5G IANA: Lessons Learned and the Future Potential of 5G and Network Applications in the Automotive sector

Dissemination level: Public (PU)

White Paper lead: ICCS and UBITECH

Version: 1.0

Version Issuing Date: 15/05/2025

Partners:

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 101016427

5g-iana.eu

Authors		
Name	**Organisation**	**Email**
Xenofon Vasilakos	ICCS	x.vasilakos@iccs.gr
Dimitrios Klonidis	UBITECH	dklonidis@ubitech.eu
José Isola	BYLOGIX	jose.isola@bylogix.it
Markus Wimmer	NOKIA	markus.wimmer@nokia.com
Peter Zidar	Telecom Slovenije	Peter.Zidar@telekom.si
Rudolf Susnik	Internet Institute	rudolf.susnik@iinstitute.eu

Version history			
Version	**Date**	**Modified by**	**Summary of changes**
v0.1	09/04/2025	Xenofon Vasilakos (ICCS)	Initial ToC
v0.2	09/04/2025	Dimitris Klonidis (UBI), Xenofon Vasilakos (ICCS)	Revised ToC and Skeleton guidelines
v0.3	02/05/2025	Dimitris Klonidis (UBI), Xenofon Vasilakos (ICCS)	Revised ToC and Skeleton guidelines
v0.4	05/05/2025	Dimitris Klonidis (UBI), Xenofon Vasilakos (ICCS)	Contributions to Sections 1 and 2.
v0.5	08/05/2025	Xenofon Vasilakos (ICCS)	Executive summary draft; elaborated introduction with generic information on objectives. Use cases section 3.
V0.6	12/05/2025	Xenofon Vasilakos, Jose Isola	External experimenters' validation; Section 3.1.2.1., contributions to Section 4.
V0.7	14/05/2025	Markus Wimmer, Peter Zidar, Xenofon Vasilakos	Section 4.3 on feedback by stakeholders.
V1.0	15/05/2025	Reviewed by all partners owning Use Cases	Finalised version of the white paper before getting published on our website.

Legal disclaimer

Table of Contents

List of Figures

ABBREVIATIONS

Abbreviation	Definition
5G	Fifth Generation (mobile networks)
5G-IANA	5G Intelligent Automotive Network Applications
AF	Application Function
AI	Artificial Intelligence
AO	Application Orchestrator
AOEP	Automotive Open Experimentation Platform
API	Application Programming Interface
AR	Augmented Reality
AV	Autonomous Vehicle
CAF	Cloud-native Application Function
CAV	Connected and Automated Vehicle
CCAM	Cooperative, Connected, and Automated Mobility
CD	Continuous Delivery
CI	Continuous Integration
CI/CD	Continuous Integration / Continuous Deployment
CNF	Cloud-native Network Function
C-ITS	Cooperative Intelligent Transport Systems
CO	Confidential

CPU	Central Processing Unit
DL	Downlink
DML	Distributed Machine Learning
DMLO	Distributed Machine Learning Orchestrator
DN	Data Network
DoA	Description of Action
E2E	End-to-End
EC	European Commission
EEO	Extreme Edge Orchestrator
ETSI	European Telecommunications Standards Institute
FL	Federated Learning
FOV	Field-of-View
GNSS	Global Navigation Satellite System
GPS	Global Positioning System
GUI	Graphical User Interface
HMD	Head Mounted Display
HMI	Human Machine Interface
HTTP	Hypertext Transfer Protocol
ID	Identifier
KPI	Key Performance Indicator
LCM	Life Cycle Management
MANO	Management and Orchestration
MCAD	Manoeuvre Coordination for Autonomous Driving
MEC	Multi-access Edge Computing
ML	Machine Learning
MOS	Mean Opinion Score
MoM	Minutes of Meeting
MTBF	Mean Time Between Failure
MTTR	Mean Time To Repair
nApp	Network Application (a composition of atomic application and network function components)
NEST	Network Slice Type
NF	Network Function
NFV	Network Function Virtualisation
NOD	Network Orchestration Domain
NSD	Network Service Descriptor
NSI	Network Slice Instance
NW	Network
OBU	On-Board Unit
OEM	Original Equipment Manufacturer

OS	Operating System
OTT	Over The Top
PAF	Physical Application Function
PNF	Physical Network Function
QoE	Quality of Experience
QoS	Quality of Service
RAM	Random Access Memory
RAN	Radio Access Network
RSTP	Rapid Spanning Tree Protocol
RSU	Road-Side Unit
RTT	Round-Trip-Time
SBA	Service-Based Architecture
SDK	Software Development Kit
SME	Small and Medium Sized Enterprise(s)
TC	Test Case
TRL	Technology Readiness Level
UC	Use Case
UDP	User Datagram Protocol
UE	User Equipment
UL	Uplink
UPF	User Plane Function
V2I	Vehicle-to-Infrastructure
V2N	Vehicle-to-Network
V2V	Vehicle-to-Vehicle
V2X	Vehicle-to-Everything
VAF	Virtual Application Function
VM	Virtual Machine
VNFD	Virtual Network Function Descriptor
VNF	Virtual Network Function
vOBU	Virtual On-Board Unit
VR	Virtual Reality

Executive Summary

The 5G-IANA project establishes an open, standards-compliant 5G experimentation platform tailored to the needs of the automotive vertical. Central to this platform is the Automotive Open Experimentation Platform (AOEP), which integrates computational and communication infrastructure with orchestration capabilities, offering SMEs a robust environment to develop, test, and deploy services. The platform supports multiple virtualisation technologies and integrates advanced MANO frameworks to enable deployment across vehicles, infrastructure, edge, and cloud resources. A key asset is the enhanced nApp Toolkit, which is coupled with a repository of automotive VNFs and nApp templates, facilitating accelerated development cycles. Additionally, the project incorporates a Distributed Machine Learning (DML) framework, enabling privacy-preserving ML applications within automotive use cases. Seven real-world use cases across two 5G SA testbeds form the core of the project's demonstration and validation activities, underpinning its roadmap toward sustainable pan-European 5G automotive service deployment.

Experimentation under 5G-IANA highlights the effectiveness of a unified platform in addressing the end-to-end deployment of automotive services—from service onboarding to lifecycle management. The platform's compatibility with heterogeneous infrastructure segments (vehicles, Road Side Units (RSUs), edge, cloud) and support for varied virtualisation formats (VMs, containers, physical deployments) ensures flexibility in deployment scenarios. The availability of pre-tested nApp packages and automotive VNFs has proven critical in reducing integration time and enabling rapid prototyping. Furthermore, the adoption of secured APIs, service-level descriptors, and performance KPIs streamlines operational validation. The integration of a DML layer enhances adaptability and enables edge-based intelligence with privacy-conscious designs. Overall, the platform enables low-latency, high-reliability service deployment, supporting CCAM applications while providing practical pathways for SMEs to enter the 5G ecosystem.

Network Applications (nApps) present significant potential to transform connected mobility by enabling modular, scalable, and reusable software components that deliver both application-level logic and network-oriented functions. On the 5G-IANA platform, nApps are packaged with rich metadata*, designed for lifecycle orchestration*, and *optimised for deployment across cloud, edge, and far-edge i*nfrastructures. This flexibility allows for the creation of *tailored services*—from safety-critical V2X applications to infotainment, predictive maintenance, and real-time navigation. By lowering technical barriers through *reusable templates* and pre-integrated *toolkits*, 5G-IANA positions nApps as a practical entry point for third-party innovation in the automotive sector. The framework supports rapid prototyping and *shortens time-to-market*, while its integration with ML and slicing technologies opens new opportunities for intelligent, data-driven services aligned with evolving industry demands.

1 Introduction

1.1 Outline of Project Objectives

Funded under H2020-ICT-41-2020, the 5G Intelligent Automotive Network Applications (5G-IANA) project ran from June 2021 to November 2024, bringing together 16 partners across 8 EU countries, including 10 SMEs. It also supported external third-party SMEs via two Open Calls, with five out of these external parties succeeding in being awarded with funding support for experimenters.

FIGURE 1. OVERVIEW OF THE 5G-IANA PROJECT SCOPE AND STRUCTURE.

In the context of the automotive sector, services and applications are delivered through targeted Use Cases (UCs), each designed to demonstrate how network-enabled functionalities can meet real-world needs. These services are composed by combining distinct 5G capabilities—akin to assembling Lego bricks—into cohesive, end-to-end network applications. This stitching process, while technically complex, is greatly simplified through the 5G-IANA platform, which provides the tooling and orchestration logic needed to "glue" these building blocks together efficiently and reliably.

