17:07.



21. Subsequently, the data was visualized through a graphical user interface connected to the IoT cloud.

6.2 Setup and Execution

The following subsections provide description of the setup and execution of Ship UC.

6.2.1 Part I

The end-to-end demonstration setup of the Part I of the live over-the air final demo (where the iNGENIOUS container is discharged from the vessel and stored at the Port of Valencia) is illustrated in Figure 43. It was built upon the following elements by the respective iNGENIOUS project partners:

- SES: Provided end-to-end managed services, powered the space segment with its existing ASTRA 2F geostationary satellite system (28.2oE), and delivered seamless connectivity between the remote and the hub platform located at its teleport in Betzdorf, Luxembourg. Furthermore, SES provided the satellite terminal and the Smart IoT Gateway.
- iDR: Provided iDirect’s 5G-enabled Velocity™ Intelligent Gateway which incorporates SDN/NFV and MEC capabilities and enables the satellite integration into a 3GPP 5G core network architecture as a 5G access network. Furthermore, iDR provided MEC server and VSAT modems along with the support required to provide the satellite backhaul connectivity.
- FV: Provided the IoT devices, the iNGENIOUS shipping container, access to the Port of Valencia, and the access point for establishing internet connection.
- COSSP: Provided a Ship for shipping the iNGENIOUS container from Valencia to Piraeus and vice versa and prepared all the documents for maritime transports.

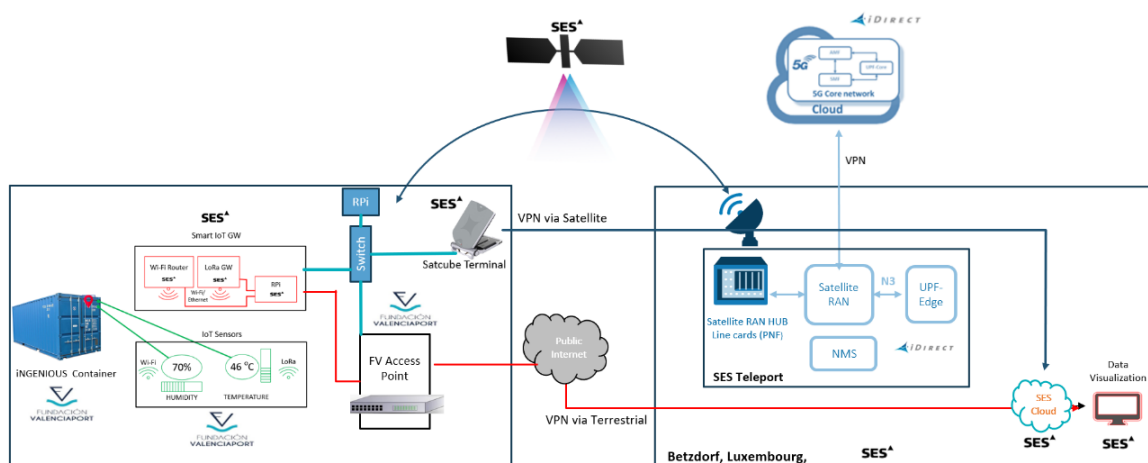


Figure 43: End-to-end architecture of the final demo.

More details are provided in the following sections.

Satellite Capacity and Space Segment: Occasional Use (OU) Satellite capacity has been provided by SES over ASTRA 2F satellite. As can be seen in 80, satellite capacity was provided for the live over-the-air demonstrations as well as other periods in preparation for the live demonstrations to test the end-to-end system including the iDirect 5G-enabled Velocity™ IGW and modem.

OU Book ID	Start Date	End Date	Satellite	Freq. Band	BW (MHz)	Satellite Hub Activities
499159	25/05/21	31/05/21	SES ASTRA 2F	Ku	6	Initial testbed testing over live satellite
199335	01/11/21	05/11/21	SES ASTRA 2F	Ku	6	Feasibility testing using SatCube
214210	07/02/22	11/02/22	SES ASTRA 2F	Ku	6	End to end testing with SatCube and IoT network.
222888	28/03/22	01/04/22	SES ASTRA 2F	Ku	6	SatCube testing in preparation for mid-term demo
222889	25/04/22	29/04/22	SES ASTRA 2F	Ku	6	Mid-term demo
258712	07/11/22	25/11/22	SES ASTRA 2F	Ku	6	Final Demo

Table 19. SES’s ASTRA 2F Space Segment.

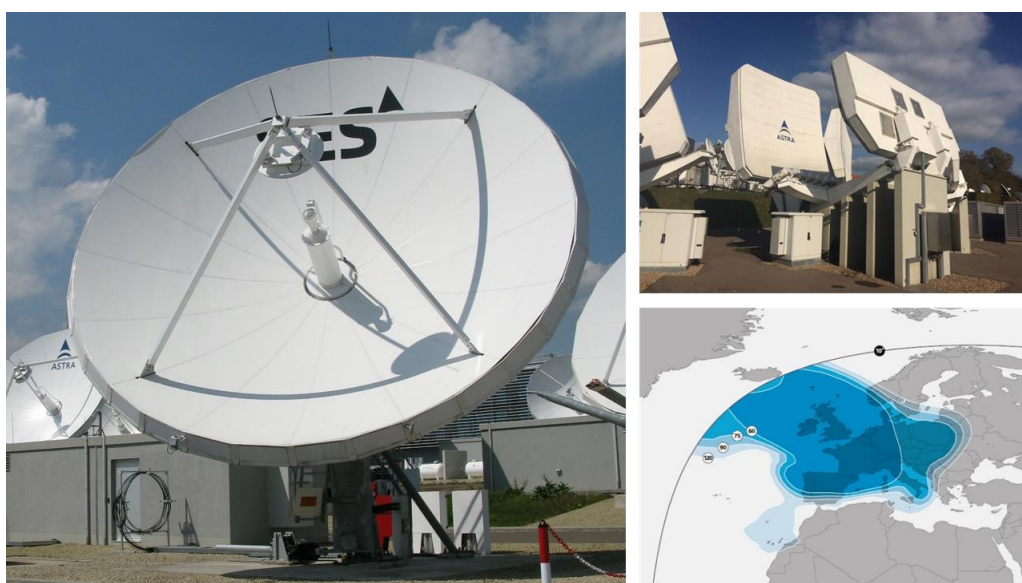


Figure 44: i) RF Uplink ground Station: ATF #33 Antenna, Diameter: 9m, Vertex, Tx/Rx, Ku-band, ii) RF Downlink Ground Station: MBA#102 Antenna, Diameter: 4.5m, Multi-Beam Antenna, Rx only, Ku-band, and iii) SES GEO Satellite ASTRA 2F (28.2oE) - Europe Ku-band beam

Uplink Ground Station: A 9m Ku band antenna located at SES’s teleport in Betzdorf, Luxembourg was used for the RF uplink (Figure 44).

Downlink Ground Station: Furthermore, a 4.5m Ku band, multi-beam antenna located at SES’s teleport in Betzdorf, Luxembourg was used for the RF downlink reception, and it can be seen in Figure 44.

Satellite Terminal: The SatCube Ku-band small-factor transportable terminal (see Figure 45) is a light weight, compact, portable satellite terminal



that enables broadband connectivity almost anywhere on earth. More information can be found in D6.2 [2].

Smart IoT Gateway: The Smart IoT GW is pictured in Figure 45. The main platform of the Smart IoT GW is the Raspberry Pi 4 and it is protected by a universal and modular metal case. More information can be found in D6.2 [2].



Figure 45: i) Satcube transportable satellite terminal and ii) Smart IoT Gateway

5G-enabled Velocity™ IGW hub: iDirect’s 5G-enabled Velocity™ IGW hub (see Figure 46) is installed at the SES teleport facility in Luxembourg. This test system is hosted on a Dell R630 server which is specifically configured and administered to support the iNGENIOUS use case demonstrations.



Figure 46: Front and back of the iDirect’s 5G-enabled Velocity™ Intelligent Gateway hub

Modems: iQ200, iQ Desktop and 9350 modems (see Figure 47) were used to verify and test satellite connectivity prior to the live over-the-air demonstrations.



Figure 47: iQ200, iQ Desktop and 9350 Modem

MEC Server: For the live demonstration at the Port of Valencia a MEC server (Raspberry Pi 3 model B and a network switch) were used to facilitate the integration of the Smart IoT GW and the satellite network and also to make the deployment easier to manage.

INGENIOUS Container: FV purchased a twenty-foot dry shipping container following 22G1 International Organization for Standardization (ISO) standard, where specific external dimensions (Length: 6,058 m; Width: 2,438 m and Height: 2,591 m) are defined (see Figure 48).

The container was purchased for demonstrating inter-modal asset tracking through the installation of IoT tracking and monitoring sensors, which can gather measurements and transmit the information via LoRa and NB-IoT.

For performing the live over-the-air final demo, the container was customized and painted by FV, adding the project logo along with the logos of all partners involved in the UC, as shown in Figure 48.



Figure 48: i) 22G1 purchased container and ii) iNGENIOUS Container

During the final demonstration, the container was monitored and transported from Valencia to Piraeus in a round trip by combining terrestrial (inside the ports of Valencia and Piraeus) and maritime transportation (from Valencia to Piraeus).

Sensors: FV outsourced the acquisition of a set of IoT Sensors (more information in D6.2 [2]) which were integrated together with LoRa and NB-IoT communication modules and then installed on the shipping container for monitoring cargo conditions and container status when terrestrial and maritime transportation is performed. Thanks to the integration of LoRa and NB-IoT communication modules, IoT sensors were able to communicate data with the Smart IoT GW developed by SES. All sensors were integrated together with the communication modules and assembled in a single IoT tracking device. After ensuring this integration and the communication with the Smart IoT GW, the device was installed on the iNGENIOUS container.



Figure 49: Final demo device installation

Terrestrial Communication: FV ensured the availability of Commercial NB-IoT coverage for performing IoT tracking tests inside the Port facilities. The coverage of these technologies was tested by UPV and FV through the execution of car-driven tests inside the facilities of Valencia Port. IoT tracking devices use this coverage for reporting the tracking and monitoring measurements.

Port Facilities: FV and COSCO ensured the access to Valencia Port and COSCO terminal facilities for performing this demonstration. COSCO terminal at the Port of Valencia was the place where iNGENIOUS container was loaded and discharged as part of the trip to Piraeus. As for the mid-term demonstration, FV also managed the access to a specific depot inside the Port.

Ship: A different ship was used for each voyage during the first part of the final demo. For the first trip from Valencia to Piraeus, the iNGENIOUS container was loaded on a COSCO Vessel, CSCL Venus, which is a 14074 TEU vessel. For the return trip from Piraeus to Valencia, another COSCO vessel was used, particularly, a 13114 TEU vessel called COSCO Glory.

Vessel Trip Transport Documentation: The documentation, needed for the vessel trip, was prepared by COSCO. More information can be found in D6.2 [2].

6.2.2 Part II

The end-to-end demonstration setup of the second part of the live over-the-air final demo (the iNGENIOUS container was also loaded to the truck) is illustrated in Figure 50.

In particular, it was built upon the following elements by the respective iNGENIOUS project partners.

- FV: Provided the IoT devices, the iNGENIOUS container, the access to the Port of Valencia and the terrestrial communication.
- COSSP: Provided the transport of the iNGENIOUS container by rail from Valencia to Madrid as well as a truck for transporting the iNGENIOUS container from Madrid to Valencia. Furthermore, COSSP prepared all the documents for the rail and truck transports.

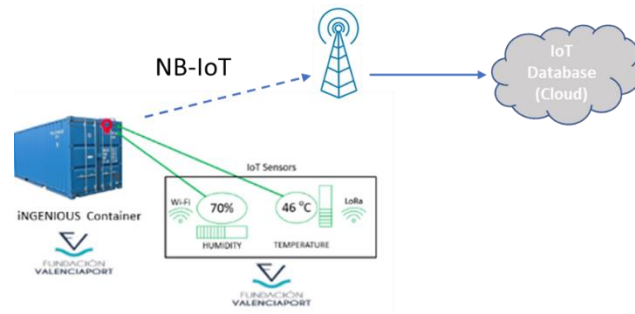


Figure 50: End-to end architecture of the final demo (part II)

More details are provided in the following sections.

Truck: For the inland transport from Madrid to Valencia during the second part of the final demo, the container was transported on a truck by a COSCO’s transport provider.

Truck Transport Documentation: The documentation needed for the inland transport by truck was prepared by COSCO. This included the release order, the acceptance order and the transport order.

Rail: During the second part of the final demo, the container was transported in the Valencia-Madrid Rail corridor by train. The propulsion of the locomotive was diesel-based. The train had 18 wagons of 90 feet, having the capacity to transport 72 TEU containers. In Figure 51 you can see the iNGENIOUS container during the second part of the Final Demo.



Figure 51: iNGENIOUS container starting rail transport from Valencia to Madrid.

Rail Transport Documentation: The documentation needed for the terrestrial rail transport was prepared by COSCO. This included the release order, the acceptance order and the transport order.

6.2.3 ISSUES ON EXECUTION

The following subsections provide description of issues encountered during the demonstration and mitigation actions to solve them.



Description of the issue	Mitigation measures
<p>The satellite terminal was not installed on the ship for several reasons: 1) very high cost (more than 40K) for buying a tracking GEO satellite terminal, 2) a lot of effort (site survey, installation, etc.). In this particular UC, the effort and the cost would have been doubled because we had to use two different ships for the shipping of the iNGENIOUS container from Valencia to Piraeus and vice versa and 3) equipment installation follows very strict time schedules aligned with maintenance periods.</p>	<p>In line with the proposal, a fixed satellite terminal was installed in the Port of Valencia for carrying out the demonstrations. Approval for the transmission regulatory licence of the satellite terminal at the Port of Valencia was obtained after a formal request to the Spanish Ministry.</p>
<p>The iNGENIOUS container, equipped with the sensors, was transported from Valencia to Piraeus and vice versa. During the trip, the Smart IoT GW, installed on the bridge of the ship, and the sensors faced connectivity issues.</p>	<p>The sensors had the capability to store the data collected during the trip from Valencia to Piraeus and vice versa. When the ship arrived at the Port of Valencia and the connectivity issues were resolved the data was transmitted to the Smart IoT GW.</p>
<p>Sensors were not able to gather GPS data while the container was being transported in the maritime segment. This issue was faced because the container was hidden by several layers of containers and the GPS antenna did not have direct visibility with the GPS satellite for establishing GPS communication and retrieving tracking information.</p>	<p>GPS tracking information could be provided in the maritime segment by integrating AIS data with the stream of information provided by the IoT sensors. AIS data provides tracking information linked to the vessel position. As an alternative, GPS tracking information could also be provided if the Smart IoT GW integrated a GPS antenna, following a similar approach as for the AIS.</p>

Table 20. Ship UC Issues on execution

6.3 Validation and Results

In this section, the results the test cases identified in D6.1 [1] are summarized.

Part I

Data collected during the trip from Valencia to Piraeus and vice versa

The data collected from the IoT devices during the trip from Valencia to Piraeus and vice versa should have been sent in real time to the Cloud through satellite backhaul. However, as mentioned earlier, the installation of the satellite terminal on a COSCO vessel could not be carried out during this project. For this reason, the Smart IoT GW was capable of storing the data during the trip and transmitting it over satellite when the ship arrived again at the Port of Valencia, where a satellite terminal was installed.

However, the Smart IoT GW and the IoT devices faced some connectivity issues during the trip and hence the Smart IoT GW was not able to gather and store the data. But the IoT devices, themselves, also had the capability to store their data and transmit it when the communication with the Smart IoT GW could be set up. And this is exactly what happened on 21 November, where the satellite terminal was installed at the Port of Valencia, as well as the Smart IoT GW, while the iNGENIOUS container was placed in the near vicinity (<20m) (see Figure 52).





Figure 52: SatCube, Smart IoT GW and iNGENIOUS container at the Port of Valencia

When the connectivity issues between the IoT devices and Smart IoT GW were resolved, the collected data during the trip was transmitted to the Smart IoT Gateway, and then it was sent to the SES Cloud through satellite backhaul and we were able to observe the collected data in the Grafana dashboards.

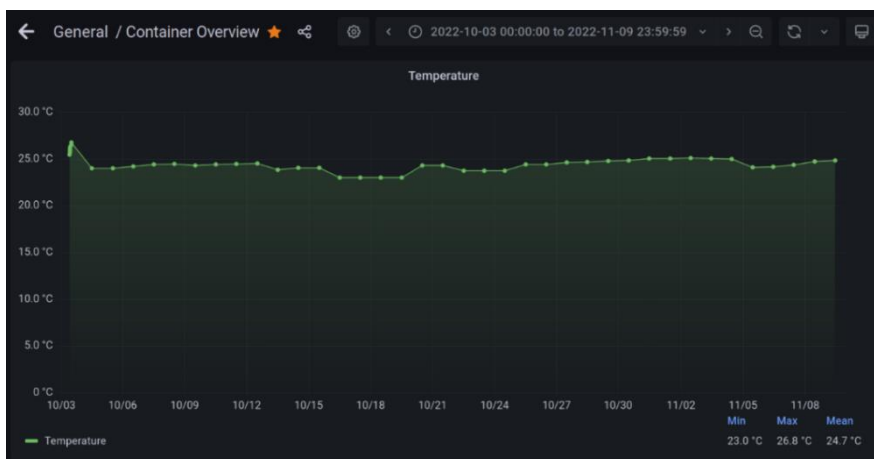


Figure 53: Temperature of the iNGENIOUS container during the trip from Valencia to Piraeus and vice versa.



Figure 54: Humidity of the iNGENIOUS container during the trip from Valencia to Piraeus and vice versa



Data collected on 21 November at the Port of Valencia

After the transmission of the collected data, from the round trip to Piraeus, to the SES Cloud, we continued the tests at the Port of Valencia. This time, we transmitted in real-time the data from the IoT devices to the SES Cloud. As mentioned earlier and shown in Figure 52, on 21 November 2022, the satellite terminal was installed at the Port of Valencia, as well as the Smart IoT GW, while the iNGENIOUS container, equipped with the IoT devices, was placed in close proximity (<20m).

With the communication between the IoT devices and the Smart IoT Gateway as well as the satellite connection established, the IoT devices were sending in real time the measured data. The Smart IoT GW then collected it and pushed it towards the SES Cloud via satellite backhaul. This automatic process worked flawlessly, and we were able to observe the collected data in the Grafana dashboards of the SES Cloud servers. From that point on, all periodic transmissions coming from the IoT devices were immediately forwarded to the Cloud server.

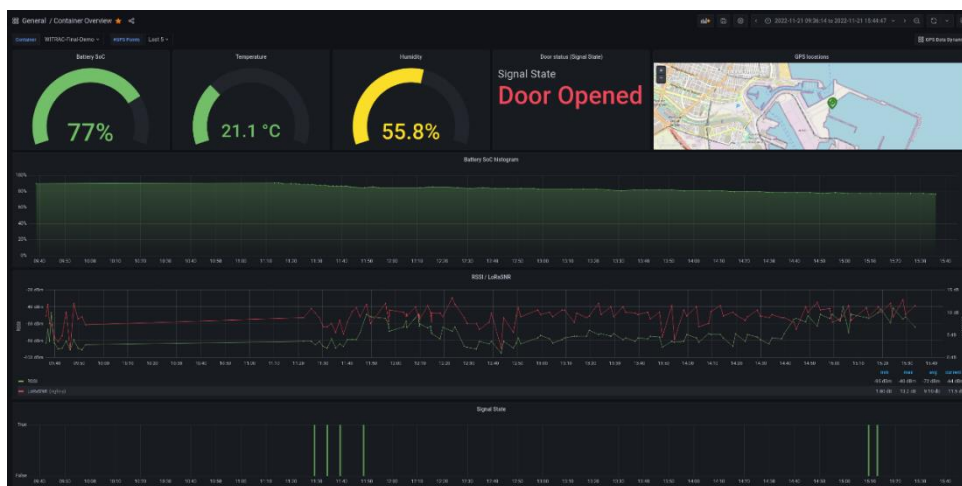


Figure 55: Overview screenshot of the Cloud-side dashboard, giving a general impression of the received data during the real-time measurements at the Port of Valencia on 21 November 2022



Figure 56: GPS location of the IoT devices during the real-time measurements at the Port of Valencia on 21



Figure 55 provides an overview screenshot of the Cloud-side dashboard, giving a general impression of the received data, where we noticed consistent transmissions of temperature, humidity, battery state and GPS values. The door state (or signal state) was only transmitted if triggered, which did not work consistently. Furthermore, the accelerometer values were only included in three sensor messages.

Figure 56 presents the GPS location of the IoT devices at the Port of Valencia. The measured GPS points showed an approximate accuracy of around 50m, being distributed in a radius of roughly 25m around the actual location of the IoT devices. This might be partially caused by the fact that at some point during the demonstration, the IoT devices were removed from the iNGENIOUS container and placed closer to the Smart IoT GW.

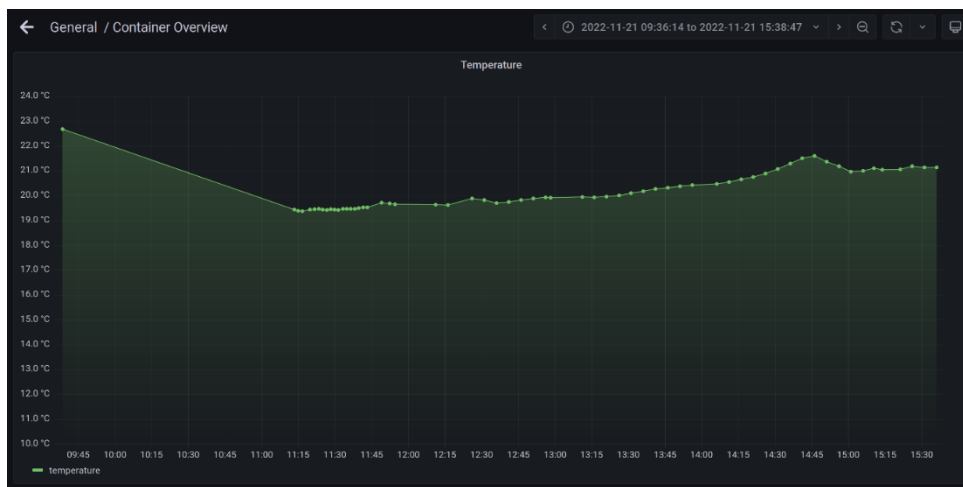


Figure 57: Temperature, measured in real-time from the IoT devices, in the Port of Valencia on 21 November 2022

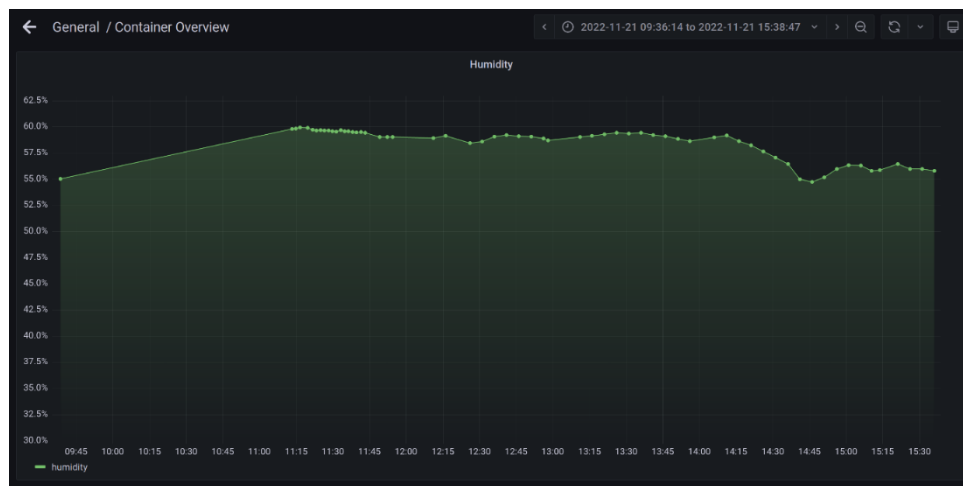


Figure 58: Humidity, measured in real-time from the IoT devices, in the Port of Valencia on 21 November 2022

In the beginning, the IoT devices were sending status updates every 90 seconds and after 30 minutes of consistent transmissions, the frequency changed to every 5 minutes. This can be seen in Figure 57, where we can also see that the temperature is over 19°C at around 11:15AM and steadily increased to 21°C around 15:00PM, fairly realistically representing a temperature trend, when the IoT devices were wind-protected during the demonstration.



Like the temperature, the humidity was measured consistently, ranging from 55% to 60%, as can be seen in Figure 58. It seems adequate, considering the proximity to the sea and the perceived humid weather conditions during the demonstration.

Furthermore, Figure 59 illustrates the battery state of charge, where the power steadily declined from 90% to 77% over the demonstration period of 4 hours and 15 minutes. This represents a drop of around 13% for 61 transmissions made at an average rate of 4.25 transmissions per minute. Considering that the intended transmission rate for the actual trip was one transmission per day, the battery of the sensor device should have more than enough capacity to cover the whole duration of the trip.



Figure 59: Battery state of charge of the IoT devices, measured in real-time in the Port of Valencia on 21 November 2022

Moreover, as the door state is only transmitted when triggered by a change event of the physical state of the door, Figure 60 does not show a consistent graph as for the previous metrics. We can see a total of six opening events received and three closing events. This is an indication that some intermediate state change events have been dropped, as the data should represent an alternating pattern between the opened and closed state.



Figure 60: Door state of the iNGENIOUS container, measured in real-time in the Port of Valencia on 21 November 2022

In addition, over the period of the demonstration, we received three data transmissions containing the accelerometer payload. The values are within the



expected range, as one axis has an acceleration value of around 1G (here: Z-axis), while the remaining two have a value close to 0G (see Figure 61).



Figure 61: Accelerometer measured in real-time in the Port of Valencia on 21 November 2022

Figure 62 illustrates the end-to-end latency for transmitting the measured data from the IoT devices to the SES Cloud through satellite, where the average latency was 613 ms.

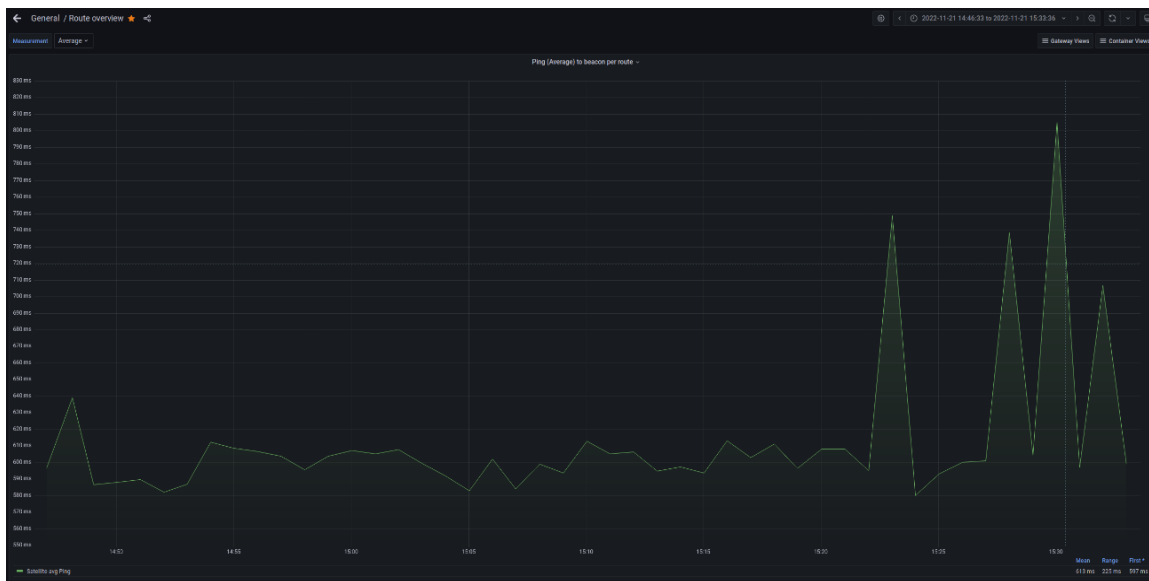


Figure 62: End-to-end latency for transmitting the measured data from the IoT devices to the SES Cloud through satellite at the port of Valencia on 21 November 2022

Finally, Table 21 below shows the results of the end-to-end round-trip time (RTT) (ICMP) tests carried out between the MEC server (FV) and the satellite network edge gateway at the satellite hub. The RTT can vary depending on the location of the remote terminal but the typical RTT for the satellite hop of the data path is between 560ms and 570ms. In this example, it would mean the extra hops contributed to an increased overhead of approximately 23ms to 33ms of the overall RTT.



Betzdorf Teleport egress point -> Satellite Terminal
--- 192.168.252.100 ping statistics ---
1050 packets transmitted, 1048 received, 0% packet loss, time 1049013ms
rtt min/avg/max/mdev = 570.345/593.199/1185.147/23.257 ms
Satellite Terminal -> Betzdorf Teleport egress point
--- 192.168.252.73 ping statistics ---
1042 packets transmitted, 1041 received, 0% packet loss, time 1040968ms
rtt min/avg/max/mdev = 577.292/593.949/677.548/15.083 ms

Table 21. ICMP RTT of Satellite Link

Part II

The data collected from the IoT devices during the trip from Valencia to Madrid by train, and back from Madrid to Valencia by truck, was sent in real time to the Cloud through the terrestrial (commercial) network using the device’s NB-IoT antenna.

The messages received at the IoT Cloud can be visualized in the Figure 63. The message includes the measured temperature, humidity, accelerometer, and GPS location at the date and time 2023-03-03 09:03:32. This figure also shows that the location captured by the IoT devices corresponds to the middle point of the entire trip of the demo’s Part II (the Madrid Dry port).

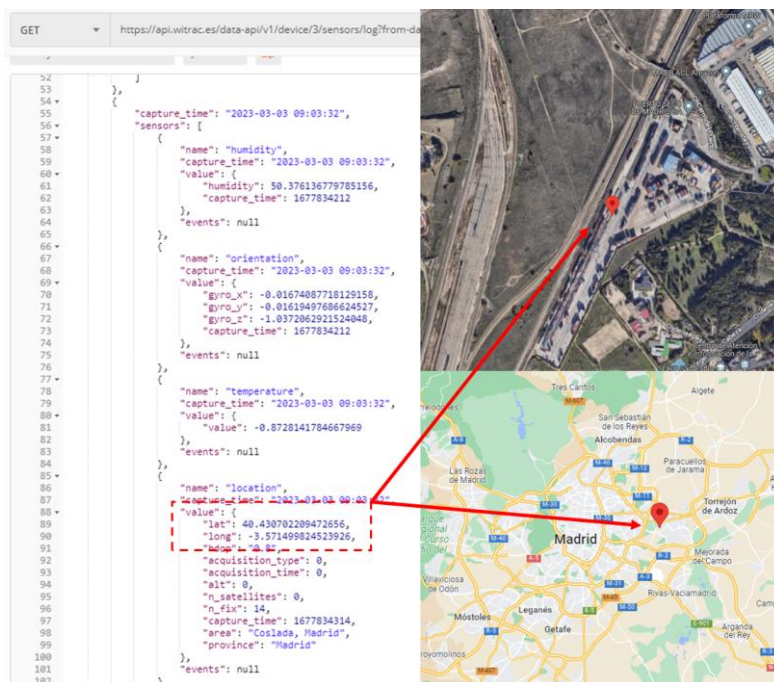


Figure 63: Ship UC Demo part B – IoT message received.



In the Figure 64, it is shown an overview screenshot of the Cloud-side dashboard (provided from FV), giving a general impression of the received data, where we noticed consistent transmissions of the GPS values.

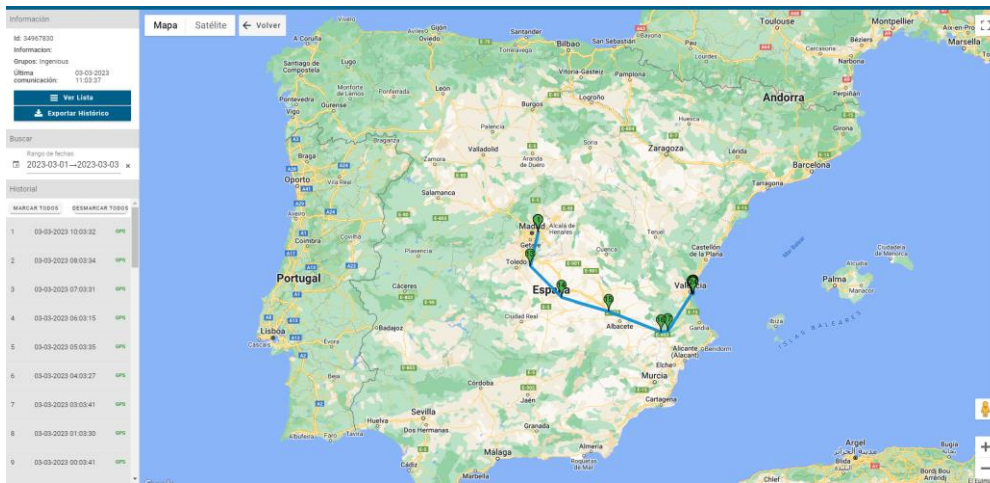


Figure 64: GPS location reported by the sensor in Part B trip

As for the rest of the parameters, the Figure 65 shows an overview screenshot of the Cloud-side dashboard (provided from FV) listing and plotting the historical values of temperature, humidity and accelerometer parameters sent by the sensor during the trip done by the container.

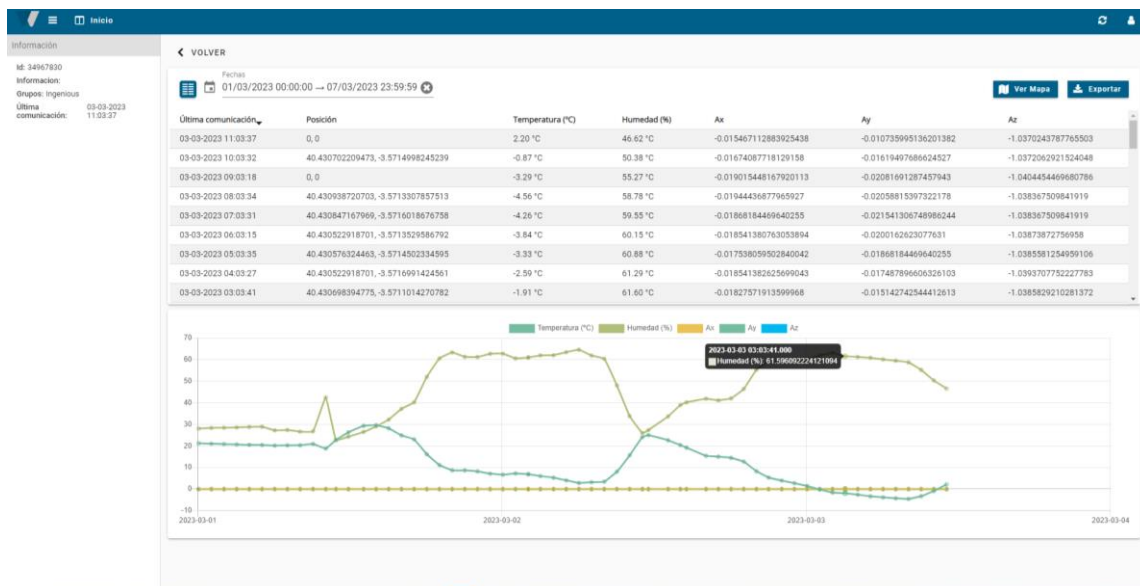


Figure 65: Temperature, humidity and accelerometer values by the sensor in Part II trip

6.3.1 TEST CASES VERIFICATION

Several Test Cases were identified in the D6.1 [1]. In this section, we present their actual results, while more information can be found in Annex V – Validation and Results.

Test Case ID	Result
UC4_TC_01: Integration and installation of sensors and communication modules on iNGENIOUS container	Passed



UC4_TC_02: Over-the-air tests for evaluating LoRa and LTE connectivity at the container in maritime and terrestrial scenarios at the Port of Valencia	Passed
UC4_TC_03: Develop an application where data gathered by IoT sensors and actuators is stored and visualized	Passed
UC4_TC_04: Container transport from the Port of Valencia to the Port of Piraeus, including storage at the Port of Piraeus until next loading	Passed
UC4_TC_05: Container transport from the Port of Piraeus to the Port of Valencia	Passed
UC4_TC_06: Terrestrial transport by truck from Port of Valencia to hinterland and vice versa	Passed
UC4_TC_07: Site Survey for exploring the practical viability of accommodating and installing the Smart IoT Gateway aboard, as well for exploring the theoretical viability of installing VSAT antenna on the vessel	Failed
UC4_TC_08: Validate proposed satellite backhaul infrastructure	Passed
UC4_TC_09: Validate end to end connectivity using Satellite backhaul	Passed
UC4_TC_10: Verify uplink and downlink Satellite backhaul capacity meets Use Case KPI requirements	Passed
UC4_TC_11: Verify uplink and downlink Satellite backhaul latency	Passed
UC4_TC_12: Validate confidentiality of satellite backhauled sensor data	Passed
UC4_TC_13: Connectivity of the Smart IoT GW with sensors	Passed
UC4_TC_14: Connectivity of the Smart IoT GW with M2M space (direct)	Passed
UC4_TC_15: Connectivity of the Smart IoT GW with M2M space (VSAT)	Passed
UC4_TC_16: Smart IoT GW will receive and process sensor data	Passed
UC4_TC_17: Smart IoT GW configuration via remote management	Passed
UC4_TC_18: Smart IoT GW will receive and process sensor data during outages	Passed
UC4_TC_19: Smart IoT GW Security	Passed
UC4_TC_20: Smart IoT GW Integration with other systems	Passed

Table 22. Ship UC Test case verification

The main aim of the test cases execution and verification was to guarantee that the core elements of the Ship use Case were able to act as initially defined and planned. According to the table above, this verification covered the following aspects:

- The verification of the functionalities of the Smart IoT GW.
- The integration of the IoT devices with the Smart IoT GW.
- The integration of the Smart IoT GW with the satellite terminal and the M2M platform.
- The verification of the satellite connectivity.
- The verification of the container transportation.
- We should also mention that the UC4_TC_07 has failed, because as we reported in the D6.2 [2], the site survey did not take place. The verification of the container transportation.

We should also mention that the UC4_TC_07 has failed, because as we reported in the D6.2 [2], the site survey did not take place. COSSP contacted Marine Operating Center Department for authorization, however due to COVID restrictions, COSSP headquarters prohibited boarding the ship if not necessary. As a mitigation action, the partners contacted a feeder service provider, that agreed to do the site survey in the feeder vessel, however due to high level of work, provider availability and timing, the site survey was finally cancelled.

6.3.2 KPIS

The KPIs of the Ship Use Case as defined in the Deliverable D2.1 [11]:

The KPIs of the Ship Use Case as defined in the Deliverable D2.1 [11]:

KPI	Test Case Reference	Target	Actual
Availability	UC4_TC_02, UC4_TC_03, UC4_TC_04, UC4_TC_05, UC4_TC_06, UC4_TC_08, UC4_TC_09, UC4_TC_13, UC4_TC_14, UC4_TC_15, UC4_TC_16, UC4_TC_18, UC4_TC_20	≥ 99.9%	≥ 99.9%
Reliability	UC4_TC_02, UC4_TC_04, UC4_TC_05, UC4_TC_06, UC4_TC_08, UC4_TC_09, UC4_TC_13, UC4_TC_14, UC4_TC_15, UC4_TC_16, UC4_TC_18, UC4_TC_20	≥ 99.9%	≥ 99.9%
Battery life	UC4_TC_01, UC4_TC_03	> 12 years	5 years
Coverage	UC4_TC_02, UC4_TC_08, UC4_TC_09, UC4_TC_13, UC4_TC_16, UC4_TC_18	GEO	GEO
Typical message size	UC4_TC_01, UC4_TC_10	200 bytes	110 bytes
Maximum message size	UC4_TC_01, UC4_TC_10	2500 bytes	250 bytes
Typical frequency (messages per day)	UC4_TC_01, UC4_TC_02, UC4_TC_04, UC4_TC_05, UC4_TC_06, UC4_TC_10, UC4_TC_16, UC4_TC_18	Maximum at every 10 minutes	Once per day during the trip from Valencia to Piraeus and vice versa. Once per 5 minutes during the real-time over-the-air demo at the Port of Valencia
Connectivity of heterogeneous IoT devices	UC4_TC_02, UC4_TC_04, UC4_TC_05, UC4_TC_06, UC4_TC_14, UC4_TC_15, UC4_TC_16, UC4_TC_18, UC4_TC_20	LoRa, Wi-Fi, Bluetooth and wired	LoRa and Wi-Fi
Latency	UC4_TC_02, UC4_TC_11, UC4_TC_16, UC4_TC_18	≤ 1 s	613 ms
Mobility	UC4_TC_02, UC4_TC_04, UC4_TC_05, UC4_TC_06, UC4_TC_14	≤ 90 km/h(truck) 45 km/h (ship)	≤ 90 km/h(truck) 45 km/h (ship)
Positioning accuracy	UC4_TC_01, UC4_TC_04, UC4_TC_05, UC4_TC_06	≤ 5 m	25 m

Table 23. Ship UC KPIs

All the KPIs defined for the Ship Use Case are intended to guarantee the set technical requirements. Such requirements include the availability and reliability of the satellite connectivity, as well as the capability the capability of the Smart IoT GW to gather and process data from heterogeneous IoT devices.