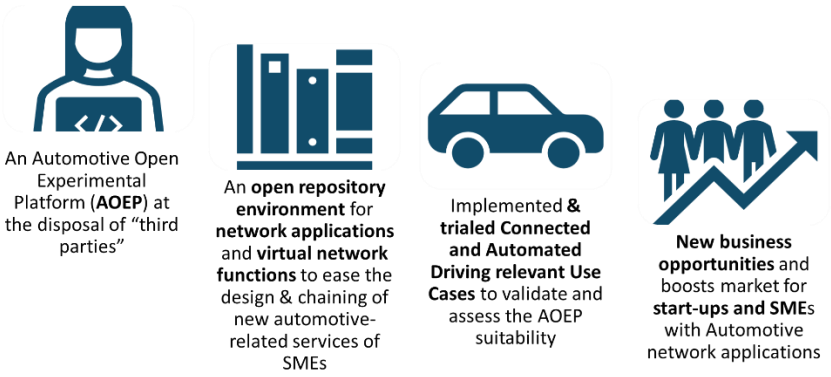


FIGURE 2. 5G-IANA OBJECTIVES.

As portrayed in Figure 2, The 5G-IANA project is driven by four core objectives that underpin its overall mission. At its centre is the Automotive Open Experimentation Platform (AOEP), a dedicated environment for accelerating the development, deployment, and validation of 5G-based network applications within the automotive domain. AOEP empowers third-party developers, especially SMEs, to design, chain, and test smart solutions through an open repository environment. It supports user-friendly integration of services within relevant UCs, facilitating their validation in realistic, trial-based scenarios. Ultimately, 5G-IANA seeks to stimulate innovation and unlock new business opportunities, establishing a strong and sustainable SME ecosystem across the European automotive landscape.

1.2 Background on the 5G-IANA initiative.

Fifth Generation-based (5G) automotive services, encompassing Connected and Automated Mobility (CAM), leverage 5G's high-speed, reliable, and versatile connectivity to enable a wide array of safety-critical and commercial applications in and around vehicles. 5G poses a unified, multi-service platform supporting digital transformation across industries, and creating significant opportunities for innovation and economic growth.

In response to both European Commission initiatives and market demands, the 5G Intelligent Automotive Network Applications (5G-IANA) project poses a push to establish *open, cooperative 5G experimentation platforms* where *third-party developers—especially SMEs—can test and deploy tailored applications*. The provision of accessible, open-source 5G resources, such as Virtual Network Functions (VNFs) and Network Applications (nApps), is essential to empower these smaller players to participate in the 5G ecosystem without needing the resources of major industry stakeholders. The latter, in particular, poses the cornerstone novelty and contribution of 5G-IANA, with details provided in Section 1.4.1.

Specifically, the 5G-IANA project establishes such an open experimentation environment enabling third-party developers to experiment with and test automotive-related applications efficiently. Central to this initiative is the design of the AOEP, a comprehensive multilayered system *spanning from the application layer to the infrastructure layer*, integrating both context-aware and network-aware capabilities to support the deployment of advanced services composed of chained virtualised Application Functions (Afs) and Network Functions (NFs):

- **Catalogue of pre-configured nApps**: 5G-IANA provides a *curated catalogue of pre-configured nApps*, featuring reusable and extendable AF and NF chains that Verticals can adapt for delivering 5G-enabled services.

- **Full suite of hardware and software aligned with Automotive sector**: Additionally, the project delivers a *full suite of hardware and software resources*—including computational, communication, and orchestration components—alongside a nApp Catalogue specifically aligned with Automotive sector needs. A key offering within this catalogue is a set of Starter Kits, which are baseline nApps grouped by service categories such as vehicle movement and infotainment, designed to support third parties in creating and evolving their own Automotive services.

- **Functionalities for simplifying design, chaining, provisioning, validation, and benchmarking:** The project also delivers functionalities aimed at simplifying the design, chaining, provisioning, validation, and benchmarking of new Automotive services. These capabilities are made *accessible to third parties* through an enhanced GUI, which further allows the specification of high-level QoS requirements related to each service.

- **Deployment, orchestration, and dynamic adaptation of Automotive services:** The project supports the deployment and orchestration of Automotive services and their associated components (i.e., nApps and AFs) from both application and networking perspectives, with capabilities for real-time monitoring and dynamic adaptation. Particular emphasis is placed on lightweight orchestration over On-Board Units (OBUs) and RSUs, ensuring flexible and scalable resource management.
- **Support for ML model definition and reusability:** The platform facilitates the definition and reuse of ML models by any Vertical service that incorporates model training into its business logic, such as in Federated Learning processes.
- **Abstraction of infrastructure complexity for interoperability:** The project abstracts underlying infrastructure and technology complexities, providing uniform, domain-agnostic deployment mechanisms that enable seamless cross-domain and cross-platform interoperability.

1.2.1 Novelty in far-edge integration and Distributed ML support

The project's key innovations focus on two main areas. First, it enables the integration and provisioning of *far-edge resources* for network management and orchestration,

empowering third-party developers to leverage the full potential of far-edge nodes such as On-board Units (OBUs) and Road-side Units (RSUs). The platform supports programmable OBUs/RSUs capable of hosting containerised services.

Second, 5G-IANA incorporates a *Distributed Machine Learning (DML)* framework within the 5G-IANA platform, offering *collected intelligence as a service* to Over The Top (OTT) and Original Equipment Manufacturer (OEM) third parties. For instance, data traffic monitoring can be used to generate Quality of Service (QoS) predictions that inform and optimise service provisioning strategies.

1.2.2 Compared to other HORIZON-2020 projects and approaches

The landscape of nApp-based research [SNS2024] within the 5G H2020 research ecosystem involves various projects, each of which brings unique perspectives and methodologies to their objectives, ambition and, hence, development and deployment.

While all aim to harness the capabilities of 5G networks to support vertical industries, their approaches differ in terms of architecture, lifecycle management, and integration strategies. For instance, the 5GASP [GAL2022] project emphasizes a unified methodology for designing, developing, and onboarding nApps, focusing on facilitating rapid development and testing of innovative 5G services. In contrast, the 5G-ERA[5GE2024] initiative concentrates on providing reference nApps tailored for specific verticals, highlighting the importance of middleware deployment strategies and lifecycle management. The EVOLVED-5G [EVO2022] project introduces a comprehensive framework for nApp development and evaluation, integrating validation environments and certification processes to ensure compliance with 5G standards. In the meanwhile, the 5G-PPP Software Network Working Group [5GP2022, 5GP2023] seeks to clarify the concept of nApps, advocating for a middleware layer that simplifies the implementation and deployment of vertical systems on a large scale.

Below, we engage into a concise comparative analysis focusing specifically on their approaches to nApps. This examination aims to *elucidate the similarities and differences* in their methodologies, particularly in relation to the detailed definition of nApps as adopted by the 5G-IANA project (see Section 1.4.1). But before proceeding with the comparative analysis, we note that the aforementioned projects are *not* the only ones, but at the time of authoring the current version of this document, they are the ones providing publicly available clear definitions. Also – as a general notice – 5G-IANA does not contradict but *rather extends* these projects nApp adaptation concepts by introducing detailed packaging standards encompassing operational requirements,

test cases, and performance metrics. This comprehensive approach enhances the ability to validate, deploy, and manage nApps effectively across diverse 5G environments.

1.2.2.1 5GASP

The 5GASP methodology defines a Network Application (nApp) as a modular and reusable software component developed to deliver specific functionalities within 5G verticals [GAL2022]. These nApps are constructed following a Service-Based Architecture (SBA) paradigm and may include both software and hardware elements. Designed to operate within 5G infrastructures, they adopt Network Function Virtualization (NFV) principles and expose standardised APIs for seamless integration with other network services.

Comparison to 5G-IANA: Both 5GASP and 5G-IANA emphasise modularity and reusability in nApp design. However, 5G-IANA further formalises the concept by introducing a comprehensive packaging strategy that embeds service-level metadata, interface definitions, and operational requirements—key additions that streamline rapid prototyping and enable scalable deployment across heterogeneous 5G domains.

1.2.2.2 5G-ERA

The initial deliverable from the 5G-ERA project defines nApps as modular software components that deliver specific functionalities tailored for vertical services in 5G systems [5GE2022]. These components are explicitly designed to support ease of integration and deployment across varying 5G infrastructures, aligning with the goal of accelerating service development in vertical domains.

In its updated reference framework, 5G-ERA refines the definition of nApps to span the full development lifecycle, from design and deployment to validation and operational adaptation [5GE2024]. A key characteristic is the dynamic manageability of nApps within 5G infrastructures, addressing vertical-specific demands with runtime flexibility.

Comparison to 5G-IANA: While both approaches converge on modularity and support for vertical enablement, 5G-IANA introduces an extended specification that incorporates detailed operational descriptors, testing procedures, and performance monitoring parameters. This enriches the reusability and integration capabilities of its nApp model, especially in complex or constrained deployment environments. Regarding the refined 5G-ERA's lifecycle-oriented view of [5GE2024], this aligns closely with the 5G-IANA approach. However, 5G-IANA complements this by embedding test cases, KPIs, and

metrics directly into the nApp packaging structure. This structured packaging not only accelerates integration and validation efforts but also enhances the capacity to ensure consistent performance across deployments.

1.2.2.3 EVOLVED-5G

The EVOLVED-5G project frames nApps as modular entities that undergo a full development, validation, and deployment pipeline within 5G infrastructure environments [EVO2022]. The approach places emphasis on support tooling, including CI/CD processes and certification environments, to ensure alignment with 5G standards and interoperability requirements.

Comparison to 5G-IANA: EVOLVED-5G and 5G-IANA share a strong alignment in supporting the complete nApp lifecycle. Nonetheless, 5G-IANA enhances this model with a robust packaging framework that defines operational and performance parameters upfront, enabling more deterministic deployments and clearer integration paths for third-party developers.

1.2.2.4 Definition of nApp by the 5G-PPP Software Network Working Group

The 5G-PPP Software Network Working Group presents Network Applications as modular software units that can be developed by third-party stakeholders and integrated into 5G networks to provide specific functionalities [5GP2022, 5GP2023]. The paper advocates for an open and accessible middleware layer to foster service innovation and simplify deployment across verticals.

Comparison to 5G-IANA: While both perspectives aim to democratise access to 5G capabilities via nApps, 5G-IANA introduces a more formalised and operationally grounded packaging structure. Its inclusion of interface specifications, slice requirements, and validation tools positions it as a more production-ready framework for vertical integration and service assurance.

1.3 Role of 5G in transforming connected mobility.

5G is a key enabler in the transformation of connected mobility [AUT2025, NOK2023, WEF2025, ERI2025, TRI2025], offering the real-time, high-bandwidth, and highly reliable communication required to link vehicles, infrastructure, and cloud environments. Its low-latency characteristics are foundational to advanced Vehicle-to-Anything (V2X) use cases and applications; this refers to Vehicle-to-Vehicle (V2V), Vehicle-to-Infrastructure (V2I), and Vehicle-to-Network (V2N) communications [NOK2023, TRI2025], communications. As such, 5G is positioned as a critical component in the broader context of Cooperative, Connected, and Automated Mobility (CCAM).

Through integration, 5G supports a range of safety-critical functionalities, including collision avoidance, blind-spot alerts, and emergency braking, all of which contribute directly to improving road safety and optimising traffic flow. In parallel, its compatibility with edge computing architectures brings compute resources closer to where data is generated, allowing for latency-sensitive processing and decision-making at the network edge—an essential requirement for reliable operation in highly dynamic and time-constrained automotive environments. Together, these capabilities place 5G at the core of next-generation autonomous and intelligent mobility systems.

But beyond this pivotal role in safety and automation, 5G fundamentally reshapes the *in-vehicle experience* by enabling high-speed, low-latency connectivity that supports infotainment systems, real-time navigation, and seamless over-the-air updates. With enhanced bandwidth and network reliability, vehicles are now capable of delivering uninterrupted high-definition media, interactive digital services, and cloud-based functionalities—effectively becoming fully connected platforms. In parallel, 5G is being actively utilised in the automotive manufacturing domain through the deployment of private network infrastructures that support real-time system monitoring and predictive maintenance. This shift not only enhances operational efficiency, but also *reduces downtime*.

Taken together, the developments arrayed above underscore the central role of 5G in advancing connected mobility, positioning 5G as a cornerstone for digital evolution - and ultimately the *transformation* - of the automotive sector.

1.4 Network Application and related cornerstone terminologies

Below, we introduce the *fundamental* software and platform definitions—most prominently the nApp (Section 1.4.1)—before returning to expand on these concepts in greater detail Section 0.

1.4.1 Network Applications (nApps) - The 5G-IANA approach

1.4.1.1 Core concept

A Network Application (nApp) in the context of 5G-IANA [ZEN2022] is defined as a **modular, reusable, and deployable software component** that can be orchestrated across 5G infrastructure to deliver targeted, network-enabled functions or services. Designed to support both standalone and composite service architectures, an nApp encapsulates specific logic and functionality that can be rapidly integrated into vertical applications.

On the 5G-IANA platform, nApps are deployed and managed through an advanced orchestration layer that supports full lifecycle management—from onboarding and configuration to runtime execution and scaling. Each nApp can be composed of one or more AFs and NFs, where AFs implement application logic and NFs handle network-specific tasks such as communication, data routing, or mobility management. These functions are deployable using multiple virtualisation formats: VM-based (VAFs/VNFs), container-based (CAFs/CNFs), or statically provisioned (PAFs/PNFs), allowing flexibility in deployment models depending on the target environment.

1.4.1.2 nApp Deployment Formats

In the 5G-IANA platform, nApps are designed for flexible deployment across diverse virtualisation environments. They can be instantiated in various formats, each tailored to specific infrastructure requirements:

- **Virtual Application Functions (VAFs)** and **VNFs**: These are deployed within virtual machines (VMs), providing isolation and compatibility with traditional virtualised infrastructures.
- **Cloud-native Application Functions (CAFs)** and **Cloud-native Network Functions (CNFs)**: These are containerised components designed for deployment

within container orchestration platforms, such as Kubernetes, enabling microservices-based architectures and dynamic scaling.

- **Physical Application Functions (PAFs)** and **Physical Network Functions (PNFs)**: These are dedicated hardware-based components that are statically deployed, offering high performance for specific use cases where virtualisation may not be optimal.

This multi-faceted deployment strategy ensures that nApps can be effectively orchestrated across the 5G-IANA infrastructure, accommodating a wide range of application scenarios and performance requirements.

1.4.1.3 nApp Packaging and Reusability

To support modularity, reusability, and composability, each nApp within the 5G-IANA framework is delivered as a self-contained package enriched with service-level metadata, interface definitions, and operational descriptors. This includes details such as the required 5G slice profile and associated compute resource specifications, enabling efficient integration into the wider orchestration environment. The packaging approach is designed to facilitate rapid prototyping, cross-domain portability (across cloud, edge, and far-edge tiers), and elastic scalability, thus reducing time-to-market for third-party developers and Vertical service providers. Additionally, each nApp Package contains detailed documentation, including predefined test cases, relevant KPIs, and monitoring metrics, supporting streamlined validation and seamless functional integration into target deployment scenarios.

1.4.2 Application Orchestration

The application-level orchestration process is responsible for managing the deployment and lifecycle of NetApps, with native support for elasticity and policy compliance. This orchestration approach decouples the application layer from underlying network orchestration, maintaining compatibility with slice management solutions while enabling slice-aware application deployments. It aligns with modern distributed vertical applications that leverage edge and far-edge resources, built from independently scalable cloud-native components. Each component is orchestrated as a VM, container, or unikernel, preserving compatibility with industry standards while enabling telco-specific features such as resource-constrained slicing, application profiling, and policy enforcement. The AO interfaces with the front-end GUI or directly with the NetApp repository to onboard application components. On the southbound interface, it

communicates with the slice management layer to translate application requirements into operational parameters that fulfil the functional demands of the NetApps.

1.4.3 Virtualised Infrastructure Domain

The virtualised infrastructure domain serves as the execution environment for Vertical Services and NetApps, spanning across cloud, edge, and far-edge tiers. In the Automotive context, the far-edge domain includes OBUs installed on vehicles and RSUs deployed roadside, forming part of the traditional C-ITS framework. OBUs connect to the edge cloud via 5G, while RSUs connect either through 5G or wired links when available. Both OBUs and RSUs integrate compute, storage, and networking resources locally. A critical component of the far-edge domain is the localised MANO framework—termed "on-vehicle MANO" in this context—to distinguish it from the MANO framework orchestrating central and edge cloud resources in AOEP.

1.4.4 Slice Management and Multi-domain Orchestration

Slice management within 5G-IANA is driven by high-level service intents defined at the Development (NOD) layer during Vertical Service design. The process includes translating these requirements into an appropriate 5G Network Slice profile (NEST) to select the optimal NSI and orchestrate the provisioning of compute resources across virtualised domains. Multi-domain orchestration handles resource allocation for NetApps and their constituent AFs/NFs, respecting the compute constraints (e.g., vCPU, RAM, storage) and additional domain-specific parameters like geographical scope or platform-specific capabilities.