During the validation phase, no significant deviations from the target values were faced and all the KPIs were considered fulfilled.



6.3.3 IMPACT ASSESSMENT

Ship container tracking is an essential part of the supply chain and logistics to make them more efficient. Monitoring and seamlessly tracking the container in near real-time provides all the supply chain players and stakeholders full traceability and optimises the transport and storage of container goods. Any event related to a container is quickly reported and analysed and acted on e.g., alternative sourcing plans if needed.

By tracking and tracing the cargo, the operator will monitor the asset movement, record the actual routes, transit times, stationing in the facilities and congestion points for every transport mode. By analysing the transit performance, the operator can make informed decisions by choosing preferred routes, carriers or even modes of transport.

Real time monitoring of temperature, humidity, accelerometers and even simple contact sensors allow the operator to assess additional critical information for various goods in transport. Temperature and humidity are relevant for perishable goods and abnormal variations in the values will indicate to customers will not receive the goods in adequate condition or that they need the immediate maintenance to avoid the loss of goods e.g., prepare a new transport to receive goods needed on time or solve possible future disputes.

Furthermore, the accelerometer output will provide real-time indication about the integrity of the goods and the container. Similarly, abnormal variations may trigger subsequent inquiries which may conclude for example, that an accident occurred. Knowing where this accident happened helps to know the responsibility and if an intervention is required.

Continuous contact sensors data may certify that the goods are transported securely in their containers, and no unauthorized access occurred. If a door alarm event took place, the operator will alert security entities to counteract the potential illegal action.

In summary, the continuous monitoring and awareness of the goods as they pass through the various supply chain steps, from beginning to end, will allow all suppliers and consumers to have confidence in the quality and safety of the products they are supplying and consuming. Furthermore, the continuous monitoring of goods, over land and sea, will ensure early intervention if something goes wrong which could result in major cost savings for the suppliers and avoid loss of good.

6.4 Lessons Learned and Potential Improvements

Shipping companies want to track and trace containers. To do so, in this Ship Use Case, IoT devices were installed on the containers that can report location and other parameters (e.g., temperature, humidity, etc. in the container) to a central server. As containers travel in areas where there is no terrestrial coverage, satellite communication was provided to ensure that the containers are tracked when they are travelling by ship at sea or are travelling by train/truck through remote areas without terrestrial network coverage.

Furthermore, a Smart IoT GW was used which ensures efficient connectivity from heterogeneous IoT devices, by harmonizing different IoT technologies and application protocols and formatting the data to be transferred in an intelligent and efficient way across the network.

The design of the Ship Use Case, the description of each individual component and the trial results presented in the previous sections provide evidence to the significant team effort made by the partners to meet the iNGENIOUS project objectives, as well as those set forth by the specific iNGENIOUS Ship Use Case.

The over-the-air live demonstration of the Ship Use case was successfully delivered and produced new insights and useful results for the way forward. However, the use case also faced several challenges. Overall, it was learned that over-the-air live demos are very much different and more complex than lab simulations, especially in the maritime domain, for several reasons:

- Equipment installation on a ship follows very strict time schedules aligned with maintenance periods, which means that it is challenging to get the authorization and align with the ship timetables.
- A ship site survey should be planned from the very beginning of the project, as it adds important value. The site survey will explore the potential locations for the installation of the VSAT antenna system on the ship as well as will identify how the communication between the Smart IoT GW and the container will be obtained, where the Smart IoT GW will be installed, etc.
- Formal requests for getting the approval for the transmission regulatory licence of the satellite terminal at the Ports should be submitted at least four months before the over-the-air live demos.
- Direct line of sight is desirable for the communication of the Smart IoT GW with the IoT devices. That is not always available (e.g., when a container is at the bottom of a stack on a container ship) and in this case the communication is lost or is quite poor.

Regarding the IoT network several improvements were identified. For example, the Smart IoT GW is quite effective and scalable. Assuming the IoT devices inside the container should send updates every ten minutes and that a LoRa WAN session lasts less than one second, the Smart IoT GW is able to gather and process data simultaneously from around 600 IoT devices. However, some additional improvement can be researched, including:

- Implementation of additional sensor space interfaces is needed in order to be able to support different use cases. At the moment, LoRaWAN and Wi-Fi are the sensor space interfaces supported by the Smart IoT GW. However, we plan to add support for additional communication technologies, such as Bluetooth, Serial, etc.
- Improve the configurability of the Smart IoT GW. Currently, the Smart IoT GW features a basic configuration interface, allowing to manipulate its behaviour in a limited way. It is planned to extend these capabilities, allowing it to be fully configurable and controllable by an operator.
- Remote management improvements. The implemented cloud-integration of the Smart IoT GW allows for downstream commands to be sent towards the Smart IoT GW. Using these commands, remote management and action triggering can be implemented to influence the behaviour of the Smart IoT GW from anywhere in the world.



- Improved software packaging and over-the-air updates are needed. The software packaging should be improved to allow for straight-forward deployment of the whole system, for example using Debian packages. In combination with the previous point, this could also be extended to allow the installation of over-the-air updates to a deployed Smart IoT GW.

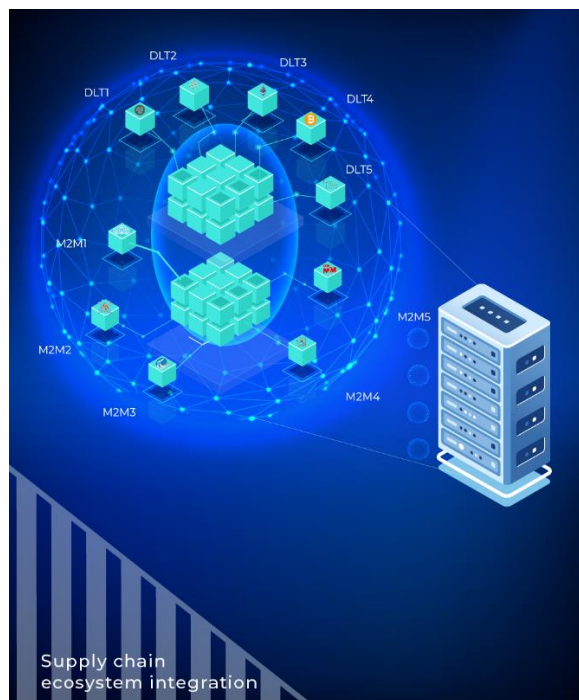
An overall learning from the project was that the continuous monitoring and analysis of the shipping container or goods requires large scale coordination and oversight which cannot be performed by one entity alone along the supply chain. Today, each area is working on improving their own tracking however it requires oversight of many different areas and technologies to meet the continuous monitoring goal that was set out by this use case. There may be an opportunity here for new entrants to coordinate and provide this service.



7 PoC - Supply Chain Ecosystem Integration

7.1 Objective and Description

Standard approaches for efficient and secure data management are still missing and the interoperability across heterogeneous machine-to-machine (M2M) platforms still needs to be tackled on a case-by-case and platform-by-platform basis due to a wide number of possible applications, design choices, formats and configurations within the IoT domain. Moreover, many of available M2M solutions have been developed in the form of application silos where the interoperability is limited by the scope of the solution. On the other hand, Distributed Ledger Technologies (DLTs) industry is completely fragmented with different alternatives: there is still a lack of consistent standardization across different available DLT solutions that do not interoperate with each other. DLT's security capabilities are not fully exploited. The DVL/DTL UC is focused on the interoperability between different M2M platforms as well as different DLT solutions.



Standard approaches for efficient and secure data management are still missing and the interoperability across heterogeneous machine-to-machine (M2M) platforms still needs to be tackled on a case-by-case and platform-by-platform basis due to a wide number of possible applications, design choices, formats and configurations within the IoT domain. Moreover, many of available M2M solutions have been developed in the form of application silos where the interoperability is limited by the scope of the solution. On the other hand, Distributed Ledger Technologies (DLTs) industry is completely fragmented with different alternatives: there is still a lack of consistent standardization across different available DLT solutions that do not interoperate with each other. DLT's security capabilities are not fully exploited. The DVL/DTL UC is focused on the interoperability between different M2M platforms as well as different DLT solutions.

The main aim of this use case is to provide two different interoperable layers in order to abstract the complexity of the underlying M2M platforms and DLT solutions, guaranteeing at the same time data privacy by means of most common pseudonymization techniques.

The main components of this use case include the Data Virtualization Layer (DVL) and the cross-DLT layer (TrustOS). The DVL allows to federate the underlying M2M platforms as well as external data sources (e.g., the Port Community System), while the TrustOS components allows to federate a set of DLTs on top.

The demonstration of this use case took place in February 2023 and it was performed by considering four different scenarios where several software components and platforms are involved, as summarized in Table 24:



Scenario Id	Description	Related UC
1	This scenario is focused on GateIn, GateOut, VesselArrival and VesselDeparture events in Livorno and Valencia seaports. Such events are retrieved from the DVL and stored in a form of TrustPoints in different DLTs through TrustOS component.	DVL/DLT UC
2	This scenario is focused on sealRemoved event in Valencia seaports. Such event is retrieved from the DVL and stored in a form of TrustPoints in different DLTs through TrustOS component.	Ship UC
3	This scenario is focused on tracking of trucks in Livorno seaport. Geolocation real-time data are retrieved from the DVL and visualized in a map through a Web Application.	Port Entrance UC
4	This scenario is focused on the pseudonymization of personal data (truck plate number) through a Pseudonymization Function integrated with the DVL.	Port Entrance UC

Table 24. Scenarios used for the demonstration of the DVL/DLT UC

A detailed description and the main architecture used for each scenario, is described in Annex VI – Objective and Description. The first demonstration of this DVL/DLT UC was performed during the mid-term review meeting in May 2022 with a limited set of functionalities.

The final part of the demonstration took place in February 2023 by relying on a remote interaction with the following software components hosted in partners' premises (e.g., staging environments and cloud-based infrastructures):

- TrustOS (owned by TIOTBD).
- DVL (owned by CNIT).
- TPCS (owned by AdSPMTS).
- Awake.AI platform (owned by Awake).
- Tracking Web Application (owned by UPV).
- M2M platforms (owned by CNIT, NXW, SES and FV).
- DLTs (owned by CNIT, TIOTBD, PJATK and FV).
- DLT Events Visualizer (owned by PJATK and TIOTBD).

7.2 Setup and Execution

According to the DVL/DLT UC's scenarios listed in the previous section, the following setup was used for the validation and demonstration of each scenario.

7.2.1 SCENARIO 1

Setup

In this scenario, the following software components were used and properly configured for the execution of the demonstration:

TPCS: an instance of the Port Community System hosted in a staging environment in Livorno and managed by CNIT. The underlying SQL Server was



not synchronized with the production environment and the datasets used for the demonstration of the scenario (for the case of the Port of Livorno) were historical ones (2021). This component provided datasets to DVL for the GateIn, GateOut, VesselArrival and VesselDeparture events' implementation in Livorno seaport.

PISystem M2M Platform: an instance of the M2M platform used by the Port of Valencia and hosted in FV facilities. This component provided datasets for the GateIn and GateOut events' implementation in Valencia seaport.

DVL: an instance of the Data Virtualization Layer (Teiid) running in a dedicated virtual machine within the Staging Farm in Livorno seaport. The platform is managed by CNIT. It retrieves data stored in TPCS and PISystem M2M Platform and exposes such data (properly aggregated and formatted in a form of GateIn, GateOut, VesselArrival and VesselDeparture events) through a RESTful API to the Integration Bridge.

Integration Bridge: a microservice hosted in TIOTBD facilities in Spain which acts as an intermediate component between DVL and TrustOS. It asks the DVL, every 60 seconds, if there are new GateIn, GateOut, VesselArrival or VesselDeparture events. If the information belongs to a new event, the event's digital asset is created in TrustOS, otherwise the existing digital asset is updated accordingly. The access to the DVL (by invoking the API of considered events) is provided with "ReadOnlyRole" enabled on DVL side, so that the entity is not able to perform any changes to the underlying datasets.

TrustOS: an instance of the Cross-DLT platform deployed in TIOTBD facilities in Spain. It allows to distribute the information of the TrustPoints among available DLTs by means of a common API.

DLTs: testnets of different DLTs the TrustOS is integrated with. These includes: Ethereum and Polygon (deployed in TIOTBD facilities in Spain), IOTA Private Tangle (deployed in CNIT facilities in Livorno), Bitcoin (both testnet and mainnet deployed in PJATK facilities in Gdansk) and Hyperledger Fabric (deployed in FV and TIOTBD facilities in Spain). The DLTs store the TrustPoint of the GateIn, GateOut, VesselArrival or VesselDeparture events.

DLT Events Visualizer: a web-based application hosted in PJATK facilities in Gdansk (dedicated server) which is integrated with TrustOS. It allows end-users to visualize the different events recorded on TrustOS and DLTs as well as to verify the TrustPoints. It provides then two main functionalities: Asset View (information representing the GateIn, GateOut, VesselArrival or VesselDeparture events) and TrustPoint View (information of a TrustPoint stored in a specific DLT).

Execution

The aim of the demonstration of this scenario is to test the integration between the DVL, TPCS, TrustOS, PISystem M2M platform and the DLTs in the context of semantic and syntactic interoperability of the heterogeneous Machine-to-Machine platforms, as defined by the Challenge 2 (advancements on security, privacy and interoperability) of the iNGENIOUS project.

Moreover, it addresses the DLT interoperability topic by relying on a cross-DLTs layer that abstracts the complexity of the underlying DLTs and serving as a

standard interface between DLT networks and the higher layers of the infrastructure.

The demonstration of this scenario consisted of several steps as described below (and depicted in the sequence diagram in Annex VI – Objective and Description).

Step 1: the Integration Bridge was able to correctly consume implemented APIs at DVL for the retrieval of the GateIn and GateOut events (for the Port of Valencia) as well as of the GateIn, GateOut, VesselArrival and VesselDeparture events (for the Port of Livorno).

The GateIn and GateOut events from the Port of Valencia were retrieved from the PISystem M2M Platform which is integrated with the DVL. All the other events were correctly retrieved from the TPCS platform used in Livorno by means of implemented RESTful APIs.

Step 2: for each retrieved event from DVL (through the Integration Bridge component), TrustOS was able to create a DigitalAsset as well as the associated Trustpoint.

The Trustpoint was then stored among integrated DLTs by means of a common API which allowed TrustOS to write and read the information in/from a given DLT.

In the Annex VI – Setup and Execution, the pictures depict both the DigitalAssets and Trustpoints for each considered event stored either on TrustOS and on the specific DLT (identified by the attribute “networkId”).

Step 3: the Integration Bridge was able to ask the DVL, every 60 seconds, if there were new GateIn, GateOut, VesselArrival or VesselDeparture events. When the information belonged to a new event, the corresponding DigitalAsset was correctly created in TrustOS.

When the information belonged to an existing event, the existing DigitalAsset was updated accordingly.

Step 4: once the TrustPoint related to a given DigitalAsset was stored both in TrustOS and in a given DLT, the DLT Events Visualizer application allowed the end-users to correctly visualize their own DigitalAssets and Trustpoints.

The Figure 66 depicts how a DigitalAsset and the associated Trustpoint were represented through the DLT Events Visualizer (assetId 001 linked to the VesselArrival event in Livorno seaport):

Transactions	
Hash	Timestamp
xbGoa+uifOM1DB/x0TsxpVGPR0LdsO7KBU5KNWv+0Ms=	1/20/1970, 9:18:56 AM
Metadata	
billOfLadingNumber	MEDUDM329279
carrierBookingNumber	
equipmentNumber	MEDU4654152
eventOccurrenceTime8601	2021-10-14T10:58:00Z
eventSubmissionTime8601	2021-10-03T11:00:00Z
location	ITLIV
originatorId	ITLIV
originatorName	Port of Livorno
terminal	LRN
transportEquipmentId	NULL
transportEquipmentRef	NULL
transportationPhase	Import
vehicleId	9141297
vehicleName	Ship
voyageId	19814

Figure 66: DLT Events Visualizer representing the DigitalAsset and the associated Trustpoint for the VesselArrival event in Livorno seaport.

7.2.2 SCENARIO 2

Setup

In this scenario, the following software components were used and properly configured for the execution of the demonstration:

IoT Sensor: physical IoT device installed on iNGENIOUS container stored in Valencia for monitoring purposes, which sends data to the Smart IoT Gateway.

Smart IoT Gateway: physical gateway which connects to the IoT sensor mounted on the iNGENIOUS container over the wireless LoRa interface as well as to the M2M space on the network/cloud side (Eclipse OM2M Platform).

Eclipse OM2M Platform: an instance of the machine-to-machine platform deployed in a cloud-based environment in Luxembourg owned by SES which allowed to store data coming from the Smart IoT Gateway.

DVL: an instance of the Data Virtualization Layer which retrieves data stored in Eclipse OM2M M2M platform and exposed such data (properly formatted in a form of sealRemoved event) through a RESTful API to the Integration Bridge.

Integration Bridge: a microservice which acts as an intermediate component between DVL and TrustOS as described in Scenario 1. In this case the sealRemoved event is considered.

TrustOS: an instance of the Cross-DLT platform, as described in Scenario 1.

DLTs: test nets of different DLTs the TrustOS is integrated with, as per Scenario 1. The DLTs store the TrustPoint of the sealRemoved event.

DLT Events Visualizer: a web-based application, as described in Scenario 1.

Execution

The aim of the demonstration of this scenario is to test the integration between the DVL, Eclipse OM2M Platform, TrustOS and DLTs. It consisted in the test steps described below (and depicted in the sequence diagram in Annex VI – Setup and Execution).

Step 1: the IoT sensor (see Figure 67) was removed from the iNGENIOUS container (to simulate the door opening event) and the data was sent to Eclipse OM2M Platform through the Smart IoT Gateway (according to in-field tests performed in Valencia in November 2022).



Figure 67: IoT device used for sealRemoved event.

Step 2: the Integration Bridge component interacted with DVL in order to check if a new sealRemoved event was available. The DVL was able then to retrieve data from the Eclipse OM2M Platform. The data was aggregated at DVL level according to a given data model so that the sealRemoved event was correctly made available to the Integration Bridge. The event was obtained by combining static and dynamic information.

Step 3: TrustOS component retrieved sealRemoved event from the DVL through the Integration Bridge and correctly created both a Digital Asset and a TrustPpoint with the same procedures described in Scenario 1.

Step 4: the TrustPpoint was then stored among the DLTs, as per Scenario 1.

Step 5: once the TrustPpoint related to a given DigitalAsset was successfully stored both in TrustOS and in a given DLT, the DLT Events Visualizer application allowed the end-users to visualize their own DigitalAsset (related to the sealRemoved event with assetId 005) and the associated TrustPpoint, as depicted in Figure 68:

Transactions	
Hash	Timestamp
h2pgLfuFEZessXd5bz1QXqDRYPnL2GkZIoLEjEwoEDM=	20/1/1970, 10:48:28
Metadata	
equipmentNumber	ZIMU1381282
eventOccurrenceTime8601	2022-04-06T10:15:43
latitude	49.70000076293945
longitude	6.340000152587891
originatorId	VLC
originatorName	Port of Valencia
sealNumber	a8a178daf4c03631
sealState	true
sealType	Carrier
smdgTerminal	CSP Iberian Valencia Terminal
Evidences	
jgQ5oirCSuYqOmamgsI7AMiKFAXbDGloathSn7labBc=	bitcoin Go to blockchain explorer
jgQ5oirCSuYqOmamgsI7AMiKFAXbDGloathSn7labBc=	iota
jgQ5oirCSuYqOmamgsI7AMiKFAXbDGloathSn7labBc=	hyperledger fabric vp
jgQ5oirCSuYqOmamgsI7AMiKFAXbDGloathSn7labBc=	goerli Go to blockchain explorer

Figure 68: DLT Events Visualizer representing the DigitalAsset and the associated Trustpoint for the sealRemoved event in Valencia seaport.

7.2.3 SCENARIO 3

Setup

In this scenario, the following software components were used and properly configured for the execution of the demonstration:

IoT Tracker: physical IoT device (Micktrack MT821) installed on a testing vehicle in Livorno seaport. It sends data to Symphony M2M platform.

Symphony M2M Platform: an instance of the M2M platform running in a dedicated virtual machine within the Staging Farm in Livorno seaport (remotely accessible through a VPN). The platform is managed by NXW and process and stores data coming from the IoT device.

DVL: an instance of the Data Virtualization Layer which retrieves data stored by Symphony M2M platform and exposes such data through a RESTful API (when the Tracking Application requests it).



Tracking Application: a web-based application hosted in a private server in Valencia and managed by UPV. It visualizes data provided by DVL by means of a GUI. The access to the DVL is provided with “ReadOnlyRole” enabled on DVL side, so that the entity is not able to perform any changes to the underlying datasets.

Execution

The aim of the demonstration of this scenario is to test the integration between the DVL and Symphony M2M Platform. It consisted of the test steps described below (and depicted in the sequence diagram in Annex VI - Setup and Execution).



Figure 69: Service vehicle in the Port of Livorno with the IoT tracking device installed on board.

Step 2: a dedicated HAL southbound plugin (Tracker SBI Plugin implemented in Symphony M2M platform) successfully received, filtered and transformed structured data from the IoT tracking sensor into the internal format supported by the HAL. A custom HAL northbound plugin (NBI Plugin implemented in Symphony M2M platform) allowed then to integrate the Symphony Data Storage by storing data collected from the underlying HAL southbound plugin by using a message broker based on RabbitMQ. The data was then exposed through a RESTful interface which is integrated with DVL.

Step 3: on one hand the DVL integrated the interface exposed by Symphony M2M platform and on the other hand it exposed a RESTful interface which was used by a Tracking Application to consume datasets coming from the IoT tracking device. This interface was integrated with the Tracking Application which correctly performed requests to DVL to retrieve such data.

Step 4: the DVL retrieved GPS data from the Symphony M2M platform by performing a data mapping according to the Tracking Application requirements. The picture included in Annex IV – Setup and Execution, depicts the structure of data available at DVL level by using Postman tool to perform the POST request.

Step 5: the data was correctly made available to the Tracking Application as a result of the HTTP request.

Step 6: the Tracking Application provided a GUI for a graphical representation of the main path undertaken by the service vehicle with the tracking device on board within the Port of Livorno area, as shown in Figure 70:

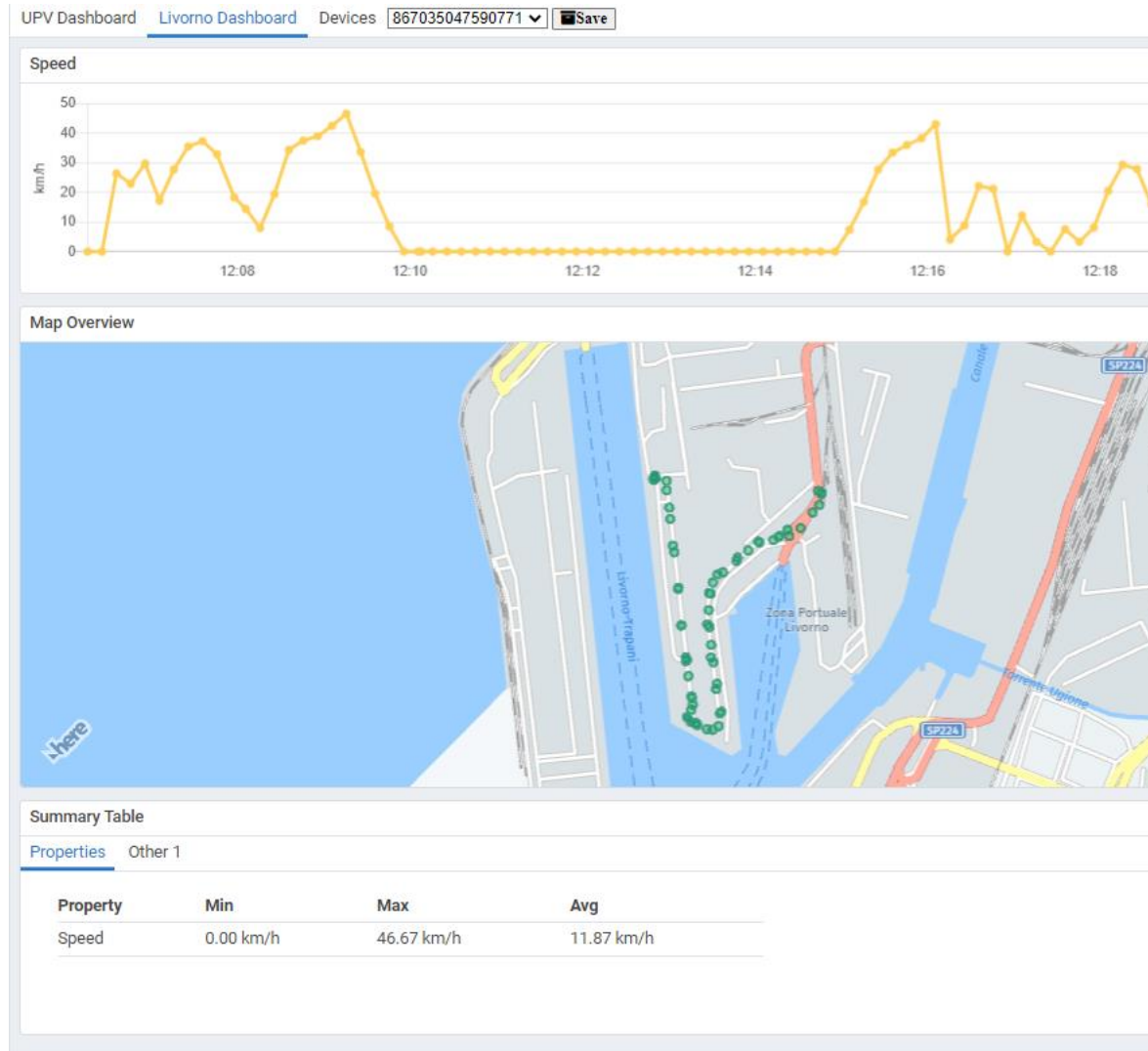


Figure 70: Tracking Application - Livorno Dashboard.

7.2.4 SCENARIO 4

Setup

In this scenario, the following software components were used and properly configured for the execution of the demonstration:

TPCS: an instance of the Port Community System which provides datasets to DVL for the GateIn and GateOut events occurring in Livorno seaport.

Mobius OneM2M Platform: an instance of the Machine-to-Machine platform hosted and running in a staging environment in Livorno seaport, managed by CNIT. The platform is responsible for collecting data coming from the IoT

devices installed within the seaport. In the context of this scenario, the platform stores the meteorological data coming from two distributed monitoring stations within the seaport.

DVL: an instance of the Data Virtualization Layer which retrieves data stored by Mobius OneM2M and TPCS platforms and exposes such data through a RESTful API (when requested by the Predictive Analytics Application).

Pseudonymization Module: a microservice-based application hosted in a dedicated virtual machine (managed by TEI) accessible only via VPN in a staging environment in Livorno seaport. The microservice allows to retrieve GateIn and GateOut events from DVL (for the Port of Livorno), pseudonymize and store them by using available pseudonymization techniques and expose a pseudonymized dataset to DVL by a RESTful interface so that external applications may consume such datasets through the DVL component. It also provides interfaces for the management of all stored datasets (e.g., deleting of the pseudonyms which retention period has expired). The access to the DVL (by invoking the API of considered events) is provided with “ReadOnlyRole” enabled on DVL side, so that the entity is not able to perform any changes to the underlying datasets.

Predictive Analytics Application: a cloud-based application managed by Awake for performing predictive analysis in the scope of the Port Entrance use case for both the Port of Valencia and the Port of Livorno. Further details are given in the Chapter 6 of this document. The access to the DVL (by invoking the API of considered events) is provided with “ReadOnlyRole” enabled on DVL side, so that the entity is not able to perform any changes to the underlying datasets.

Execution

The aim of the demonstration of this scenario is to test the integration between the DVL and Mobius OneM2M Platform. In addition, the demonstration aims at extending the DVL’s capabilities by providing a Pseudonymization Module with a pseudonymization functionality for personal data management (e.g., truck’s plate number) in line with GDPR requirements. Pseudonymized data can be then used by third-party applications for analytics and/or analysis purposes by guaranteeing data confidentiality.

The demonstration of this scenario consisted of several steps as described below (and depicted in the sequence diagram in Annex VI – Setup and Execution):

Step 1: using two distinct RESTful interfaces implemented in the DVL (HistoricalGateInEvent() and HistoricalGateOutEvent()) the Pseudonymization Module was able to read once per day all historical GateIn and GateOut events (in Livorno seaport) by considering a time window set to 24h for the demonstration scope.

Step 2: the Pseudonymisation Module successfully applied the selected pseudonymization algorithm (Hashing Without Key - HWK) on the fetched personal data in GateIn/GateOut events (truck’s plate number attribute), before storing them within an encrypted database. The Figure 129 and Figure 130 depict the above-mentioned interactions.

Step 3: the pseudonymized events were then retrieved through the DVL (which exposes an additional RESTful interface called FetchGateEvents()), by the

Predictive Analytics Application, and were used for the AI-based algorithms training which are part of the Port Entrance UC (further details and the main outcomes of this scenario are given in the Demo – Situational Understanding in Smart Logistics Scenario of this document).

7.2.5 ISSUES ON EXECUTION

During the DVL/DLT use case demonstration, no significant issues occurred. Nevertheless, the main issues faced during the preparation of the execution were based on technical aspects, and they have been properly addressed through appropriate development activities. Such issues are briefly summarized in 108.

Description of the issue	Mitigation measures
Lack of a mechanism to keep TrustOS synchronized when new events are available in DVL (both are passive components)	Implementation of an intermediate layer (namely Integration Bridge) which allows checking whether new events are available in DVL.
DVL does not support the response format (XML) of the Eclipse OM2M platform API	Implementation of an additional service within the DVL which allows parsing the API's response correctly.
A token-based authentication is not natively supported by the DVL to interact with the PISystem APIs	Implemented the authentication procedure within the VDB (which defines the API to be integrated with TrustOS) to interact with PISystem.
Unsupported communication protocol between the tracking IoT sensor and Symphony M2M Platform	Development of a dedicated southbound plugin for proper hardware abstraction in Symphony M2M platform.

Table 25. DVL/DLT UC Issue on execution.

7.3 Validation and Results

In this chapter, the main results and outcomes of the DVL/DLT UC are provided. The verification process was based on the validation of the defined test cases which allowed fulfilling the user and the system requirements respectively. Moreover, the KPIs' assessment is described according to the main expectations initially set for the use case. Finally, the impact assessment of the solution as well as potential improvements to be done in the future are briefly presented and discussed.

7.3.1 TEST CASES VERIFICATION

In the context of the DVL/DLT UC, the following test cases were performed so that both the user and system requirements were fulfilled as defined and described in D2.1 [1]. The test cases were initially defined in D6.1 [1] while the description of all related development and integration activities with technical details are reported in D6.2 [2]. The following table provides a list of all test cases for this use case that were performed and verified (further details are included in Annex VI – Validation and Results):



Test Case ID	Result
UC6_TC_01 - Interaction between OneM2M platform and Data Virtualization Layer	Passed
UC6_TC_02 - Interaction between OM2M platform and Data Virtualization Layer	Passed
UC6_TC_03 - Interaction between PISystem platform and Data Virtualization Layer	Passed
UC6_TC_04 - Interaction between DVL, Integration Bridge, TrustOS and the set of DLT providers	Passed
UC6_TC_05 - Mapping of the access roles for Data Virtualization Layer consumers	Passed
UC6_TC_06 - All personal data received by Data Virtualization Layer has to be pseudonymized	Passed
UC6_TC_07 - DVL (authorized entity) can fetch data, in pseudonymized format, from PF module	Passed
UC6_TC_08 - Personal Data storage	Passed
UC6_TC_09 - Data Owner can request to the platform to cancel personal data	Passed
UC6_TC_10 - Views and query results caching capability	Passed
UC6_TC_11 - Interaction between TPCS and Data Virtualization Layer	Passed
UC6_TC_12 - Integration between Symphony M2M Platform and Data Virtualization Layer	Passed

Table 26. DVL/DLT UC Test case verification.

The main aim of the test cases execution and verification was to guarantee that the core elements of the iNGENIOUS Interoperable Layer (namely DVL and TrustOS) were able to act as initially defined and planned in the scope of this use case. According to the table above, this verification covered the following aspects:

- The integration between the DVL and the underlying M2M platforms as well as data sources for data aggregation;
- The integration between the DVL and TrustOS by means of an Integration Bridge for maritime events retrieval;
- The integration between the TrustOS and the DLTs on top for maritime events storage;
- The integration between the DVL and a Pseudonymization Module for personal data pseudonymization based on HWK technique.

7.3.2 KPIS

In this section, the list of identified KPIs is provided for the DVL/DLT use case. Each KPI is further described in KPIs with additional technical details used for their assessment. For each KPI, the related testing activities are also reported in a form of test cases. In addition, the target values (the ones set at the beginning of the project) are benchmarked against the actual values resulted from the validation and demonstration of the DVL/DLT use case. The Table 27 provides a summary of such comparison.

KPI	Test Case Reference	Target	Actual
-----	---------------------	--------	--------



Data Virtualization Layer scalability	UC6_TC_01 UC6_TC_02 UC6_TC_03 UC6_TC_11 UC_TC_12	≥5 heterogeneous and simultaneous M2M platforms as data sources	4
Data Virtualization Layer data processing	UC6_TC_01 UC6_TC_02 UC6_TC_03 UC6_TC_10 UC6_TC_11 UC6_TC_12	Real-time	Real-time
Data Virtualization Layer access control	UC6_TC_05	Role-based access control	RBAC
Cross-DLT layer access control	UC6_TC_04	Role-based access control	Dedicated TrustOS identity for iNGENIOUS
Cross-DLT layer scalability	UC6_TC_04	At least 4 simultaneous DLT technologies	Integration with 8 simultaneous DLT providers
Availability of the DLT connectivity layer	UC6_TC_04	The DLT connectivity layer should be highly available	High availability (8x5 environment)
Data processing time in DLTs	UC6_TC_04	Each request for the given DLT should be processed in less than one minute	Less than 30 sec
Cross-DLT concurrent requests	UC6_TC_04	At least 4 concurrent requests	8 concurrent requests
Confidentiality and integrity protection of personal data	UC6_TC_06 UC6_TC_07 UC6_TC_08 UC6_TC_09	100%	100%
Logs of privacy events	UC6_TC_06 UC6_TC_07 UC6_TC_08 UC6_TC_09	100%	100%

Table 27. DVL/DLT UC KPIs.

All the KPIs defined for the DVL/DLT use case are intended to guarantee technical requirements set for the proposed solution (namely iNGENIOUS Interoperability Layer). Such requirements include the scalability and availability of both the DVL and TrustOS (Cross-DLT layer) components, as well as the capability to properly manage the access to all considered data sets by addressing data privacy and confidentiality aspects.

Originally, it was planned to have at least five heterogeneous M2M platforms among all iNGENIOUS use cases, but one of them (namely NB-IoT Platform) was never used as M2M platform in the context of the project. Nevertheless, this deviation did not impact the overall scalability of the DVL component, which was additionally integrated with an external data source (namely TPCS).

During the validation phase, no significant deviations from the target values were faced and all the KPIs were considered fulfilled.



7.3.3 IMPACT ASSESSMENT

First of all, the DVL/DLT use case allowed to tackle the lack of standardization, which is one of the issues that are currently hindering massive consumer uptake of IoT technologies. This was achieved by providing a new approach for the interoperability based on the federation of different IoT platforms within heterogeneous domains, overcoming the compatibility issues between both standard and non-standard, proprietary and custom M2M solutions. Secondly, the use case addressed the abstraction of different available DLTs, going beyond the distinction between public and private DLTs, by providing a common layer for their interoperability within a heterogeneous environment and ensuring an immutable data storage as well as removing (or reducing) the need of the third trusted party that holds records of events (during the lifetime of the project, the most important aspect of the DLTs in the context of supply chain was data immutability and accountability rather than smart-contract compatibility). By means of this approach, the organizations and companies could spend less on building and managing data integration processes for connecting distributed data sources, benefiting in terms of costs and time savings by quickly validating new business models using an agile approach to data integration. In addition, the lack of interoperability between independent blockchain-based systems and use cases, is preventing DLTs from being applied in large industrial ecosystems and unleashing their full potential and benefits. The implementation of a blockchain and DLT interoperability leads to a significant breakthrough towards global blockchain use cases and systems. Instead of being forced to deploy the technology for corporate business cases with a small number of participants, iNGENIOUS interoperable layer enables the exchange of data and the orchestration between different use cases. This allows the transformation of limited use cases into global ones, with a corresponding business impact. Finally, iNGENIOUS interoperability layer enables the communication and exchange of data that allows users and companies with a way of governing their data in every network, fulfilling one of the promises of blockchain technology and decentralize identities that is to return the control of their data to users.

7.4 Lessons Learned and Potential Improvements

During the demonstration of the DVL/DLT UC and based on the requirements as well as constraints of the implementation approach that was adopted, the following aspects were identified in relation to further improvement of the considered solution (namely Interoperability Layer): i) TrustOS and DVL are implemented as a single access point for both the underlying data sources (e.g., M2M platforms) and DLTs on top (namely Bitcoin, IOTA, Hyperledger Fabric and Ethereum). This approach may, in principle, lead to a single-point-of-failure issue. In order to further improve the proposed solution, a distributed approach may be used: more than one instance of both software components can be deployed and synchronized so that in case of a failure, another instance can be

used without compromising the availability of the interoperability layer; ii) Four different scenarios have been used for the validation of the DVL/DLT use case. Only two of such scenarios (Scenario 2 and Scenario 3) were validated using real-time data due to technical constraints. This slightly limited the validation process of both TrustOS and DVL components in a real environment with more realistic conditions. In order to further test and validate the proposed solution, more than two data sources, with real-time capabilities, would be beneficial; iii) Due to the project's constraints, only five different DLTs have been used to validate the DVL/DLT use case. In a real context, the supply chain ecosystem (at least in the maritime context) includes a lot of actors: terminal operators, maritime agencies, freight forwarders, carriers, institutional bodies, etc. Considering a real scenario, the proposed solution may be tested by involving a wider range of actors and assuming each of them relies on a different DLT solution for their own business.



8 Additional Research Activities – Satellite Direct Access

The purpose of this section is to capture additional notable research carried out during the project which was outside the scope of the selected use cases. The exploratory research was always planned from a project perspective but not directly impacting the uses cases defined.

One such activity was the research carried out by iDR in the area of direct access of IoT devices over satellite. Satellite technology was used in the Transport and Ship use cases with focus on using satellite to backhaul the IoT information between the IoT gateway and the cloud. However, satellite direct access for IoT devices is where IoT devices connect directly to the satellite network, which in turn connects to the IoT cloud/data center. This allows delivery of the IoT content in a more efficient and cost-effective manner by utilizing a Direct-to-Satellite approach rather than using the satellite to backhaul IoT traffic. This section describes the setup, testing and results from the satellite direct access activities.

8.1 Objective and Description

The objective of this activity was to research satellite direct access concepts. Direct access of IoT devices over satellite can be categorized in the following three areas.

- **Non-3GPP IoT access** - Direct access of IoT devices over satellite using proprietary non-3GPP access technologies is already supported by many industry partners today. Within iNGENIOUS, iDR researched the use of their own proprietary access technologies to determine if they were suitable for connecting IoT devices over satellite.
- **3GPP 5G NR-NTN** - 5G New Radio Non-Terrestrial Network support is a new feature added in 3GPP Release 17. This offers the capability to use a standard 5G NR waveform over satellite links. This could offer new opportunities for both the direct access and satellite backhaul use cases.
- **3GPP NB-IoT NTN** - 3GPP have also made changes to NB-IoT and LTE-M to support NTN. These changes were studied in Release 16 and included in Release 17. In general, the changes are similar to the changes outlined for 5G NR-NTN but tailored for NB-IoT.

There was no use case requirement for direct access over satellite solutions within the iNGENIOUS project, however, within this project research was carried out on all three areas mentioned above. Details of the iNGENIOUS research on 3GPP 5G NR-NTN and 3GPP NB-IoT NTN are included in D3.2 [12].

The following sections contain details of Non-3GPP IoT access (direct access) research.



8.2 Setup and Execution

The satellite direct access research and development work can be further categorized into two main areas:

- Transmission of IoT data over satellite.
- Satellite radio channel characterization.

8.2.1 TRANSMISSION OF IOT DATA OVER SATELLITE

The purpose of this research was to investigate and demonstrate the possibility of generating a robust IoT data transmission message that can be encoded, transmitted and received over a satellite link and decoded at the other side by an existing iDR satellite system before forwarding to the IoT cloud. This would allow existing satellite remote terminals to receive IoT data from sensors and transmit over satellite without having to make any hardware changes to the existing satellite nodes. Importantly the IoT data transmission does not require the setup of an end-to-end data session but uses control plane messaging to transmit the information.

The research work carried out is summarized below.

Researching IoT payload transmission over GEO satellite network

This research included the enhancement of the existing commercial satellite system to add capability to transmit and receive an IoT payload. This required modifications to the access procedure and control plane on the existing VSAT terminal and satellite hub.