1.4.5 Distributed Machine Learning Framework

The DML framework is implemented via a dedicated orchestration layer that selects and provisions compute resources required by DML-driven Vertical Services. The core component, the DML orchestrator, is tightly coupled with the slice and resource management layers to inject DML-specific constraints into the provisioning pipeline. Its responsibilities include monitoring DML resource availability at edge/far-edge levels, orchestrating DML application deployment, and managing the lifecycle and delivery of trained models across the infrastructure.

1.4.6 Application Monitoring

Application monitoring enables the platform to collect and process runtime operational data from deployed Vertical Services. Metrics such as CPU usage, memory footprint, disk I/O, container status, and socket activity are captured according to the monitoring policies defined by the AO. This mechanism is typically coupled with an internal analytics engine that supports rule-based inferences—both forward and backward chaining—enabling proactive and reactive management actions based on service behaviour and performance trends.

1.5 Overview of 5G-IANA Testbed Infrastructure and Deployment Context

To support realistic experimentation and deployment of automotive-related network applications, 5G-IANA offers two distinct 5G Standalone (SA) network infrastructures used for deployment and testing[1]. The first testbed is located in Ulm, Germany, and is operated by Nokia, while the second is based in Ljubljana, Slovenije, managed by Telekom Slovenije. These environments serve as foundational infrastructure for executing use cases (UCs) and onboarding third-party applications under real-world conditions.

FIGURE 3. TESTBED SITES[1].

The two sites differ significantly in coverage characteristics and deployment configurations. The Ulm testbed offers granular coverage, with a range of low to high signal quality zones—making it ideal for UCs requiring performance and Quality of Service (pQoS) sensitivity, such as those involving mobility or handover testing. It is also equipped with RSUs and On-Board Units (OBUs), which are available for use within the platform's supported UCs. In contrast, the Ljubljana site provides binary coverage—either fully available or absent— hence being ideal for testing abrupt network changes..

[1] A detailed description of the testbeds is provided in Section 2.3.1.1.

Additionally, participants operating at the Telekom Slovenije site must supply their own RSUs or OBUs, making this site more suitable for UCs or third-party experiments with self-contained hardware configurations.

1.6 Project Use Case Overview and Deployment Context

The 5G-IANA project features seven distinct Use Cases (UCs), designed to demonstrate the application of nApps across real-world automotive and mobility scenarios. Six of the UCs are aligned with the Cooperative, Connected, and Automated Mobility (CCAM) paradigm, while UC7 stands apart as a cross-border infrastructure coordination scenario, not strictly classified under CCAM. These use cases collectively reflect the diversity and applicability of the 5G-IANA platform across safety, infrastructure, infotainment, and emergency contexts.

FIGURE 4. OUTLINE OF PROJECT USE CASES IN 5G-IANA.

Each UC has been mapped to one or both of the project's 5G SA testbeds—Nokia in Ulm, Germany, and Telekom Slovenije in Ljubljana, Slovenia—as denoted by the respective logos. While all UCs are technically deployable in either testbed, demonstration activities so far have leveraged the infrastructure most suited to the technical requirements of each use case. For example, UCs such as UC1 (Remote Driving) and UC6 (Network Status Monitoring) benefit from Ulm's heterogeneous signal conditions to validate pQoS mitigation mechanisms, whereas UCs that require consistent, stable connectivity have leveraged Telekom Slovenije's binary coverage model.

UC7 is notable for its cross-testbed nature, involving coordination between both sites and highlighting the platform's capacity for distributed, multi-domain orchestration.

This cross-border demonstration also illustrates the ability of the AOEP to handle dynamic resource provisioning and complex scenario management.

Below is a summary of each of the seven project UCs as shown in the Figure 4. This use case suite serves as a validation backbone for the 5G-IANA platform and toolkit, demonstrating its effectiveness in managing diverse service needs, deployment scenarios, and orchestration strategies. For the sake of clarity, we note that Third-Party experimenters bring their own Use Cases and testing scenarios related to the automotive sector.

- **UC1 – Remote Driving *(deployed in Nokia)*:** Tests the reliability and latency of remote driving commands in areas with variable signal quality.
- **UC2 – Manoeuvres Coordination for Autonomous Driving *(deployed in NOKIA)*:** Focuses on V2X coordination for traffic and intersection management.
- **UC3 – Virtual Bus Tour *(deployed in Nokia)*:** Demonstrates infotainment delivery and immersive user experiences during group transportation.
- **UC4 – Content Delivery for Vehicular Networks *(deployed in Telekom Slovenije)*:** Validates the high-bandwidth delivery of digital media content in moving vehicles.
- **UC5 – High-Risk Driving Hotspot Detection *(deployed in Telekom Slovenije)*:** Leverages ML to detect and analyse accident-prone areas in urban settings.
- **UC6 – Network Status Monitoring *(deployed in Nokia)*:** Tests proactive monitoring and fault detection for vehicle connectivity in low-signal environments.
- **UC7 – Situational Awareness in Cross-Border Tunnel Incidents *(deployed across both testbeds)*:** Demonstrates coordinated emergency response using cross-border network orchestration and data fusion.

2 The 5G IANA Platform

Section 2 provides an overview of the 5G IANA platform, starting with its high-level architecture and the fundamental layers that enable its functionality, and then moving to the key components, detailing the nApp Orchestration layer, the Slice Manager and Resource Orchestrator, and the Virtualised Infrastructure Segments. The section further explores the practical deployment and integration of the platform within real-world experimental infrastructures, including insights into the adopted integration processes and CI/CD mechanisms that facilitate its modular and adaptable nature.

2.1 Architecture Overview

The 5G-IANA platform is specifically conceived for simplifying and automating the management of nApps onto programmable infrastructures, including 5G systems. The adoption of a modular platform design aims to largely hide the complexity of programmable infrastructure and the 5G environment from service developers and providers. This approach intends to make the development, deployment, and operation of 5G-ready applications (nApps) similar to the well-known processes applied to cloud-native applications in cloud computing environments.

This modular approach is highlighted in the 5G-IANA architecture, as depicted in Figure 5, which showcases a two-layered Orchestration stack. These layers are:

A) The nApp Application Orchestration and Development (NOD) layer.
B) The Slice Management & Resource Orchestration (SM-RO) layer

The orchestrator manages the virtualized infrastructure segments (lower part of the figure), which expand from the edge (including vehicle and roadside elements) to the core and central cloud. The cross-layer supported functionalities (right part of the figure) include the Distributed AI/ML framework and the combined Monitoring & Analytics and Distributed Data Collection.

The separation of the 5G-IANA orchestration platform functionalities between the NOD and the SM-RO layers serves the need to operate across two different functional administrative domains: a) The Application Domain (represented in yellow in the document's figures) and b) The Infrastructure Domain (represented in blue in the document's figures). This distinction of layers targets the different "work-burden" that has to be achieved and managed. Consequently, the orchestration tools target two distinct lifecycles: a) that of the application management, and b) that of the programmable infrastructure resources and network services management. In this sense, the 5G-IANA Platform is comprised of a set of orchestration tools, with each set

devoted to its specific administrative domain (applicative or network). Each administrative domain is operated by specific stakeholders: for the Application Domain, the stakeholders are nApp developers from various automotive vertical industries, while for the Infrastructure Domain, the stakeholders are programmable infrastructure owners, including 5G network operators. Since the on-board units (OBU) and roadside units (RSU) are part of the programmable resources, their specific management is undertaken by the SM-RO. Finally, the cross-layer functionalities are important functionalities for the runtime management and decision-making processes supported by the 5G-IANA Platform. These are: a) The Data Collection functionality, which collects application and device-oriented data for the deployed services and the attached infrastructures respectively, b) The Monitoring and analytics functionality, which processes the collected data and expose them to the decision making mechanisms of the orchestrator, and c) The Distributed AI/ML feature, which supports multi-criteria client selection for Federated Learning workloads.