Building Sensor Network in iDR lab testbed

To test and verify the end-to-end IoT data transmission, a lab testbed was setup which comprised of three IoT nodes (with seven temperature sensors in total) and a network edge MEC node at the IoT sensor side. The core network side consisted of a satellite hub for IoT processing and the cloud endpoint which stored and displayed the received IoT information. An example of the IoT devices can be seen in Figure 71.



Figure 71: Heat sensors installed in iDR lab in Killarney.

IoT Data Payload Optimisation

Once the initial investigations were complete and the test lab was setup, the next task was to work on the sensor data encoding at the IoT network edge to encode the data securely and robustly for transmission over the satellite link. Once the data was encoded at the network edge it was forwarded onto the satellite modem for encapsulation and transmission. On the satellite hub side, the IoT data transmission had to be received, detected and decoded before being forwarded onto the IoT cloud for storage and analysis.

Satellite Network Validation

The final part of this research was testing in the lab testbed and over the live satellite network. This was done in the lab by setting up a GEO satellite network using satellite channel emulators and in the live network using SES' GEO satellite connectivity over Astra 2F. In both cases the remote terminal and satellite hub needed to be upgraded to support the transmitting and receiving of the IoT data.

Figure 72 below provides an overview of the lab testbed and live over the air setups both of which were used to test and validate the IoT data transmission. The same IoT sensors and MEC nodes were used for collecting, concatenating and encoding the IoT data. Using the same edge network nodes allowed for seamless switching between the lab testbed and live network when capacity was made available. Another useful element to the lab setup was the ability to capture and replay IoT burst information which is highlighted in red in Figure 72.

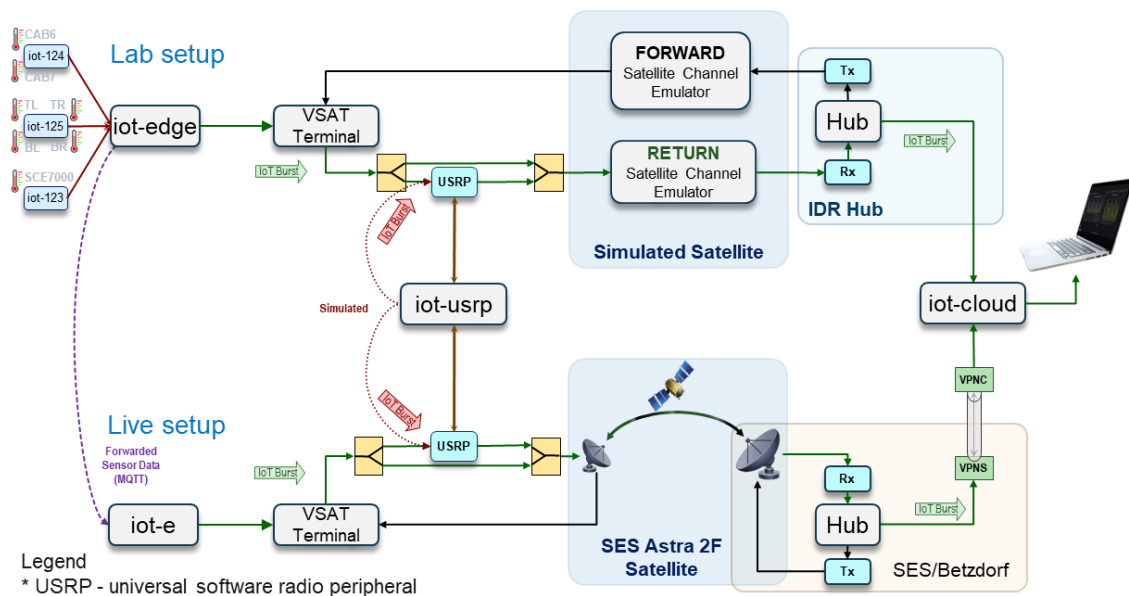


Figure 72: Transmission of IoT data over satellite lab and live testbed setups

One important characteristic of how the direct access IoT over satellite network operates is that the IoT data is sent over the satellite link from the VSAT satellite terminal to the satellite hub without ever creating a data session over the satellite. The information is sent using control signaling messages that are typically used for requesting access to the satellite network prior to a data session being setup. This means that no data session is required to pass the IoT



data over the satellite network which allows for a more efficient use of resources.

On the IoT Cloud side, the initial integration was done with the Microsoft Azure IoT cloud system which was used to store, analyze and display the communicated IoT information. An example of the Azure IoT dashboard can be seen in Figure 73.

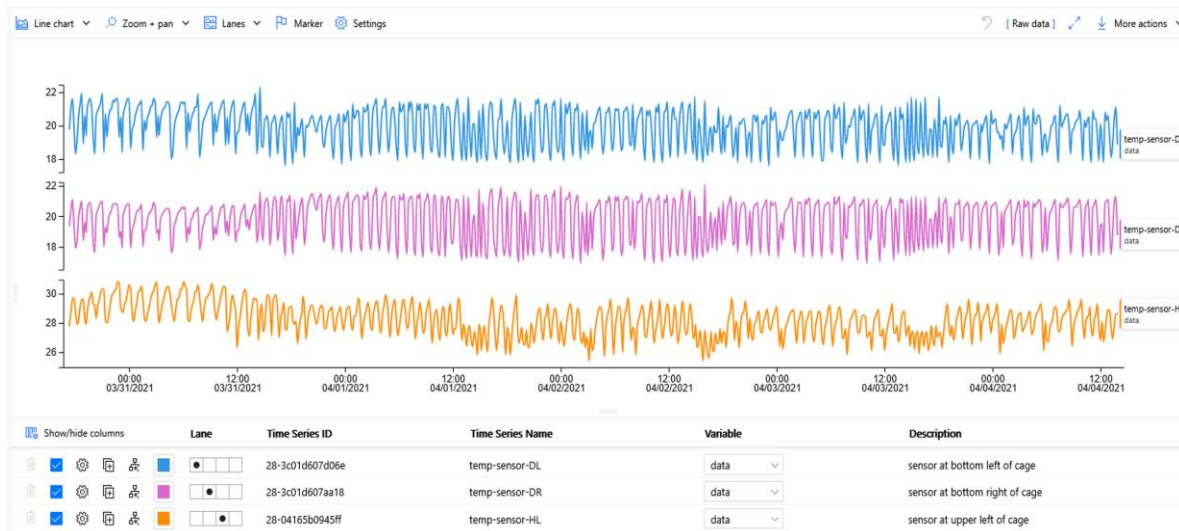


Figure 73: Microsoft Azure IoT cloud dashboard showing IoT information.

Maintaining the Microsoft Azure IoT cloud service was proving costly, so it was decided to move to an in-house IoT cloud system based on Grafana coupled with an influxDB timeseries databases to provide data storage and visualization. An example screen shot can be seen in Figure 74 below.

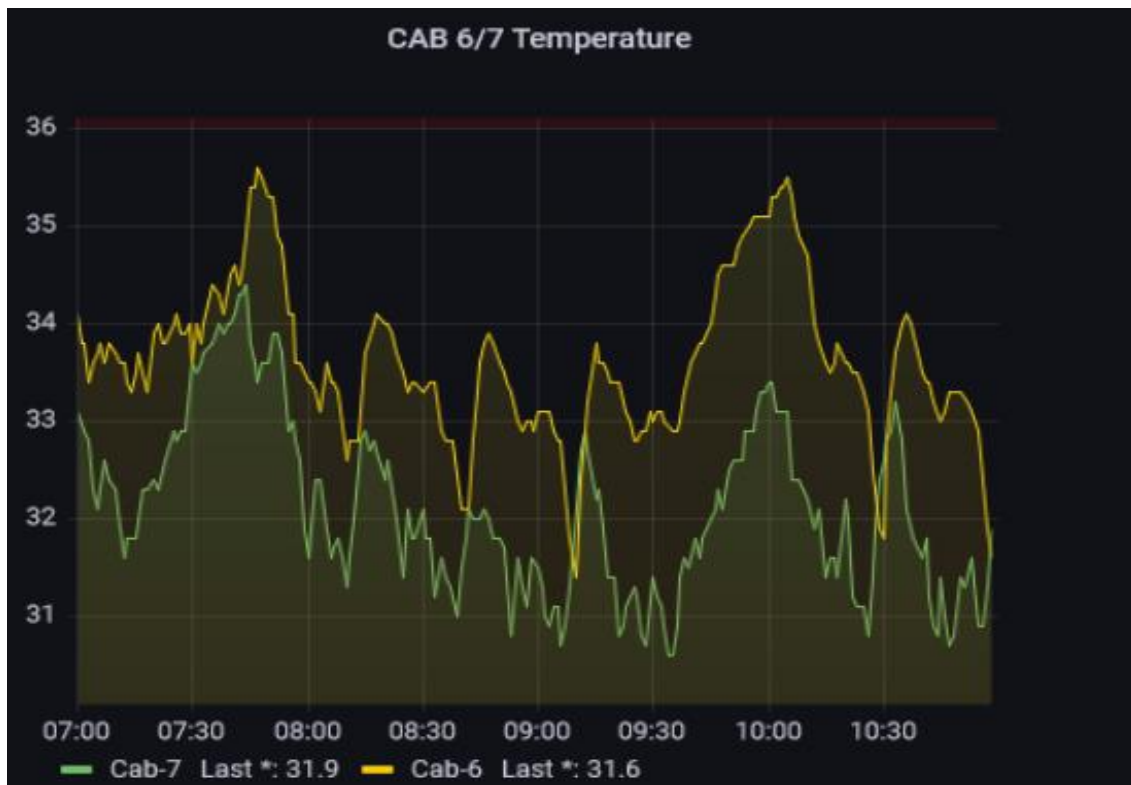


Figure 74: Example of in-house IoT cloud dashboard, based on Grafana, showing IoT information.

8.2.2 SATELLITE CHANNEL CHARACTERIZATION

The purpose of this research was to identify the lowest possible signal to noise ratio (SNR) that an IoT device could successfully send and receive IoT data over an existing iDR satellite network using the direct access setup described earlier. This would be a major consideration when determining the type of IoT devices and configuration that can be supported on the existing satellite network for connecting IoT devices using direct access over satellite connectivity.

The research included identifying and analyzing the power and SNR limits for IoT data transmission for various configurations. This was performed using the lab test setup shown in Figure 72. The return link (link from the remote terminal/IoT device to the satellite hub) satellite emulator was used to reduce the SNR step by step until the IoT data was not received by the satellite hub any longer. At each step the SNR was recorded and several IoT data messages were sent to the satellite hub. If the information was received successfully the SNR was dropped further until the lower limit was reached.

8.3 Validation and Results

8.3.1 TEST CASES VERIFICATION

Transmission of IoT data over satellite

Successful transmission of IoT data over satellite using direct access was validated and confirmed by the IoT sensors data (temperature sensor data) being received by the Grafana system and displayed correctly on the Grafana user interface. An illustration of this can be seen in Figure 75 below which shows the temperature of the iDR lab in Killarney where the sensors were installed (see Figure 73).

The correct capture, transmission and display of the IoT sensor information on the Grafana system confirmed the following steps were completed correctly (see Figure 72 for system overview diagram):

1. IoT Sensor information was received correctly by the MEC server.
2. MEC server correctly encoded the IoT information in order forward the data over the satellite link.
3. MEC server forwards the encoded IoT information to the satellite modem.
4. Satellite terminal receives IoT data from the MEC server and transmits it over the satellite within an IoT burst.
5. IoT burst containing IoT data is received on the satellite hub and extracted successfully.
6. IoT data is decoded correctly and passed to influxDB for storage. It is then retrieved by Grafana for analysis and display.
7. IoT information is displayed correctly on Grafana system.

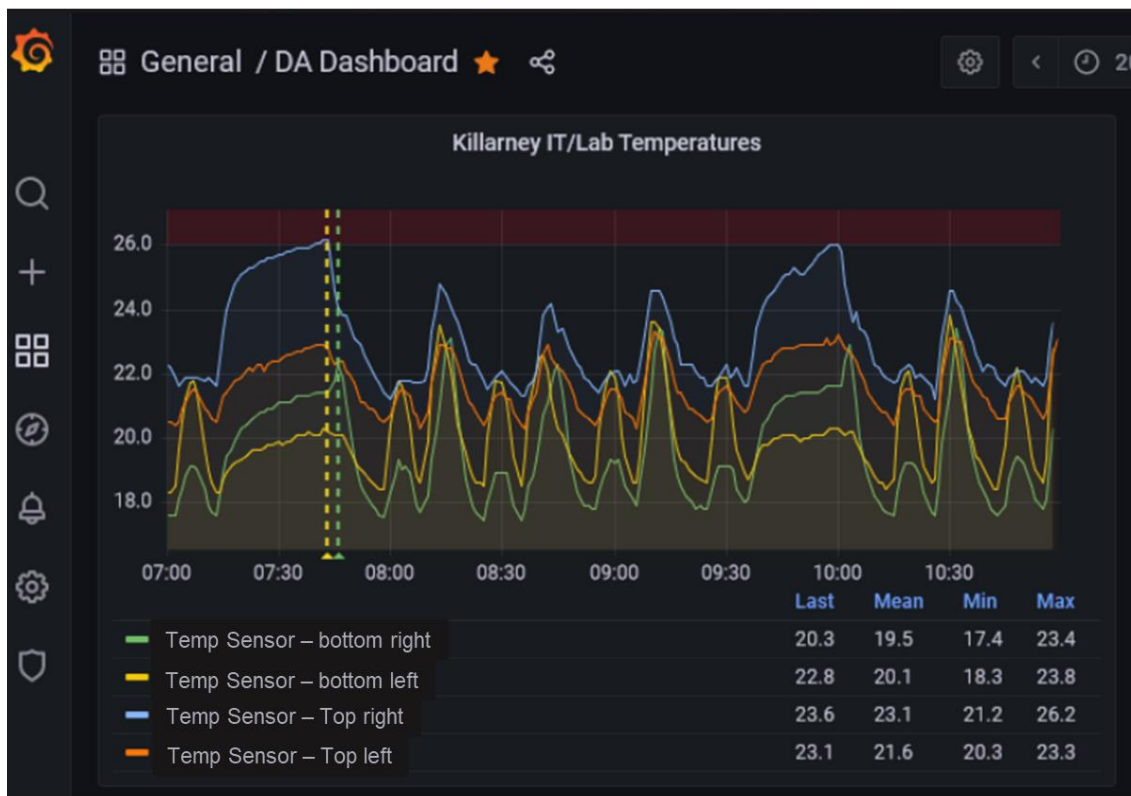


Figure 75: Example of in-house IoT cloud dashboard, based on Grafana, showing IoT information.

Satellite channel characterization

The satellite channel characterization testing required detailed setup and testing of the satellite system at different power and noise levels to identify the lowest possible SNR where an IoT burst could still be successfully received by the satellite hub. The testing was performed on the modified lab and live networks and a summary of the results are provided in Table 28 below.

IoT Config	iDR Lab dB (SNR)	SES Astra2F Live dB (SNR)
Narrowband + min bandwidth	-6.01	-5.71
Narrowband + min-low bandwidth	-5.81	-5.71
Narrowband + mid bandwidth	-5.96	-5.82
Spread spectrum + min-low bandwidth	-15.06	-10.12
Spread spectrum + mid bandwidth	-15.73	-13.32

Table 28. Satellite channel characterization SNR values

The table shows the lowest recorded SNR value where an IoT burst was received by the satellite hub. The SNR values recorded for the narrowband setup were similar for both the lab and live systems which is a good indication that the test setup was representative of the live network. The satellite remote terminal and hub equipment used were similar for both, so this was expected.

For the spread spectrum setup, the difference was larger and it may be possible that the environmental conditions may have been a bigger factor here. Further investigation is required to determine such difference.

The number of successful IoT transmission received at the various SNR levels was seen to reduce as the SNR level reduced which is also expected. However, it proved difficult to get a consistent measurement of the success rate at the



lower SNR values. For the latter, it should be noted that the setup was a standard satellite network VSAT deployment that was not tuned for low SNR or IoT data transmission which was the focus of the testing in this case. Further research and investigations are required on what the SNR level may be if a different waveform was used over the satellite e.g. LoRa or NB-IoT.

8.3.2 IMPACT ASSESSMENT

The ability for an existing satellite system to successfully transmit IoT data over satellite using the existing control plane mechanisms and without having to setup a dedicated radio bearer (end-to-end data session) is a powerful capability that has been identified and tested with this research. It allows the transmission of IoT data very efficiently as there is no resource overhead required to set up a dedicated radio bearer. It means there is potential for the existing satellite system to immediately become an IoT Sensor network without any major hardware upgrades apart from deploying the IoT sensors and connecting to the satellite terminal.

It was also shown that the IoT data transmission is robust and can be transmitted and received at low SNR values which is an important aspect of any IoT sensor network where the IoT sensors may be located in very remote areas. The ability to successfully transmit and receive data at low SNR values also means that there is very little interference with the wider satellite network which is an important network planning consideration.

The results of this research will provide valuable input to the existing product deployment capabilities and future planning for IoT solutions over satellite.

8.4 Lessons Learned and Potential Improvements

The satellite channel characterization testing took a large effort to setup, test and analyse the large amount of generated data. This is an area that could be explored further and is so large that it could warrant a dedicated project.

The live over-the-air testing was very useful and confirmed the lab testing results. In the future more time could be spent on this testing, especially to test with different hardware configurations which would be more representative of the IoT sensors network connecting directly to satellite.

There is also potential for improvement on the SNR levels achieved for successful reception of the IoT burst information. As mentioned above, the test setup used was not modified or tuned for IoT data transmission apart from the addition of the IoT burst encoding and detection feature which was added at the application level. Improvements could be made on the existing waveform to reach lower SNR levels and improve the success rate of received IoT bursts for a given SNR. Also investigating the use of other waveforms over satellite (e.g. LoRa, NB-IoT) could lead to better performance as these waveforms are designed for low SNR IoT data transmission.

9 Conclusion

The document has described the activities related to measurement campaigns and trials developed in the PoCs and Demos and their validation against KPIs and requirements defined in WP2 and gathered in D2.1 [11]. For each PoC and Demo, the objectives of the demonstration have been defined as well as the set-up and execution activities. Results and test cases validation have been described and KPIs calculated and accordingly compared with the target values pursued and defined in D2.1 [11]. The trials conducted in the use cases allowed to identify main lessons learned and potential improvements that can be summarized as follows.

The **Factory UC** focuses on cooperative automated robots for future smart factory production lines or warehouses. Within the iNGENIOUS project it has been demonstrated how wireless communications systems based on 3GPP standards are able to provide services for industrial scenarios. The advantages highlighted are related to cost-saving and benefits of virtualization, that allow to improve efficiency, flexibility and quality of the supply chain and production processes handled by robots, AGVs, transport vehicles and people. All of them could be equipped with devices, capturing real-time data on temperature, humidity, noise, presence of particles in the air, etc. In addition, all of this data can be used to train predictive maintenance system before any anomalies occur. The results obtained in the tests and trials of this use case allowed to identify possible improvements in the protocol for sensor data sending sensor by using an event-based sampling approach, in order to send information only when an event is detected. Additionally, the 5G LAN use has been identified to bring added value when creating private groups of devices to be connected within the industrial network. The integration, testing and demonstration activities have shown the importance of the availability of well-defined and accurate management and control APIs for the support of full automation in service and slice deployment and operation. This has been identified as a crucial aspect and lesson learned, especially when software and hardware components are provided by different vendors or institutions in general.

The **Transport UC** focuses on safe and secure micro edge sensors for monitoring wheels and axles of cargo train carriages. Within the iNGENIOUS project, it demonstrated eight improvement areas when compared with conventional IoT sensors, such as energy harvesting, data cloud and secure data authorized, etc. One of the key driving factors of the Transport UC is the optimized and efficient communication energy paths for sensors tested, that always run on batteries or energy-starved harvesters. Along the development of these components, it has been detected its possible further usage with the corresponding updates for joint medical edge applications. The usage of remote attestation to ensure security and confidentiality for this future application is under discussion and further cooperation with partners involved is almost ensured. One of the most important learnings for the future applicability of these developments is the need of clear legislations for accident and rail track damage prevention as well as penalties on damaged infrastructure due to poor maintenance. Furthermore, it is of primary importance the cooperation among many relevant stakeholders to share not only benefits from the applied innovative technology but also implementation cost.



The **Port Entrance UC** focuses on enhancing the situational understanding of events in maritime ports by ingesting multiple data. Within the iNGENIOUS project, it has been demonstrated how prediction capabilities enable to optimize truck turnaround times (TTT), therefore minimizing truck congestion, consumption, noise, emissions and time waste. The use case demonstrated how the inclusion of eco-efficient management criteria is increasing in modern port management and market competition. One of the most important learnings obtained is the need of historical datasets availability and updated data integration allowing to feed machine learning models. The test results with historical data indicated that having access to some additional features already existing in port systems and to real-time data improves the prediction accuracy. In fact, the long-term prediction was demonstrated with minimal data available and additional inputs such as temporary high congestion were not taken into account during the testing. These aspects represent the most important challenges to improve the current use case models and are considered to be performed prolonging the cooperation among the involved partners after the project's end. Such cooperation ensures the viability of a future integration of the iNGENIOUS work into real operation processes in smart ports.

The **AGV UC** focuses on improving the driver's safety by combining the use of mixed reality and haptic solutions for controlling AGVs (autonomous vehicles) in a real scenario, to solve the problem with the actual autonomous vehicles without remote driver control. Within iNGENIOUS, an innovative Tele-operative Driving has been demonstrated ensuring a proper connectivity for the AGV, validating the haptic gloves, and developing digital twin for improving the remote cockpit. The safe driving is possible with the established components if the network performance is optimal, under the limit of the latency established. It has been observed that both tested gloves work well, but completely natural feeling is not yet achieved. Another observed aspect to be improved for the technology behind is the immersion experienced by the teleoperator and therefore safety, exploring the use of new cameras and their overlapping in the cockpit. For the 5G network administration, in order to obtain a better distribution of network traffic, it will be necessary to separate the uplink and downlink traffic into different slices, as well as to assign different priorities depending on the needs.

The **Ship UC** focuses on providing end-to-end (E2E) container tracking via IoT devices, Smart IoT gateway and satellite technology. Within iNGENIOUS project, it has been demonstrated how shipment information is available across all connected platforms and interested parties in real-time. The live demonstration of the Ship UC produced new insights and useful results on container track and tracing, reporting location and other parameters (e.g., temperature, humidity, etc. in the container) to a central server. It was learned that live demos are very much different and more complex than lab simulations, especially in the maritime domain, due to the different and several protocols, regulations and restrictions to be fulfilled. Regarding the IoT network several improvements were identified, such as implementation of additional sensor space interface, improvement in configurability and remote management. As overall learning, the continuous monitoring and analysis of the shipping container or goods requires large scale coordination and oversight among multiple entities along the supply chain.

The **DVL/DTL UC** focuses on the interoperability between different M2M platforms as well as different DLT solutions for efficient and secure data management. Within iNGENIOUS, it has been demonstrated a novel approach to ensure an immutable data storage and privacy capabilities. This was achieved by providing federation of different IoT platforms within heterogeneous domains and common interoperability layer within a heterogeneous environment. The interoperability layer enables the communication and exchange of data that allows users and companies to govern their data in every network, fulfilling one of the premises of blockchain technology and decentralized identities: to return the control of data to users. During the demonstration, improvement on the interoperability layer has been identified in order to avoid a single-point-of failure issue and use a distributed approach that allows the applications and users to switch to another instance in case of failure, without compromising the interoperability layer. Moreover, in order to further test and validate the proposed solution, more than five data sources, with real-time capabilities, would be beneficial. Considering a real scenario, where lots of actors are involved in the supply chain ecosystem, the proposed solution can be tested by considering a wider range of actors and assuming each of them relies on a different DLT solution for their own business.

Additional research activities have been carried out during the project, satellite direct access concepts of IoT devices, to demonstrate the ability to transmit IoT data over satellite using the existing control plane mechanisms and without having to setup a dedicated radio bearer. This powerful capacity allows a very efficiently transmission of IoT data without actual resource overhead required. There is potential for the existing satellite system to immediately become an IoT Sensor network without any major hardware upgrades apart from deploying the IoT sensors and connecting to the satellite terminal. It was also shown that the IoT data transmission is robust and can be transmitted and received at low SNR, demonstrating that IoT sensors may be located in very remote areas and the very little interference with the wider satellite network. This area could be explored further, improving the existing waveform or investigating the use of other waveforms over satellite.

As conclusion, we can say that with the different Demos and PoCs developed in the iNGENIOUS project, the following objectives and challenges were achieved and properly addressed according to the project's ambition:

- new Cellular IoT solutions were developed, using innovative 5G systems at both New Radio and 5G Core for enabling the enhanced Mobile Broadband and Ultra-reliable and Low-latency Communications capabilities that the tactile use cases are demanding.
- AI/ML technologies were exploited across all iNGENIOUS architectural layers, from neuromorphic sensor level to smarter applications passing through network automation enablers.
- a semantic and syntactic interoperability between the heterogeneous Machine-to-Machine platforms as well as DLT solutions currently used in the supply chain sector, was enabled and tested.

Finally, the security aspects of IoT systems were enhanced, by developing IoT devices based on new hardware paradigms that enable strong isolation by default.

References

- [1] iNGENIOUS Consortium, "D6.1 Initial Planning for Testbeds", 2021.
- [2] iNGENIOUS Consortium, "D6.2 PoC development, platform and test-bed integration", 2023.
- [3] iNGENIOUS Consortium, "D3.3 Secure, private and more efficient HW solutions for IoT devices", 2022.
- [4] HERE Technologies, "Start building customizable maps and spatial intelligence content using your data," 2023. [Online]. Available: <https://www.here.com/get-started/start-building>.
- [5] G. E. P. Box and G. M. Jenkins, Time Series Analysis: Forecasting and Control, San Francisco: Holden-Day, 1976.
- [6] J. Perktold, S. Seabold and T. Jonatan, "Statsmodels User Guide: SARIMAX," 02 November 2022. [Online]. Available: <https://www.statsmodels.org/stable/generated/statsmodels.tsa.statespace.sarimax.SARIMAX.html>.
- [7] T. Smith, "API Reference: pmdarima.arima.auto_arima," 2022. [Online]. Available: https://alkaline-ml.com/pmdarima/modules/generated/pmdarima.arima.auto_arima.html.
- [8] L. Breiman, "Random Forests. Machine Learning," pp. 5-32.
- [9] A. Ronacher, "Flask: web development, one drop at a time," 2022. [Online]. Available: <https://flask.palletsprojects.com/en/2.2.x/>.
- [10] OpenJS Foundation, "Node-Red: Low-code programming for event-driven applications," [Online]. Available: <https://nodered.org/>.
- [11] iNGENIOUS Consortium, "D2.1 - Use cases, KPIs & requirements", 2021.
- [12] iNGENIOUS Consortium, "D3.2 Proposals for next generation of connected IoT modules," 2022.
- [13] 5G-eve, "5Probe GitHub repository," [Online]. Available: <https://github.com/5GEVE/5Probe>. [Accessed 2023].
- [14] R. Curnow, "Chrony official web page," 12 2021. [Online]. Available: <https://chrony.tuxfamily.org/index.html>. [Accessed 2023].
- [15] iNGENIOUS Consortium, "D6.3 Final Evaluation and validation", 2023.
- [16] "Tradelens platform," [Online]. Available: <https://platform-sandbox.tradelens.com/documentation/swagger/?urls.primaryName=Event%20Publish%20API>. [Accessed 2023].
- [17] iNGENIOUS Consortium, "D5.3 Final iNGENIOUS data management platform", 2023.



Annex I: Factory UC - Automated Robots with Heterogeneous Networks

Below information about Factory UC Setup and execution and validation and results.

Setup and Execution

Part I

The Factory UC was tested with different AGV and operation modes in each one, using the 5G network for different purposes. The deployments used for the use case implementation are shown below:

Tribot AGV

This AGV (see Figure 79) has one Fivecomm modem which is connected to the 5G-LAN and it has one RPI with can bus via ethernet. This RPI receives the data which are been sent by a gamepad. It can see in the next architecture.

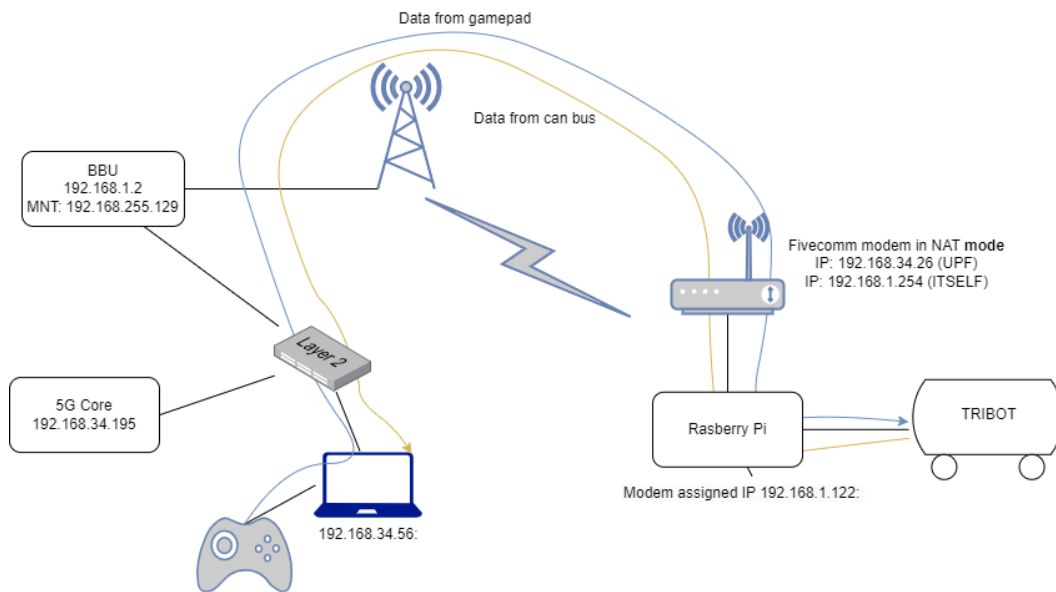


Figure 76: Tribot architecture

The information is sent from the gamepad to the RPI thanks to 5G-LAN and the information from the AGV goes from the AGV to the laptop (192.168.34.56). This information are the internal variables of the Tribot such as linear speed, rotation speed, level battery, errors states and others.

Sender	Receiver	Data
Gamepad	Tribot	Motion actions
Tribot	Laptop	Variables from AGV

Table 29. Information flows for Tribot AGV



EasyBot AGV

This AGV (see Figure 80) is moved automatically thanks a black magnetic band on the floor of our facility. It gives to the laptop the information the temperature and humidity of the environment.

In this case, we have the modem connected to the 5glan and it has connected one RPI to send the information, which is collected by the sensor DHT11 with Arduino. The architecture is shown in figure bellow:

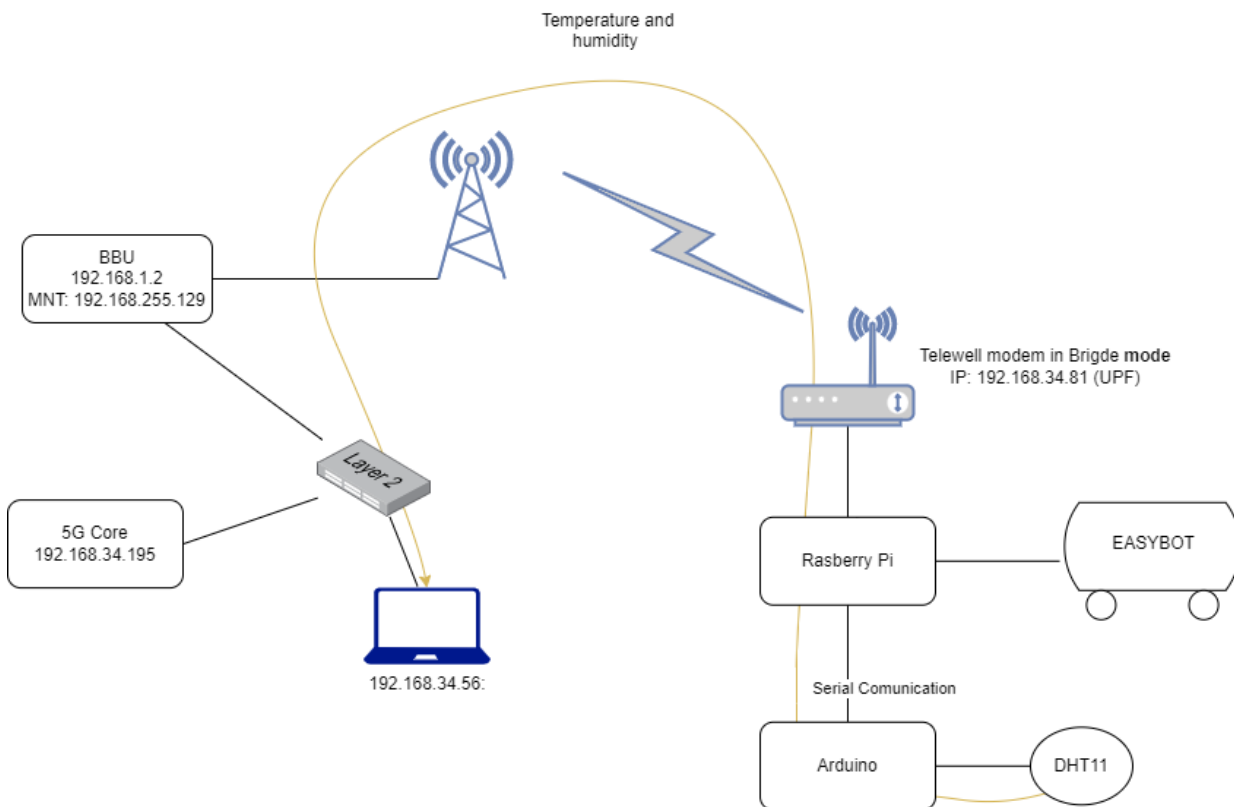


Figure 77: EasyBot architecture

Sender	Receiver	Data
Arduino	Laptop	Temp. & Humd.

Table 30. Information flows for EasyBot AGV

Ebot

This AGV (see Figure 81), has one Fivecomm modem which is connected to the 5GLAN and it has one rpi with can bus via ethernet. This rpi receives the data which are been sent by a laptop, which has a script to generate the data that the AGV needs it. It can see in the next architecture. It also has connected a depth camera, d435i specifically. This camera sends the information by the 5GLAN.



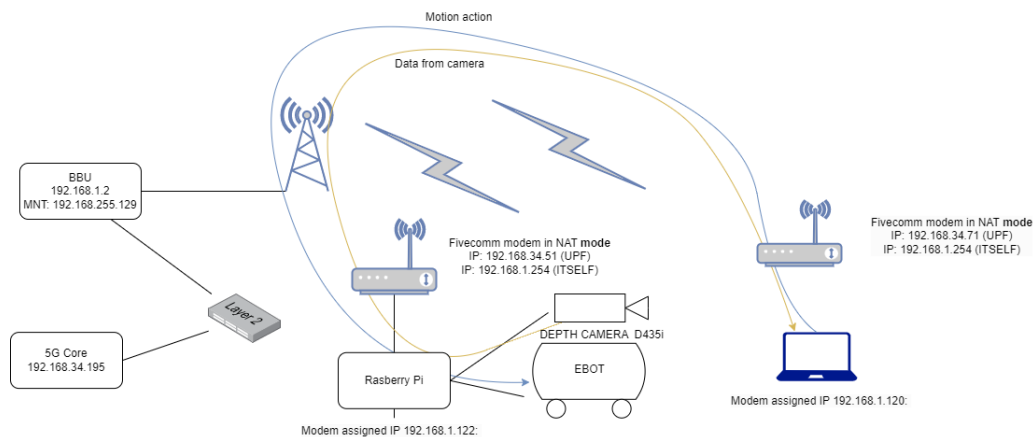


Figure 78: Ebot architecture

Sender	Receiver	Data
Laptop	Ebot	Motion actions
D435i	Laptop	Camera frames

Table 31. Information flows for Ebot AGV

Equipment

The main characteristics of these AGVs are detailed in Table 32.




AGV		Specifications
Tribot	 <p>Figure 79: Tribot AGV</p>	<ul style="list-style-type: none"> • Towing capacity: 3200 N • Maximum payload: 5000 Kg • Dimensions (LxWxH): 1221 x 695 x 762 mm • Movement: Unidirectional • Speed range: From 0.035 to 2 m/s • Battery: Li-Ion 24V 120Ah
Easybot	 <p>Figure 80: EasyBot AGV</p>	<ul style="list-style-type: none"> • Towing capacity: 600 N • Maximum payload: 1200 Kg • Dimensions (LxWxH): 1700 x 520 x 370 mm • Movement: Unidirectional • Speed range: From 0.01 to 1.2 m/s • Battery: Li-Ion 24V 40Ah
Ebot	 <p>Figure 81: Ebot AGV</p>	<ul style="list-style-type: none"> • Maximum payload: 350 Kg • Dimensions (LxWxH): 1052 x 660 x 352 mm • Movement: Omnidirectional • Speed range: From 0.05 to 2.2 m/s • Battery: Li-Ion 48V 40Ah

Table 32. AGVs employed demonstration in Burgos.

The following table shows the main parameters and configurations of the 5G network used.

Component	Model	Features
Antenna	AAFGHC	
Radio Unit	AZNA	<ul style="list-style-type: none"> • Work on n40 band (2370-2390 MHz) • 20 MHz of bandwidth • 4T4R
Baseband Unit	Airscale System Module	Rel. 16
5G core	Cumucore	Rel 17 inc. network slicing and UPF with 5GLAN, TSN functions
5G modem	Fivecomm	<ul style="list-style-type: none"> • Works in both 5G SA and 5G NSA • Bands supported: n41, n77, n78, n79, n1, n3, n5, n7, n8, n20, n28, n38, n40 • Ethernet connection • Up to 2.1 Gbps (DL)/900 Mbps (UL) in SA

Table 33. Main parameters and configuration of 5G network

The figure below shows the gNB, deploy at UBU premises, was used to perform the use case trial.



Figure 82: 5G base station

Other devices

Other devices were used during the demo for communication or sensing purposes. These devices are listed on the following table.







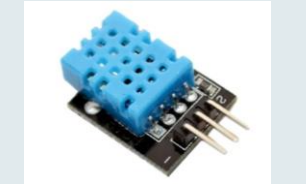
Device		Specifications
<p>Raspberry Pi 3 Model B Rev 1.2 with can bus hat (connected to Tribot)</p>		<ul style="list-style-type: none"> • SOC Type: Broadcom BCM2837 • Core Type: Cortex 453 64 bit • No of cores : 4 • CPU Clock: 1,2 GHz • USB 4xUSB2.0. • Ethernet: 10/100M. • SPI: YES. • I2C: YES. • 2.4GHz 802.11n. • 4.1 BLE. • Oscilator: 160000. • Bitrate: 10000000 Mbits/s
<p>Raspberry Pi 4 Model B with can bus hat (connected to Easybot)</p>		<ul style="list-style-type: none"> • SOC Type: Broadcom BCM2711. • Core Type: Cortex-A72 (ARM v8) 64 bit. • No of cores: 4. • CPU Clock: 1,5 GHz. • USB: 2xUSB2.0 + 2xUSB3.0 + USB-C OTG. • Ethernet Gigabit. • SPI: YES. • I2C: YES. • Wi-Fi: 2.4 GHz and 5GHz 802.11. • Bluetooth 5.0. • Oscilator: 120000. • Bitrate: 250000 Mbits/s. • 1 CAN port.
<p>Raspberry Pi 4 Model B with can bus hat (connected to Ebot)</p>		<ul style="list-style-type: none"> • SOC Type: Broadcom BCM2711. • Core Type: Cortex-A72 (ARM v8) 64 bit. • No of cores: 4. • CPU Clock: 1,5 GHz. • USB: 2xUSB2.0 + 2xUSB3.0 + USB-C OTG. • Ethernet: Gigabit. • SPI: YES. • Wi-Fi: 2.4 GHz and 5GHz 802.11. • Bluetooth: 5.0. • Oscilator: 120000. • Bitrate: 250000 Mbits/s. • 2 CAN ports.
<p>Arduino Uno</p>		<ul style="list-style-type: none"> • Microcontroller ATmega38P – 8-bit AVR family. • DC Current on I/O Pins: 40 mA. • Flash Memory 32 KB. • SRAM 2 KB. • EEPROM: 1 KB. • Frequency (Clock Speed): 16 MHz.
<p>DHT11</p>		<ul style="list-style-type: none"> • Operation Voltage: 3.5 to 5.5 V • Output Serial data. • Temperature Range: 0 °C to 50 °C • Humidity Range: 20% to 90% • Resolution Temperature and Humidity are 16 bit. • Accuracy +- 1°C and +- 1%

Table 34. Equipment for factory UC demonstration in Burgos



Validation and Results

TEST CASES VERIFICATION

Test Case Id	UC1_TC_01
Test case description	Hardware and software implementation
System requirements covered	UC1_SR_08
Expected result	<ul style="list-style-type: none"> E2E latency for remote control: 10-50 ms E2E latency for control/human-in -loop control: 1-5 ms Data rate per robot: 10 Mbps Data rate for IoT sensors: 0.1 Mbps
Actual result	<ul style="list-style-type: none"> E2E latency: max: 2.9 ms, min: 1.6 ms Throughput: max: 2.94 Mbps, min: 0.34 Mbps
Passed/Failed	Partially

Table 35. UC1_TC_01 verification

Test Case Id	UC1_TC_02
Test case description	Core network integration testing
System requirements covered	UC1_SR_08
Expected result	<ul style="list-style-type: none"> E2E latency for remote control: 10-50 ms E2E latency for control/human-in -loop control: 1-5 ms Data rate per robot: 10 Mbps Data rate for IoT sensors: 0.1 Mbps
Actual result	<ul style="list-style-type: none"> E2E latency: max: 3.1 ms, min: 1.6 ms Throughput: max: 2.45 Mbps, min: 0.33 Mbps
Passed/Failed	Partially

Table 36. UC1_TC_02 verification

Test Case Id	UC1_TC_03
Test case description	Gateway test
System requirements covered	UC1_SR_08
Expected result	Successful data transmission with different RAN standards
Actual result	Successful integration among Flexible PHY/MAC and 5G core using UERANSIM
Passed/Failed	Passed

Table 37. UC1_TC_03 verification



Test Case Id	UC1_TC_04
Test case description	Onboard industrial IoT network slice templates and NF descriptors
System requirements covered	UC1_SR_06
Expected result	The onboarded network slice templates and related descriptors are successfully maintained by the cross-layer MANO to create new vertical services and network slices instances.
Actual result	The network slice templates (NSTs) have been successfully onboarded into the NSMF catalogue and are visible from the NSMF web GUI. Two NSTs describing a video streaming and device-to-device communication services have been onboarded. This test case has been validated first in the NXW lab, for the mid-term review demo in UPV testbed, and finally in the TUD testbed. The test case is fully achieved.
Passed/Failed	Passed

Table 38. UC1_TC_04 verification

Test Case Id	UC1_TC_05
Test case description	Automated deployment of industrial IoT network slice instance
System requirements covered	UC1_SR_03, UC1_SR_07, UC1_SR_08
Expected result	A new network slice instance is created, all the related network and computing resources have been allocated and the 5G Core NFs are up and running and ready to be configured. Moreover, the cross-layer MANO maintains the information related to the network slices instance and the NFs information related.
Actual result	Based on the outcome of UC1_TC_04, the automated deployment of the industrial IoT network slice instance has been completed. This test case has been validated first in the NXW lab, then for the mid-term review demo in the UPV testbed, and finally into the TUD testbed. In the last case, the automated end-to-end slice deployment includes also resource management on the RAN segment, assigning the correct amount of radio resources interacting with the flexible PHY-MAC control APIs. This test case can be considered passed.
Passed/Failed	Passed

Table 39. UC1_TC_05 verification

Test Case Id	UC1_TC_06
Test case description	Automated termination of industrial IoT network slice instance
System requirements covered	UC1_SR_03, UC1_SR_07, UC1_SR_08
Expected result	The network slice instance is terminated, all the related network and computing resources have been de-allocated and the 5G Core virtualized NFs are terminated and the related virtual resources freed. Moreover, the cross-layer MANO still maintains the information of the network slice instance terminated.
Actual result	Based on the outcome of UC1_TC_05, the automated termination of industrial IoT network slice instance has been completed. As for



	previous tests, this one has been validated initially in the NXW lab, then demonstrated in the mid-term review in the UPV testbed, and finally executed in the TUD testbed. The resources allocated in the 5GC and in flexible RAN are freed as expected upon termination of the end-to-end network slice through the NSMF APIs. Moreover, the NSMF keeps track of the terminated end-to-end network slice. Therefore, this test case can be considered passed.
Passed/Failed	Passed

Table 40. UC1_TC_06 verification

Test Case Id	UC1_TC_07
Test case description	Manual scaling of an industrial IoT network slice instance
System requirements covered	UC1_SR_07
Expected result	The network slice instance is modified, all the related network and the 5G Core virtualized NFs are modified (or new ones are created) and the related virtual resources as well
Actual result	Based on the outcome of UC1_TC_05, the manual scaling of the industrial IoT network slice instance has been validated initially on the NXW lab and then on the TUD testbed. As result, downlink or uplink throughput of the end-to-end network slice (according to the network slice data model used in the NSMF) can be automatically scaled, resulting in a re-configuration of the 5GC subnet-slice (according to the CumuCore 5GC APIs), combined with a re-configuration of the flexible RAN resource allocation to meet the application requirements.
Passed/Failed	Passed

Table 41. UC1_TC_07 verification

Test Case Id	UC1_TC_08
Test case description	Automatic slice configuration through 5GC NSM
System requirements covered	UC1_SR_07
Expected result	The network slice is correctly configured by the NSM NF as requested.
Actual result	Based on the outcome of UC1_TC_07, the process of slice configuration through the CumuCore 5GC APIs has been automated within the NSMF and 5GC NSSMF, and validated initially on the NXW lab and then in the TUD testbed. For this reason, this test case is being considered passed.
Passed/Failed	Passed

Table 42. UC1_TC_08 verification

Test Case Id	UC1_TC_09
Test case description	Automated deployment of industrial IoT network slice instance and of an edge robot control application as part of network slice instance



System requirements covered	UC1_SR_04
Expected result	A new network slice instance is created, all the related network and computing resources have been allocated and the 5G Core NFs are up and running and ready to be configured. As part of network slice instance, a robot control application is deployed at the edge and the related computing resources have been correctly allocated. Moreover, the cross-layer MANO maintains the information related to the network slices instance and the related NFs information.
Actual result	The automated deployment of an industrial IoT network slice instance is already being validated in the UC1_TC_05 test case. However, in the TUD testbed the edge application is considered already deployed. The edge application consists of a video streaming application sending data relying on the end-to-end network slice deployed. However, in the NXW lab, the automated deployment of the video streaming edge application integrated with an end-to-end network slice has been validated, through a dedicated additional NSSMF for the management of virtualized edge functions and applications. For this reason, this test case can be considered passed.
Passed/Failed	Passed

Table 43. UC1_TC_09 verification

Test Case Id	UC1_TC_10
Test case description	Automated termination of industrial IoT network slice instance and of edge robot control application as part of network slice instance
System requirements covered	UC1_SR_04
Expected result	The network slice instance is terminated, all the related network and computing resources have been de-allocated and the 5G Core virtualized NFs are terminated and the related virtual resources freed. As part of network slice instance, also the computing resources at the edge are de-allocated. Moreover, the cross-layer MANO still maintains the information of the network slice instance terminated.
Actual Result	The automated termination of an industrial IoT network slice instance is already being validated in the UC1_TC_06 test case. In the NXW lab, the automated termination of the end-to-end network slice, integrated with a video streaming edge application deployed for UC1_TC_10 has been validated. For this reason, this test case can be considered passed.
Passed/Failed	Passed

Table 44. UC1_TC_10 verification

Test Case Id	UC1_TC_11
Test case description	Subscription to either Network Data Analytics Function (NWDAF) or Network Exposure Function (NEF) for collecting monitoring and analytics information related to the network slices, NFs and UEs.
System requirements covered	UC1_SR_03, UC1_SR_05, UC1_SR_07, UC1_SR_08
Expected result	The cross-layer MANO is able to receive the notifications it is subscribed to.
Actual Result	The automated termination of an industrial IoT network slice instance is already being validated in the UC1_TC_06 test case. In the NXW lab, the automated termination of the end-to-end network



	slice, integrated with a video streaming edge application deployed for UC1_TC_10 has been validated. For this reason, this test case can be considered passed.
Passed/Failed	Passed

Table 45. UC1_TC_11 verification

Test Case Id	UC1_TC_12
Test case description	Deletion of either Network Data Analytics Function (NWDAF) or Network Exposure Function (NEF) active subscription.
System requirements covered	UC1_SR_03, UC1_SR_05, UC1_SR_07, UC1_SR_08
Expected result	The cross-layer MANO is no longer able to receive the notifications related to the just removed subscription
Actual Result	As already mentioned in the previous test case validation, there is no actual subscription to either NWDAF or NEF. However, the monitoring platform exposes dedicated APIs to control the data collection, and is integrated with the NSMF that can activate and deactivate the monitoring jobs. This test case can be considered passed, and executed in both NXW lab and TUD tested.
Passed/Failed	Passed

Table 46. UC1_TC_12 verification

Test Case Id	UC1_TC_13
Test case description	Automated slice scaling triggered by AI\ML platform using NWDAF data.
System requirements covered	UC1_SR_07, UC1_SR_11
Expected result	The cross-layer MANO correctly and automatically scales with the support of the AI\ML platform the network slice instance.
Actual Result	<p>This test case has been executed and validated in two flavours:</p> <ul style="list-style-type: none"> UC1_TC_13a (NXW lab), the AI/ML engine/agent implements a decision logic for scaling UPF network functions as a whole upon UPF load and congestion prediction provided by the ML model, thus triggering towards the NSMF a UPF scaling action. This is processed and translated by the NSMF into the creation of new UPFs instances for the UEs to connect to, and distributing the data plane traffic. UC1_TC_13b (TUD testbed): the AI/ML engine/agent implements a decision logic for scaling generically the entire end-to-end network upon slice congestion prediction provided by the ML model, thus triggering towards the NSMF a network slice scaling action. This is processed and translated by the NSMF into consistent 5GC subnet slice re-configuration (increase of slice uplink or downlink throughput) and flexible RAN resource allocation (increase of PHY-MAC radio resource allocation).
Passed/Failed	Passed

Table 47. UC1_TC_13 verification

Test Case Id	UC1_TC_14
Test case description	Robot interface connectivity

System requirements covered	UC1_SR_01
Expected result	The devices in the robot can utilize standard Ethernet RJ45 ports to connect to 5G communication module and connect to 5G network.
Actual result	The AGVs and the camera were controlled through a Raspberry Pi, which was responsible to process the control messages from the controller application to the remote device. The Raspberry Pi was connected through Ethernet RJ45 to the 5CMM 5G modem, providing a successful connection with others UEs and 5G network. On the other hand, the AGVs were successfully connected to the Raspberry Pi through the CAN bus. Several KPIs (present in Table 5) were taken to measure the quality of the connection between these devices using the 5G private network.
Passed/Failed	Passed

Table 48. UC1_TC_14 verification

Test Case Id	UC1_TC_15
Test case description	Test of API for application development
System requirements covered	UC1_SR_10
Expected result	Simple application implemented using the available devices and the connectivity among them should be demonstrated
Actual result	Emulated applications have their data transmission configured via the Tactile API, which abstracts the network resources. The API was successfully used to integrate the Flexible PHY/MAC in a 5G compliant network configured by the MANO.
Passed/Failed	Passed

Table 49. UC1_TC_15 verification

KPIs

This section explains the procedures taken in order to validate the performance of the network and measure the KPIs.

The coverage of the 5G signal was measured with the scanner R&S® TSME6 from UPV, which can analyse the environment and decode mobile communication signals, obtaining the main information of the gNB, such as RSRP (Reference Signal Receive Power), RSRQ (Reference Signal Receive Quality), SINR (Signal Interference Noise Ratio), PCI (Physical Cell Identifier), SCS (Sub-Carrier Spacing), among others. The scanner was used around the entire trial area, verifying the proper configuration and functionality of the installed gNB (transmitting in band n40, 2370-2390MHz). The walk test was performed around the industrial unit where the AGVs were circulating. The signal power received by the scanner (RSRP) was between -50 dBm and -75 dBm (Figure 83), thus the UEs could receive the 5G signal without a problem. SINR and RSRQ values measured were adequate for a successful 5G transmission. Also, we could verify the proper frequency transmission, PCI (54), and the SCS (30 kHz), which had been configured previously in the gNB.

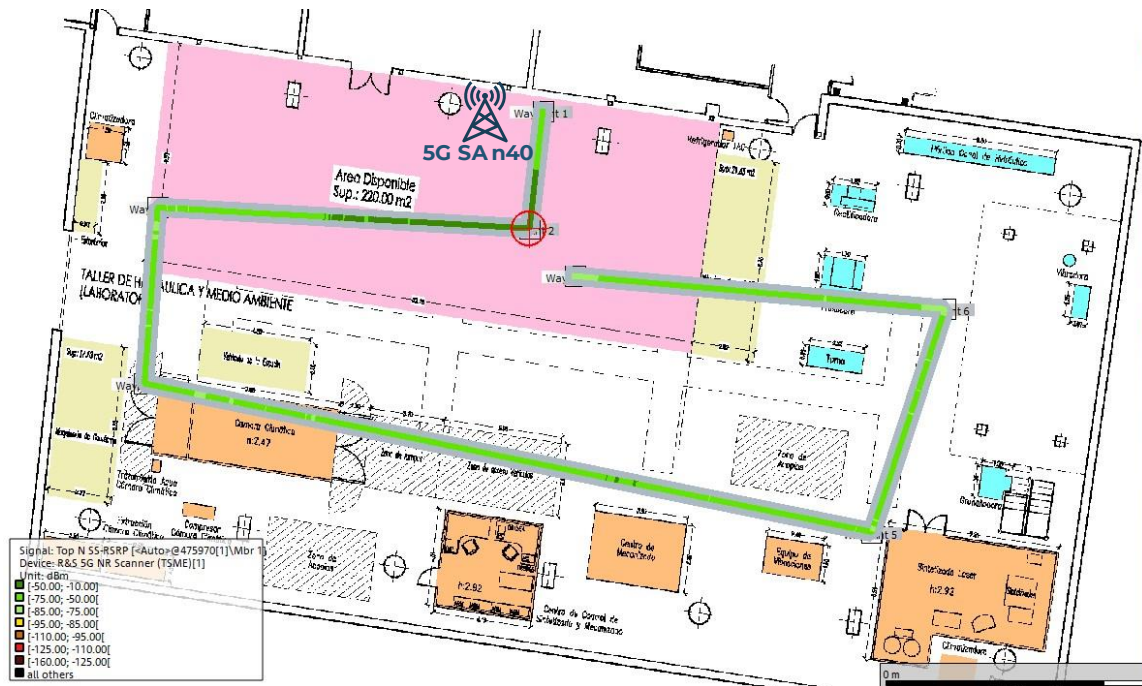


Figure 83: RSRP values obtained through the walk test around the industrial unit.

Once the coverage test was performed successfully and the received signal was proper to operate the AGV, the different devices and robots were connected to the 5G SA network through 5CMM modems. While the operators control the AGVs and the stream of data from cameras and robots is transmitted through the network, some KPIs such as mobility (AGV’s speed), data rate (from the AGV to the operator and vice versa), latency (Round-Trip-Time, RTT) and, connection density for robots, and reliability for remote control.

The speed of the AGV is obtained from the internal logs of the robot. At the same time, the connection density for the robots was simulated by connecting several modems (3 modems connected to AGVs and another one connected to the PCs which control the AGV) to the network at the same time and simulating the same traffic, in order to analyse the performance of the network and the evolution of the KPIs.

To perform the data rate and latency measurement, 5Probe [13] tool has been used. This tool has been developed by the consortium of the 5G-EVE project, which permits measuring the uplink and downlink throughput, RTT and One-Way-Delay (OWD) of the network. The tool was installed on the Raspberry Pi connected to the AGV. Also, InfluxDB was used to store the parameters discussed above by the tool, which is installed on an external virtual machine on a laptop. This virtual machine is time-synchronized through NTP (Network Time Protocol) protocol using Chrony tool [14]. While the 5Probe tool was executed, several iperf3 and ICMP tests were performed, two scenarios were tested: i) Communication between two different UEs of the network (through 5GLAN technology) Figure 84 and, ii) Communication between a UE and a laptop connected directly to the 5G core Figure 85. The first scenario would emulate the communication machine-to-machine (M2M) in an industrial environment where “things” have to exchange data between them using the 5G radio interface. The second scenario emulates the communication of a machine connected with 5G to a resource outside the 5G network. The most remarkable difference between these two scenarios is that, in a 5G network, the

uplink throughput (UE to 5G network) is usually the most limiting factor in the communication due to the power of the signal the UEs transmit. In the first scenario, this “UE to 5G network” traffic is always present in the communication acting as the bottleneck.

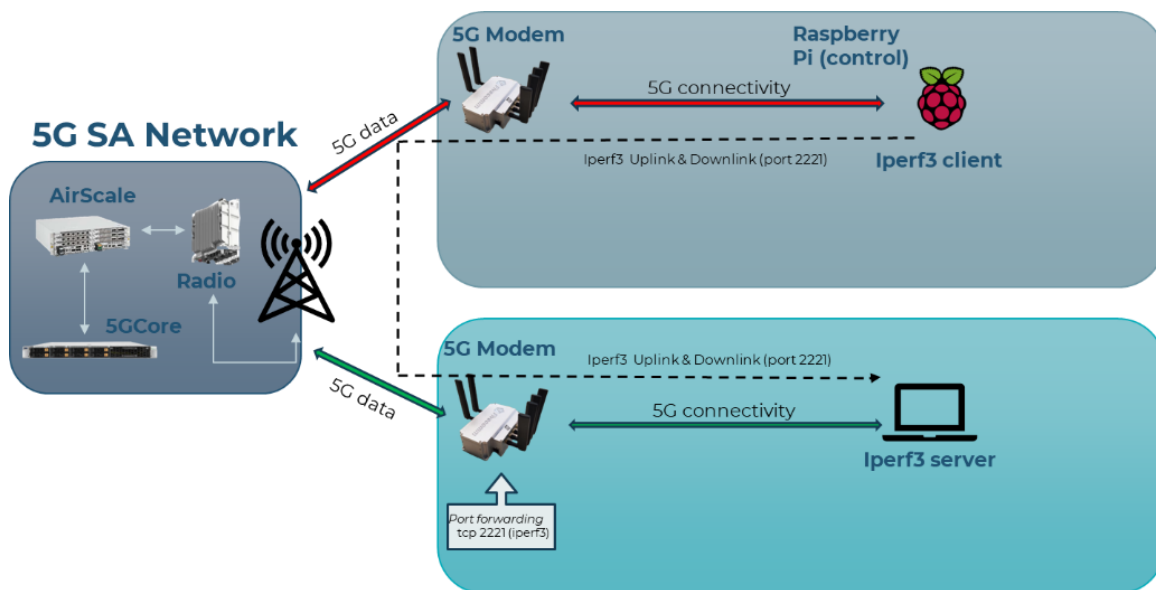


Figure 84: End-to-end architecture iperf3 test UE to UE.

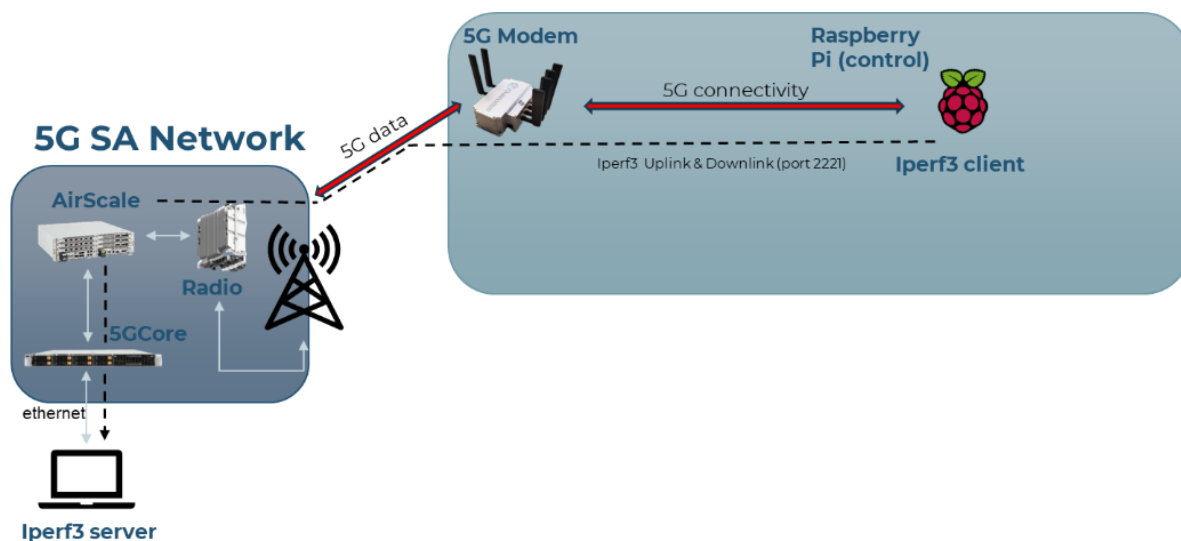


Figure 85: End-to-end architecture iperf3 test UE to core.

Grafana has been used to analyse the data stored in the database. The following Figure represents the full architecture to measure the KPIs using the 5Probe tool. The results of these tests were shown in Section 2.3 of the main document.

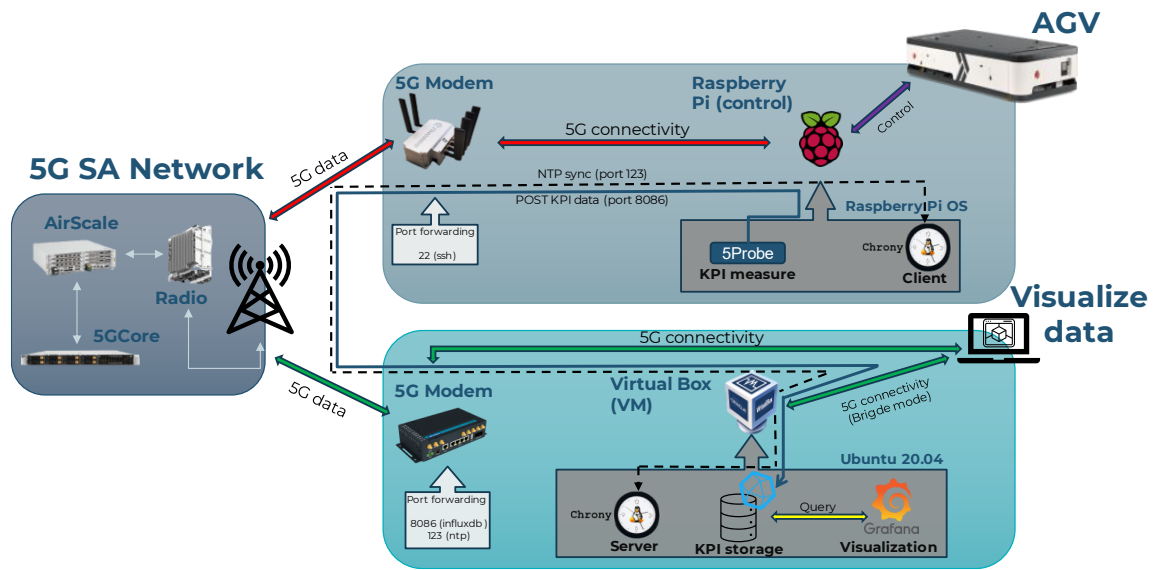


Figure 86: End-to-end architecture used for the KPI measurement setup with 5Probe.

Annex II: Transport UC - Transportation Platforms Health Monitoring

Below information about Transport UC Validation and results

Validation and Results

TEST CASES VERIFICATION

Test Case Id	UC3_TC_01
Test case description	Lifetime Operation - Battery Life Load Cycles Battery Spec
System requirements covered	UC3_SR_01
Expected result	Lifetime Operation Value
Actual result	See description below
Passed/Failed	Passed*

Table 50. UC3_TC_01 verification

12 years+ Lifetime Operation @ -20 to +60°C environmental conditions with 5 broadcast per day is theoretically possible. 24-30 years of operation is not possible, but essential for the business case to avoid sensor replacement.

Test Case Id	UC3_TC_02
Test case description	Minimum Communication Content and Frequency Requirements
System requirements covered	UC3_SR_02
Expected result	Communication Frequency and Fault Latency for Bearing & Critical Flat Spot detection.
Actual result	See description below
Passed/Failed	Passed*

Table 51. UC3_TC_02 verification

Micro-Edge Sensor to Gateway communication @5x per day is possible for 12 years. Gateway GSM communication @5x per day is only possible without extended periods of snow and ice coverage on solar harvester. Changing from GSM communication from each gateway to LORA communication between gateways and GSM2Cloud communication from the energy heathiest node reduces weather-based communication outage by a factor of 10.



Test Case Id	UC3_TC_03
Test case description	Connectivity Coverage
System requirements covered	UC3_SR_03
Expected result	Communication Frequency and Fault Latency for Bearing & Critical Flat Spot detection.
Actual result	See description below
Passed/Failed	Passed

Table 52. UC3_TC_03 verification

GSM connectivity coverage is generally acceptable for 5x daily status reporting. Regional communication outage is rare and not prolonged. Extended communication outage is more likely occur in stational situations due to infrastructure interference. This type of interference can be nearly completely avoided via Lora-Wan Gateway2Gateway plus healthiest/best connectivity node GSM2Cloud communication.

Test Case Id	UC3_TC_04
Test case description	TC3 Connectivity Coverage
System requirements covered	UC3_SR_04
Expected result	Memory Requirements and Strategy for Data Reduction/Compression
Actual result	See description below
Passed/Failed	Passed

Table 53. UC3_TC_04 verification

Micro-Edge Sensor meta data storage is trivial due to the small data volume required. The same applies to Gateway fusion data storage. The edge storage capability is used to bridge communication outage between Edge2Gateway and Gateway2Cloud.

Test Case Id	UC3_TC_05
Test case description	Functional Safety Requirements, Connectivity Frequency, Connectivity Coverage
System requirements covered	UC3_SR_05
Expected result	Multimodal Connectivity Opportunities
Actual result	See description below
Passed/Failed	Passed

Table 54. UC3_TC_05 verification



Multimodal connectivity results in significant performance (online availability) improvements. Changing Micro Edge Sensor communication from BLE_SensorEdge2Gateway to BLE_SensorEdge2 SensorEdge2Gateway allows to lower communication power, while adding Lora-Wan communication between Gateways allows reduced gateway power consumption and improved online availability as described in UC3_TC_03.

Test Case Id	UC3_TC_06
Test case description	Functional Safety Requirements, Customer Requirements
System requirements covered	UC3_SR_06
Expected result	Monitoring Resolution
Actual result	See description below
Passed/Failed	Passed*

Table 55. UC3_TC_06 verification

Ideally, increasing fault/speed levels should be time-stamped with the first level of occurrence. This information can be used to determine when and which operator is likely to have caused the fault. 5 measurements per day are the minimum resolution requirement. Wake on dynamic defect energy (Piezo-Electric Energy Gradient) is good indicator for fault detection. If subsequent fault intensity computations at a given speed result in higher than previously recorded value, than a new, even more intense fault event than previously recorded has occurred. While the speed is typically not known in fault-free conditions, it can be easily determined once a fault is present. What cannot be detected (or at least not with high reliability) are new faults with same or lower intensity than previous faults.

Test Case Id	UC3_TC_07
Test case description	Functional Safety Requirements, Customer Requirements, Energy Resources, Application Results
System requirements covered	UC3_SR_07
Expected result	Monitoring Capability
Actual result	See description below
Passed/Failed	Passed*

Table 56. UC3_TC_07 verification

The biggest functional challenge of commercial- vs. passenger-rail condition monitoring is the differentiation of non-critical faults and the one critical fault buried in a mass of non-critical faults and noise. After two years of continuous algorithm development, the results exceed expectations. The approach is completely different from reviewed publications, and the fault resolution is phenomenal. Fault severity levels can be reliably classified into up to 64 and possibly more severity levels for flat spots and 4 or more levels for bearing



defects. This far exceeded the targeted 3-5 severity levels for flat spots and the OK/NOK classification of bearing defects.

Test Case Id	UC3_TC_08
Test case description	Edge Classification & Edge Pre-Processing Capabilities & Results
System requirements covered	UC3_SR_08
Expected result	Cloud Defect Validation Capability
Actual result	See description below
Passed/Failed	Passed

Table 57. UC3_TC_08 verification

Cloud or alternatively Gateway fault validation is not really important if simple statistical approaches can be applied. This is good news, as defect validation via Cloud based SVM Engine or similar require more data intensive Feature-Vectors which require higher than desired levels of communication energy resources. Since the implemented micro-edge fault classification is based on a physical-model (validated by real-world data, empirical and physical fault data, and hypothetical signal injection), it will beat any neural network-based approach. The statistical cloud validation checks for continuity and cross-checks neighbouring axles and wheels.

Test Case Id	UC3_TC_09
Test case description	Edge Classification & Edge Pre-Processing Capabilities & Results
System requirements covered	UC3_SR_09
Expected result	Gateway Defect Validation Capability
Actual result	See description below
Passed/Failed	Passed

Table 58. UC3_TC_09 verification

UC3_TC_09 is analogue to US3_TC_08. The validation of condition monitoring defects @Gateway level is at most statistical, or simply data-fusion of Micro-Edge Sensor Meta values.

Test Case Id	UC3_TC_10
Test case description	System Architectural Design and Communication Architecture
System requirements covered	UC3_SR_10
Expected result	Confidentiality and integrity of connection via TLS



Actual result	TLS has been determined to be the suitable solution for protecting confidentiality and integrity of the communication between IoT sensor and the cloud.
Passed/Failed	Passed

Table 59. UC3_TC_10 verification

This test case is a not a functional test, but a review activity performed at the beginning of the project to assess the suitability of TLS in the context of the Transport use case. Due to its flexibility, industry-wide usage, and strong security guarantees, TLS has been found to be an excellent foundation for securing IoT communication security beyond the state of the art.

Test Case Id	UC3_TC_11
Test case description	Security (Listening)
System requirements covered	UC3_SR_11
Expected result	Confidentiality and integrity of connection between sensor endpoint and cloud server protected via TLS
Actual result	TLS v1.3 is used and integrated with remote attestation, ensuring both the security of the communication channel and strong identity and integrity of endpoint devices.
Passed/Failed	Passed

Table 60. UC3_TC_11 verification

This test case shows that the connection between cloud sensor endpoint and server is established and confidentiality and integrity of this communication channel is protected via.

Test Case Id	UC3_TC_12
Test case description	Security (Flash)
System requirements covered	UC3_SR_12
Expected result	Only cryptographically signed M3 operating system and applications can start on BI platform
Actual result	Applications and service programs (part of the operating system) are measured during started for reporting via remote attestation.
Passed/Failed	Passed*

Table 61. UC3_TC_12 verification

Measured startup before execution of applications and service programs ensures that remote attestation can report the identity and integrity of said applications and service programs. The original goal of secure startup for the entire M³ OS turned out too ambitious and could not be realized, because hardware root-of-trust support could not be implemented (as documented in Issues on execution). Hence, as an intermediate step, only a specific application scenario consisting of a set of programs is supported for measurement and

remote attestation. Full support for measured startup of all software will be worked on after the end of the project.

Test Case Id	UC3_TC_13
Test case description	Security (Commanding)
System requirements covered	UC3_SR_13
Expected result	Connection only established, if remote attestation of sensor endpoint passed; connection refused, if the sensor endpoint does not pass remote attestation
Actual result	Connection between cloud sensor endpoint and server is established only if remote attestation of the IoT sensor and cloud endpoint passed verification. The connection is refused, if either the sensor or cloud endpoint fail verification during remote attestation.
Passed/Failed	Passed

Table 62. UC3_TC_13 verification

A connection between cloud sensor endpoint and server is established only if remote attestation of the IoT sensor and cloud endpoint passed verification. The connection is refused, if either the sensor or cloud endpoint fail verification during remote attestation. Both the IoT sensor and the cloud server will abort the connection attempt, if the software running on the other device is not recognized (based on a cryptographic hash fingerprint).

Test Case Id	UC3_TC_14
Test case description	Data Encryption
System requirements covered	UC3_SR_14
Expected result	Sensor data is encrypted and digitally signed, cloud server can decrypt and verify signature.
Actual result	Not pursued
Passed/Failed	N/A

Table 63. UC3_TC_14 verification

This test case was aimed at confirming that data provided by the sensor can be encrypted offline and stored in local memory/storage of the FPGA/M³ platform for later transmission to sensor endpoint. However, the very ambitious goal of implementing a root-of-trust for the M³ platform in FPGA-based hardware could not be reached during the duration of the project. The associated risk has first been documented in deliverable D6.1 [1] and mitigations were described in D6.1 [1], D6.2 [2] and D3.3 [3]. As a result of a scaled-back software-only simulation of a root-of-trust, support for encrypting data at rest has been delayed and will be pursued after the end of INGENIOUS. Data encryption support is not essential to demonstrate the improved security guarantees enabled by remote attestation, which has been the main goal; this limitation therefore does not impact the use case significantly.



Test Case Id	UC3_TC_15
Test case description	Rail Health Functional Safety Requirements, Customer Requirements
System requirements covered	UC3_SR_15
Expected result	Fault Detection Diagnostics Coverage
Actual result	See description below
Passed/Failed	Passed

Table 64. UC3_TC_15 verification

Functional Safety considers the integrity of the signal measurement. Is it possible to detect a broken sensor, and what safeguards are in place to assure that critical faults can be detected reliably? A broken sensor can mean no signal or completely random signal noise. The relative white noise signal energy between adjacent axles sensors follows both expected and relative ranges. If this white noise level exhibits unexpected permanent changes in one or the other direction, then the sensor element must be broken. This can be detected at both micro-edge device and gateway or cloud level. Likewise, a communication failure with a micro-edge sensor, or gateway is detected at cloud level. Therefore, the sensor signal integrity can be assured for all critical fault modes.

Test Case Id	UC3_TC_16
Test case description	Explosion Safety Requirements, Customer Requirements
System requirements covered	UC3_SR_16
Expected result	ATEX Compliance Gaps
Actual result	See description below
Passed/Failed	Passed

Table 65. UC3_TC_16 verification

Fire/Explosion Safety is a specialty requirement for hazardous goods transport vessels. Such rail carriages typically lack electric connection, because the technical solutions are cost intensive. Therefore, all sensors and gateways must be battery and harvester powered, and any energy reserves must be physically designed not to cause uncontrolled thermal dissipation if physically damage. ATEX is an example of a standard guiding fire and explosion safety. The rail market is very segmented and governed by mostly national/regional and not international standards. ATEX compliant batteries are readily available, as are ceramic capacitors for sufficient energy reserve to make ATEX compliance achievable. Therefore, Fire/Explosion compliance is not a technically challenging compliance issue.



Test Case Id	UC3_TC_17
Test case description	The radio access should be able to run local application processing when user selects low latency for selected applications
System requirements covered	UC3_SR_17
Expected result	The data received from the device will be processed as closer as possible to the device and returned with lower delay than processing the data in another UPF in the cloud.
Actual result	N/A
Passed/Failed	N/A

Table 66. UC3_TC_17 verification

Not pursued in this use case.

Test Case Id	UC3_TC_18
Test case description	Extended Satellite Coverage: Confidentiality of satellite backhauled sensor data.
System requirements covered	UC3_SR_18
Expected result	Captured sensor data is indecipherable between Sensor Gateway and teleport IP egress point
Actual result	Same as expected result
Passed/Failed	Passed

Table 67. UC3_TC_18 verification

This test was aimed at validating the confidentiality of sensor data as it transited the satellite network. Sensor data was sent between two BI endpoints over iDR’s simulated satellite network. Traces were taken at the ingress and egress points of the simulated satellite network and they confirmed the packets sensor data was encrypted correctly.

Test Case Id	UC3_TC_19
Test case description	Communication Load Optimization: The platform shall be able to use the most appropriate radio technology depending on network access and communication demands.
System requirements covered	UC3_SR_19
Expected result	Communication Load Optimization
Actual result	N/A
Passed/Failed	N/A

Table 68. UC3_TC_19 verification

Not pursued in this use case.



Test Case Id	UC3_TC_20
Test case description	OTA upgradeability
System requirements covered	UC3_SR_20
Expected result	After reboot in A/B configuration, the signature-checked software update is B is running; A is started if signature checks failed on B
Actual result	Not pursued
Passed/Failed	Passed

Table 69. UC3_TC_20 verification

This test case aimed to validate robustness against failed or compromised software updates. Only digitally signed (i.e., authorized, correct, and not manipulated software) is started on the IoT sensor device. As considered in D6.1 [1], this test case was about and A/B software configuration, where the currently running and correctly signed software (A) is kept as a fallback in case an updated, new version (B) fails to start due an incorrect code signature. Due to the unavailability of a fully-integrated root-of-trust, this capability could not be implemented during the project. However, the measured startup capabilities of M³ that are covered by test case UC3_TC_11 ensure that only correct applications and service programs can be started using a signature check; fallback to previous version will be worked on by BI after the end of iNGENIOUS project, when the necessary root-of-trust support has been implemented.

Test Case Id	UC3_TC_21
Test case description	Extended Satellite Coverage: Satellite Multi-Protocol Support; Validate confidentiality of satellite backhauled sensor data
System requirements covered	UC3_SR_21
Expected result	All protocols tested are shown to work over satellite
Actual result	Same as expected result
Passed/Failed	Passed

Table 70. UC3_TC_21 verification

This test was created to show that multi-protocol support was possible over a satellite network. Multiple protocols were tested and verified over iDR’s simulated satellite network including; arp, tcp, udp, mqtt, sctp, http and ftp.

Test Case Id	UC3_TC_22
Test case description	Extended Satellite Coverage: IP Connectivity
System requirements covered	UC3_SR_22
Expected result	Sensor data is received successfully at data centre/cloud



Actual result	Same as expected result
Passed/Failed	Passed

Table 71. UC3_TC_22 verification

This test was carried out as part of the preparation for the mid-term and final demonstrations. IP connectivity was established between the edge network MEC server and hub side gateway server via the simulated satellite network. Sensor data was successfully sent and received in both directions using the simulated forward and return satellite links.

Test Case Id	UC3_TC_23
Test case description	Extended Satellite Coverage: Verify uplink and downlink Satellite backhaul latency
System requirements covered	UC3_SR_21, UC3_SR_23
Expected result	Perform enough tests and preparation to ensure sources of connectivity issues are known and resolved
Actual result	Same as expected result
Passed/Failed	Passed

Table 72. UC3_TC_23 verification

This test was created to simulate the latency of a GEO stationary satellite using iDR’s lab testbed and verify connectivity between the sensors and cloud/data center with added latency applied. The typical latency of a GEO stationary satellite link ranges between 560-580ms depending on environmental conditions and other factors. In the lab setup the satellite delay was set to 560ms which was verified during multiple testing windows.



Annex III: Port Entrance UC - Situational Understanding in Smart Logistics Scenario

Below information about the Port Entrance UC Setup and execution and validation and results.

Setup and execution

Part I

b.1) Offline Data Preparation

Examples of the features used from historical data are included in Figure 87, which illustrates the relevant content in distinct datasets, and how they are used in the model development. When these are present in the data, basic data engineering methods (e.g. database queries, filtering, transformations, merge operations) are used to combine the datasets as necessary for subsequent model development. Specifically, the smaller datasets related to port operations were processed offline using mainly the Python Pandas library.

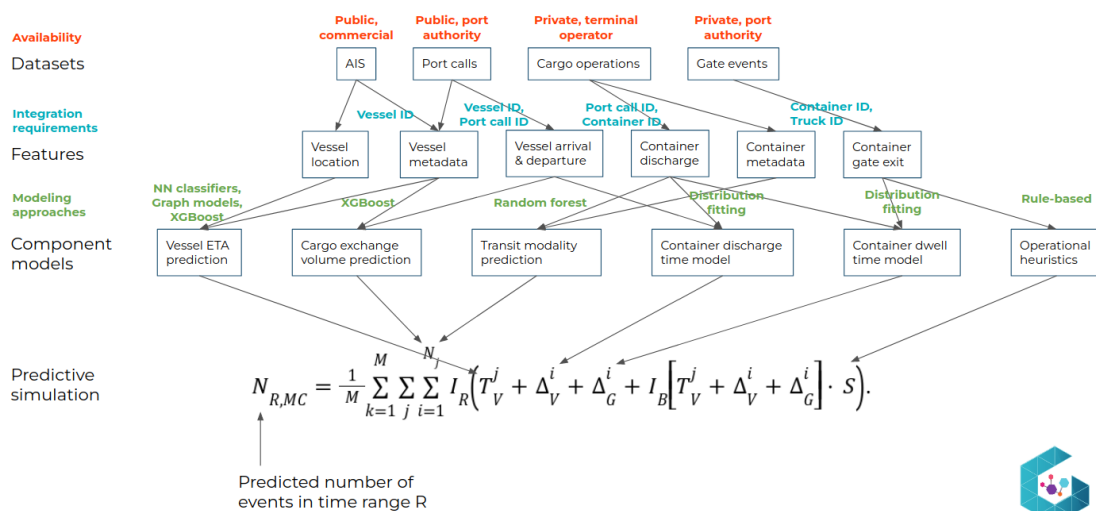


Figure 87: Overview of prediction model components, required features, and source datasets.