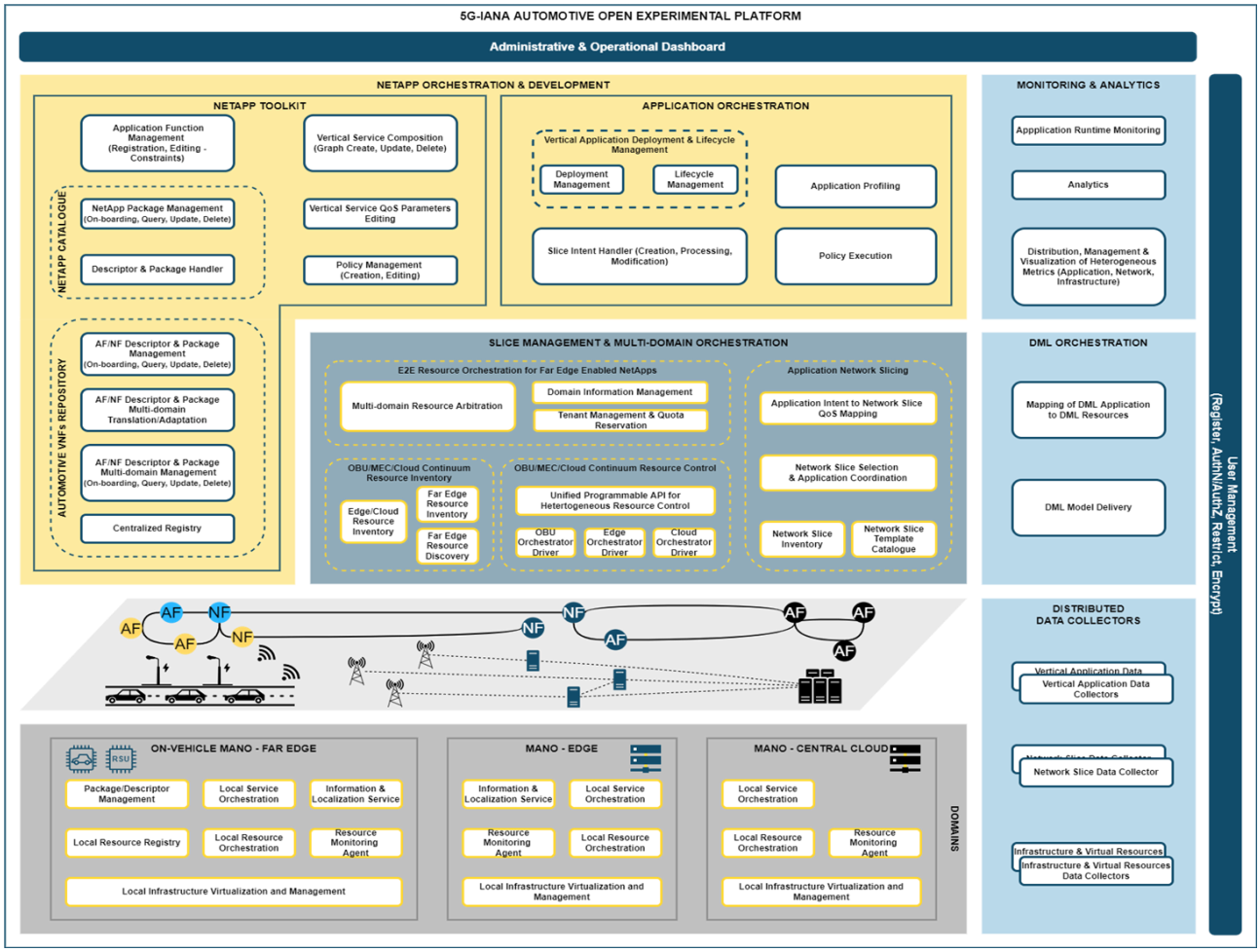


FIGURE 5. 5G-IANA ORCHESTRATION LAYERS ABSTRACTION.

2.2 Key platform components and interfaces

2.2.1 nApp Orchestration and Development (NOD)

The nApp Orchestration and Development (NOD) layer is a fundamental part of the 5G-IANA platform, designed to undertake the deployment and real-time management of Network Applications (nApps). Its primary objective is to interface with the end-user, typically an automotive industry application developer, for managing deployable applications and their features. A key design principle of the NOD layer is to decouple application layer management procedures from network layer management. This allows it to interface with underlying slice creation and management systems, ensuring compatibility with various network orchestration solutions and their respective slice management subsystems.

The NOD layer consists of several software modules responsible for registering a nApp and its components, along with cloud-related and slice-related metadata. It also handles the authoring of deployment and runtime policies, negotiates slices, and manages the operational state of the nApp within a materialized slice. The main components in the NOD layer are the following:

- **nApp Toolkit**: The nApp Toolkit offers an abstraction for registering and onboarding nApp components, allowing users to compose a nApp without needing to interfere with infrastructure and network configuration details. For example, an application developer can tag a container for OBU deployment without the need to know any technology-specific information like metadata that are required for a deployment in Kubernetes or in OpenStack or through docker-compose or Helm charts. It assists vertical service providers in wrapping cloud-native nApps in a proper format, so as to be publishable in the Container Registry. Each nApp consists of multiple containers that are chained in the form of a service graph. The nApp Toolkit provides design-time validation in the granularity of component model and guarantees that required cloud-native properties are maintained through metadata regarding minimum infrastructural requirements, configuration parameters per atomic component, mutable configuration parameters during runtime, exposed and required interfaces – dependencies for a component to be functional, and deployment preferences. The nApp Toolkit provides input to the Deployment Management module.
- **Deployment Management**: The Deployment Manager is the software module that carries out the task to realise the initial actual deployment of a nApp Graph on top of programmable resources by undertaking the task of communication

with the underlaying registered infrastructure. To do so, the Deployment Manager has to be fully aware of the available resources, their state and directive-indications (declared through the 5G-IANA specific metadata) for the actual deployment. The knowledge of the resource-state derives from the interaction of the Deployment Manager with the Resource Inventory (see Section 4.1.2 of the referenced document). It is important to state that the ability of this module to perform deployments in various virtualization environments derives from a built-in capability to interact with various virtualization endpoints. Hence, the Deployment Manager provides an abstract interface for basic management virtualization operations (e.g., select/create tenant, create/modify namespaces, etc.) by incorporating some of the industry-dominant virtualization technologies (OpenStack, Kubernetes) and their respective APIs.

- **Lifecycle Management**: The Lifecycle Manager is a closed control loop that assesses continuously the existing resources, the deployment requests and the reconfiguration requests of the already deployed nApps. This software module encapsulates the orchestration business logic and can be seen as a centralized logical entity that affects the "scheduling" of nApp components that are distributed at various virtualizations levels i.e., at the cloud-edge-OBU level. In a nutshell, the Lifecycle Manager is responsible to maintain the operational state of the nApp. The state is mapped to an amount of resources that are allocated per component of the nApp, the slice parameters that have been selected, the geographical constraints of the deployment, etc. Any change on these aspects has to be handled by the control plane that the Lifecycle Manager oversees.
- Slice Intent Handler: The purpose of the Slice Intent Handler module is to handle the complete lifecycle of the Network Slice creation process. The conceptual flow of this module includes a) the slice request phase, b) the slice instantiation, c) the slice operation and d) the slice deprovision. During the instantiation phase, the creation and verification of the necessary virtualized environment is performed. This process programmatically allocates and assigns resources to virtualized environments to host the lifecycle of the nApp. These environments will be adopted to satisfy the nApp requirements coming from the Service Composition module (which belongs to the nApp Toolkit). The translation of the requirements to a proper slice configuration plan is a primary objective of the Slice Intent Handler (see a formalized slice intent expressing the user-defined requirements in json in Annex B of the referenced document). Upon completion of the instantiation phase, all resources shared/dedicated to the slice should have been created and configured.
- **Policy Execution**: This module works in conjunction with the Lifecycle Manager, which is responsible for maintaining the operational state of all nApp components. Policies are instructions regarding how the overall nApp should behave prior to the deployment and during runtime. In 5G-IANA the Policies are

covering only the runtime phase (not the prior to deployment phase) and are expressed as a set of rules and actions. Each rule is defined as Service Level Agreement (SLA) expressions (formally SLOs) that conceptualize the desired state at the nApp graph-level or at the nApp component-level. In a nutshell, policies are realized through a rule-based framework that attempts to derive execution instructions based on the current set of data and the active rules; rules associated with the deployed service graphs at each point of time. The SLAs can be authored through the Policy Editor (which belongs to the functionalities provided by nApp Toolkit) and the data is fed to the Policy Execution through the Monitoring module that is responsible to collect data based on a set of active monitoring probes. The evaluation of the rules according to incoming monitoring data takes places in the Policy Execution as well as the triggering of specific actions that the Lifecycle Manager will need to realize.

- **Application Profiling**: The Application Profiling module is designed for supporting various profiling aspects at nApp component and nApp graph level. The scope of this software entity is the examination of the behaviour of the application characteristics (e.g., resources' usage efficiency, scaling or potential scaling actions and profiles, identification of bottlenecks, identification of capacity limits and breaking points upon stressing the software processes, reliability aspects taking into account time series data with identified problems and failures) under various conditions. Application Profiling consumes monitoring metrics from the Monitoring & Analytics component with specific criteria e.g., fetching nApp ID or nApp component monitoring data, historical data timeslot and more.