The most complex offline data preparation needed in this development was required for obtaining voyage information and timestamps from global AIS data. This involved decoding the raw AIS data transmitted globally by vessels in the National Marine Electronics Association (NMEA) format, analyzing the messages to determine voyage start and stop times and locations (classified according to ports following e.g. global UNECE UN/LOCODE definitions), and labelling data points along voyages accordingly. Due to the size of the dataset (as noted above, as a tabular dataset this would contain of order 10^{11} rows), this is a computationally demanding task not well suited for offline batch processing. In the AWA pipeline, this processing was implemented using continuously operating cloud-based microservices (implemented in a scalable Kubernetes



environment), which performed the above described processing operations over streaming data and store the obtained metadata as logs for use in a dedicated machine learning operations (MLOps) pipeline performing machine learning model training and validation, as outlined in the following subsection.

b.2) Model Development

The monte Carlo simulation used to predict container traffic rates can be described with the following formulation for the number of containers transported by trucks out of the port during a selected time range R:

$$N_{R,MC} = \frac{1}{M} \sum_{k=1}^M \sum_j \sum_{i=1}^{N_j} I_R(T_V^j + \Delta_V^i + \Delta_G^i + I_B[T_V^j + \Delta_V^i + \Delta_G^i] \cdot S). \quad (1)$$

Here, $I_R(x)$ and $I_B(x)$ are indicator functions defined as:

$$I_R(x) := 1 \text{ if } x \in \{R_{min}, R_{max}\}, 0 \text{ if } x \notin \{R_{min}, R_{max}\},$$

where R_{min}, R_{max} are the limits of the time range for which events are simulated and

$$I_B(x) := 1 \text{ if } x \in B, 0 \text{ if } x \notin B,$$

where B is a set of blocked days when no events are allowed (for example, there is typically no truck traffic in the Port of Valencia on Sundays).

Δ_V^i and Δ_G^i are i.i.d. random variables corresponding to the delay between vessel arrival and container discharge and the delay between container discharge and gate exit, respectively. The distributions of these variables are estimated empirically by fitting to observed events using a Kolmogorov-Smirnov metric for goodness of fit over 60 candidate distributions.

The combination of the random variables $\Delta_V^i + \Delta_G^i$ estimates the total dwell time of a container in the port. The figure below illustrates the distribution of the simulated dwell times compared to the actual observed events.

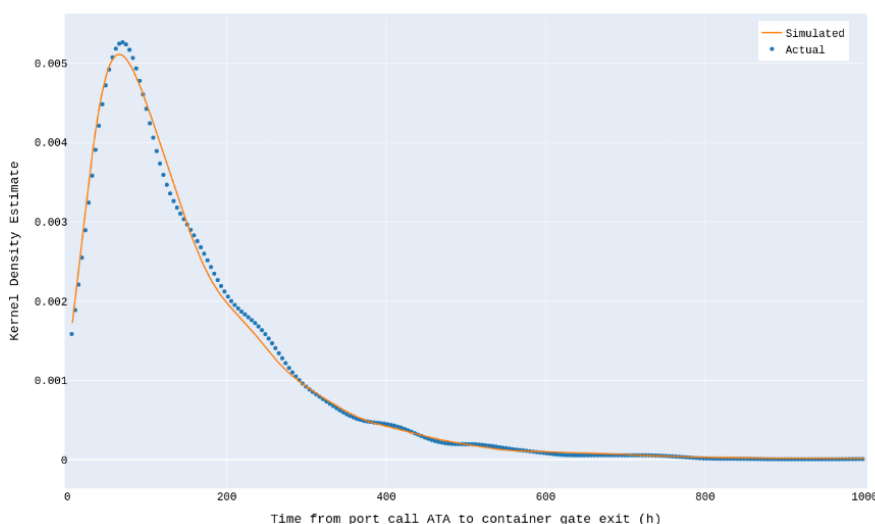


Figure 88: Kernel density estimates of the empirical distributions of actual and simulated total container dwell times in the port of Valencia.

T_V^j is the arrival time of vessel j, where $T_V^j \leq R_{max}$. These times are obtained using estimated time of arrival (ETA) prediction models, which are composed of separate predictions for the vessels' current destination, the geographical



voyage trajectory to this destination, and the duration of the voyage along this trajectory. These component models applied various machine learning techniques trained using extensive historical vessel traffic data. Destination prediction was performed using a combination of neural network embedding, classification models, and lookup tables. Trajectory prediction was based on training a global graph model representing vessel movements around the world, allowing use of graph algorithms such as Dijkstra’s algorithm to estimate the minimum cost route between any two locations based on history. Finally, sea voyage durations were estimated based on a combination of the vessel’s current speed and regression models trained with historical voyage data to compensate for typical location-specific variations in vessel speeds. The regression model training was performed similarly as described above for cargo volume prediction. An overview of the models and steps used in vessel ETA prediction is shown in Figure 89.

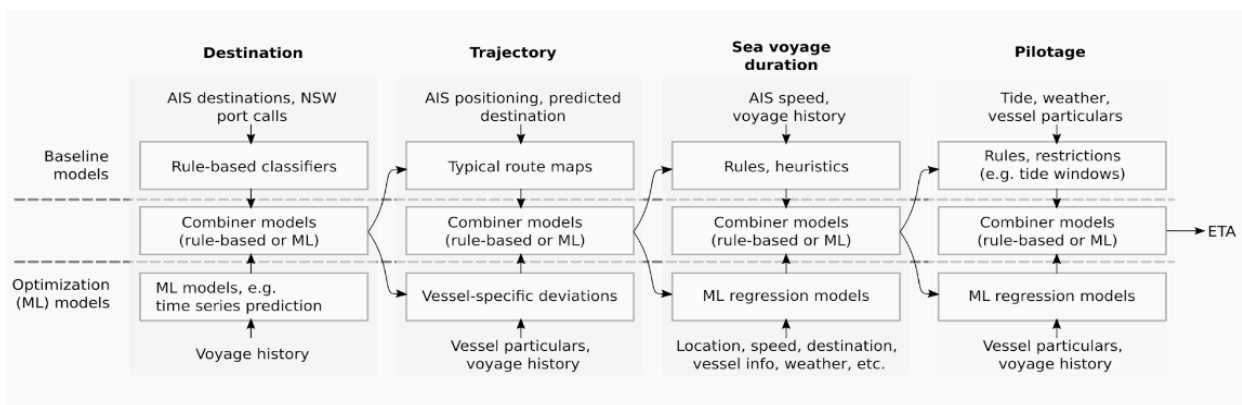


Figure 89: Vessel ETA prediction model pipeline

S is a random variable used to shift generated exit times falling on blocked days. This is a heuristic meant to disperse values more evenly around the blocked days instead of e.g. simply shifting the events to the next possible time. In the considered examples, a uniform distribution is used with a range of approximately one week.

N_j is the number of containers discharged from vessel j estimated to be leaving the port by trucks. This is a critical parameter for the simulation, which should ideally be obtained from operators or port authorities. In the final demonstration, these were estimated using dedicated ML models. Input features used in predicting the cargo exchange volumes included (in order of estimated feature importance) the estimated port call duration, vessel gross tonnage, arrival hour, vessel length, arrival weekday, vessel beam, and vessel maximum draught.

M is the Monte Carlo simulation parameter specifying how many trials are performed to obtain the cargo flow estimates.

Using the above described models, Monte Carlo simulations of the gate traffic rates $N_{R,MC}$ are implemented as described in $N_{R,MC} = \frac{1}{M} \sum_{k=1}^M \sum_j N_j \sum_{i=1}^{N_j} I_R(T_V^j + \Delta_V^i + \Delta_G^i + I_B[T_V^j + \Delta_V^i + \Delta_G^i] \cdot S)$. (1). The figure below shows weekly simulated vs. true container exit numbers over the year 2019 for a development data subset.



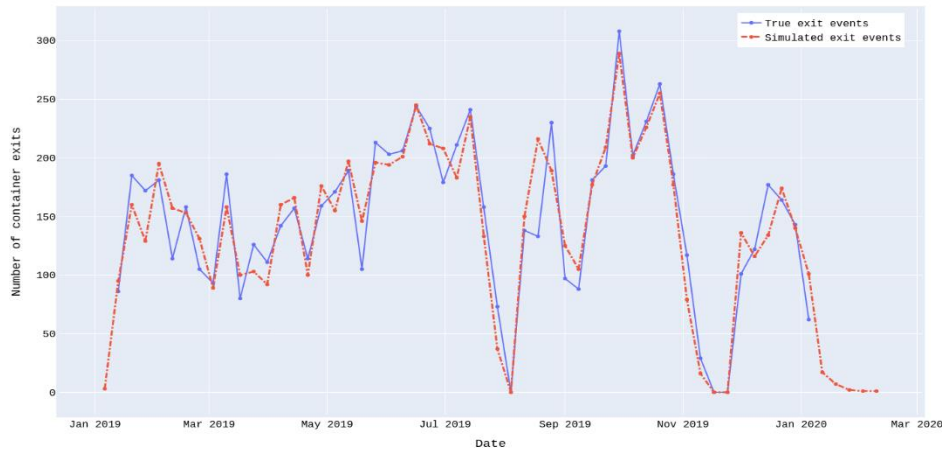


Figure 90: Simulated vs. actual weekly numbers of containers leaving port of Valencia by truck

To enable automated training and updates of those machine learning models for which source data is continuously available (such as vessel ETA prediction), a machine learning operations pipeline was implemented. This allowed orchestrating the entire model training and validation process in a cloud environment using automatically ingested and labelled training data, enabling a self-learning ML system. Figure 91 illustrates the data processing flow and computational environments used in the MLOps pipeline on a high level.

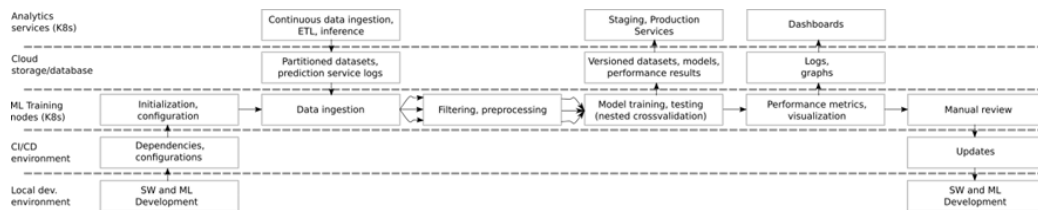


Figure 91: MLOps pipeline overview.

c.1) Offline Data Preparation:

After identifying the data sources required to approach the TTT prediction, data needed to be prepared and merged to create the final datasets to be injected as input to the ML-models. In this case, since TTT can be directly influenced by maritime and terrestrial events, the data sets exploited to feed TTT models were port calls dataset and gate access data (including gate in and gate out events).

For estimating truck entry events, some data preparation work was carried out over the Gate In raw dataset. In this process, the independent variables considered for the TTT prediction from this dataset were the following:

- Num_trucks: variable showing the number of trucks sampled for the given time and date.
- WeekDay: variable indicating the day of the week, from 0 (Monday) to 6 (Sunday), for the new sample
- Hour: the hour of the day (1-24) in which the sample was taken

To get a dataset with the number of gate-in events in regular time slots a resample of the “dataTruckEntryGates” dataset was performed. In addition to the existing rows, two extra variables “Hour” and “WeekDay” were also included. The dataset shown in Figure 92 was used as an input for the SARIMA model.



	QRP_FECHA_PASO	Hour	WeekDay	Num_trucks
0	2019-01-01 00:00:00	0	1	2
1	2019-01-01 01:00:00	1	1	1
2	2019-01-01 02:00:00	2	1	1
3	2019-01-01 03:00:00	3	1	2
4	2019-01-01 04:00:00	4	1	0
...
17515	2020-12-30 19:00:00	19	2	249
17516	2020-12-30 20:00:00	20	2	152
17517	2020-12-30 21:00:00	21	2	154
17518	2020-12-30 22:00:00	22	2	49
17519	2020-12-30 23:00:00	23	2	18

17520 rows x 4 columns

Figure 92: Gate-in dataset resampled

For estimating TTT, the preparation of the port calls dataset focused on quantifying the cargo volume that each vessel could load and discharge. Since cargo information is not available in our port call historical data, the type of vessel is identified as the most relevant factor to address the impact of vessel arrivals and departures at the port of Valencia. To execute this approach, Vessel port calls and Vessel master datasets were merged to link the vessel port calls to the characteristics of the vessels (gross tonnage, length, draught, TEU capacity). The column "category" is a classification of the vessel into 6 groups (from A to F) considering the size and capacity of the vessel (calculated from the rest of parameters). After merging both data sets, new variables are created to quantify the number of vessels per category. These variables were filled after performing a resampling process where the number of vessels for each category is calculated per hour, assuming the existing port calls.

	Fecha_Entrada	catA	catB	catC	catD	catE	catF	total
0	2020-01-02 06:00:00+01:00	1	0	0	1	0	1	3
1	2020-01-02 07:00:00+01:00	1	0	0	1	0	1	3
2	2020-01-02 08:00:00+01:00	1	1	0	1	0	1	4
3	2020-01-02 09:00:00+01:00	1	1	0	1	0	1	4
4	2020-01-02 10:00:00+01:00	1	1	1	1	0	1	5
...
2606	2020-12-30 13:00:00+01:00	7	2	1	1	0	0	11
2607	2020-12-30 14:00:00+01:00	8	2	1	1	0	0	12
2608	2020-12-30 15:00:00+01:00	8	2	1	1	0	0	12
2609	2020-12-30 16:00:00+01:00	8	2	1	1	0	0	12
2610	2020-12-30 17:00:00+01:00	7	2	1	1	0	0	11

Figure 93: Port Call Dataset

In addition, to quantify the impact of terrestrial operations in TTT estimation, the gate in and gate out dataset was also ingested and transformed. Initially, this dataset provided the timestamp of ingress, the truck plate, the timestamp of exit, the truck plate, and the container number for each combination of gate in and gate out per truck.

Considering that one of the most relevant factors for quantifying TTT is the volume of trucks inside the port facilities, the information related to truck plates



was dropped. To calculate the exact number of trucks located inside the port facilities per hour, a resampling process was again carried out. Additionally, since the timestamp for both ingress and exit was available, truck turnaround time for past events can also be calculated. To complement this approach, the time frame (hour) and the day of the week were also included in the terrestrial dataframe.

Since maritime and terrestrial dataframes provided information per hour, a final dataframe was assembled by combining both of them. Consequently, the resultant dataframe to be injected as an input for generating the TTT model was the following:

	Fecha_Entrada	Hour	WeekDay	Num_trucks	TTT	catA	catB	catC	catD	catE	catF	total
0	2020-01-02 06:00:00+01:00	6	3	138	101.212500	1	0	0	1	0	1	3
1	2020-01-02 07:00:00+01:00	7	3	533	118.761696	1	0	0	1	0	1	3
2	2020-01-02 08:00:00+01:00	8	3	654	142.066942	1	1	0	1	0	1	4
3	2020-01-02 09:00:00+01:00	9	3	375	135.889007	1	1	0	1	0	1	4

Figure 94: Final TTT data frame

c.2) Model Development:

Gate-IN forecast model

After representing the Gate-in dataset in a chart – by executing the `seasonal_decompose()` function from the python’s `statsmodels` library – it could be seen that the data has a strong seasonal component (see Figure 95). To plot this chart, the weekends were extracted from the prepared dataset.

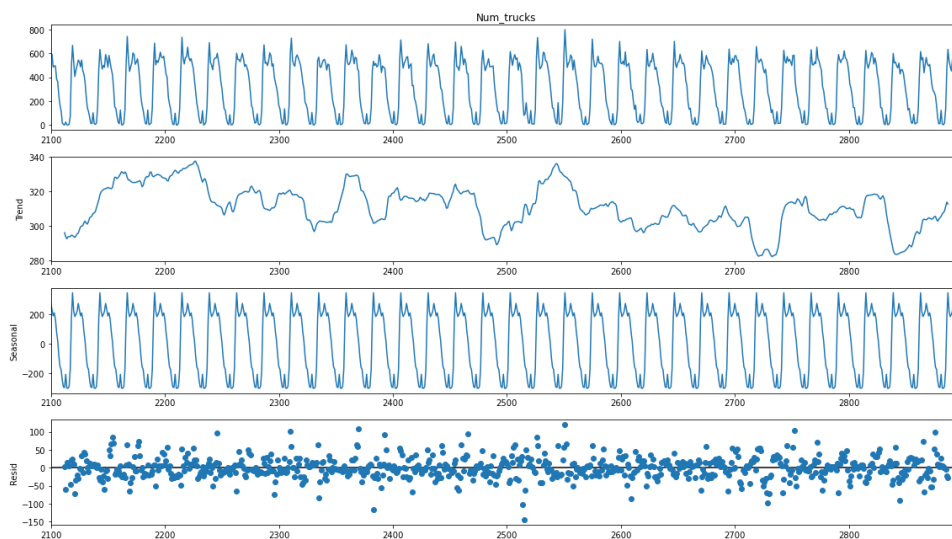


Figure 95: Gate In time series analysis

In time series forecasting, autoregressive models (a.k.a ARMA models) are used to give good results. For the gate-in prediction, the SARIMA [x] (Seasonal Autoregressive Integrated Moving Averages) model was used. The model development phase consists of finding out the $(p,d,q) \times (P,D,Q)m$ parameters [x] that allow us to generate the model and train it with the dataset prepared. The term p represents the number of autoregressive, the term d refers to the number of times the differencing between lags is applied to make the time series stationary, and the q parameter indicates the number of moving average lags to be used. The parameters P , D and Q represent the seasonal regression,



differencing and moving average orders, and m represents the number of data points (rows) in each seasonal cycle.

First, the time-series was analyzed to get the degrees of stationarity and seasonality in data as well as the level of autocorrelation and partial autocorrelation of the different time-series lags. To check if our time-series is stationary we run Augmented Dickey-Fuller Test using the `adfuller()` function of the `statsmodels` library. As we obtained a value of $4.582775e-16$, the data seems to be stationary, and we do not need to apply differencing to make time-series stationary (i.e. $d = 0$ of SARIMAX). To check the level of seasonality order (i.e. parameter m) we used the `plot_acf()` and `plot_pacf()` functions which plot the correlation of time-series by lag and the direct relationship between an observation and its lag respectively. From these charts, we verified that the time-series has a seasonality of 24 (hours) as expected.

Finally, `auto_arima()` method of the `pmdarima` library [x] was run to get the rest of SARIMA parameters (i.e. p, q, P and Q). The best combination of parameters for the model is $(2,0,2)(1,0,0)_{24}$.

```

model1 = pm.auto_arima(port_entries_week_sub, #time series
                      seasonal=True, # is the time series seasonal
                      m=24,
                      max_p=7, # max value of p to test
                      max_q=7, # max value of q to test
                      max_P=7, # max value of P to test
                      max_Q=7, # max value of Q to test
                      information_criterion='aic', # used to select best mode
                      trace=True, # prints the information_criterion for each model it fits
                      error_action='ignore', # ignore orders that don't work
                      stepwise=True, # apply an intelligent order search
                      suppress_warnings=True)

Best model: ARIMA(2,0,2)(1,0,0)[24] intercept
Total fit time: 334.577 seconds

```

Figure 96: SARIMA hyperparameter tuning for the Gate In model

Finally, as mentioned in section 4.2.1, the model was generated and fitted using the `SARIMAX()` class and its fit function of `statsmodels` library.

Part II

Devices - MT821 Specifications

The MT821 device has the following characteristics:

- Connectivity: GSM, LTE-M and NB-IoT.
- Dimensions : 86 mm x 60 mm x 33 mm.
- Weight: 286 gr.
- Battery backup: 3.7 V 7800mAh.
- Power consumption: 60uA (standby).
- Operating temperature: between -40°C and 85°C .
- Minimum accuracy: 10 m.
- GPS receiver sensitivity: -162dBm .
- Sensor: 3D accelerometer.

- Cold start: 30 s.
- Hot start: 15 s.
- Classification: IP65 against water.
- SIM slot: Micro.

Dashboard - Installation and execution process

Pre-requisites for the installation and the execution of the IoT tracking solution is the following:

- Linux operating system like Ubuntu or some similar Debian Distribution.
- Python (version >= 3.8) and ****pip**** installed.
- PostgreSQL server installed
- pgAdmin 4 (Optional).
- Docker and Docker-Compose

Once the prerequisites for implementation were established, the installation of a specific data base instance (called uc_5) was deployed with the following table structure:

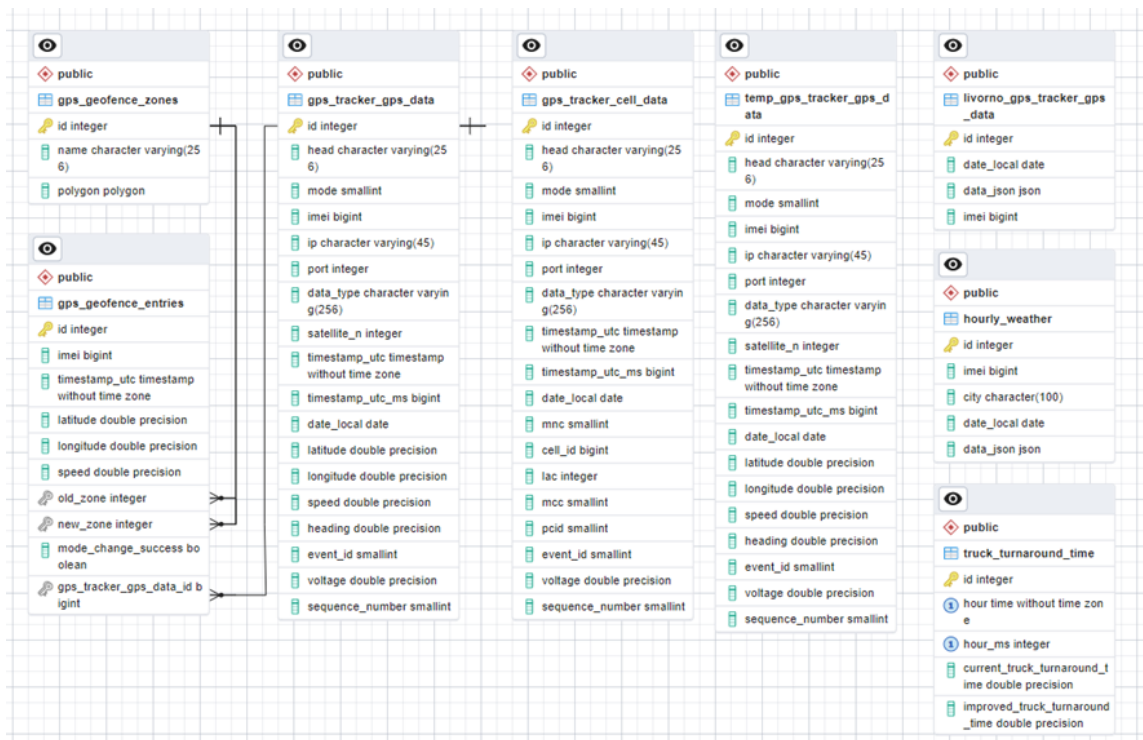


Figure 97: Port Entrance UC database structure

For the installation of the database, the executable 'install_project.sh' contained in the 'script' folder of the project was used. This script checks if PostgreSQL is installed and generates the database together with the ingenious users and data. Once the data was available, it was necessary to install a series of libraries in Python with the pip tool. Pip was used together with the list of libraries written in the file requirements.txt for this purpose.

For the execution of the dashboard and 'comm_protoco' services the developed start_projectUC5.sh script was used. The successful launch of the script was checked with the status.sh script, which should generate the following output:



```
[INFO] - APPLICATION SERVICES STATUS, IF EMPTY MAYBE NOT RUNNING
dashboa+      7022      0.0      1.6      548692      79604      ?
/home/dashboard/newdashboard/code/iot-logistics-
platform/venv/bin/python3 /home/dashboard/newdashboard/code/iot-
logistics-platform/venv/bin/flask run --host=0.0.0.0

dashboa+ 7023 0.0 0.5 108256 27332 ? python3 -m comm_protocol.mt82x

dashboa+      7042      6.1      1.7      978288      86384      ?
/home/dashboard/newdashboard/code/iot-logistics-
platform/venv/bin/python3 /home/dashboard/newdashboard/code/iot-
logistics-platform/venv/bin/flask run --host=0.0.0.
```

On the other hand, the GAD service was launched via a Docker container using the following commands inside of GAD directory:

```
docker build -t gad .
docker run -d --env-file ./GAD.list.env --name gad_container --network host -v /home/dashboard/gad:/gad gad
```

All the different parameters such as addresses, ports, folders and credentials were loaded as environment variables. These environment were declared in a file called '.env' and this file was used by all necessary scripts such as 'install_project.sh' and 'start_projectUC5.sh'.

The project was deployed via Docker-compose, through the command 'docker-compose up -d'. This command was launched from the terminal while being in the project folder inside './test_code/dc-project'. Once launched, the deployment was shown in Figure 101.

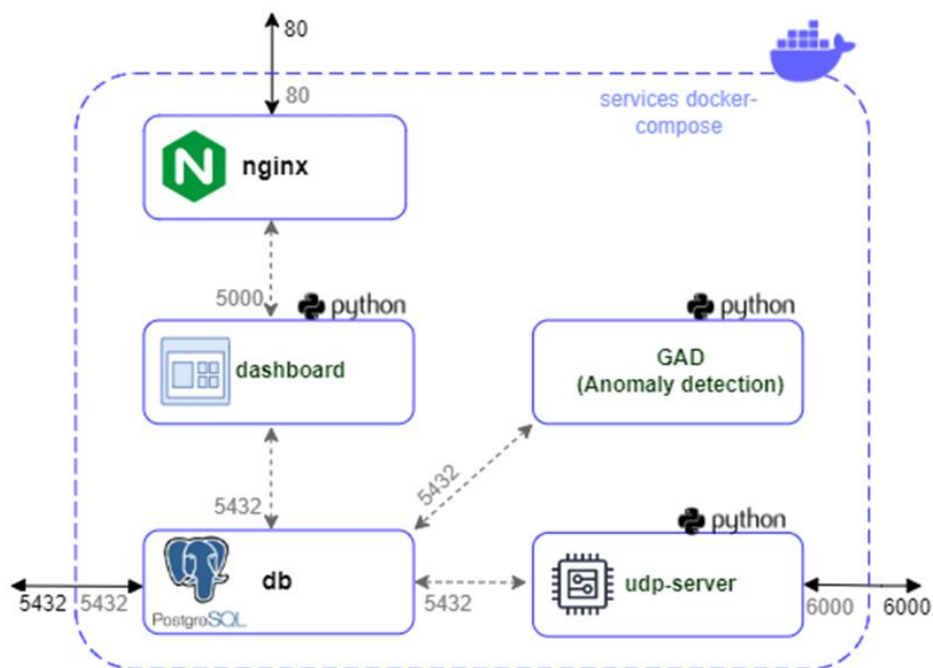


Figure 98: IoT tracking service deployment infrastructure.



Validation and Results

ADDITIONAL TEST CASES

Regarding the present use case, in section 7.2 of Deliverable D6.1 [1] a total of 12 test cases were defined. Apart from these test cases, in the course of the development of the use case, it was detected the need to create nine additional test cases with the aim to maximise the coverage of the validation of the implemented solutions. In this section, these test cases are defined and described respecting the format established in the Deliverable D6.1 [1] (see Table 73-Table 80).

Test Case Id	UC5_TC_13
Responsible Partner	AWAKE
Use Case	Situational Understanding and Predictive Models in Smart Logistics Scenarios
Test case description	Verification of availability and correctness of time event information in historical datasets
Prerequisites	Historical datasets are available on vessel, container, and truck traffic in the ports
Type of test	Data quality test
Reference standards used	None
Test Environment	Local processing environment using Python libraries, e.g. Pandas, Numpy.
Input to the system	Historical data sets from text files (e.g. CSV), databases or similar
Output of the system	Report quantifying availability of timestamps in input data sets, evaluation of timestamp validity
Data involved in the test	Datasets about: <ul style="list-style-type: none"> • History of port calls at the port • History of cargo flows at the port • History of trucks' entry/exit events • History of meteorological data • AIS data • History of vessels that arrived at the port and their characteristics
System requirements covered	UC5_SR_17
Related KPIs	Data Source Sufficiency, Data Quality
Are UC's users involved in the test?	Yes: FV and Port of Livorno
Who will perform the test?	FV and AWAKE
Test Steps	<ol style="list-style-type: none"> 1. Check availability of timestamps for events. 2. Check formal validity of timestamps. 3. Check timestamps for anomalies and outliers (e.g. clearly incorrect times or event durations).



	4. If possible cross-reference event times between related datasets (e.g. port calls and AIS data).
Risks	Timestamps are missing or incorrect.
Mitigation	Revise datasets, find alternative references for event data.
Expected result	Correct timestamps are available for majority (e.g. over 95 %) of events.

Table 73. UC5_TC_13 description

Test Case Id	UC5_TC_14
Responsible Partner	AWAKE
Use Case	Situational Understanding and Predictive Models in Smart Logistics Scenarios
Test case description	Verification of availability and correctness of resource ID information in historical datasets.
Prerequisites	Historical datasets are available on vessel, container, and truck traffic in the ports.
Type of test	Data quality test
Reference standards used	None
Test Environment	Local processing environment using Python libraries, e.g. Pandas, Numpy.
Input to the system	Historical data sets from text files (e.g. CSV), databases or similar
Output of the system	Report quantifying availability of timestamps in input data sets, evaluation of timestamp validity
Data involved in the test	Datasets about: <ul style="list-style-type: none"> • History of port calls at the port • History of cargo flows at the port • History of trucks' entry/exit events • History of meteorological data • AIS data • History of vessels that arrived at the port and their characteristics
System requirements covered	UC5_SR_18
Related KPIs	Data Source Sufficiency, Data Quality
Are UC's users involved in the test?	Yes: FV and Port of Livorno
Who will perform the test?	FV and AWAKE
Test Steps	<ol style="list-style-type: none"> 1. Check availability of resource IDs (e.g. vessel IMO or MMSI numbers, container and truck IDs (possibly anonymized). 2. Check formal validity of IDs. 3. If possible cross-reference IDs between related datasets (e.g. port calls and AIS data).
Risks	IDs are missing or incorrect.
Mitigation	Revise datasets, find alternative references for identifying and matching resources between datasets.



Expected result	Correct IDs are available for majority (e.g. over 95 %) of events and resources.
-----------------	--

Table 74. UC5_TC_13 description

Test Case Id	UC5_TC_15
Responsible Partner	AWAKE
Use Case	Situational Understanding and Predictive Models in Smart Logistics Scenarios
Test case description	Verification of vessel estimated time of arrival (ETA) prediction model performance.
Prerequisites	Historical datasets are available on vessel traffic in the ports, prediction models have been trained.
Type of test	Statistical error analysis.
Reference standards used	None
Test Environment	Local processing environment using Python libraries, e.g. Pandas, Numpy, XGBoost, Scikit-learn.
Input to the system	ETA prediction model, historical dataset containing necessary input features for prediction (model-dependent) and actual times of arrival to target areas.
Output of the system	Statistical analysis of prediction errors, e.g. mean absolute prediction error (MAE) and standard deviation over remaining time to arrival.
Data involved in the test	Datasets about: <ul style="list-style-type: none"> • History of port calls at the port • AIS data • History of vessels that arrived at the port and their characteristics
System requirements covered	UC5_SR_20
Related KPIs	Time Prediction Accuracy
Are UC's users involved in the test?	No
Who will perform the test?	AWAKE
Test Steps	<ol style="list-style-type: none"> 1. Extract subset of historical data not used in prediction model training (test data). 2. Find actual times of arrival to the target prediction areas (analysis of historical AIS data). 3. Inference of predicted ETAs using test data subset. 4. Statistical analysis of error statistics for obtained predictions.
Risks	Prediction accuracy does not meet expected criteria.
Mitigation	Collection of more or better training data, selection of new model features or model architectures, model hyperparameter optimization.
Expected result	Prediction results meet set criteria (e.g. less than 10 % MAE relative to remaining time to arrival).

Table 75. UC5_TC_15 description



Test Case Id	UC5_TC_16
Responsible Partner	AWAKE
Use Case	Situational Understanding and Predictive Models in Smart Logistics Scenarios
Test case description	Web application functionality for visualizing analytics.
Prerequisites	Data integrations, models, backend services, and user interfaces have been developed.
Type of test	Usability testing.
Reference standards used	None
Test Environment	Web application accessed by users.
Input to the system	Web service application providing graphic visualization of truck turnaround analytics.
Output of the system	User feedback on usability of the application.
Data involved in the test	N/A
System requirements covered	UC5_SR_08
Related KPIs	N/A
Are UC's users involved in the test?	Yes, ports of Valencia and Livorno.
Who will perform the test?	AWAKE, FV, CNIT.
Test Steps	<ol style="list-style-type: none"> 1. Test users are provided access and instruction for using web application. 2. Feedback is collected from users on application usability.
Risks	N/A
Mitigation	N/A
Expected result	Users either confirm usability of application or provide feedback on necessary improvements.

Table 76. UC5_TC_16 description

Test Case Id	UC5_TC_17
Responsible Partner	AWAKE
Use Case	Situational Understanding and Predictive Models in Smart Logistics Scenarios
Test case description	Web application alert functionality on truck traffic levels.
Prerequisites	Data integrations, models, backend services, and user interfaces have been developed.
Type of test	Automation testing



Reference standards used	None
Test Environment	Web application front- and backend services.
Input to the system	Configuration of alert and notification conditions.
Output of the system	Verification that system produces alerts and notifications as expected.
Data involved in the test	N/A
System requirements covered	UC5_SR_09
Related KPIs	N/A
Are UC's users involved in the test?	No
Who will perform the test?	AWAKE
Test Steps	<ol style="list-style-type: none"> 1. Test automation scripts are configured to trigger necessary alerts and notifications. 2. Automation tests verify that required alerts and notifications are shown in the web application.
Risks	N/A
Mitigation	N/A
Expected result	Alerts and notifications are produced according to specifications

Table 77. UC5_TC_17 description

Test Case Id	UC5_TC_18
Responsible Partner	AWAKE
Use Case	Situational Understanding and Predictive Models in Smart Logistics Scenarios
Test case description	Web application authentication functionality.
Prerequisites	Data integrations, models, backend services, and user interfaces have been developed.
Type of test	Automation testing
Reference standards used	None
Test Environment	Web application front- and backend services.
Input to the system	Test user logs in to system.
Output of the system	Verification that log in procedure operates correctly.
Data involved in the test	N/A



System requirements covered	UC5_SR_24
Related KPIs	N/A
Are UC's users involved in the test?	No
Who will perform the test?	AWAKE
Test Steps	<ol style="list-style-type: none"> 1. Test automation scripts are configured to log in to the web application using correct and incorrect credentials. 2. Automation tests verify that the authentication procedure works correctly.
Risks	N/A
Mitigation	N/A
Expected result	Users with correct credentials can log in, others cannot.

Table 78. UC5_TC_18 description

Test Case Id	UC5_TC_19
Responsible Partner	CNIT
Use Case	Situational Understanding and Predictive Models in Smart Logistics Scenarios
Test case description	Testing the functioning of the communication interface between the DVL and the Tuscan Port Community System (TPCS) and between DVL and the M2M Platforms.
Prerequisites	A DVL connecting all data sources is up and running.
Type of test	This is mainly a software test to check the information retrieving from the M2M platform and TPCS by means of DVL.
Reference standards used	None
Test Environment	Tools such as Postman, Soap UI or similar.
Input to the system	HTTP REST request with the necessary parameters.
Output of the system	Data coming from the sources being queried.
Data involved in the test	Trucks/Vessels arrival/departure data, Meteorological data, AIS data.
System requirements covered	UC5_SR_21 and UC5_SR_22
Related KPIs	Data Availability, Data Source Sufficiency, Data Quality
Are UC's users involved in the test?	Yes, Port of Livorno.
Who will perform the test?	CNIT

Test Steps	<ol style="list-style-type: none"> Trucks/Vessels arrival/departure data, Meteorological data, AIS data are sent to a local server at the laboratory; Through DVL which uses specific APIs, it will be possible to read the information.
Risks	Impossibility to reach the information sources.
Mitigation	Periodical check of the connection between DVL and data sources.
Expected result	All data provided by the sensor are recovered by means of DVL.

Table 79. UC5_TC_19 description

Test Case Id	UC5_TC_20
Responsible Partner	CNIT
Use Case	Situational Understanding and Predictive Models in Smart Logistics Scenarios
Test case description	Testing the functioning of the communication interface between the AI-based Platform and the DVL.
Prerequisites	A DVL connecting all data sources is up and running.
Type of test	This is mainly a software test to check the connection between the AI-based platform and DVL..
Reference standards used	None
Test Environment	Tools such as Postman, Soap UI or similar.
Input to the system	HTTP REST request with the necessary parameters.
Output of the system	Data coming from the sources being queried.
Data involved in the test	Trucks/Vessels arrival/departure data, Meteorological data, AIS data.
System requirements covered	UC5_SR_23
Related KPIs	Data Availability, Data Source Sufficiency, Data Quality
Are UC's users involved in the test?	Yes, Port of Livorno.
Who will perform the test?	CNIT
Test Steps	<ol style="list-style-type: none"> Trucks/Vessels arrival/departure data, Meteorological data, AIS data are sent to a local server at the laboratory; Through DVL which uses specific APIs, AI-based platform can retrieve the information to perform its operations (predictions made by the models In production).
Risks	Impossibility to establish a connection with the DVL.
Mitigation	Periodical check of the connection between DVL and AI-based platform.
Expected result	All necessary data for AI-based platform are obtained by means of DVL.

Table 80. UC5_TC_20 description

Test Case Id	UC5_TC_21
Responsible Partner	UPV
Use Case	Situational Understanding and Predictive Models in Smart Logistics Scenarios
Test case description	Testing the dashboard visualisation with real time and historical data in the Valencia Port.
Prerequisites	The dashboard and tracker devices are fully working. Historical data is available.
Type of test	Software (dashboard) and hardware (tracker device) tests.
Reference standards used	None
Test Environment	Web application - front and backend services.
Input to the system	Tracker dataframes with geolocation and speed.
Output of the system	Data visualisation on the Dashboard.
Data involved in the test	Truck geolocation and speed.
System requirements covered	UC5_SR_07, UC5_SR_08, UC5_SR_11, UC5_SR_17
Related KPIs	Truck Turnaround Time, Idling Time, Time Prediction Accuracy, IoT Position Accuracy.
Are UC's users involved in the test?	Yes, Port of Valencia.
Who will perform the test?	UPV
Test Steps	<ol style="list-style-type: none"> 1. Attach the tracker device to a truck that will enter the port of Valencia. 2. Confirm that geofences and real time data are working.
Risks	Tracker may lose connectivity.
Mitigation	Use alternative Cellular technology (GSM).
Expected result	The truck position and speed is shown correctly inside the port.

Table 81. UC5_TC_21 description

TEST CASES VERIFICATION

Test Case Id	UC5_TC_01
Test case description	Test the quality of historical datasets for the development of predictive and simulation models
System requirements covered	UC5_SR_04
Expected result	Sufficient historical data is available to develop predictive models.

Actual result	<p>The purpose of this test case was to verify that sufficient historical data is available to develop predictive models according to the targets of the Use Case. The main dataset requirements identified during the work were:</p> <ul style="list-style-type: none"> • History of port calls at the port. • History of cargo flows at the port. • History of trucks' entry/exit events. • AIS data. • History of vessels that arrived at the port and their characteristics. <p>The primary criterion in estimating the sufficiency of data for modeling was whether it was possible to meet model performance KPIs using the available data. Model performance is estimated in test case UC5_TC_03; to summarize, it was found that in port of Valencia, data for port calls, cargo flows, AIS, and vessel information were sufficient to enable model development according to performance targets, but access to trucks' entry and exit events was limited both in time coverage (data available only for 2019) and vehicle coverage (not all vehicles exiting the port were captured in the data). This causes some inaccuracy both in training models and evaluating their true accuracy. For port of Livorno, there was better coverage of truck gate events over time, but lack of reliable baseline data for truck turnaround time estimation was more significant, as less than 50 % of truck gate exits with containers could be associated with truck gate entries within a realistic turnaround time margin (8 h).</p> <p>From the perspective of ISO/IEC 25012 data quality characteristics and related ISO/IEC 25024 data quality properties, the relevant characteristics for historical data include accuracy, completeness, consistency, and credibility. For the purposes of the modeling performed in the project, we find that the data accuracy, consistency, and credibility are sufficient (time event data is obtained from official IoT sensor systems, port call records, or mandatory positioning systems, and no issues e.g. regarding format, semantic consistency, or detectable inaccuracy in recorded values were observed). However, as discussed above, the historical datasets available for the development are partially incomplete regarding the measurement of truck turnaround times, which should be addressed in further exploitation of the results.</p> <p>All in all, there were some issues with data coverage, but this did not prevent development of models according to the Use Case objectives.</p>
Passed/Failed	Passed

Table 82. UC5_TC_01 verification

Test Case Id	UC5_TC_02
Test case description	Integration of different data sources
System requirements covered	UC5_SR_04, UC5_SR_06
Expected result	Necessary data sources are available for operating the developed predictive models
Actual result	<p>The purpose of the test is to verify that necessary data sources are available for operating the developed predictive models. These include the following data types:</p> <ul style="list-style-type: none"> • Port call data. • Cargo flow data. • Trucks' entry/exit events. • Meteorological data.



	<ul style="list-style-type: none"> • Vessel's characteristics. • AIS data. <p>For port of Valencia, all of the listed data sources except cargo flow data are available and integrated to relevant components of the developed application (however, meteorological data is not used, as it was considered not to be essential for the use case). For port of Livorno, current port call data, cargo flow data, and truck entry/exit event data are not available through online APIs allowing integration, preventing online demonstration of the developed models. Regarding ISO/IEC 25012 and 25024 data quality characteristics and properties for online service deployment, in addition to the issues described for UC5_TC_01, data currentness including frequency and timeliness of updates is critical. For the data sources available in the online service, these are sufficient. Overall, the service output time resolution is on a time scale of hours or days at minimum, and all data sources are updated sufficiently frequently and with sufficiently up to date information to produce the target predictions.</p> <p>All in all, although there were some issues with data availability, this did not prevent online demonstration of the developed models according to the Use Case objectives.</p>
Passed/Failed	Passed.