2.2.2 Slice manager and resource orchestrator

The Slice Manager and Resource Orchestration layer implements the mechanisms to manage and orchestrate the available Edge and Far-Edge resources, along with the Network Slice Instances (NSIs) that can be used to support the system. This layer is composed of three main functionalities, each implemented by its subcomponents, designed to handle service intent requests, manage resource inventories, and map service requirements to appropriate network slices.

E2E Resource Orchestration for Far Edge enabled nApps (RO): The E2E Resource Orchestration for Far Edge enabled nApps is responsible for resource allocation across the entire continuum from Far Edge to Cloud. It processes Service Intent requests received from the Slice Intent Handler (part of the nApp Orchestration and Development Layer) and manages these requests by allocating the most suitable namespaces and quotas to Far Edge devices. This allocation also considers resource availabilities provided by the OBU/Edge/Cloud Continuum Resource Inventory (CCRI). The RO works in

conjunction with the Slice Manager to provide the nApp Orchestration & Development layer with a comprehensive description of the computing and networking resources required to instantiate the service on the target segment and host. Internally, the E2E Resource Orchestration for Far Edge enabled nApps comprises two software submodules:

- **Multi-segment Resource Arbitration**: This function coordinates the provisioning of computing resources. It translates information received in the Slice Intent from the nApp Orchestration & Development Layer into a set of computing constraints. By combining these computing requirements with the set of available Edge and Far Edge resources obtained from the CCRI, the Multi-segment Resource Arbitration module constructs a "Quota Reservation Request" which is then passed to the Tenant Management & Quota Reservation component to apply the operation to the actual devices.
- **Tenant Management & Quota Reservation**: This submodule is responsible for implementing the client interface to the Management and Orchestration (MANO) layer. It applies the actions formulated by the Multi-segment Resource Arbitration, such as creating tenants and reserving quotas on the physical infrastructure. This support for multitenancy allows multiple automotive vendors to deploy multi-vendor services on the same Edge and Far Edge resources.

OBU/Edge/Cloud Continuum Resource Inventory (CCRI): The OBU/Edge/Cloud Continuum Resource Inventory, also referred to as CCRI or RI, implements a continuously updated inventory of available Edge and Far Edge computing resources. This inventory is crucial for targeting resources when a service provisioning action is addressed by the Slice Intent Handler. It exposes an interface to retrieve the current list of available Edge and Far Edge devices, along with their static capabilities such as computing resources and the presence of GPUs. The CCRI can also leverage external platforms, like monitoring platforms or information services, to query its internal device registry based on dynamic filtering criteria, such as location-based queries. The CCRI is based on three internal software modules:

- **Far Edge/Edge/Cloud Resource Inventory**: These two components (often treated as a cohesive unit) implement a continuously updated registry of the available resources from the Far Edge to the Cloud, including their static capabilities. External components can query this inventory to get a complete view of the status of 5G-IANA devices ready to host nApps or vertical services.
- **Far Edge Resource Information Service**: This service works in conjunction with the Far Edge/Edge/Cloud Resource Inventory to enrich device information with dynamic data, like the location of the device or its remaining computing

resources. It relies on external monitoring platforms (such as a Prometheus-based monitoring platform as described in Section 3.1.6 of the document for Application Profiling's data source) to act as a client for the Monitoring Service. This allows it to correlate availability information stored in the Resource Inventory with dynamic information, exposing a filtering-based interface for Far Edge Resources, which is particularly useful for components like the Distributed Machine Learning Orchestrator (DMLO) to select appropriate OBUs based on refined criteria.

Slice Management: The Slice Management component provides the functionalities to map a Service Intent request, received from the Slice Intent Handler, with the most suitable Network Slice Template (NEST) according to a set of translation rules. It relies on a catalogue of pre-defined 5G Network Slice Instances that are chosen based on the requested Quality of Service (QoS) indicators. This component implements the functionalities for storing available Network Slice Instances (NSIs) and validating Service Provider requests for Vertical Services in terms of compatible 5G service profiles. The Slice Management component is composed of four internal functions:

- **Application Intent to Network Slice QoS Mapping**: This function receives the Slice Intent from the Slice Intent Handler. Relying on a set of translation rules, it converts this information into recommendations for the most suitable Network Slice Instance available on the system. It particularly uses the Networking Constraints specified in the intent-based request to select the most appropriate NEST for instantiating the 5G Network Slice for the requested Vertical Application Slice. To achieve this, it interacts with the NEST catalogue of the 5G Network Slice Management Function module.
- **Network Slice Verification, Selection & Application Coordination**: This function is responsible for choosing the 5G Network Slice from the Network Slice Management & Inventory based on the recommendations received from the Application Slice Intent to Network Slice QoS Mapping component. Verification is performed against the networking requirements from the Slice Intent Management module; it checks if an appropriate Network Slice Template exists that can meet all specified requirements.
- **Network Slice Template Catalogue**: This component contains a static inventory of the Network Slice Templates available on the system.
- **Network Slice Management & Inventory**: This function implements a static database holding a set of 5G Network Slice Instances that can be selected for the instantiation of Network Services.

2.2.3 Virtualised infrastructure segments

The virtualised infrastructure segments in the 5G-IANA platform encompass the on-vehicle, edge, and cloud environments. The design and implementation ensure that these segments can support the orchestration of virtualized applications, particularly in the demanding far-edge (On-Board Units - OBUs, Roadside Units - RSUs) and edge/cloud domains.

On-vehicle MANO, Edge and Cloud MANO: The management and orchestration (MANO) for virtualized applications across all segments, including the far-edge (OBU, RSU) and the edge/cloud, have been standardized on Kubernetes. This decision aligns with the specified requirements for On-vehicle MANO (OVM) and Edge and Cloud MANO as outlined in the 5G-IANA deliverable D2.1. Kubernetes offers several crucial features such as scalability, resilience, and portability, which are highly beneficial in the 5G-IANA context. It allows applications to dynamically scale the number of working nodes (container replicas) based on runtime needs like CPU load, ensuring optimal performance and resource utilization. For far-edge devices, which are often hardware-constrained, a lightweight Kubernetes solution is essential. Kubernetes also provides resilience, enabling applications on far-edge devices to continue running even during network connection failures. Furthermore, its portability simplifies management by allowing a single MANO solution across different segments.

Within the Kubernetes ecosystem, the MicroK8s distribution by Canonical has been selected. MicroK8s is a certified Kubernetes distribution designed to be lightweight and easy to install, providing the main features of Kubernetes. These characteristics make it an optimal choice for 5G-IANA, where devices with limited hardware capabilities (e.g., ARM-based) are primary targets, as it avoids exhausting available computational resources. While other distributions like minikube and K3s were considered, MicroK8s was preferred due to minikube being more suited for testing/development and MicroK8s appearing more mature and complete than K3s for the project's needs. Deploying MicroK8s requires preliminary steps to configure the far-edge device to support containerized applications, including GPU support if available. The deployment and lifecycle management of services on OBUs/RSUs are handled by the nApp Orchestration & Development layer, specifically the Deployment Manager, which utilizes an abstract interface for MicroK8s APIs.

Information and Localization Service: This service is crucial for providing the 5G-IANA platform with information about far-edge device's hardware capabilities and its current geographical position, addressing requirement SFR-GVI-3 from D2.1. Information regarding hardware capabilities (e.g., total CPU, memory, GPU availability) is retrieved by

the 5G-IANA platform directly from the APIs exposed by MicroK8s on the far-edge device. The Resource Inventory interacts with the far-edge device using these interfaces. For providing the device's position, a custom service named the localization service has been developed. On OBUs, this service interacts with the vehicle's GNSS receiver (using the GPSd service) to retrieve the current position. For RSUs, the position is typically a static input parameter configured during installation. The localization service then provides this position information to a Prometheus instance located on the edge server. Subsequently, the Resource Inventory retrieves the position information of these far-edge devices by querying the Prometheus instance. The localization service on far-edge devices is deployed as a system Linux service and supports two modes for GPSd: a monitor mode using real GNSS data and a simulated mode using pre-recorded or manually crafted NMEA tracks. This simulated mode is particularly useful for third-party experimenters, offering RESTful APIs to manage simulations of vehicle movement, allowing nApp testing without actual vehicle movement.