Table 83. UC5_TC_02 verification

Test Case Id	UC5_TC_03
Test case description	Evaluate prediction model accuracy analysis as part of training process
System requirements covered	UC5_SR_01, UC5_SR_02
Expected result	KPIs for time prediction accuracy are met
Actual result	<p>The purpose of the test is to quantify how closely the performance KPIs (primarily Time Prediction Accuracy) for the predictive analytics are met. This testing also enables quantifying the effects of insufficient historical data as considered in UC5_TC_01, as data problems affect model performance.</p> <p>Model accuracy testing was performed by generating test data for ML-based prediction components using nested cross-validation. In this process, cross-validation is used in model hyperparameter optimization. Here for each potential model configuration, the dataset is split e.g. to three groups of approximately equal size, and the model training is repeated three times using a different subset each time to provide an accuracy metric for the tested model configuration. Furthermore, in nested cross-validation, the entire dataset is first divided e.g. into ten subsets, and the hyperparameter optimization (using cross-validation) is repeated ten times with a different 1/10 subset reserved as test data in each round. This process enables using a maximal amount of data for model training, while producing a maximal amount of test data on model performance. The drawback is that this is computationally complex, e.g. if there are 300 combinations of model hyperparameter candidates defined for a model, its training is performed 9000 times with the above outlined procedure.</p> <p>In the AWA model pipeline, the component ML models were tested separately using nested cross-validation as described above, and the total pipeline was tested also by using the resulting model validation data as inputs to subsequent pipeline models. This enables evaluating how inaccuracies in intermediate models affect the outputs of later steps in the pipeline. Regarding the performance of</p>



	<p>the final output of the modeling pipeline (long-term truck turnaround time predictions), the median absolute relative prediction error is 10 %, which meets the original KPI for Time Prediction Accuracy. In this metric, the comparison was done with daily maximum values because especially high turnaround times are of interest for the use case, and generally the reference turnaround times can be close to 0 hours, causing large outliers in computing relative errors. It is possible that this prediction model could be further improved by extending the truck turnaround reference data to cover all truck traffic in the port; as noted regarding UC5_TC_01, currently part of truck gate events in port of Valencia are not included in the data, and the time coverage of data available for model training was limited to 2019.</p> <p>When the observed rates are replaced by predictions from the preceding pipeline model components (predicted discharge volumes per port call, predicted container dwell times in port), some accuracy loss is observed due to compounding error in the predictions. Specifically, the median absolute relative prediction error when using only predicted cargo volumes and traffic rates in the turnaround time model was 13 %, i.e. the accuracy was reduced by three percentage points. For reference, if the traffic rate input is completely removed from the turnaround prediction model (i.e. the full model pipeline is not used), the predictions have significant bias, and the median absolute relative prediction error is 21 %, or 11 percentage points higher than with the full model, highlighting the benefit from the developed prediction pipeline.</p> <p>Similar performance analyses were performed for other prediction models in the prediction pipeline regarding Time Prediction Accuracy; the results are summarized for the KPIs in the main document.</p> <p>In general, developed models meet the KPIs when sufficient input data is available.</p>
Passed/Failed	Passed

Table 84. UC5_TC_03 verification

Test Case Id	UC5_TC_04
Test case description	Performance evaluation of models deployed in production
System requirements covered	UC5_SR_01, UC5_SR_02, UC5_SR_03, UC5_SR_10
Expected result	Test service output corresponds with test results using historical data and current observations.
Actual result	<p>The purpose of the test is to evaluate the performance of predictions made by the models running in an online service. Main performance tests for the developed models are performed using historical datasets to provide sufficient statistical coverage for estimating the results. However, as there may be differences in the statistics of the modeled processes as seen in historical data versus current inputs, smaller subsets of data are collected from the online service running the developed models, and the current model inputs and outputs are compared to the historical data analysis results. Such testing can be used to adjust or reconfigure/retrain the models as needed in case of discrepancies between history and current processes.</p> <p>All in all, the service is up and running, functioning as expected.</p>
Passed/Failed	Passed

Table 85. UC5_TC_04 verification



Test Case Id	UC5_TC_05
Test case description	Validate the reception of trucks' geoposition data in the M2M platform
System requirements covered	UC5_SR_07, UC5_SR_19
Expected result	All data provided by the sensor is returned by the M2M platform in near real-time.
Actual result	At the port of Valencia scenario, data collected by IoT tracking devices installed on trucks for obtaining geopositioning data were not finally integrated in the M2M platform of the Port of Valencia due to the internal port strategy. To enable the storage and processing of this information, data was directly stored in UPV server where tracking information is processed and represented in a dashboard interface. For the port of Valencia Livorno scenario, the geopositioning data collected by IoT tracking devices is received and stored in Symphony M2M Symphony platform. Two different plugins were developed allowing on one hand to collect data coming from the IoT tracking device and on the other hand to expose such data to DVL. The DVL is able to aggregate such data according to a given data model and expose it through a RESTful interface so that a Web Application (developed by UPV) can use it by providing a graphical representation of the main track recorded during the on-field tests.
Passed/Failed	Passed

Table 86. UC5_TC_05 verification

Test Case Id	UC5_TC_06
Test case description	Onboard supply chain network slice templates and NF descriptors
System requirements covered	UC5_SR_13
Expected result	The onboarded network slice templates and related descriptors are successfully maintained by the cross-layer MANO to create new vertical services and network slices instances
Actual result	The original goal of this set of test cases was to develop an AI/ML algorithm for closed-loop slice optimization based on the combination of application/M2M (from DVL) collected and processed data with network related data (from the 5GC). However, as explained in D6.2 [2] for the related development activities, at the Port Entrance UC the cross-layer MANO does not control and manage any 5G network, and the DVL deployed on the field cannot provide insightful data for the network slice optimization purposes. For these reasons, it was agreed with the FV and CNIT to not provide such AI/ML driven network slice optimization capabilities. This test case validation can be referred to the validation of the Automated Robots with Heterogeneous Network test case whose identifier is UC1_TC_04. The onboarding of supply chain network slice templates (NSTs) and NF descriptors was demonstrated during the mid-term review. An NST was onboarded on the NSMF catalogue and the Virtual Network Functions (VNFs) along with the corresponding Network Service Descriptors (NSDs) were onboarded on the Open-Source MANO (OSM) through its GUI.
Passed/Failed	N/A* *For above reasons the test was not run

Table 87. UC5_TC_06 verification

Test Case Id	UC5_TC_07
Test case description	Automated deployment of supply chain network slice instance
System requirements covered	UC5_SR_14, UC5_SR_15
Expected result	A new network slice instance is created, all the related network and computing resources have been allocated and the 5G Core NFs are up and running and ready to be configured.
Actual result	Please refer to actual result of UC5_TC_6. Additionally, this test case validation can be referred to the validation of the Automated Robots with Heterogeneous Network test case whose identifier is UC1_TC_05. However, as explained in the activity "VALENCIA_LIVORNO_UC5_development_09" in Deliverable D6.2 [2], the Situational Understanding in Smart Logistic Scenario makes use of a commercial mobile network which cannot be controlled and managed by the cross-layer MANO. For this reason, this test cases cannot be considered applicable.
Passed/Failed	N/A

Table 88. UC5_TC_07 verification

Test Case Id	UC5_TC_08
Test case description	Automated termination of supply chain network slice instance
System requirements covered	UC5_SR_14, UC5_SR_15
Expected result	A new network slice instance is created, all the related network and computing resources have been allocated and the 5G Core NFs are up and running and ready to be configured.
Actual result	Please refer to actual result of UC5_TC_6. This test case validation can be referred to the validation of the Automated Robots with Heterogeneous Network test case whose identifier is UC1_TC_06. However, as explained in the activity "VALENCIA_LIVORNO_UC5_development_09" in Deliverable D6.2 [2], the Situational Understanding in Smart Logistic Scenario makes use of a commercial mobile network which cannot be controlled and managed by the cross-layer MANO. For this reason, this test cases cannot be considered applicable.
Passed/Failed	N/A

Table 89. UC5_TC_08 verification

Test Case Id	UC5_TC_09
Test case description	Manual scaling of a running supply chain network slice instance
System requirements covered	UC5_SR_14, UC5_SR_15
Expected result	The network slice instance is manually modified, all the related network and computing resources have been scaled and the 5G Core resources accordingly.
Actual result	Please refer to actual result of UC5_TC_6
Passed/Failed	N/A

Table 90. UC5_TC_09 verification



Test Case Id	UC5_TC_10
Test case description	Interaction with the Data Virtualization Layer (DVL) to start collecting IoT application data from deployed M2M platforms
System requirements covered	UC5_SR_12
Expected result	The cross-layer MANO is able to collect data from DVL
Actual result	With the reference to UC5_TC_06, the integration between the MANO platform and the DVL component was not performed. As explained in UC5_TC_06, DVL deployed on the field cannot provide insightful data for the network slice optimization purposes, so this test case cannot be considered applicable. Nevertheless, the DVL was integrated with available M2M platforms and it was validated in the DLT's UC (Scenario 1, Scenario 2, Scenario 3 and Scenario 4) as described in Setup and Execution and Validation and Results of this deliverable.
Passed/Failed	N/A

Table 91. UC5_TC_10 verification

Test Case Id	UC5_TC_11
Test case description	Interaction with the Data Virtualization Layer (DVL) to stop collecting IoT application data from deployed M2M platforms.
System requirements covered	UC5_SR_12
Expected result	The cross-layer MANO does not collect DVL data anymore
Actual result	Please refer to actual result of UC5_TC_6. As explained in UC5_TC_06, DVL deployed on the field cannot provide insightful data for the network slice optimization purposes, so it cannot be considered applicable.
Passed/Failed	N/A

Table 92. UC5_TC_11 verification

Test Case Id	UC5_TC_12
Test case description	Automated slice scaling triggered by AI\ML platform using data application collected from DVL.
System requirements covered	UC5_SR_12, UC5_SR_14, UC5_SR_15, UC5_SR_16
Expected result	The cross-layer MANO correctly scales the network slice instance
Actual result	Please refer to actual result of UC5_TC_6. The original goal of this activity was to develop an AI/ML algorithm for closed-loop slice optimization-based on the combination of application/M2M (from DVL) collected and processed data with network related data (from the 5G Core). However, as already explained the cross-layer MANO cannot control and manage any 5G network. Therefore, this test case, cannot be considered applicable.
Passed/Failed	N/A

Table 93. UC5_TC_12 verification



Test Case Id	UC5_TC_13
Test case description	The purpose of this test case was to verify that necessary time event information for model development is available, that the available time information is correct. This is closely connected with test case UC5_TC_01, where some missing data relevant for modeling was identified. For port of Valencia, it was determined during development that necessary timestamps were available for development, these were formally valid, there were not a significant number of outliers such as clearly incorrect times or durations. For port of Livorno, the truck gate exit data was missing a large percentage of events which could be associated to gate entries, causing uncertainty and significant outliers in estimating true truck turnaround times (as it is not possible to know whether the available entry and exit events for a given vehicle actually correspond to the same port visit).
System requirements covered	UC5_SR_17
Expected result	Necessary data for model development is available.
Actual result	There were some issues with data coverage, but this did not prevent development of models according to the Use Case objectives.
Passed/Failed	Passed

Table 94. UC5_TC_13 verification

Test Case Id	UC5_TC_14
Test case description	The purpose of this test case was to verify that resources in different datasets can be connected correctly.
System requirements covered	UC5_SR_18
Expected result	Necessary IDs are available for combining different datasets such as vessel voyages, port calls, and cargo events.
Actual result	As expected.
Passed/Failed	Passed

Table 95. UC5_TC_14 verification

Test Case Id	UC5_TC_15
Test case description	This test case regards performance evaluation of the vessel schedule prediction components in the AWA model pipeline. The models were evaluated during training using a similar nested cross-validation procedure as described for UC5_TC_03. In addition, the models were operated as online services with live input data sources and the produced predictions were logged for performance analysis. Finally, the log data collected from online predictions was used to perform a retraining of the ETA optimization model. Figure 99 shows the performance of the final ETA model in terms of absolute prediction error mean and median vs. remaining travel time to port. With the training set of 2154 voyages to port of Valencia, the final optimized model reaches a median absolute error of approximately 5 % relative to remaining travel time up to 200 h before arrival (this limit corresponds to > 30 ongoing voyages for computing the error statistics).



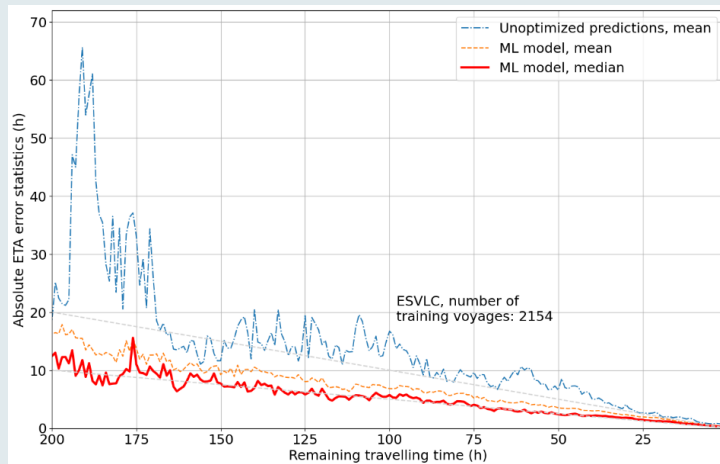


Figure 99: Final vessel ETA prediction model accuracy statistics*.

* Blue dash-dot curve: unoptimized predictions, orange dashed curve: ML-optimized predictions mean, red curve: ML-optimized predictions median. Gray dashed lines indicate the 5 % and 10 % relative error levels.

System requirements covered	UC5_SR_20
Expected result	Relative prediction error less than 10 %.
Actual result	Relative prediction error between 5-10 %.
Passed/Failed	Passed

Table 96. UC5_TC_15 verification

Test Case Id	UC5_TC_16
Test case description	The web application visualizing the developed model outputs is made available for test users, and features such as visualization of the output data are improved according to feedback as needed.
System requirements covered	UC5_SR_08
Expected result	Web application functionalities work according to requirements.
Actual result	As expected.
Passed/Failed	Passed

Table 97. UC5_TC_16 verification

Test Case Id	UC5_TC_17
Test case description	According to the service requirements, an alert functionality was included in the web service to highlight predicted scenarios where the output variables (e.g. truck turnaround time) exceed pre-set thresholds. This functionality was tested by adjusting the threshold and verifying that the alerts are visualized correctly in the web application.



System requirements covered	UC5_SR_09
Expected result	Alerts are generated in the service according to specification.
Actual result	As expected.
Passed/Failed	Passed

Table 98. UC5_TC_17 verification

Test Case Id	UC5_TC_18
Test case description	The service authentication functionality was implemented using the Keycloak identity and access management solution. Tests were performed to verify that users are able to sign in with correct credentials, are not able to sign in with incorrect credentials, access without a user account is not allowed, new credentials can be created by valid users, and users can correctly log out of the service.
System requirements covered	UC5_SR_24
Expected result	Service authentication functions according to specification.
Actual result	As expected.
Passed/Failed	Passed

Table 99. UC5_TC_18 verification

Test Case Id	UC5_TC_19
Test case description	Testing the functioning of the communication interface between the DVL and the Tuscan Port Community System (TPCS) and between DVL and the M2M Platforms
System requirements covered	UC5_SR_21 and UC5_SR_22
Expected result	All data provided by the sensor are recovered by means of DVL
Actual result	<p>As part of the Scenarios from the DLT's UC, several RESTful interfaces were developed at DVL in order to allow the cross-DLT layer, based on TrustOS, to retrieve data for the maritime events from both seaports (e.g., GateIn, GateOut, VesselArrival, VesselDeparture and sealRemoved). More precisely, the DVL was integrated with the following M2M platforms and data sources:</p> <ul style="list-style-type: none"> • Mobius OneM2M M2M Platform: provides meteorological data for the Port of Livorno; • PISystem M2M Platform: provides GateIn and GateOut data for the Port of Valencia; • Symphony M2M Platform: provides data coming from the IoT tracking device used in the Port of Livorno; • Eclipse OM2M M2M Platform: provides data coming from the sensors installed on iNGENIOUS Smart Container; • TPCS (Tuscan Port Community System): provides GateIn, GateOut, VesselArrival and VesselDeparture data for the Port of Livorno. <p>For all listed data sources, several wrappers were developed to allow the communication with the DVL. The DVL extracts all data sets according to a given data model and exposes them through a set of</p>



	RESTful interfaces. The usage and testing of such interfaces is further described in Setup and Execution and Validation and Results of Deliverable 6.3.
Passed/Failed	Passed

Table 100. UC5_TC_19 verification

Test Case Id	UC5_TC_20
Test case description	Testing the functioning of the communication interface between the AI-based Platform and the DVL
System requirements covered	UC5_SR_23
Expected result	All necessary data for AI-based platform are obtained by means of DVL
Actual result	Please refer to actual result of UC5_TC_6
Passed/Failed	N/A

Table 101. UC5_TC_20 verification

Test Case Id	UC5_TC_21
Test case description	Testing the dashboard visualisation with real time and historical data in the Valencia Port
System requirements covered	UC5_SR_07, UC5_SR_08, UC5_SR_11, UC5_SR_17
Expected result	The truck position and speed is shown correctly inside the port
Actual result	Predictions and real TTT times can be visualized for real-time and historical data in the graphs integrated in the visualization framework developed by AWA, as specified in Issues on execution of Deliverable 6.3. On the other hand, historical IoT tracking data can be visualized in the dashboard developed by UPV and explained in Section 4.2.2 of this deliverable.
Passed/Failed	Passed

Table 102. UC5_TC_21 verification



Annex IV: AGV’s UC - Improved Drivers’ Safety with MR and Haptic Solutions

Below is additional information about setup, execution, validation, and results of the AGV use case.

Setup and Execution

Part I

The specifications about the 5G connection of each device using the mobile phone via tethering are the following:

- Mobile phone model: ASUS Smartphone for Snapdragon Insiders I007D. EXP21 Smartphone.
- Ethernet Connector: Tethering by USB-C Interface to Ethernet.



Figure 100: 5G Network connection setup

The details of each 5G modem and the relation with the AGVs and other devices are described below.

Tests Platform	AGV	IMSI friendly name	IMSI	Profile 5QI	IP	Comment	Modem Label
AGV-B	Nokia	IMSI-AGV-B-1	214380000000013	6	10.45.100.13	Mounted in AGV	ASUS2
AGV-B	Nokia	IMSI-AGV-B-2	214380000000014	6	10.45.100.14	Connected to Cockpit	ASUS3
AGV-A	ABB	IMSI-AGV-A-1	214380000000010	6	10.45.100.10	Mounted in AGV	ASUS4
AGV-C	5COMM	IMSI-AGV-C-1	214380000000012	6	10.45.100.12	Mounted in AGV	ASUS5
ALL	ALL	IMSI-P-1	214380000000006	9	10.45.100.6	KPIs Operations Center Connectivity	ASKEY-1
ALL	ALL	IMSI-P-2	214380000000009	9	10.45.100.9	Used for Load Test during Use Cases	ASUS1

Figure 101: Relation between modems and devices

The different profiles 5QI had been used to give different priorities to the different devices in order to test how this affect to the use of 5G network resources.

Part II

Details of GloveSense haptic gloves are:

- Motion capturing
 - **9-axis orientation** sensor in the wrist
 - **4 flexion and extension** sensors (thumb, index, middle, ring)
 - **Computer-vision** algorithms to enhance finger tracking in the field of view of HMDs cameras
- Haptic feedback
 - **2 Linear Resonant Actuators** (thumb, index) of vibration magnitude 1.8G
 - **1 Voice Coil** actuator in the palm of vibration magnitude 4.3G
- Force feedback
 - 4 passive modules (thumb, index, middle, ring) that **restrict flexion**
 - Programmable force magnitude until a maximum of **20N** at the fingertips, with a resolution of 0.2N



Figure 102: SenseGlove haptic gloves

Validation and Results

TEST CASES VERIFICATION

Test Case Id	UC2_TC_01
Test case description	Perform measurements of 5G millimeter wave coverage in Segovia.
System requirements covered	UC2_SR_03, UC2_SR_11

Expected result	Low latency measurements.
Actual result	Low latency measurements were obtained in the Segovia tests.
Passed/Failed	Passed

Table 103. UC2_TC_01 verification

Test Case Id	UC2_TC_02
Test case description	AGV teleoperation via 5G millimeter wave in Segovia.
System requirements covered	UC2_SR_02, UC2_SR_04, UC2_SR_06, UC2_SR_09
Expected result	Low latency. Proper AGV teleoperation
Actual result	The AGV had a proper teleoperation thanks to the low latency. The KPIs were measured during the final test in Valencia Port.
Passed/Failed	Passed

Table 104. UC2_TC_02 verification

Test Case Id	UC2_TC_03
Test case description	Immersive cockpit.
System requirements covered	UC2_SR_07, UC2_SR_08, UC2_SR_12
Expected result	Good performance of the visualization of the AGV environment through virtual reality glasses. Good software capture of the hardware interaction of the cockpit.
Actual result	The visualization of the AGV environment through virtual reality glasses and the software capture of the hardware interaction of the cockpit were good enough for AGV teleoperation.
Passed/Failed	Passed

Table 105. UC2_TC_03 verification

Test Case Id	UC2_TC_04
Test case description	Fivecomm's cockpit integration for AGV teleoperation.
System requirements covered	UC2_SR_02, UC2_SR_03, UC2_SR_04, UC2_SR_07, UC2_SR_09
Expected result	Successful teleoperation with fully integrated VR glasses and haptics gloves. Haptic feedback from the AGV. Future integration with Nokia.
Actual result	Successful teleoperation with fully integrated VR glasses and haptics gloves (including new gloves from Senseglove). Haptic feedback from the AGV.
Passed/Failed	Passed

Table 106. UC2_TC_04 verification



Test Case Id	UC2_TC_05
Test case description	Perform measurements of 5G millimeter wave coverage in Valencia Port.
System requirements covered	UC2_SR_03, UC2_SR_11
Expected result	Low latency measurements.
Actual result	Low latency measurements were obtained in the Valencia Port tests.
Passed/Failed	Passed

Table 107. UC2_TC_05 verification

Test Case Id	UC2_TC_06
Test case description	ASTI AGV teleoperation via 5G millimeter wave in Burgos.
System requirements covered	UC2_SR_02, UC2_SR_04, UC2_SR_06, UC2_SR_09
Expected result	Low latency. Proper AGV teleoperation.
Actual result	The ASTI AGV was successfully teleoperated in the Burgos thanks to the appropriate latency conditions.
Passed/Failed	Passed

Table 108. UC2_TC_06 verification

Test Case Id	UC2_TC_07
Test case description	ASTI AGV teleoperation via 5G millimeter wave in Valencia Port.
System requirements covered	UC2_SR_02, UC2_SR_04, UC2_SR_06, UC2_SR_09
Expected result	Low latency. Proper AGV teleoperation.
Actual result	The ASTI AGV was successfully teleoperated in Valencia Port thanks to the appropriate latency conditions.
Passed/Failed	Passed

Table 109. UC2_TC_07 verification

KPIS

Coverage

Defined as the maximum distance at which the 5G network coverage continues to be good.

The measured have been performed in Segovia and Valencia Port, both places where the 5G system have been deployed.

End-to-end latency



The time it takes for a data packet to travel through the 5G network from source to destination.

In this use case we have measures latencies of two types: related to the driving of the AGVs by calculating the RTT in the driving commands; and referring to the network itself by sending pings from a mobile device.

These latencies have been measured in the different scenarios proposed in the test cases: AGV-A, AGV-B, AGV-C in Valencia Port and AGV-B in Burgos; also, for the 5G network in general in Valencia Port and Segovia.

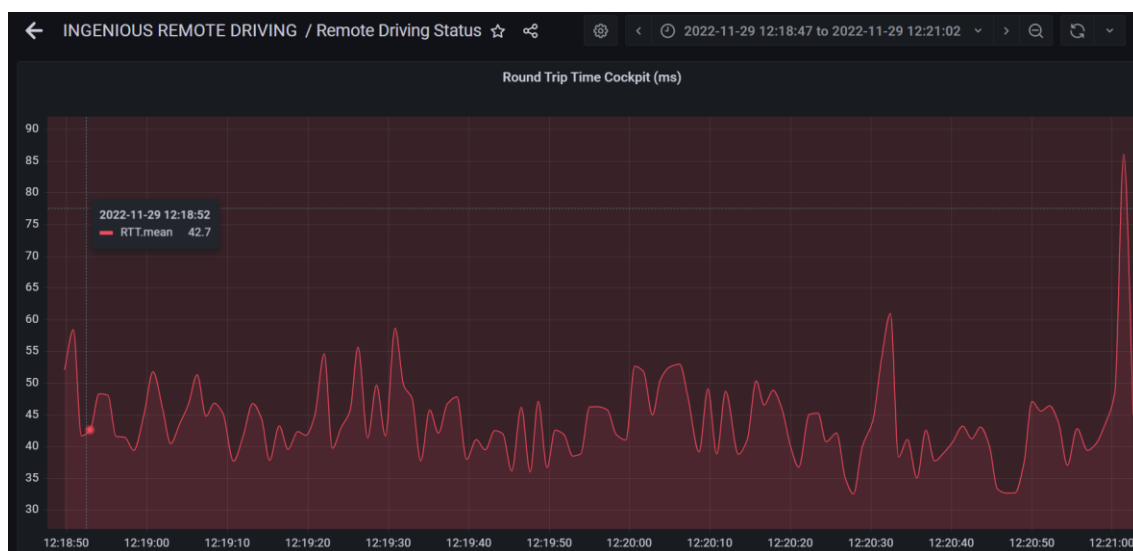


Figure 103: Example of RTT during Valencia Port tests.

Availability

The amount of time that the 5G network is fully operational in the Valencia Port network system during the test performance.

During the tests carried out, the 5G network was always available, which makes this measure reach the value of 100%.

Reliability

The ability of the 5G network to minimize the scope and frequency of incidents, continue operations while under pressure and recover as quickly as possible.

To measure this value, the number of decoded frames in the cockpit every second had been considered. The transformation to a percentage measure has been made with the number of seconds that the acceptable threshold for driving (25 FPS) is exceeded over the total duration of the test.



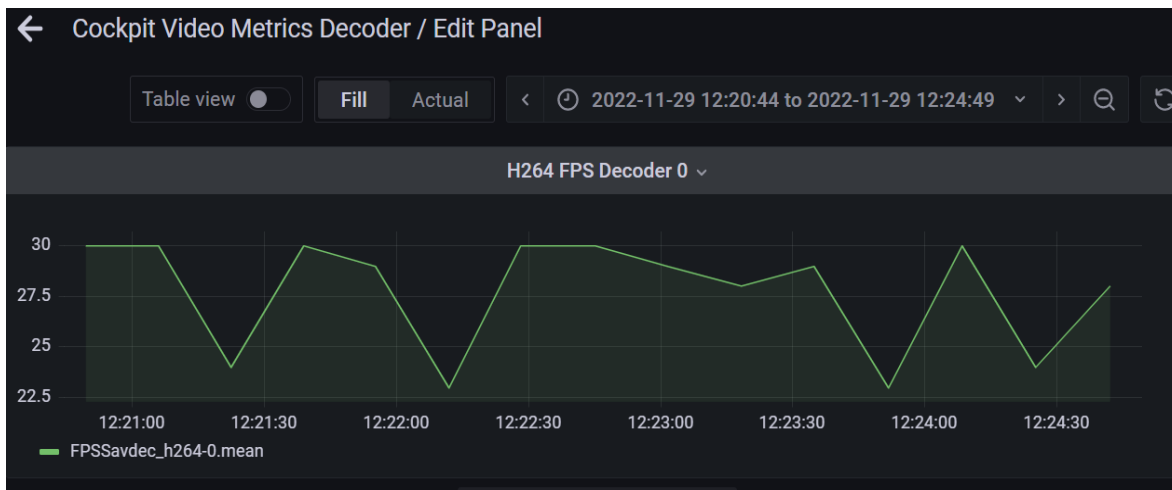


Figure 104: Example of the decoded frames during Valencia Port tests.

Mobility

The maximum AGV speed at which the 5G connectivity is guaranteed by providing a suitable QoS.

Data rate

The number of bits transmitted from one device to another or over the 5G network per second.

As in the case of latency this KPI has been measured in the different scenarios proposed in the test cases: AGV-A, AGV-B, AGV-C in Valencia Port and AGV-B in Burgos.

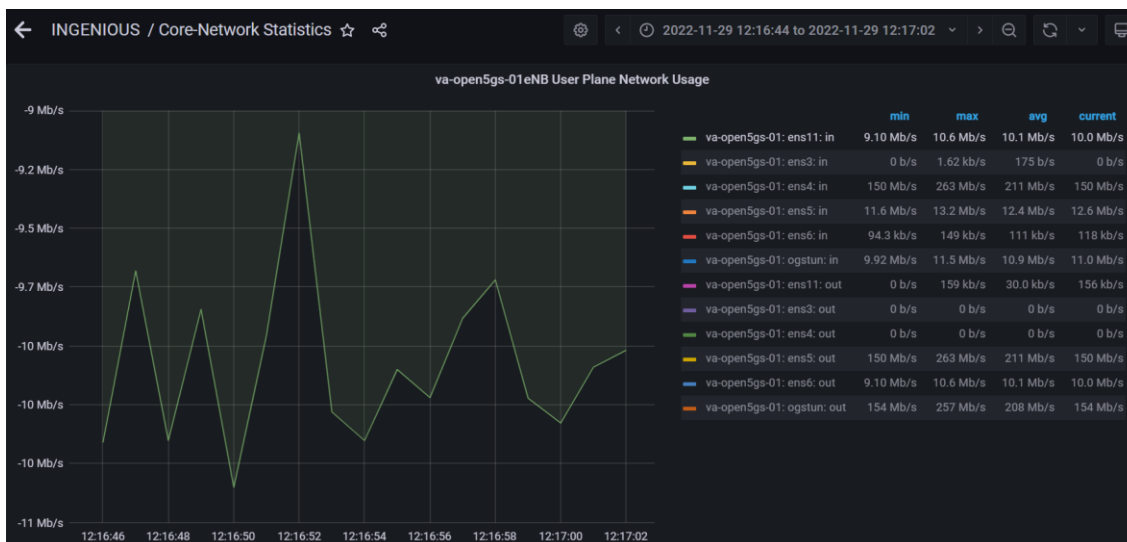


Figure 105: Example of the downlink data rate for AGV-B during Valencia Port tests.

A Bluetooth coverage test have been carried out with the two pair of gloves: Neurodigital and SenseGlove. This test is represented below, where we can see the difference of performance in an indoor and outdoor scenario. It is easy to check that the Neurodigital haptic gloves have a bigger range of movement, as it still can perform far from the server they are connected to. On the other hand, the SenseGlove ones lose connection earlier.



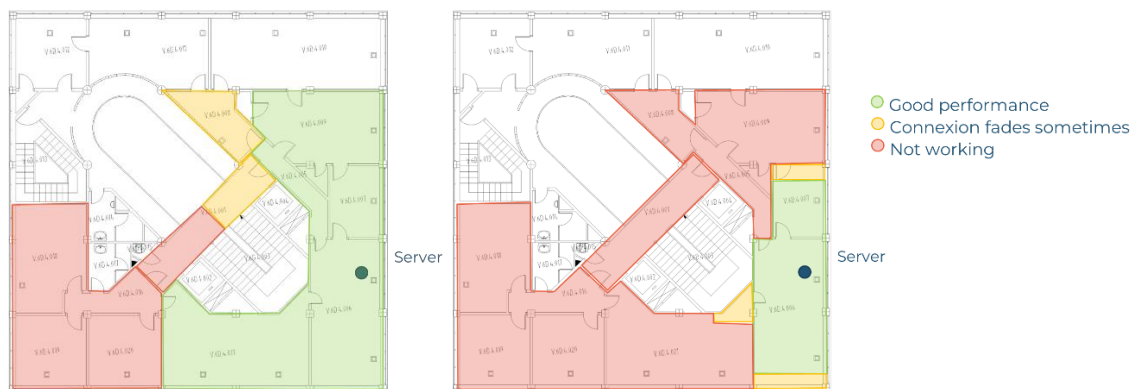


Figure 106: Indoor bluetooth range for Neurodigital (left) and SenseGlove (right) haptic gloves



Figure 107: Outdoor bluetooth range for Neurodigital (left) and SenseGlove (right) haptic gloves.

For indoor, the Neurodigital coverage is 12 m, and 5 m in the case of SenseGlove. For outdoor, the Neurodigital coverage is 32 m, and the SenseGlove coverage is 9 m.

Annex V: Ship UC - Intermodal Asset Tracking via IoT and Satellite

As mentioned in Demo – Intermodal Asset Tracking via IoT and Satellite, along with the Intermodal Asset Tracking via IoT and Satellite live over-the-air demonstrations, we also conducted lab simulations. This Annex provides more information on the lab testbed along with an overview of the verification testing and a more detailed description of the KPIs for the use case in general.

iDR Lab Testbed

The iDR Lab Testbed was setup to allow iDR test and validate different configurations in advance of any live satellite occasional use capacity received from SES for live demonstrations. iDR built the lab to closely resemble the live testbed environment hosted at SES' Betzdorf facility thus allowing minimum disruption to the live testbed. The testbed proved to be an invaluable asset throughout the project as a testing environment. The workflow of the iDR lab testbed setup and demonstrations were as follows:

1. iDirect's 5G-enabled Velocity™ Intelligent Gateway (IGW) hub was installed which included an integrated cloud based 5G core network.
2. An NMS server was installed to manage the satellite system.
3. An edge-UPF was integrated and hosted at SES's teleport allowing for a local breakout of user plane traffic in Betzdorf.
4. Two satellite channel emulators were deployed to simulate the forward and return carrier of a satellite link.
5. Three different models of modems were installed (iQ DT, iQ200 & 9350) for testing.
6. A Raspberry Pi model 3 was used as an IoT Gateway.
7. Generic IoT sensors were installed in the iDR lab to emulate the functionality of the container sensors.
8. Lab demonstrations were used for staging, testing and validation of the end-to-end system in preparation for the live demonstrations.

From a satellite system perspective, the iDR lab setup is very similar to the live testbed environment described in section 5 apart from the satellite link which is provided by two satellite emulators that simulate the forward and return satellite links. Also, the modem in use was a fixed Very Small Aperture Terminal (VSAT) modem rather than the SatCube modem used for the live demonstrations.

Similar to the live setup, the lab setup also used iDirect's 5G-enabled Velocity™ IGW hub which connected to the same Cloud hosted 5G core network used by the live network.

Furthermore, the iDR lab setup included an in-house IoT test network which was used for staging, testing and validation of the end-to-end system in preparation for the live demonstrations.



Figure 108 provides a high-level overview of the iDR lab testbed used throughout the project, while Figure 109 shows pictures of the equipment installed in the lab.

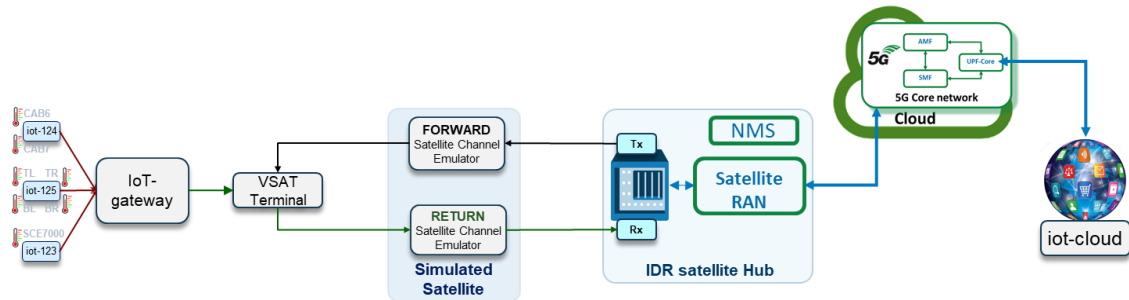


Figure 108: iDR lab testbed system overview



Figure 109: i) iDR Lab testbed including an iQ200, iQ Desktop, 9350 modems, IoT GW & Satellite Channel Emulators x2, ii) iDR Lab Testbed generic sensor used to measure temperature and humidity of the lab and iii) iDR Lab Testbed iDirect's 5G-enabled Velocity™ IGW hub

IDR LAB TESTBED - RESULTS

The iDR lab testbed was used for staging and testing on an ongoing basis including testing iDR's end-to-end IoT network. Another major focus of the iDR testbed was to de-risk the live mid-term and final demonstrations by validating the configurations and connectivity in advance. Table 110 below provides an overview of the testing carried out during the initial testbed setup and in preparation for the mid-term and final live demonstrations.

Live testing booking ID	Dates	Live satellite testing activities	Lab testing	Status
499159	25/05/21	Initial testbed testing over live satellite	Verify initial test setup including satellite lab system, GEO satellite simulator and IoT sensors. Introduce satellite delay of	Passed

			560mS and test live configuration.	
199335	01/11/21	Feasibility testing using SatCube	Test SatCube test configuration.	Passed
214210	07/02/22	End to end testing with SatCube and IoT network.	Verify SatCube test configuration and verify end to end IoT connectivity and operation.	Passed
222888	28/03/22	SatCube testing in preparation for mid-term demo	Verify configuration setup for mid-term demo.	Passed
222889	25/04/22	Mid-term demo	Verify configuration setup for mid-term demo.	Passed
258712	07/11/22	Final Demo	Verify configuration setup for final demo.	Passed

Table 110. Overview of IDR lab testbed activities

Validation and Results

TEST CASES VERIFICATION

The actual result of the test cases was presented in the Section 6.3.1, while here more information is provided.