Resource Monitoring Agent: The Resource Monitoring Agent is implemented to collect and provide information about the status of virtualized resources on the far-edge devices, fulfilling requirement SFR-GVI-3. This agent consists of two main software modules: the Hardware Monitoring Agent and the Data and Metrics Exporter. The Hardware Monitoring Agent is a software component that periodically checks the runtime utilization of far-edge resources, such as CPU utilization, memory usage, and storage used. Different versions of this agent can be deployed depending on the specific hardware of the far-edge device, particularly concerning the presence of a GPU, which requires specific commands for metric retrieval. This agent provides the collected information in a JSON formatted message to the Data and Metrics Exporter via a custom RESTful API.

The Data and Metrics Exporter is then responsible for processing these received metrics and providing them to a Prometheus instance running on the edge server. This module also exposes RESTful APIs for Application Functions (AFs) that wish to collect data through the 5G-IANA monitoring platform. The Exporter sends both the hardware metrics and any AF-specific data to the Prometheus instance on the edge server, which then forwards them to the central monitoring platform in the cloud. The Resource Monitoring Agent, therefore, interacts primarily with the Distributed Data Collection component of the 5G-IANA platform through the interfaces made available by Prometheus.

2.2.4 Cross Layer functionalities

The 5G-IANA platform incorporates cross-layer functionalities that span across the other architectural layers, providing essential support services. These include the Distributed AI/ML framework and the Monitoring & Analytics capabilities, along with Distributed Data Collection.

Distributed AI/ML framework mechanisms: The Distributed AI/ML (DML) orchestration layer is primarily designed to support multi-criteria client selection for Federated Learning (FL) workloads. This allows FL applications to dynamically adapt their set of training nodes (e.g., On-Board Units - OBUs) based on dynamically varying parameters. The goals are to decouple the acquisition of application-agnostic selection criteria from the application itself, simplifying FL application development, and to realize client selection decisions through lifecycle management and orchestration primitives. This means OBU resources are reserved and consumed only when necessary, in contrast to traditional FL clients that run constantly. The core component here is the Distributed Machine Layer Orchestrator (DMLO), which provides a DML-service specific user interface and a backend for interaction with other platform components and Application Functions (AFs). Involved AFs include the AggrNode AF (FL aggregation node), Training Node AF (FL client), and Data Collection AF. The selection of OBUs for FL training is based on static criteria (e.g., access rights, hardware capabilities like GPU, sensing capabilities), dynamic ML-agnostic criteria (e.g., CPU/GPU utilization, RAM, storage, energy availability, location information managed by Monitoring & Analytics and the Compute Continuum Resource Inventory - CCRI), and dynamic ML-specific criteria (data availability/quality at the node, and ML performance KPIs like convergence rate).

The client selection process involves several steps. Initially, the DMLO collects ML-specific configurations (model, training scheme) and deploys the AggrNode and Data Collection AFs using the platform's baseline functionality. The Data Collection AFs are instantiated on vehicles matching static criteria. The DMLO then collects dynamic ML-specific data criteria from these Data Collection AFs and queries the CCRI for nodes matching dynamic ML-agnostic criteria. By combining this information, the DMLO identifies a filtered set of suitable nodes. For synchronous FL, the AggrNode retrieves this filtered list from the DMLO, applies any further ML-specific KPI criteria, makes the final selection, and communicates this to the Policy Execution module (via DMLO) to instantiate Training Node AFs on the selected OBUs. The FL training round then commences. Finally, the DMLO visualizes status reports, combining ML process statistics from the AggrNode and resource utilization from Monitoring & Analytics.

Application Functions (AFs) and FLOWER integration: The FLOWER framework is integral to handling FL task-specific operations. The AggrNode AF implements the FLOWER server functionality, and the Training Node AF implements the FLOWER client. The FLOWER server consists of a FLOWER strategy (customizing the ML process like client selection methods, aggregation methods), a FLOWER client manager (monitoring clients), and an RPC server for communication (gRPC over HTTP). These are configured during the initial FL task setup based on DMLO input. In each training round, the FLOWER server queries the DMLO for available nodes, performs client selection based on its configured strategy (using ML-specific criteria), and then informs the DMLO so that the corresponding Training Node AFs (FLOWER clients) can be activated via the Policy Manager. Once clients are active, the FLOWER server manages the training round. The FLOWER client (Edge Client) includes the training pipeline, access to training data (acquired in real-time from the Data Collection AF on each OBU), and an RPC client for server communication. FLOWER acts as a facilitator, providing wrappers to control functionalities and communication, allowing flexible development and dockerization. It focuses on ML-specific tasks, while more generic tasks like client connectivity and data availability monitoring are handled by the DMLO.

Monitoring & Analytics, Distributed Data Collection: This cross-layer functionality aims to collect and make readily available monitoring data regarding resource utilization from the underlying virtualized infrastructure (including on-vehicle MANO for compute, memory, network, as detailed in Section 5.3 of the document) and the behavior of deployed nApps through tailored application-level metrics (e.g., throughput, active users). The Monitoring & Analytics component enables core services like Orchestration, Policy Execution, and Application Profiling. It is responsible for collecting data based on active monitoring probes and supporting data management operations (e.g., calculating average values over specific time windows). It also provides access to both historic and real-time monitoring data, which can be centrally accessed by tenants through the 5G-IANA Dashboard and programmatically via its API (e.g., Prometheus API).

Design Details of Monitoring & Analytics: The metrics collection model follows a distributed approach, fetching and processing monitoring data and requests for specific deployed nApps over an overlay mesh network that interconnects the respective nodes running the nApp components. The monitoring engine leverages an industry-grade telemetry system, Prometheus, to store and manage monitoring metrics. To populate the Prometheus database, Netdata is used as a monitoring agent on the nodes, providing real-time streams of system and application metrics that are collected by Prometheus. The developments in this area concern setting up different monitoring instances (cloud, edge) to collect both on-vehicle MANO metrics and application-specific

metrics. As described in Section 5.1.3, far-edge devices interact with a Prometheus instance at the edge server to report their metrics.

2.3 Deployment and Integration

The 5G-IANA platform, encompassing the comprehensive suite of orchestration and management modules previously detailed, has been successfully deployed and validated within real-world experimentation infrastructures. These crucial validation and testing activities are conducted across two primary sites: the Nokia testbed located in Ulm, Germany, and the Telekom Slovenije testbed in Ljubljana, Slovenije. These facilities provide the necessary 5G Standalone (SA) network capabilities and diverse hardware resources, enabling thorough testing of the platform's components and the execution of various automotive use cases.

2.3.1 The 5G-IANA deployment infrastructures

2.3.1.1 Nokia Testbed (Ulm, Germany)

Infrastructure Details:

The Nokia testbed, located in Ulm, Germany, features a 3GPP Release 15 compliant 5G Standalone (SA) architecture. The radio access network operates primarily on the n38 band (2.6 GHz, Time Division Duplex - TDD) with a 40 MHz bandwidth and a 30/70 uplink/downlink split. Initially, outdoor coverage was provided by four antenna sites, which was later augmented with an indoor antenna at the Nokia office (Lise-Meitner-Str. 7/1, 89081 Ulm) to streamline testing processes; this indoor setup mirrors the radio frequency and network slice configurations of the outdoor sites. The core network is also 5G SA, with control plane functions managed remotely (from Finland) and User Plane Function (UPF) connectivity to the gNodeB (gNB) established via a direct link. The Public Land Mobile Network (PLMN) is identified by MCC 262 and MNC 731. Network slicing is a key feature, with pre-configured V2X (Slice/Service Type - SST 4) and eMBB (SST 1) slices, both demonstrating average Round-Trip Times (RTT) to the EDGE Server of less than 20ms under low load conditions.

Hosting Environment Characteristics:

The 5G-IANA platform components, along with EDGE and Far-EDGE server functionalities, are hosted within the Nokia testbed. The hosting environment relies on

Linux Kernel-based Virtual Machines (KVMs). Each virtual machine is provisioned with Ubuntu Server 22.04 LTS and runs MicroK8s for container orchestration. The VMs designated for various platform components (such as the nApp Toolkit, Application Orchestrator, Extreme-Edge Orchestrator, Monitor, and Slice Manager) are typically configured with 4 vCPUs, 8GB of RAM, and substantial storage (e.g., 50GB for primary disk and 100GB for secondary disk), each equipped with dual network interfaces for connectivity within the testbed's network segments. Secure remote access for partners and third parties is facilitated through VPN connections.