Test Case Id	UC4_TC_01
Test case description	Integration and installation of sensors and communication modules on iNGENIOUS container
Test Steps	Integration of sensors in main board Integration of communication modules in main board Installation of main board module on iNGENIOUS container
System requirements covered	UC4_SR_25, UC4_SR_27
Expected result	Module with all sensors installed on iNGENIOUS container
Actual result	Module with all sensors was installed on iNGENIOUS container as can be also seen in Figure 49.
Passed/Failed	Passed

Table 111. UC4_TC_01 verification

Test Case Id	UC4_TC_02
Test case description	Over-the-air tests for evaluating LoRa and LTE connectivity at the container in maritime and terrestrial scenarios at the Port of Valencia
Test Steps	<ol style="list-style-type: none"> LoRa connectivity tests at the laboratory LoRa connectivity tests in the maritime segment (when the container is on the vessel) LoRa connectivity tests at the Port of Valencia LTE connectivity tests at the laboratory

	5. LTE connectivity tests in the terrestrial segment (when the container is on the truck)
System requirements covered	UC4_SR_27, UC4_SR_28
Expected result	LoRa and LTE connectivity ensured with the container at the terrestrial and maritime segments
Actual result	LoRa and LTE connectivity was validated
Passed/Failed	Passed

Table 112. UC4_TC_02 verification

Test Case Id	UC4_TC_03
Test case description	Develop an application where data gathered by IoT sensors and actuators is stored and visualized
Test Steps	<ol style="list-style-type: none"> 1. Structuring and storage of data in a database 2. Development an application for visualization
System requirements covered	UC4_SR_04
Expected result	Application where data gathered by IoT sensors and actuators is stored and visualized
Actual result	As we can see in several figures, Figure 53 – Figure 62, we were able to observe the collected data in the Grafana dashboards of the SES Cloud servers
Passed/Failed	Passed

Table 113. UC4_TC_03 verification

Test Case Id	UC4_TC_04
Test case description	Container transport from the Port of Valencia to the Port of Piraeus, including storage at the Port of Piraeus until next loading
Test Steps	<p>Transport from Valencia port to Piraeus port:</p> <ol style="list-style-type: none"> 1. Container loading to the ship at the Port of Valencia 2. Transport to the Port of Piraeus by vessel 3. Container unloading at the Port of Piraeus 4. Storage at the Port of Piraeus
System requirements covered	UC4_SR_01, UC4_SR_03
Expected result	The iNGENIOUS container should be transported from the Port of Valencia to the Port of Piraeus, including storage at the Port of Piraeus until next loading
Actual result	The iNGENIOUS container was transported from the Port of Valencia to the Port of Piraeus (05-08 October 2022) and it was stored there until the next loading (27 October 2022). More information about the details of the transportation have been provided in Section 6.2.1.
Passed/Failed	Passed

Table 114. UC4_TC_04 verification

Test Case Id	UC4_TC_05
---------------------	------------------

Test case description	Container transport from the Port of Piraeus to the Port of Valencia
Test Steps	Transport from the Port of Piraeus to the Port of Valencia 1. Container loading to the ship at the Port of Piraeus 2. Transport to the port of Valencia by vessel 3. Container unloading at the Port of Valencia 4. Storage at the Port of Valencia
System requirements covered	UC4_SR_01, UC4_SR_03
Expected result	The iNGENIOUS container should be transported from the Port of Piraeus to the Port of Valencia, including storage at the Port of Valencia until we conduct the over-the-air demo using the satellite connection
Actual result	The iNGENIOUS container was transported from the Port of Valencia to the Port of Piraeus (27 October – 08 November) and it was stored there until 21 November 2022 where the over-the-air demo took place. More information about the details of the transportation have been provided in Section 6.2.1.
Passed/Failed	Passed

Table 115. UC4_TC_05 verification

Test Case Id	UC4_TC_06
Test case description	Terrestrial transport by truck from Port of Valencia to hinterland and vice versa
Test Steps	Inland segment from the Port of Valencia to Madrid area and vice versa: 1. Container loading on the truck 2. Inland transport by truck 3. Container unloading
System requirements covered	UC4_SR_01, UC4_SR_03, UC4_SR_23, UC4_SR_27
Expected result	The iNGENIOUS container should be transported by truck from the Port of Valencia to hinterland and vice versa
Actual result	The iNGENIOUS container was transported from the Port of Valencia to the Madrid Dry Por (01 March – 09 March). More information about the details of the transportation have been provided in Section 6.2.2.
Passed/Failed	Passed

Table 116. UC4_TC_06 verification

Test Case Id	UC4_TC_07
Test case description	Site Survey for exploring the practical viability of accommodating and installing the Smart IoT Gateway aboard, as well for exploring the theoretical viability of installing VSAT antenna on the vessel

Test Steps	Drafting of list of activities for performing the site survey Validation with COSCO, the captain and the owner of the ship Execution of the site survey Drafting of document summarizing the main outcomes of the survey
System requirements covered	UC4_SR_01, UC4_SR_08, UC4_SR_12, UC4_SR_13, UC4_SR_15
Expected result	Assessment and validation of power supply requirements, environment and physical dimensions required, electromagnetic compatibility, LoRa, Wi-Fi and BT coverage, accessibility and deployment constraints for installing the Smart IoT GW on board. Theoretical assessment of a potential installation of the VSAT antenna onboard
Actual result	The site survey did not take place as we have described in the D6.2 [2].
Passed/Failed	Failed

Table 117. UC4_TC_07 verification

Test Case Id	UC4_TC_08
Test case description	Validate proposed satellite backhaul infrastructure A number of iterations of testing will take place as satellite capacity is made available, in order to guarantee that the infrastructure will meet the KPI requirements of the live demonstration
Test Steps	<ol style="list-style-type: none"> 1. Setup – SatCube terminal connectivity to SES live network. 2. Send measurement data from device co-located with SatCube via satellite to host at teleport side (and vice-versa). 3. Verify receipt of test data in both directions.
System requirements covered	UC4_SR_01, UC4_SR_18, UC4_SR_29
Expected result	Test data exchanged successfully over satellite between terminal and teleport. Achieved bandwidth and latency results should indicate sufficient performance to meet use case requirements
Actual result	Testing completed successfully with SatCube and Fixed VSAT terminals. Several iterations took place based on the provided satellite capacity (see Table 19)
Passed/Failed	Passed

Table 118. UC4_TC_08 verification

Test Case Id	UC4_TC_09
Test case description	Validate end to end connectivity using Satellite backhaul
Test Steps	<ol style="list-style-type: none"> 1. Setup – Data path establishment between Sensor and Data centre using satellite backhaul. 2. Publish sensor data from device to data centre/cloud. 3. Verify successful receipt of sensor data.
System requirements covered	UC4_SR_01, UC4_SR_18, UC4_SR_24, UC4_SR_26, UC4_SR_29
Expected result	Sensor data is received successfully at data centre/cloud
Actual result	Sensor data is received successfully at data centre/cloud as we can see also in several figures, Figure 53 – Figure 62.
Passed/Failed	Passed



Table 119. UC4_TC_09 verification

Test Case Id	UC4_TC_10
Test case description	Verify uplink and downlink Satellite backhaul capacity meets Use Case KPI requirements
Test Steps	<ol style="list-style-type: none"> 1. Setup – Data path establishment between Sensor and Data centre using satellite backhaul. 2. Using iperf or similar utilities, measure UDP and TCP downlink bandwidth between Satellite Terminal location and Betzdorf egress point. 3. Using iperf or similar utilities, measure UDP and TCP downlink bandwidth between Satellite Terminal location and Betzdorf egress point.
System requirements covered	UC4_SR_01, UC4_SR_05, UC4_SR_07, UC4_SR_08, UC4_SR_18
Expected result	Uplink and downlink capacity should exceed the minimum requirements defined in Use Case KPIs
Actual result	The uplink and downlink capacity (see Table 19) was enough for a successful over-the-air demo
Passed/Failed	Passed

Table 120. UC4_TC_10 verification

Test Case Id	UC4_TC_11
Test case description	Verify uplink and downlink Satellite backhaul latency
Test Steps	<ol style="list-style-type: none"> 1. Setup – Data path establishment between Sensor and Data centre using satellite backhaul. 2. Send test TCP/UDP data uplink and downlink between host at/co-located with satellite terminal and Betzdorf teleport egress point, to measure latency observed over satellite segment of backhaul.
System requirements covered	UC4_SR_01, UC4_SR_05, UC4_SR_07, UC4_SR_08, UC4_SR_18
Expected result	Latency should be within the limits specified for the use case
Actual result	Using ICMP packets, an average latency of 593 ms was observed between the satellite terminal and Betzdorf teleport egress point which is shown in Table 21.
Passed/Failed	Passed

Table 121. UC4_TC_11 verification

Test Case Id	UC4_TC_12
Test case description	Validate confidentiality of satellite backhauled sensor data
Test Steps	<ol style="list-style-type: none"> 1. Setup – Data path establishment between Sensor and Data centre using satellite backhaul. 2. Capture sensor data in transit, at point between Smart IoT Gateway and Teleport IP egress point. 3. Analyse captured sensor data to verify encrypted status.
System requirements covered	UC4_SR_01, UC4_SR_05, UC4_SR_07, UC4_SR_18, UC4_SR_21

Expected result	Captured sensor data is indecipherable between IoT Gateway and teleport egress point
Actual result	Pcaps were taken of the sensor data at the ingress and egress of the satellite network to confirm the data was indecipherable as it was contained within a VPN which was setup between the Smart IoT GW and the IoT Cloud.
Passed/Failed	Passed

Table 122. UC4_TC_12 verification

Test Case Id	UC4_TC_13
Test case description	Connectivity with sensors
Test Steps	<ol style="list-style-type: none"> 1. Configure GW and sensors (IDs, security...). 2. Sensors start transmitting meaningful data. 3. GW receives the messages. 4. GW processes the messages. 5. GW stores the messages.
System requirements covered	UC4_SR_03, UC4_SR_06
Expected result	GW and sensors can communicate and exchange data
Actual result	The Smart IoT GW and the IoT devices communicated and exchanged data successfully, as we can see in several figures, Figure 53 – Figure 62.
Passed/Failed	Passed

Table 123. UC4_TC_13 verification

Test Case Id	UC4_TC_14
Test case description	Connectivity with M2M space (direct)
Test Steps	<ol style="list-style-type: none"> 1. Configure oneM2M CSE. 2. Trigger messages on the IoT GW that needs to be routed directly.
System requirements covered	UC4_SR_01, UC4_SR_02
Expected result	Messages are correctly routed to the oneM2M CSE
Actual result	Messages were correctly routed to oneM2M CSE and this can be seen in several figures, Figure 53 – Figure 62, as the data are observed from the SES Cloud through Grafana dashboard
Passed/Failed	Passed

Table 124. UC4_TC_14 verification

Test Case Id	UC4_TC_15
Test case description	Connectivity with M2M space (VSAT)
Test Steps	<ol style="list-style-type: none"> 1. Configure VSAT termina. 2. Configure oneM2M CSE. 3. Trigger messages on the IoT GW that needs to be route via satellite. 4. Send M2M messages through the VSAT.



System requirements covered	UC4_SR_01, UC4_SR_11, UC4_SR_16, UC4_SR_17, UC4_SR_19, UC4_SR_23
Expected result	Messages are correctly routed via satellite
Actual result	Messages were correctly routed via satellite. All figures, Figure 53 – Figure 62 presents results through satellite communication
Passed/Failed	Passed

Table 125. UC4_TC_15 verification

Test Case Id	UC4_TC_16
Test case description	Smart IoT GW will receive and process sensor data
Test Steps	<ol style="list-style-type: none"> 1. Trigger message generation for a specific route type. 2. Change message parameters (type, priority, payload...). 3. Repeat step 2 for the supported message types.
System requirements covered	UC4_SR_07, UC4_SR_08, UC4_SR_10, UC4_SR_11, UC4_SR_16, UC4_SR_17, UC4_SR_19, UC4_SR_23
Expected result	Correctly formatted messages are routed to the appropriate destination
Actual result	Correctly formatted messages were routed to SES Cloud and can be seen in several figures, Figure 53 – Figure 62.
Passed/Failed	Passed

Table 126. UC4_TC_16 verification

Test Case Id	UC4_TC_17
Test case description	Smart IoT GW configuration via remote management
Test Steps	<ol style="list-style-type: none"> 1. Log in to Smart IoT GW management endpoint. 2. Send configuration parameters. 3. Retrieve status and configuration data. 4. Sensors send alert/warning messages. 5. Verify that the Smart IoT GW sends the appropriate alerts.
System requirements covered	UC4_SR_09, UC4_SR_12, UC4_SR_20
Expected result	The Smart IoT GW changes configuration and shows status
Actual result	The Smart IoT GW changes configuration and shows status
Passed/Failed	Passed

Table 127. UC4_TC_17 verification

Test Case Id	UC4_TC_18
Test case description	Smart IoT GW will receive and process sensor data during outages
Test Steps	<ol style="list-style-type: none"> 1. Trigger message generation for a specific route type. 2. Change message parameters (type, priority, payload...). 3. Verify that messages are being routed. 4. Disconnect destination endpoint. 5. Verify that messages are being stored. 6. Connect back the destination.



	7. Verify that the stored messages are (re)sent again.
System requirements covered	UC4_SR_12, UC4_SR_12, UC4_SR_14, UC4_SR_15, UC4_SR_16
Expected result	During the outages, the messages are held and sent again when the destination network is available
Actual result	During the outages, the messages were held and sent again when the destination network was available. During the final over-the-air demo this happened, as in the beginning we faced some difficulties to establish the satellite connection while the Smart IoT GW was receiving data from the IoT devices
Passed/Failed	Passed

Table 128. UC4_TC_18 verification

Test Case Id	UC4_TC_19
Test case description	Smart IoT GW Security
Test Steps	<ol style="list-style-type: none"> 1. The Smart IoT GW captures and processes sensor data. 2. Analyse captured sensor data to verify encrypted status.
System requirements covered	UC4_SR_21
Expected result	Captured sensor data are sent to cloud with high level of security
Actual result	Captured sensor data are sent to cloud with high level of security
Passed/Failed	Passed

Table 129. UC4_TC_19 verification

Test Case Id	UC4_TC_20
Test case description	Smart IoT GW Integration with other systems
Test Steps	<ol style="list-style-type: none"> 1. Communication of the Smart IoT GW with the sensors. 2. Communication of the Smart IoT GW with the satellite terminal or the LTE network.
System requirements covered	UC4_SR_22
Expected result	Sensor data is received successfully at data centre/cloud
Actual result	Sensor data was received successfully at data centre/cloud as we can see in several figures, Figure 53 – Figure 62.
Passed/Failed	Pending

Table 130. UC4_TC_20 verification

KPIS VERIFICATION

The actual result of the KPIs was presented in KPIs, while here more information is provided.

Availability

- Description: The measured data from the IoT devices, installed in the INGENIOUS container, should be ubiquitously available at any time to the end users with the user's requested QoE level.
- Verification: The live over-the-air tests for the first part of the final demo took place at the Port of Valencia. The data measured by the IoT devices were sent to the SES Cloud through satellite connectivity where the used space segment was the ASTRA 2F GEO satellite. Figure 44 presents the coverage of the ASTRA 2F in Europe where we can see that the Port of Valencia is covered all the time. Furthermore, from Figure 57, Figure 58, and Figure 59, we can see that when the communication between the IoT devices and the Smart IoT Gateway and the satellite connection were established the data were sending to the SES Cloud consistently.

Reliability

- Description: The reliability is an assessment criterion to describe the quality of the radio link connection for fulfilling a certain service level
- Verification: The live over-the-air tests for the first part of the final demo took place at the Port of Valencia. From Figure 57, Figure 58 and Figure 59, we can see that all the data were sending to SES Cloud without any loss and based on the established frequency for the IoT devices to send updates.

Battery life

- Description: The battery powered IoT UE should be able to operate for the entire lifetime of the tracked container (>12 years) without large capacity battery packs and without being replaced during this period of time. Measured in years.
- Verification: The lifetime of the IoT devices deployed in the INGENIOUS container is 5 years with a reporting frequency of 1 message per day. Furthermore, the battery life of the IoT devices could be observed in real-time in the Grafana dashboards of the SES Cloud servers as shown in Figure 59.

Coverage

- Description: Satellite coverage will be provided to enable ubiquitous coverage of the shipping INGENIOUS
- Verification: The data measured by the IoT devices were sent to the SES Cloud through satellite connectivity where the used space segment was the ASTRA 2F GEO satellite. Table 19 presents the details of the provided OU) satellite capacity for the preparation and execution of the live over-the-air demo, such as booking confirmation ID, start and end dates, used satellite and frequency band, bandwidth and satellite hub.

Typical message size

- Description: The typical size of the information sent by the IoT devices should be 200 bytes.
- Verification: The values for the average and maximum message sizes are extracted from the syslog for the lora_pkt_fwd service, which is the concentrator service for the LoRa module, attached to the Smart IoT Gateway. This log contains entries for all received LoRa transmissions (including join requests and response), as seen in the screenshots below, providing of the message in bytes. Using the calculations shown in

calculation.png, the average and maximum message sizes could be determined.

```
Nov 21 17:15:10 smartiotgw lora_pkt_fwd[1317]: RXTX- [{"rxpk":{"tmst":2995151283,"tme":"2022-11-21T16:14:03.746405Z","chan":4,"rfch":0,"freq":867.300000,"stat":1,"modu":"LORA","datr":"SF7Bw125","codr":"4/5","lsnr":9.2,"rssi":-63,"size":110,"data":"QHtNgQcABwAB+hRoWMDcIKaw6lJlchZ/Q0iK47AGtFPVzL51rNpcxCPQMy3D1AJQ01Quzsd97yXGR7UyB814a80V9nygJ/Xq4XkrCix3L+LtxdXAvq00ArqbXQlgrunHmX6JRwWwD3Mq1nRWG4="}]}

pi@smartiotgw:~/syslog $ less lorawan.log | grep size | sed 's/. *size:*(.*) ,*/\1/' > message_sizes.txt
pi@smartiotgw:~/syslog $ COUNT=$(cat message_sizes.txt | wc -l)
pi@smartiotgw:~/syslog $ SUM=$(cat message_sizes.txt | awk '{sum += $1} END {print sum}')
pi@smartiotgw:~/syslog $ AVG=$(( $SUM / $COUNT ))
pi@smartiotgw:~/syslog $ MAX=$( cat message_sizes.txt | awk '{max = $1 > max ? $1 : max} END {print max}')
pi@smartiotgw:~/syslog $
pi@smartiotgw:~/syslog $ echo "COUNT: $COUNT
> SUM: $SUM
> AVG: $AVG
> MAX: $MAX"
COUNT: 424
SUM: 46301
AVG: 109
MAX: 255
pi@smartiotgw:~/syslog $
```

Maximum message size

- Description: The size of the data payload containing the status information related to a container can be up to 2500 Bytes.
- Verification: Same explanation as for the typical message size.

Typical frequency (messages per day)

- Description: Depending on the service level required by the owner of the container and the supply chain associated, the IoT devices installed in the INGENIOUS container send regular status update.
- Verification: From Figure 53 and Figure 54, we can see that the IoT devices were sending messages once per day during the trip from Valencia to Piraeus and vice versa. While, from Figure 57 and Figure 58 we can see that this frequency was set up to every 5 minutes for the live over-the-air final demo at the Port of Valencia.

Connectivity of heterogeneous IoT devices

- Description: The Smart IoT GW is responsible for the appropriate routing and sorting of sensor data, coming from one or more sensor networks. Sensors can send messages to the Smart IoT GW either wirelessly (with technologies such as Wi-Fi, LoRa, Bluetooth), or directly connected to the device (via ethernet, I2C or SPI).
- Verification: The Smart IoT GW was developed to support LoRa and Wi-Fi protocols but during the live over-the-air demonstrations only LoRa communication with the IoT devices was tested. Figure 53, Figure 54, Figure 55, Figure 56, Figure 57, Figure 58 and Figure 59 present the capability of the Smart IoT GW to receive and process the data, sent from the IoT devices.

Latency

- Description: The end-to-end latency which represents the maximum tolerable elapsed time from the instant a data packet is generated at the source application to the instant it is received by the destination application should be less than $\leq 1s$.
- Verification: Figure 62 and Table 21 and depict that the end-to-end latency is in the range of 600 ms.

Mobility

- Description: Mobility is defined as the maximum vehicle speed at which IoT connectivity is guaranteed by providing a suitable QoS.

- Verification: As the sensor was configured to send the data once every hour, checking this KPI during the demo was not applicable. As all the data was received in each report sent by the sensor, the actual values assigned to this KPI were the maximum speed achievable by each of these transport means.

Positioning accuracy

- Description: Degree of correctness provided by the real-time IoT positioning sensors when tracking the iNGENIOUS container on the ship or the truck.
- Verification: Figure 56 shows the measured GPS points with an approximate accuracy of around 50m, being distributed in a radius of roughly 25m around the actual location of the IoT devices.



Annex VI: DLT's UC - Supply Chain Ecosystem Integration

For each section of the D6.3 [15], additional technical details are reported in relation to the DVL/DLT UC.

Objective and Description

The main aim of the demonstration for this use case is to show that all data flows (from the data sources to the end users) are properly implemented and are processed as expected. More in detail, the demonstration covers the following aspects:

- For all defined scenarios, the DVL is able to retrieve and aggregate raw data from all available machine-to-machine platforms (namely OneM2M, Eclipse OM2M, Symphony and PISystem) as well as external data sources (namely Port Community System in Livorno).
- The DVL implements a set of remote procedures (exposed by means of RESTful APIs) for GateIn, GateOut, VesselArrival, VesselDeparture and sealRemoved maritime events as well as for data defined in Scenario 3 and Scenario 4 to be used by TrustOS and external applications respectively (namely Awake.AI platform and a Web Application for trucks' real-time monitoring).
- The TrustOS component is able to interact with DVL by means of a RESTful interface and to correctly retrieve the data of the maritime events as per above. A DigitalAsset (a digital representation of the data) is created for each of the supported maritime events and stored as a TrustPoint accordingly.
- The TrustOS component communicates with four different DLTs (namely Bitcoin, Ethereum, IOTA and Hyperledger Fabric) by means of a common API, and it makes sure that a given DigitalAsset (as well as the relative TrustPoint) is written among the ledgers of the DLTs as a proof of evidence for the stored data. A GUI is able then to visualize the main information related to a DigitalAsset (e.g., TrustPoint, ownership and metadata) stored on a given DLT as well as on TrustOS so that the end user can double check the content accordingly.
- Pseudonymization Function is able to detect the truck plate number (for GateIn and GateOut events in Scenario 4), pseudonymize it and make it available to DVL by means of a RESTful interface. The DVL exposes such data by means of a remote procedure and an external application (namely Awake.AI platform) can consume pseudonymized data accordingly.

The access to DVL is performed according to a role-based policies for all data consumers (CRUD operations). According to the defined scenarios, data consumers are represented by the following entities: TrustOS, Pseudonymization Module, Awake.AI platform and the Web Application for trucks' monitoring.

SCENARIO 1

In this scenario, the DVL is responsible for providing GateIn, GateOut, VesselArrival and VesselDeparture events for both seaports. In this case, TrustOS, which is part of the Cross-DLT layer, plays the role of data consumer of such events. Each event is defined by a given set of attributes according to a common data model from Tradelens platform [16]. DVL retrieves from the underlying and available data sources by aggregating available data sets. The structure of these events is described in deliverable D6.2 [2], along with further details. GateIn, GateOut, VesselArrival and VesselDeparture events for the Port of Livorno are obtained by aggregating data from the Port Community System (TPCS), whereas for the Port of Valencia, only GateIn and GateOut events are considered and retrieved from the PISystem OSISOFT M2M platform. In both cases, DVL implements two different connectors/wrappers to interact with both the PCS and the underlying M2M platform. RESTful interfaces are then exposed so that TrustOS can consume such events by associating a TrustPoint and store it across available DLTs. The overall architecture for this scenario is depicted in Figure 110 and further technical details can be found in D6.2 [2] and in D5.3 [17]:

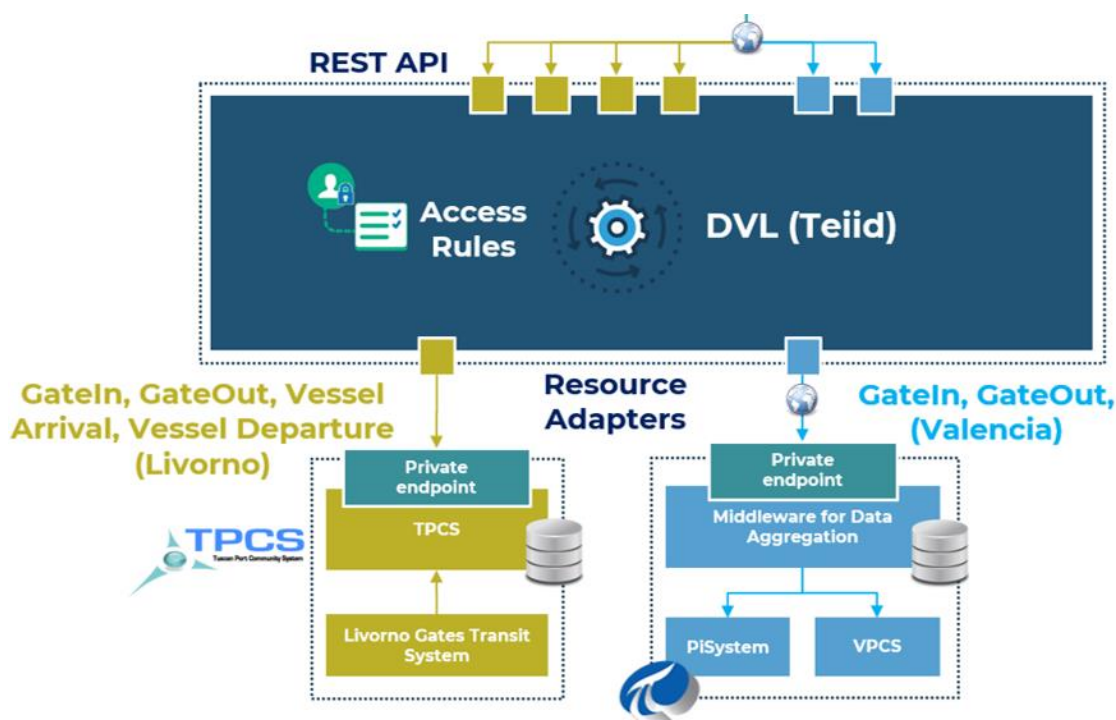


Figure 110: Scenario 1 architecture for the demonstration of the use case.

SCENARIO 2

This scenario is linked to the Ship UC where the Port of Valencia is involved. The DVL is responsible here for interacting with Eclipse OM2M platform in order to retrieve data coming from smart sensors installed on the transported container. In this case the main aim is to detect when the doors of the packed container are opened by removing the smart seal from the iNGENIOUS container. This information is retrieved from the M2M platform by means of a

RESTful connector/wrapper, processed by the DVL, and exposed to TrustOS through a consumable RESTful API. As in the Scenario 1, TrustOS can use such an API to retrieve the event, create a TrustPoint, and distribute it across all DLTs for secure storage. The overall architecture for this scenario is depicted in Figure 111: and further technical details can be found in D6.2 [2] and in deliverable D5.3 [17]:

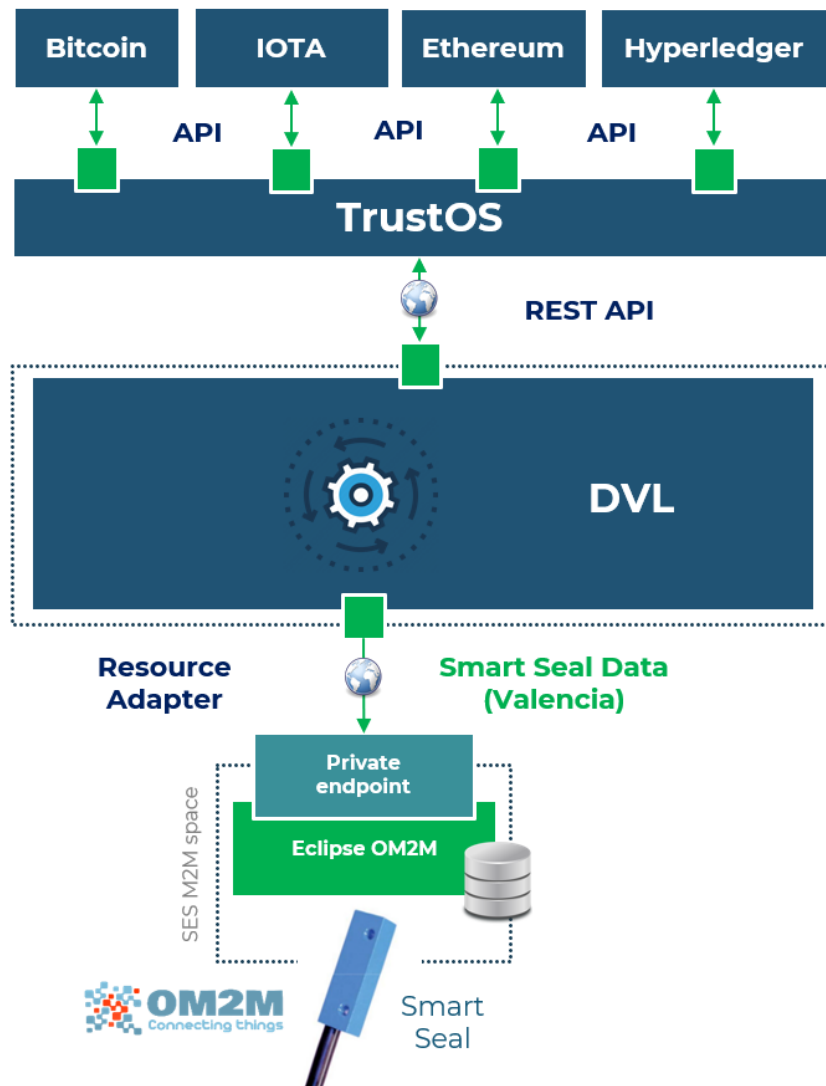


Figure 111: Scenario 2 architecture for the demonstration of the use case.

SCENARIO 3

This scenario is linked to Port Entrance use case where both Port of Valencia and Port of Livorno are involved. The DVL is responsible for interacting with the Symphony M2M platform in order to retrieve GPS data coming from tracking devices installed on trucks in the Livorno seaport. The device sends data to the M2M platform for storage. This information is then retrieved by the DVL through a RESTful connector/wrapper which allows to directly interact with the platform. Finally, a dashboard-based application consumes the tracking data by invoking a RESTful API at DVL level. The data are then visualized through a

dashboard in real time. The overall architecture for this scenario is depicted in Figure 112 and further technical details can be found in D6.2 [2] and D5.3 [17]:

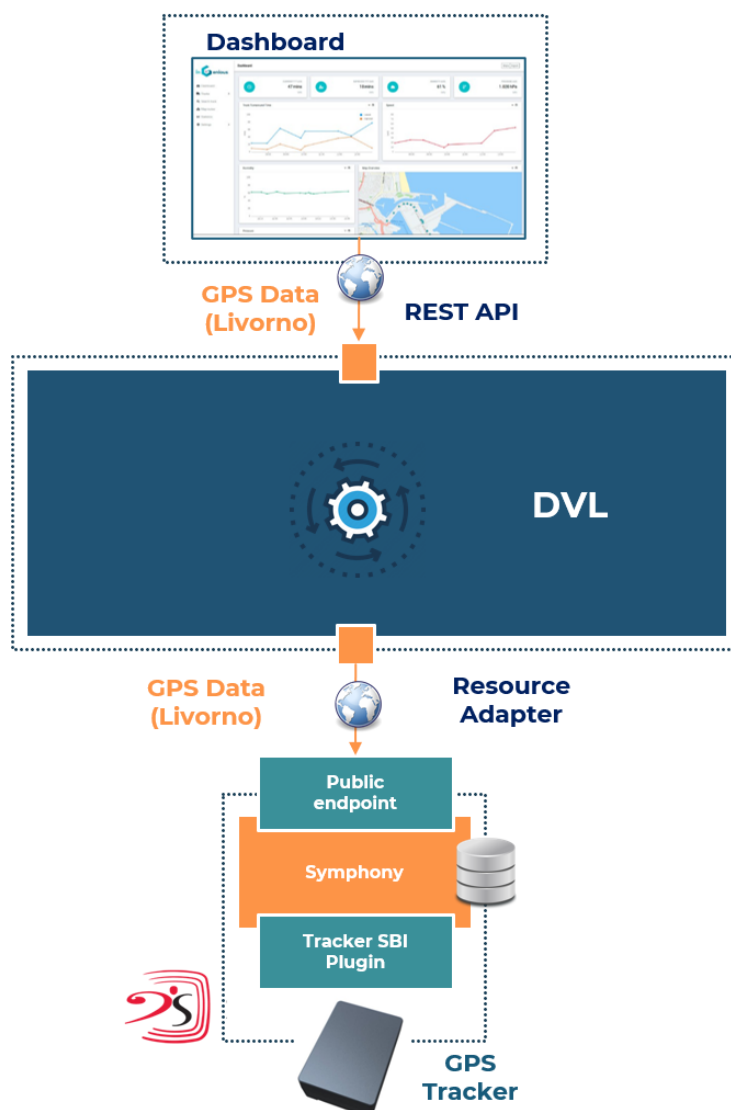


Figure 112: Scenario 3 architecture for the demonstration of the use case.

SCENARIO 4

This scenario is also related to Port Entrance use case implemented in both Port of Valencia and Port of Livorno. Two different components are involved: Mobius OneM2M platform and a Pseudonymization Module. The M2M platform is responsible for collecting meteorological data in Livorno seaport. DVL implements a RESTful connector/wrapper to interact with this platform, extracts the available data set, and exposes it by means of a RESTful API so that an AI-based platform can consume and correlate it with truck-turnaround times in the Livorno seaport. Moreover, a Pseudonymization Module component is used to process GateIn and GateOut events in order to identify personal data and pseudonymize it accordingly (trucks' plate number). The module retrieves the GateIn and GateOut events from the Port of Livorno by using existing RESTful APIs (available from Scenario 1), detects all the attributes

which may be potentially sensitive (in our case the truck plate number), pseudonymizes the attribute according to available pseudonymization techniques (e.g., hash without key), and stores it within a conversion database. The DVL is able then to expose a RESTful API to allow an AI-based platform to consume pseudonymized data sets for training of predictive AI/ML models. The overall architecture for this scenario is depicted in Figure 113 and further technical details can be found in D6.2 [2] and D5.3 [17]:

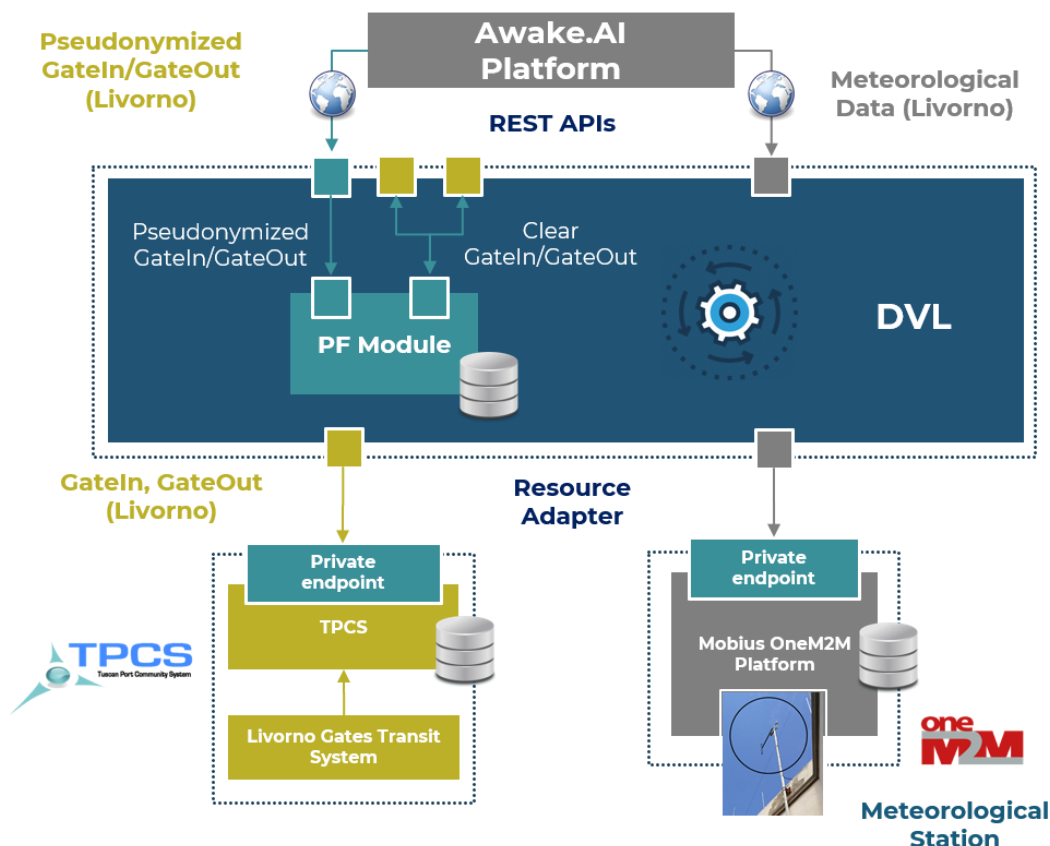


Figure 113: Scenario 4 architecture for the demonstration of the use case.

Setup and Execution

SCENARIO 1

Technical specifications of the hosting environment and the used APIs in this scenario are provided in D6.2 [2] and D5.3 [17]:

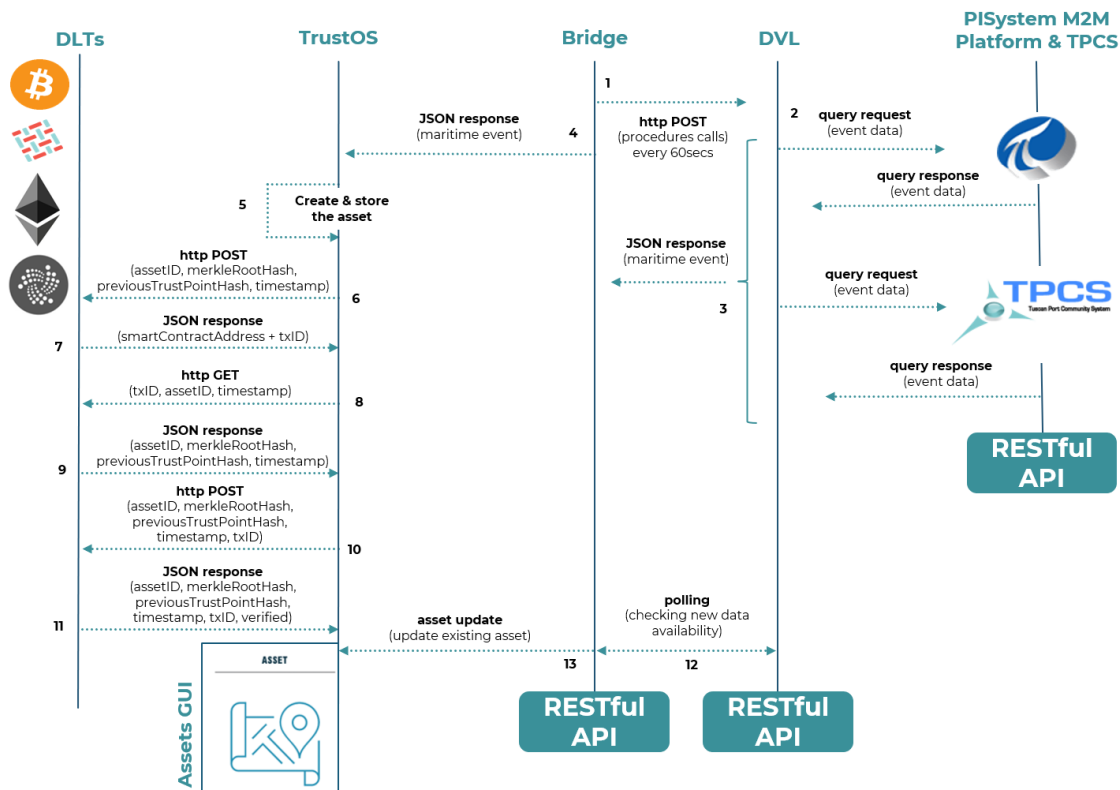


Figure 114: sequence diagram for the demonstration of the Scenario 1.

```

"output": {
  "assetId": "001",
  "data": {
    "location": "ITLIV",
    "originatorId": "ITLIV",
    "originatorName": "Port of Livorno"
  },
  "datetime": 1651160013,
  "hash": "E5RWIt1AoKR3u00b8r4EBSpy5Kt7oa5iGvNICot9uE=",
  "hfTxId": "0315b1102790bb0c4a9abec18ca86fe0cebd9528c4378ec2731c98e5b005646b",
  "metadata": {
    "billOfLadingNumber": "MEDUM329279",
    "carrierBookingNumber": null,
    "equipmentNumber": "MEDU4654152",
    "eventOccurrenceTime8601": "2021-10-14T10:58:00Z",
    "eventSubmissionTime8601": "2021-10-03T11:00:00Z",
    "location": "ITLIV",
    "originatorId": "ITLIV",
    "originatorName": "Port of Livorno",
    "terminal": "LRN",
    "transportEquipmentId": "NULL",
    "transportEquipmentRef": "NULL",
    "transportationPhase": "Import",
    "vehicleId": "9141297",
    "vehicleName": "Ship",
    "voyageId": 19814
  },
  "userOwner": "did:vtn:trustid:6776c3d9a9a6f5e33835081ce5f1c7973d0be70e22565e4767f3742f50dae909"
}
    
```

Figure 115: DigitalAsset for the VesselArrival event in Livorno seaport.




```

"output": {
  "assetId": "002",
  "data": {
    "location": "ITLIV",
    "originatorId": "ITLIV",
    "originatorName": "Port of Livorno"
  },
  "datetime": 1651160016,
  "hash": "m9ajYRSb4J2q4/6Bwye/jA5Kn0Jk8EazJFN/V1qufEI=",
  "hftxid": "2ce394c2229e27f1bc6c07cbc577dad9e64dd27a244bba685bfaaed44fc5a7f7",
  "metadata": {
    "billOfLadingNumber": null,
    "carrierBookingNumber": null,
    "equipmentNumber": "SEGU6725734",
    "eventOccurrenceTime8601": "2021-08-01T09:16:30Z",
    "eventSubmissionTime8601": "2021-08-01T09:21:00Z",
    "location": "ITLIV",
    "originatorId": "ITLIV",
    "originatorName": "Port of Livorno",
    "terminal": "LRN",
    "transportEquipmentId": "NULL",
    "transportEquipmentRef": "NULL",
    "transportationPhase": "export",
    "vehicleId": 9646665,
    "vehicleName": "Ship",
    "voyageId": 1969
  },
  "userOwner": "did:vtn:trustid:6776c3d9a9a6f5e33835081ce5f1c7973d0be70e22565e4767f3742f50dae909"
}

```

Figure 116: DigitalAsset for the VesselDeparture event in Livorno seaport.

```

"output": {
  "assetId": "003",
  "data": {
    "location": "ITLIV",
    "originatorId": "ITLIV",
    "originatorName": "Port of Livorno"
  },
  "datetime": 1651160133,
  "hash": "lDn/KSPQMqWiraVwDhp9z8R8B8z0gg0Y3Gw4be3VB34=",
  "hftxid": "04ce06fc358e0fd06893b38224cf8f77cc2369dc03085d976dff581d3e4e0634",
  "metadata": {
    "billOfLadingNumber": "908291711",
    "carrierBookingNumber": null,
    "equipmentNumber": "HASU4605254",
    "eventOccurrenceTime8601": "2021-08-01T12:00:30.8510000+01:00",
    "eventSubmissionTime8601": "2021-08-01T11:04:00Z",
    "fullStatus": "F",
    "location": "ITLIV",
    "originatorId": "ITLIV",
    "originatorName": "Port of Livorno",
    "terminal": "LRN",
    "transportEquipmentId": null,
    "transportEquipmentRef": null,
    "transportationPhase": "export",
    "vehicleId": "BV892CT",
    "vehicleName": "Truck",
    "voyageId": "20488"
  },
  "userOwner": "did:vtn:trustid:6776c3d9a9a6f5e33835081ce5f1c7973d0be70e22565e4767f3742f50dae909"
}

```

Figure 117: DigitalAsset for the GateIn event in Livorno seaport.

```

"output": {
  "assetId": "004",
  "data": {
    "location": "ITLIV",
    "originatorId": "ITLIV",
    "originatorName": "Port of Livorno"
  },
  "datetime": 1651160021,
  "hash": "N3Pe5eT+3+THEJRD3JR5Tro4uI1tX8Bqo1oKvGboTjc=",
  "hfTxId": "9d678ba881fd4fb576857a87cd96c718b262be850ed71e45b790ff6c219d858",
  "metadata": {
    "billOfLadingNumber": "GQL0225748",
    "carrierBookingNumber": null,
    "equipmentNumber": "APRU6115615",
    "eventOccurrenceTime8601": "2021-07-30T18:40:03.5430000+01:00",
    "eventSubmissionTime8601": "2021-07-30T19:04:08.671Z",
    "fullStatus": "F",
    "location": "ITLIV",
    "originatorId": "ITLIV",
    "originatorName": "Port of Livorno",
    "terminal": "TDT",
    "transportEquipmentId": null,
    "transportEquipmentRef": null,
    "transportationPhase": "Import",
    "vehicleId": "FX085FX",
    "vehicleName": "Truck",
    "voyageId": "19798"
  },
  "userOwner": "did:vtn:trustid:6776c3d9a9a6f5e33835081ce5f1c7973d0be70e22565e4767f3742f50dae909"
}

```

Figure 118: DigitalAsset for the GateOut event in Livorno seaport.

```

"output": {
  "networkId": 2,
  "hash": "0yhiJIagT9MUQskU+9QRVzegpaaowb/HGeQ0eljzRsc=",
  "previousHash": "0yhiJIagT9MUQskU+9QRVzegpaaowb/HGeQ0eljzRsc=",
  "timestamp": 1673439026,
  "init": 0,
  "end": 1651160013,
  "smartContract": "0x5c218EE49d5bd7DFB4d755Fa2552AB984d14D63C",
  "transaction": "0xf3b868b3a735069d9f3ec95e3ed0013e4fe26e43a7eed07576f78574a31481d2",
  "includedTransactions": [
    {
      "metadata": {
        "billOfLadingNumber": "MEDUDM329279",
        "carrierBookingNumber": null,
        "equipmentNumber": "MEDU4654152",
        "eventOccurrenceTime8601": "2021-10-14T10:58:00Z",
        "eventSubmissionTime8601": "2021-10-03T11:00:00Z",
        "location": "ITLIV",
        "originatorId": "ITLIV",
        "originatorName": "Port of Livorno",
        "terminal": "LRN",
        "transportEquipmentId": "NULL",
        "transportEquipmentRef": "NULL",
        "transportationPhase": "Import",
        "vehicleId": "9141297",
        "vehicleName": "Ship",
        "voyageId": 19814
      },
      "timestamp": 1651160013,
      "userOwner": "did:vtn:trustid:6776c3d9a9a6f5e33835081ce5f1c7973d0be70e22565e4767f3742f50dae909",
      "hash": "E5RWIt1AoKR3u000b8r4EBSpy5Kt7oa5iGvNICot9uE="
    }
  ]
}

```

Figure 119: Trustpoint of the VesselArrival event in Livorno seaport.

```

"output": {
  "networkId": 2,
  "hash": "Xp5psQVpNfcLsLiQ8u0IM+VgmYXQht5lFr3sPmIqAeo=",
  "timestamp": 1673439144,
  "init": 0,
  "end": 1651160016,
  "smartContract": "0x5c218EE49d5bd7DFB4d755Fa2552AB984d14D63C",
  "transaction": "0x044cf321a83389614f48d2f91d1c89b9bdb4a9aa469a3a24075419e0c9c718d2",
  "includedTransactions": [
    {
      "metadata": {
        "billOfLadingNumber": null,
        "carrierBookingNumber": null,
        "equipmentNumber": "SEGU6725734",
        "eventOccurrenceTime8601": "2021-08-01T09:16:30Z",
        "eventSubmissionTime8601": "2021-08-01T09:21:00Z",
        "location": "ITLIV",
        "originatorId": "ITLIV",
        "originatorName": "Port of Livorno",
        "terminal": "LRN",
        "transportEquipmentId": "NULL",
        "transportEquipmentRef": "NULL",
        "transportationPhase": "export",
        "vehicleId": 9646665,
        "vehicleName": "Ship",
        "voyageId": 1969
      },
      "timestamp": 1651160016,
      "userOwner": "did:vtn:trustid:6776c3d9a9a6f5e33835081ce5f1c7973d0be70e22565e4767f3742f50dae909",
      "hash": "m9ajYRSb4J2q4/6Bwye/jA5Kn0jK8EazJFN/V1qufEI="
    }
  ]
}

```

Figure 120: Trustpoint of the VesselDeparture event in Livorno seaport.

```

"output": {
  "networkId": 2,
  "hash": "0JtyArWbRhJ7pKTQe7e3ZzTjdw80PiNpVc102vsRmG8=",
  "timestamp": 1673439216,
  "init": 0,
  "end": 1651160133,
  "smartContract": "0x5c218EE49d5bd7DFB4d755Fa2552AB984d14D63C",
  "transaction": "0x07460d973f0831e0e65c84dce7ecb1badd701cf371f6eae76ff33bb9762d242",
  "includedTransactions": [
    {
      "metadata": {
        "billOfLadingNumber": "908291711",
        "carrierBookingNumber": null,
        "equipmentNumber": "HASU4605254",
        "eventOccurrenceTime8601": "2021-08-01T12:00:30.8510000+01:00",
        "eventSubmissionTime8601": "2021-08-01T11:04:00Z",
        "fullStatus": "F",
        "location": "ITLIV",
        "originatorId": "ITLIV",
        "originatorName": "Port of Livorno",
        "terminal": "LRN",
        "transportEquipmentId": null,
        "transportEquipmentRef": null,
        "transportationPhase": "export",
        "vehicleId": "BV892CT",
        "vehicleName": "Truck",
        "voyageId": "20488"
      },
      "timestamp": 1651160133,
      "userOwner": "did:vtn:trustid:6776c3d9a9a6f5e33835081ce5f1c7973d0be70e22565e4767f3742f50dae909",
      "hash": "lDn/KSPQMqWIraVWdhp9z8R8B8z0gg0Y3Gw4be3VB34="
    },
    {
      "metadata": {
        "billOfLadingNumber": "MEDUCJ362325",

```

Figure 121: Trustpoint of the GateIn event in Livorno seaport.

```

"output": {
  "networkId": 2,
  "hash": "jmTfYL0YcZ0Etl/jF0B0a0u7t4FQZX4/T8q4+cuR9dI=",
  "timestamp": 1673439366,
  "init": 0,
  "end": 1651160021,
  "smartContract": "0x5c218EE49d5bd7DFB4d755Fa2552AB984d14D63C",
  "transaction": "0xfc12d7004756f96fc3c3142402593aab041ea5383fd074440334842af028419d",
  "includedTransactions": [
    {
      "metadata": {
        "billOfLadingNumber": "GQL0225748",
        "carrierBookingNumber": null,
        "equipmentNumber": "APRU6115615",
        "eventOccurrenceTime8601": "2021-07-30T18:40:03.5430000+01:00",
        "eventSubmissionTime8601": "2021-07-30T19:04:08.671Z",
        "fullStatus": "F",
        "location": "ITLIV",
        "originatorId": "ITLIV",
        "originatorName": "Port of Livorno",
        "terminal": "TDT",
        "transportEquipmentId": null,
        "transportEquipmentRef": null,
        "transportationPhase": "Import",
        "vehicleId": "FX085FX",
        "vehicleName": "Truck",
        "voyageId": "19798"
      },
      "timestamp": 1651160021,
      "userOwner": "did:vt:trustid:6776c3d9a9a6f5e33835081ce5f1c7973d0be70e22565e4767f3742f50dae909",
      "hash": "N3Pe5eT+3+TheJRD3JR5Tro4uI1tX8Bq1oKvGboTjc="
    }
  ]
}
    
```

Figure 122: Trustpoint of the GateOut event in Livorno seaport.

SCENARIO 2

Technical specifications of the hosting environment and the used APIs for this scenario are provided in deliverable D6.2 [2] and [17]:

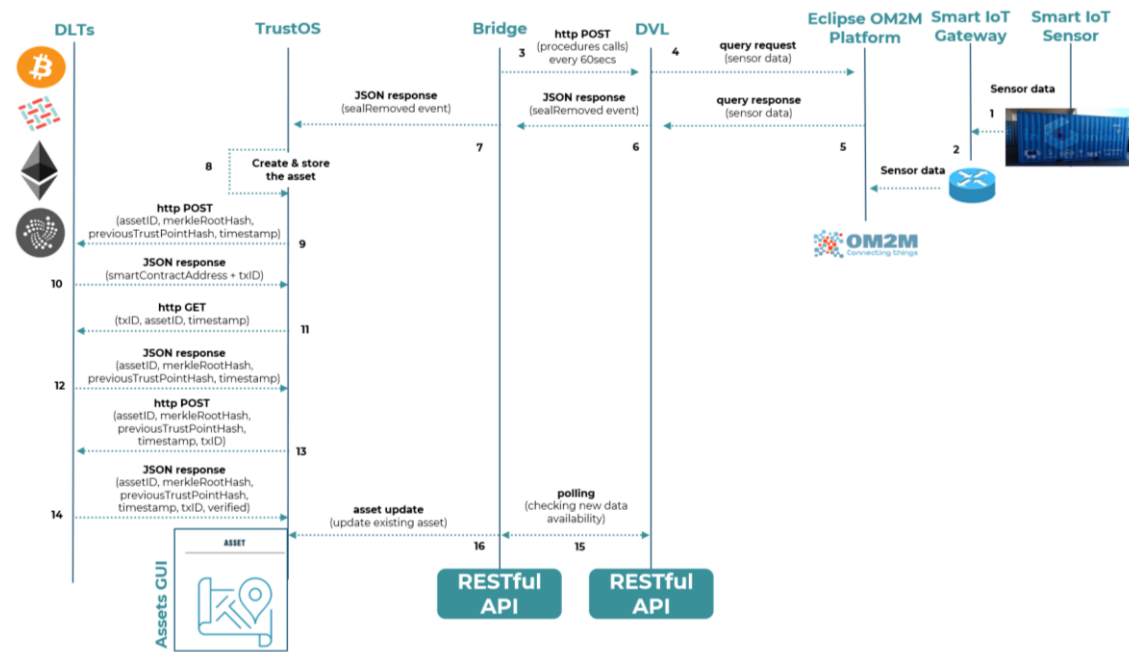


Figure 123: Sequence diagram for the demonstration of the Scenario 2.



Attribute	Type	Source
originatorName	Static	Port of Valencia
originatorId	Static	VLC
equipmentNumber	Static	ZIMU1381282
eventOccurrenceTime8601	Dynamic	SENSOR_DATA/signal_state/.../ct
smdgTerminal	Static	CSP Iberian Valencia Terminal
latitude	Dynamic	SENSOR_DATA/GPS/.../con/latitude
longitude	Dynamic	SENSOR_DATA/GPS/.../con/longitude
sealType	Static	Carrier
sealNumber	Dynamic	CONTAINER_INFO/.../con/dev_eui
signalState	Dynamic	SENSOR_DATA/signal_state/.../con/signal_state

Table 131. sealRemoved event data model.

```

{
  "@odata.context": "$metadata#Collection(ses.1.ONEM2M.SealData_RSParam)",
  "value": [
    {
      "originatorName": "Port of Valencia",
      "originatorId": "VLC",
      "equipmentNumber": "ZIMU1381282",
      "eventOccurrenceTime8601": "20220406T101543",
      "smdgTerminal": "CSP Iberian Valencia Terminal",
      "latitude": "49.70000076293945",
      "longitude": "6.340000152587891",
      "sealType": "Carrier",
      "sealNumber": "a8a178daf4c03631",
      "sealState": "true"
    }
  ]
}
    
```

Figure 124: sealRemoved event data at DVL level.

SCENARIO 3

Technical specifications of the hosting environment and APIs for this scenario are provided in deliverable D6.2 [2] and D5.3 [17]:



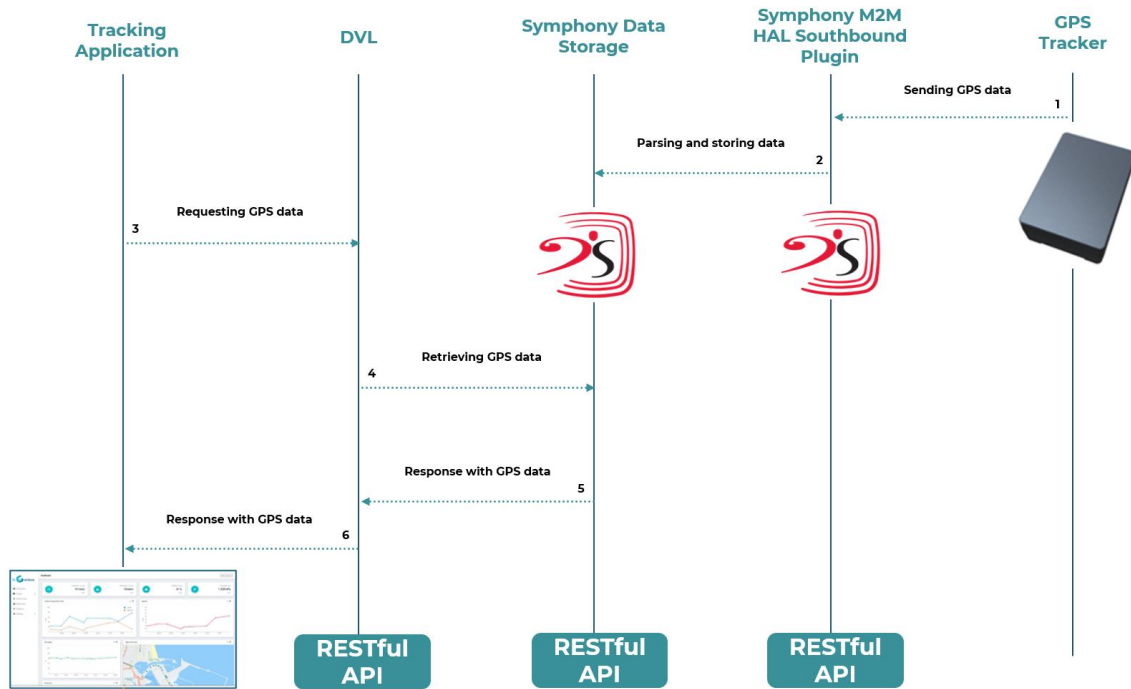


Figure 125: Sequence diagram for the demonstration of the Scenario 3.

Content	Length(Byte)
Head	2
Separator	1
Mode	1
Separator	1
IMEI	15
Separator	1
Data Type	2
Separator	1
Report Data	90

Figure 126: IoT Tracking Sensor message format.

Item	Max Length(B)
Satellite no.	2
Separator	1
Time Stamp	12
Separator	1
Latitude	9
Separator	1
Longitude	10
Separator	1
Speed	8
Separator	1
Heading	3
Separator	1
Event ID	1
Separator	1
Battery Voltage	4
Separator	1
Sequence number	3

Figure 127: IoT Tracking Sensor GPS message.

```

"@odata.context": "$metadata#Collection(symphony.1.tracker.fetchTracker_RSPParam)",
"value": [
  {
    "ts": "1651668467",
    "v": "{head=MT, data_type=R0, event_id=2, timestamp=220504124745, longitude=10.31235, mode=6, latitude=43.57805, seq_no=118, voltage=4147, imei=867035047590771, speed=7.4, heading=218, satellite_no=8}"
  },
  {
    "ts": "1651668466",
    "v": "{head=MT, data_type=R0, event_id=5, timestamp=220504124744, longitude=10.31239, mode=6, latitude=43.57808, seq_no=117, voltage=4147, imei=867035047590771, speed=11.29, heading=225, satellite_no=8}"
  },
  {
    "ts": "1651668457",
    "v": "{head=MT, data_type=R0, event_id=2, timestamp=220504124735, longitude=10.31255, mode=6, latitude=43.57794, seq_no=116, voltage=4147, imei=867035047590771, speed=15.18, heading=8, satellite_no=8}"
  }
]
    
```

Figure 128: GPS data coming from the Symphony M2M Platform and aggregated at DVL level.

SCENARIO 4

Technical specifications of the hosting environment and APIs of this scenario are provided in deliverable D6.2 [2] and D5.3 [17]:



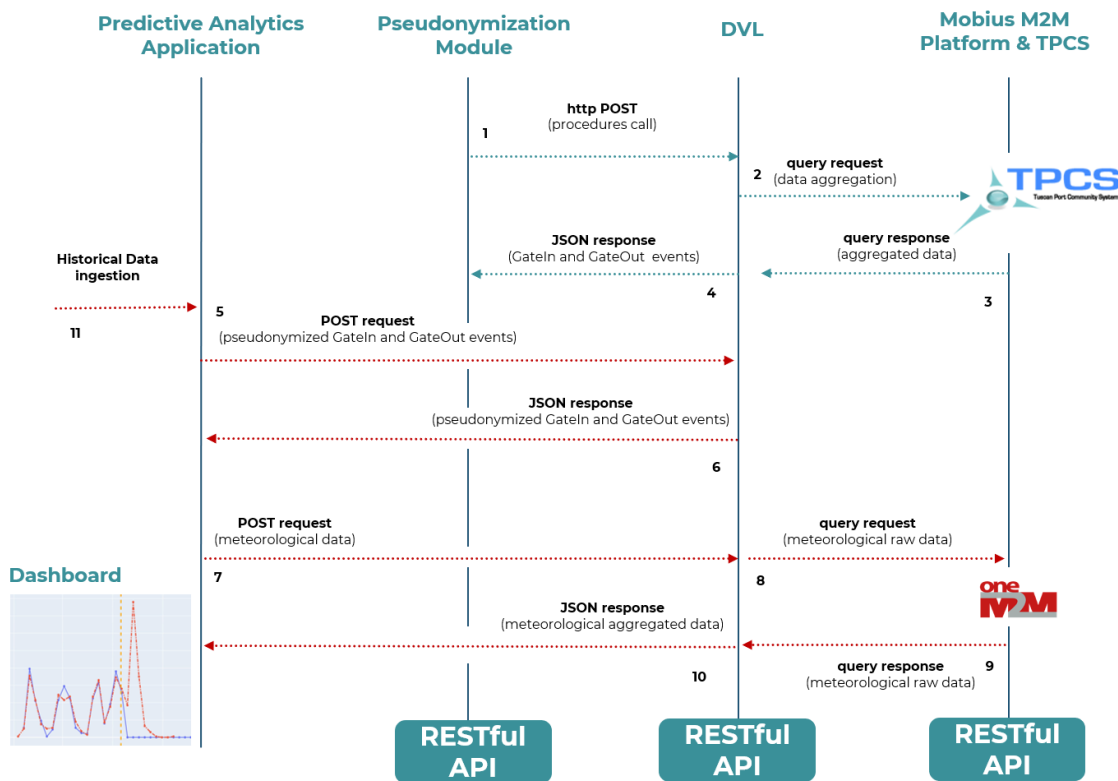


Figure 129: Sequence diagram for the demonstration of the Scenario 4.

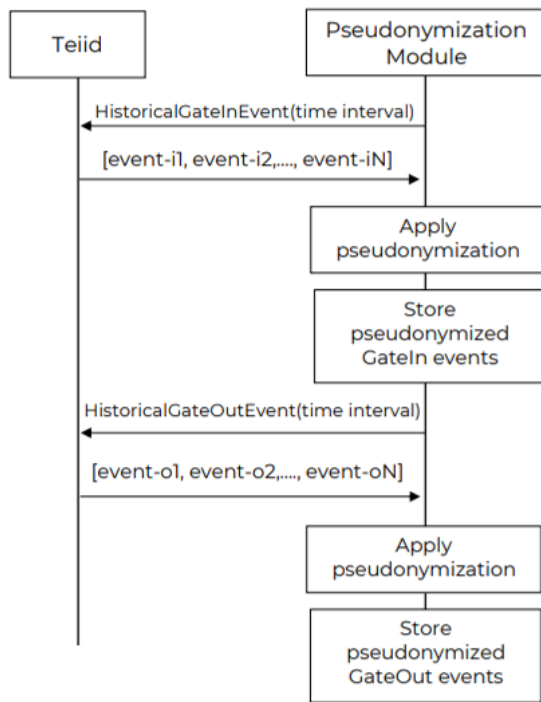


Figure 130: The main interactions between the DVL and Pseudonymized Module.



Validation and Results

TEST CASES VERIFICATION

Test Case Id	UC6_TC_01
Test case description	Interaction between OneM2M platform and Data Virtualization Layer. The test should demonstrate the interaction is working properly according to defined system requirements.
System requirements covered	UC6_SR_01, UC6_SR_02
Expected result	Correct meteorological data retrieval from OneM2M platform. Correct data processing and its availability at DVL layer.
Actual result	The implemented RESTful interface in DVL allows to retrieve meteorological data from two meteorological stations deployed in Livorno seaport when invoked by Awake.AI platform.
Passed/Failed	Passed

Table 132. UC6_TC_01 verification.

UC6_TC_01: This test case is intended to verify the integration between the DVL and OneM2M machine-to-machine platform. A RESTful interface was developed at DVL allowing data retrieval from the underlying OneM2M platform. The interface was tested by using Postman as a testing tool in order to verify that the HTTP request is correctly executed. The HTTP response was then benchmarked against the expected one, with a positive result: meteorological data was in line with the defined data model (Scenario 4). The RESTful interface and the implemented translator between the DVL and OneM2M platform behave as expected.

Test Case Id	UC6_TC_02
Test case description	Interaction between OM2M platform and Data Virtualization Layer. The test should demonstrate the interaction is working properly according to defined system requirements.
System requirements covered	UC6_SR_01, UC6_SR_02
Expected result	Correct sensor data retrieval from OM2M platform. Correct data processing and its availability at DVL layer.
Actual result	The implemented RESTful interface in DVL allows to retrieve sealRemoved event when remotely invoked by TrustOS.
Passed/Failed	Passed

Table 133. UC6_TC_02 verification.

UC6_TC_02: This test case is intended to verify the integration between the DVL and Eclipse OM2M machine-to-machine platform. A RESTful interface was developed at DVL allowing data retrieval from the underlying OM2M platform. The interface was tested by using Postman as a testing tool in order to verify that the HTTP request is correctly executed. The HTTP response was then benchmarked against the expected one, with a positive result: sealRemoved data was in line with the defined data model (Scenario 2). The RESTful interface



and the implemented translator between the DVL and Eclipse OM2M platform behave as expected.

Test Case Id	UC6_TC_03
Test case description	Interaction between PISystem platform and Data Virtualization Layer. The test should demonstrate the interaction is working properly according to defined system requirements.
System requirements covered	UC6_SR_01, UC6_SR_02
Expected result	Correct GateIn/GateOut data retrieval from PISystem platform. Correct data processing and its availability at DVL layer.
Actual result	The implemented RESTful interface in DVL allows to retrieve GateIn and GateOut events when invoked by TrustOS.
Passed/Failed	Passed

Table 134. UC6_TC_03 verification.

UC6_TC_03: This test case is intended to verify the integration between the DVL and PISystem machine-to-machine platform. A RESTful interface was developed at DVL allowing data retrieval from the underlying PISystem platform. The interface was tested by using Postman as a testing tool in order to verify that the HTTP request is correctly executed. The HTTP response was then benchmarked against the expected one, with a positive result: GateIn and GateOut data was in line with the defined data model (Scenario 1). The RESTful interface and the implemented translator between the DVL and PISystem platform behave as expected.

Test Case Id	UC6_TC_04
Test case description	Interaction between DVL, Integration Bridge, TrustOS and the set of DLT providers. The test should demonstrate the interaction is working properly according to defined system requirements.
System requirements covered	UC6_SR_05, UC6_SR_08, UC6_SR_10, UC6_SR_11, UC6_SR_12, UC6_SR_13, UC6_SR_15
Expected result	Correct storing of the data on the different DLTs.
Actual result	Correct display of event information and TrustPoints generated for each of the DLT providers.
Passed/Failed	Passed

Table 135. UC6_TC_04 verification.

UC6_TC_04: This test case is intended to verify that the interaction between the DVL, Integration Bridge, TrustOS and DLTs works as expected according to the defined system requirements. The test consisted in the following steps: i) the Integration Bridge was able to retrieve the maritime events' data from the DVL by using the available set of APIs, ii) the Integration Bridge was able to send HTTP request to TrustOS for events storage as well as for the creation of the corresponding set of Digital Assets, ii) TrustOS was able to create the corresponding trusponts and to store them in all available DLTs (Bitcoin, IOTA, Ethereum and Hyperledger Fabric) by using the common API, iii) the Integration Bridge was able to detect the availability of new events from the DVL with a polling mechanism sending a request to TrustOS to update the



corresponding Digital Asset which was correctly updated, iv) the availability of all stored trustpoints in DLTs was checked by using a GUI integrated with TrustOS.

Test Case Id	UC6_TC_05
Test case description	Mapping of the access roles for Data Virtualization Layer consumers (e.g: TrustOS, Awake.AI and PF module): RBAC – Role-Based Access Control.
System requirements covered	UC6_SR_03
Expected result	The considered Data Virtualization Layer consumer has the permission to perform only assigned operations (CRUD basic operations).
Actual result	All available maritime events are retrieved through the exposed RESTful interfaces at DVL by the following data consumers: TrustOS, PF module and Awake.AI. For all of them only reading operations are allowed.
Passed/Failed	Passed

Table 136. UC6_TC_05 verification.

UC6_TC_05: This test case is intended to verify that data consumers which interact with DVL have defined access rules properly set. First, a proper rule (according to CRUD – Create, Read, Update and Delete) is defined in the corresponding Virtual Database file (.xml), implemented in DVL. A given data consumer is then identified in a form of resource path within the Virtual database (e.g., TrustOS, PF module, Tracking Application, etc.). For the considered consumer, only READ permissions are assigned. To test the correct behaviour of the rule, a dummy schema was created, and a testing application acted as a consumer in order to verify assigned roles were properly working. This included two different scenarios: i) the data consumer attempts to invoke a specific operation he is not authorized for and ii) the data consumer attempts to invoke a specific operation he is authorized for, according to the defined virtual schema. The test result was positive: in the first scenario, the data consumer was not able to perform the HTTP request with Update SQL statement, while in the second case the requested information was correctly retrieved.

Test Case Id	UC6_TC_06
Responsible Partner	TEI
Use Case	Supply Chain Ecosystem Integration
Test case description	All personal data received by Data Virtualization Layer has to be pseudonymized so that, when stored, it is never in cleartext format.
Prerequisites	Incoming personal data
Type of test	Non-functional test
Reference standards used	N/A.
Test Environment	The test is expected to be executed in a laboratory environment (Joint Lab staging farm).



Input to the system	GateIn and GateOut data coming from the Tuscan Port Community System (TPCS) used in Livorno seaport.
Output of the system	Personal data is pseudonymized (pseudonym).
Data involved in the test	GateIn and GateOut data
System requirements covered	UC6_SR_14, UC6_SR_15, UC6_SR_16, UC6_SR_19
Related KPIs	Confidentiality and integrity protection of personal data, Logs of privacy events.
Are UC's users involved in the test?	No
Who will perform the test?	TEI/CNIT
Test Steps	Pseudonymization function is fed by sensor data; Personal data are pseudonymized (pseudonym production) and a retention period is associated to it;
Risks	No risks foreseen
Mitigation	N/A
Expected result	No personal data in cleartext format; In case a conversion table is needed for the selected pseudonymization function, it is stored in encrypted repository.
Actual result	GateIn and GateOut events are available at DVL level with the field "truck plate number" pseudonymized.
Passed/Failed	Passed

Table 137. UC6_TC_06 verification.

UC6_TC_06: This test case was intended to verify that personal data of the GateIn/GateOut events handled by DVL is processed by the pseudonymization function module so that it is not available in clear-text format within the DVL. This test uses the two interfaces HistoricalGateInEvent and HistoricalGateOutEvent. Through such RESTful interfaces, the PF module fetches the events from the DVL, and then, based on the pseudonymization technique used (for the test, Hashing-With-Key), generates pseudonyms associated with the personal data, storing all of them in an encrypted internal database. As for clarification, the only personal data handled by DVL is the vehicleId field (namely the truck plate number) of the GateIn/GateOut events (in Livorno seaport). The test case was successfully verified.

Test Case Id	UC6_TC_07
Responsible Partner	TEI
Use Case	Supply Chain Ecosystem Integration
Test case description	DVL (authorized entity) can fetch data, in pseudonymized format, from PF module
Prerequisites	Events exists in PF module
Type of test	Non-functional test



Reference standards used	N/A.
Test Environment	The test is executed in a laboratory environment (Joint Lab staging farm).
Input to the system	A request on dedicated interface, FetchGateEvents, with the following data: {startdate, enddate, gate}
Output of the system	Events in the time frame requested are returned with personal data in pseudonymized format
Data involved in the test	Pseudonym stored in encrypted DB into PF module
System requirements covered	UC6_SR_14, UC6_SR_17
Related KPIs	Confidentiality and integrity protection of personal data, Logs of privacy events.
Are UC's users involved in the test?	No
Who will perform the test?	TEI/CNIT
Test Steps	The DVL asks for events in a specified time frame; The Pseudonymization Function (PF) retrieve data events, including personal data in pseudonym format and returns them to DVL. Note: only DVL (pseudonymization entity) has the rights to fetch data from PF module.
Risks	No risks foreseen
Mitigation	N/A
Expected result	Only authorized partners can access personal data in cleartext format.
Actual result	GateIn and GateOut events are available at DVL level with the field "truck plate number" pseudonymized.
Passed/Failed	Passed

Table 138. UC6_TC_07 verification.

UC6_TC_07: This test case aims to verify that personal datasets are not provided in clear-text format to external applications that access the DVL. In particular, it has been verified that when an application requests the GateIn/GateOut events to the DVL through the FetchGateEvents interface, the DVL requests such events in pseudonymized format to the PF module via the FetchGateEvents RESTful interface. The events are properly returned with the vehicleId field (namely the truck plate number) in encrypted format by using a pseudonym. The FetchGateEvents interface exposed to the DVL is password protected, so that only the pseudonymization entity (the DVL) can use it. The test case was successfully verified.

Test Case Id	UC6_TC_08
Responsible Partner	TEI
Use Case	Supply Chain Ecosystem Integration

Test case description	Personal Data cannot be stored forever, when retention period expires personal data has to be cancelled.
Prerequisites	Events exists in PF module with date out of retention period
Type of test	Non-functional test
Reference standards used	N/A.
Test Environment	Test executed on staging environment with local instance of Pseudonymization Function module running in a dedicated Virtual Machine.
Input to the system	N/A
Output of the system	All the personal data out of retention period is canceled.
Data involved in the test	Personal data and pseudonyms in Conversion Table
System requirements covered	UC6_SR_18
Related KPIs	Confidentiality and integrity protection of personal data, Logs of privacy events.
Are UC's users involved in the test?	No
Who will perform the test?	TEI
Test Steps	Every night a process in n Pseudonymization Function module checks if personal data with expired retention period exist. In that case it personal data in conversion table.
Risks	No risks foreseen
Mitigation	N/A
Expected result	No personal data with expired retention period stored in the Interoperable Layer.
Actual result	Personal data with expired retention period are not available.
Passed/Failed	Passed

Table 139. UC6_TC_08 verification.

UC6_TC_08: The PF module introduces an auditor function that once per day checks personal data stored in encrypted DB. If the timestamp of the stored data is older than the retention period (default value set to 5 years) the personal data is removed. To test this functionality, the retention period was set to 1 day via the configTemplate API to force the auditor function to delete data. The test case was successfully verified.

Test Case Id	UC6_TC_09
---------------------	------------------



Responsible Partner	TEI
Use Case	Supply Chain Ecosystem Integration
Test case description	Data Owner can request to the platform to cancel own personal data.
Prerequisites	Events exists in PF module
Type of test	Non-functional test
Reference standards used	N/A.
Test Environment	Test executed on staging environment with local instance of Pseudonymization Function module running in a dedicated Virtual Machine.
Input to the system	N/A
Output of the system	All personal data will be canceled after Data Owner requests for deletion.
Data involved in the test	Personal data in Conversion Table, if any
System requirements covered	UC6_SR_18
Related KPIs	Confidentiality and integrity protection of personal data, Logs of privacy events.
Are UC's users involved in the test?	No
Who will perform the test?	TEI
Test Steps	A Data Owner submits a right-to-be-forgotten request, this is simulated sending a request to "deleteData" API to PF module: The request is processed by pseudonymization function removing personal data from conversion table.
Risks	No risks foreseen
Mitigation	N/A
Expected result	Personal data involved in the deleteData request is canceled from PF module.
Actual result	Personal data are removed from the conversion table.
Passed/Failed	Passed

Table 140. UC6_TC_09 verification.

UC6_TC_09: To enable the "right to be forgotten" functionality, the PF module implements the deleteData interface. Through this interface, the DVL (pseudonymization entity) can ask the PF module to remove specific personal



dataset. The test execution verified that after the deletion of the personal data via deleteData interface, it was not more available in the encrypted DB. The test case was successfully verified.

Test Case Id	UC6_TC_10
Test case description	Views and query results caching capability in case underlying data does not change frequently.
System requirements covered	UC6_SR_04
Expected result	Query results are properly cached and properly retrieved by a test application.
Actual result	The query results of the SQL statement are retrieved from the cache when the same RESTful interface is invoked more than one time (for GateIn, GateOut, VesselArrival and VesselDeparture events in Livorno seaport).
Passed/Failed	Passed

Table 141. UC6_TC_10 verification.

UC6_TC_10: This test case is intended to verify the DVL’s caching capability of the query’s results in case the data included within the response from the APIs does not change frequently. In order to retrieve and aggregate data at DVL level, a specific query was implemented and included in a virtual database configuration file (.xml). A test application was used to perform the query and the query results were cached according to the virtual database configuration file. A test application performed the same query again and the main results were taken from the cache as expected, instead of being retrieved from the underlying data source.

Test Case Id	UC6_TC_11
Responsible Partner	CNIT
Use Case	Supply Chain Ecosystem Integration
Test case description	Interaction between TPCS and Data Virtualization Layer. The test should demonstrate the interaction is working properly according to defined system requirements.
Prerequisites	TPCS and DVL instances properly configured in a staging environment.
Type of test	Non-Functional testing (integration test).
Reference standards used	N/A
Test Environment	The test is expected to be executed in a laboratory environment (Joint Lab staging farm).
Input to the system	GateIn, gateOut, VesselArrival and VesselDeparture data from TPCS.
Output of the system	Virtual view of extracted data at DVL level.
Data involved in the test	GateIn, gateOut, VesselArrival and VesselDeparture data from TPCS.



System requirements covered	UC6_SR_01, UC6_SR_02
Related KPIs	Data Virtualization Layer Scalability
Are UC's users involved in the test?	No
Who will perform the test?	CNIT/AdSPMITS
Test Steps	1. By using a tool for APIs' testing (e.g. Postman), HTTP request is sent to DVL (note that at this stage the translator for the communication with TPCS is implemented and unit tests have been performed correctly). 2. The result of the HTTP request is visualized and checked in order to make sure that the expected data are properly formatted (GateIn, GateOut, VesselArrival and VesselDeparture data in Livorno seaport).
Risks	No risks are foreseen.
Mitigation	N/A
Expected result	Correct data retrieval from TPCS. Correct data processing and its availability at DVL layer.
Actual result	The GateIn, GateOut, VesselArrival and VesselDeparture events are properly retrieved and visualized according to the adopted data model.
Passed/Failed	Passed

Table 142. UC6_TC_11 verification.

UC6_TC_11: This test case is intended to verify the integration between the DVL and the TPCS (Tuscan Port Monitoring system) platform. A RESTful interface was developed at DVL allowing data retrieval from the underlying TPCS platform. The interface was tested by using Postman as a testing tool in order to verify that the HTTP request is correctly executed. The HTTP response was then benchmarked against the expected one, with a positive result: GateIn, GateOut, VesselArrival and VesselDeparture data was in line with the defined data model (Scenario 1). The RESTful interface and the implemented translator between the DVL and TPCS platform behave as expected.

Test Case Id	UC6_TC_12
Responsible Partner	CNIT
Use Case	Supply Chain Ecosystem Integration
Test case description	Integration between Symphony M2M Platform and Data Virtualization Layer. The test should demonstrate the interaction is working properly according to defined system requirements.
Prerequisites	Symphony M2M Platform and DVL instances properly configured in a staging environment.
Type of test	Non-Functional testing (integration test).
Reference standards used	N/A
Test Environment	The test is expected to be executed in a laboratory environment (Joint Lab staging farm).



Input to the system	Trucks' tracking data from Symphony M2M Platform.
Output of the system	Virtual view of extracted data at DVL level.
Data involved in the test	Trucks' GPS coordinates.
System requirements covered	UC6_SR_01
Related KPIs	Data Virtualization Layer Scalability
Are UC's users involved in the test?	Yes
Who will perform the test?	CNIT/NXW/UPV
Test Steps	<ol style="list-style-type: none"> 1. By using a tool for APIs' testing (e.g. Postman), HTTP request is sent to DVL (note that at this stage the translator for the communication with Symphony M2M Platform is implemented and unit tests have been performed correctly). 2. The result of the HTTP request is visualized and checked in order to make sure that the expected data are properly formatted. 3. A web application consumes data through a RESTful interface implemented at DVL level and visualize them correctly.
Risks	No risks are foreseen.
Mitigation	N/A
Expected result	Correct data retrieval from Symphony M2M Platform. Correct data processing and its availability at DVL layer. Correct data visualization by means of a web based application.
Actual result	Data are correctly visualized through a GUI and the truck's path is displayed in a map.
Passed/Failed	Passed

Table 143. UC6_TC_12 verification.

UC6_TC_12: This test case is intended to verify the integration between the DVL and the Symphony machine-to-machine platform. A RESTful interface was developed at DVL allowing data retrieval from the underlying Symphony platform. The interface was tested by using Postman as a testing tool in order to verify that the HTTP request is correctly executed. The HTTP response was then benchmarked against the expected one, with a positive result: geolocation data from the IoT tracking device was in line with the defined data model (Scenario 3). The RESTful interface and the implemented translator between the DVL and Symphony platform behave as expected.

KPIS

Data Virtualization Layer scalability: The total number of simultaneous M2M platforms used during the demonstration was four. Although the Data Virtualization Layer can suffer from scalability constraints (e.g. use cases based on data trending/historical analysis), the proposed solution utilizes a caching-based mechanism to compensate for this issue.

Data Virtualization Layer data processing: Overall latency to retrieve the maritime events from DVL was measured as a sum between network latency and the time required for the execution of the specific SQL statement (1400ms as an average value over five consecutive tests). Real-time data integration requirement was achieved.

Data Virtualization Layer access control: Data roles, also called entitlements, are sets of permissions that are defined per VDB that dictate data access (create, read, update and delete). In the scope of the INGENIOUS project, for each exposed API the specific role has been assigned according to the expected operations to be performed by the applications running on top of DVL. Once the application is authenticated at DVL (e.g., TrustOS, PF module, smart applications for trucks' localization and predictive models), the "ReadOnly" role is assigned accordingly so that any unauthorized operation over the underlying data sources is prevented (e.g., create, update or delete). Therefore, the data access at DVL level is role-based.

Cross-DLT layer access control: A general identity has been generated in TrustOS to authorise all requests for registration of event information coming to the platform. It was decided to do so for simplicity, but for future versions an identity could be generated for each of the identified entities that need to log information both in TrustOS and in the different DLT providers and thus provide granularity of roles.

Cross-DLT layer scalability: TrustOS currently has integration with different DLT providers (Ethereum, Göerli, Besu, Polygon and Mumbai). In addition, integration with Bitcoin, IOTA, Hyperledger Fabric has been included for the INGENIOS project. Currently it is possible to register information (TrustPoints) of the different events coming from the DVL simultaneously in all these DLTs.

Availability of the DLT connectivity layer: TrustOS is running within an environment supported by Telefonica based on 8x5 schedule (8 hours per day, from Monday to Friday).

Data processing time in DLTs: The type of request that offers the highest latency are write requests, specifically, in this case, these requests are related to the creation of TrustPoints based on event information. Depending on the DLT provider selected, this request will have a variable latency because it is necessary to wait for the confirmation of the block in which the transaction is added. Since the block generation time of the DLT providers used is different, it has been established that the response to the request is always generated as soon as the transaction identifier is available. This never exceeds a maximum of 15 seconds.

Cross-DLT concurrent requests: Currently, it is possible to launch a number of 8 concurrent requests for the creation of a TrustPoint (one for each of the DLT providers). This is achieved through replication, high availability and load balancing of the TrustOS instance.

Confidentiality and integrity protection of personal data: Personal data are stored in the Pseudonymization Function (PF). The PF module uses password protected RESTful API, which means that only authorized user (the DVL, acting as pseudonymization entity) can have access to the data. PF uses a table for conversion and this table is encrypted.

Logs of privacy events: The personal data is handled by DVL through Pseudonymization Function module. This module aims to obfuscate the data substituting the sensible info with pseudonyms. During this process the microservices involved (of Teiid and PF modules) produce logs information for debugging purposes. This KPI want to measure, in percentage, how many logs avoid including sensible information (the personal data). To cover all the SW a static analysis on code has been performed on last SW revision, checking that personal data is never included in log in cleartext format, so this KPI reach the target (100%).