2.3.1.2 Telekom Slovenije Testbed (Ljubljana, Slovenije)

Infrastructure Details:

The Telekom Slovenije (TS) testbed is situated in Ljubljana, Slovenije, with facilities at two locations: the company's main research lab at Cigaletova 15 and the INO.LAB at Vojkova 78, which includes options for outdoor and indoor garage testing. This testbed provides a dedicated 5G infrastructure, separate from the commercial network, designed for 5G-IANA use cases and based on 3GPP Release 16. The 5G radio environment primarily utilizes the n78 band (3.5 GHz, TDD, 100MHz bandwidth), delivering an average RTT of approximately 15ms between User Equipment (UE) and the EDGE server. An LTE Band n3 is also available. The 5G SA Core is an HPE ProLiant DL360 Gen10 server. While full commercial-grade network slicing is not available, the testbed supports experimental slicing using standard S-NSSAI (Single Network Slice Selection Assistance Information) with SST and Slice Differentiator (SD) parameters. There are plans to provide eMBB, V2X, and URLLC slices, with slicing functionalities for both the 5G SA Core and Radio Access provided by Kontron.

Hosting Environment Characteristics:

The cloud and virtualization environment at the TS testbed is built upon a Cisco UCS (Unified Computing System) with 8 blade servers running a VMware cluster. This infrastructure hosts the containerized applications for control, management, and orchestration (CMO) components, supporting MicroK8s/Kubernetes and MANO functionalities, as well as the 5G-IANA nApp Toolkit if required. Similar to the Nokia setup, the 5G-IANA platform components are deployed on KVMs running Ubuntu Server 22.04 LTS and Kubernetes. The allocated resources for these VMs vary per component; for instance, the Application Orchestrator (APPORC) VM is configured with 4 vCPUs, 8GB of RAM, and 60GB of disk space. A crucial aspect of the TS testbed's integration within the broader 5G-IANA project is the site-to-site VPN connection established with the Nokia

testbed. This connection supports redundancy for platform components and is vital for testing cross-border functionalities, particularly for UC7

2.3.2 Platform integration and adaptations

The integration of the 5G-IANA platform across the distinct experimental infrastructures at Nokia (Germany) and Telekom Slovenije (Slovenia) underscores its robust and adaptable design. The platform components, including the nApp Toolkit (encompassing Vertical Application Composition and Customization), nApp Orchestration and Development (Application Orchestrator), Slice Management & Resource Orchestration, Monitoring services, and the Distributed ML Orchestrator (DMLO), are systematically deployed on dedicated servers within each testbed.

A key aspect of this integration is the consistent deployment methodology employed. In both the Nokia and Telekom Slovenije testbeds, the 5G-IANA platform components are instantiated on adequately dimensioned Kernel-based Virtual Machines (KVMs). Each VM is typically provisioned with Ubuntu Server 22.04 LTS and utilizes Kubernetes (often the lightweight MicroK8s distribution) for container orchestration. This containerized approach, where platform functionalities are realized as microservices, inherently promotes modularity.

The networking setup within these testbeds is crucial for the platform's operation and is exemplified by configurations such as the one depicted in Figure 6. This figure illustrates the various subnetworks established in the Nokia testbed, including separate networks for administrative data, application data (for eMBB and V2X slices), and internet connectivity. The 5G-IANA platform components, EDGE server, and Far-EDGE UEs are shown interconnected through these distinct network segments, highlighting how administrative traffic for platform management is separated from the user plane traffic of the deployed nApps. This structured network integration, with clearly defined interfaces and traffic flows, is replicated with similar principles in the Telekom Slovenije testbed, ensuring consistent operational logic. Secure VPN tunnels are established to allow project partners and third-party experimenters remote access to the platform components and EDGE servers hosted in both testbeds, facilitating ongoing development, integration, and nApp experimentation.

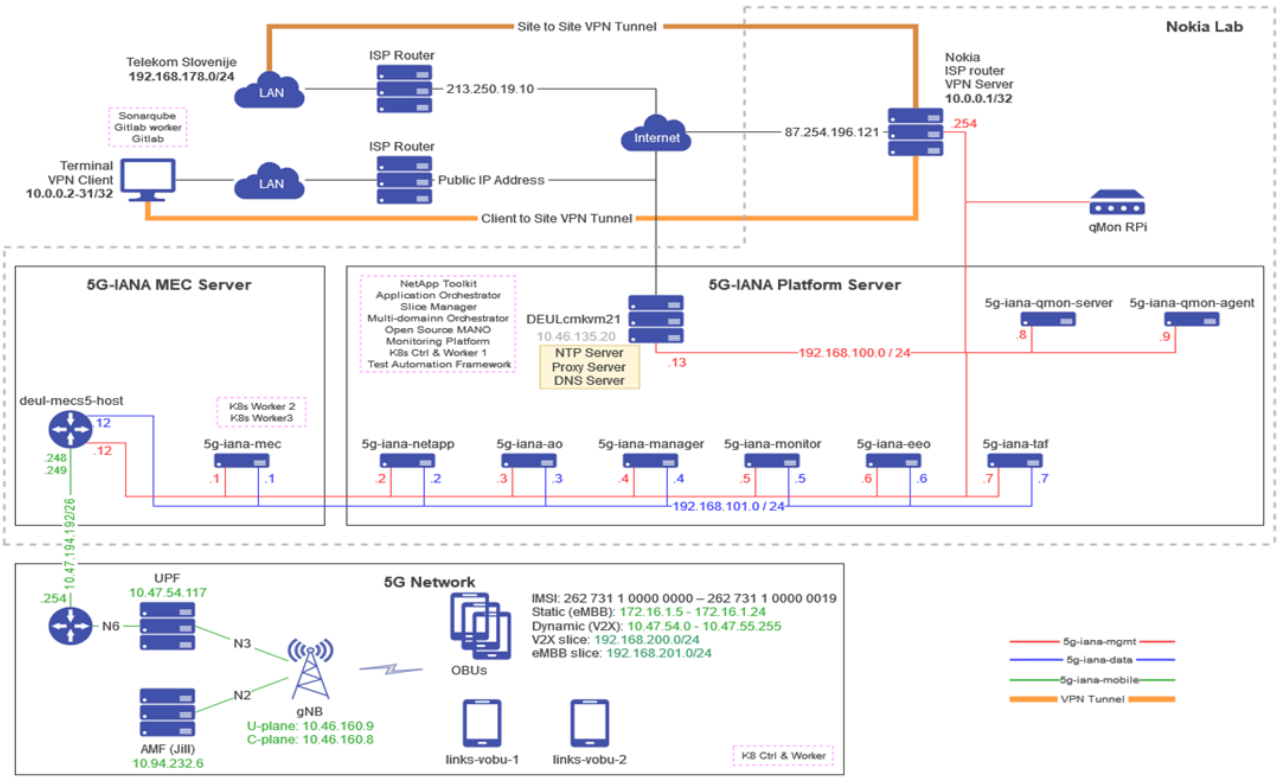


FIGURE 6. 5G-IANA PROJECT NETWORKING OVERVIEW.

This modular and container-based architecture, coupled with well-defined networking, is central to the platform's ease of deployment across diverse infrastructures. The adopted Continuous Integration and Continuous Deployment (CI/CD) functionalities and version control of generated artifacts and platform modules (see next subsection) allows for standardized deployment procedures and updates. The ability to deploy the same platform components using a common approach on the distinct underlying hardware and network configurations of the Nokia and TS testbeds demonstrates the platform's flexibility and its design for straightforward deployment over varied infrastructure environments.

2.3.3 Adopted integration processes for end users

The development, integration, and deployment of the 5G-IANA platform and its associated nApps are governed by a robust methodology emphasizing modularity, flexibility, and automation. This is supported by a structured approach to version control, software repositories, and a Continuous Integration/Continuous Deployment (CI/CD) pipeline.

2.3.3.1 Repositories, Image Sourcing, and Version Control

The 5G-IANA platform's modular design inherently supports flexibility in how application components are managed and sourced. While a specific GitLab server is utilized for the core development and experimentation purposes within the 5G-IANA project, the architecture allows for connectivity with potentially any server repository. This means that nApp images are not strictly tied to a single project-internal registry; developers can select and specify the source of application images (e.g., from external Docker registries or repositories from other projects) during the nApp definition and deployment phase.

For the 5G-IANA project's internal development and experimentation, a dedicated GitLab server is employed. This server is centrally located and managed, functioning as a cloud-like resource that is accessible by both the Nokia and Telekom Slovenije testbeds, rather than being an integral part of each specific testbed domain. This central GitLab instance provides version control for all software artifacts of the 5G-IANA platform components and also hosts the project-specific nApp components and templates. It facilitates collaboration among multiple developers and partner organizations by providing a unified platform for managing and tracking changes to the codebase. The 5G-IANA repository holds both platform software and nApp elements and is connected with virtual machines within the core cloud infrastructure of the testbeds.

2.3.3.2 CI/CD Pipeline

The GitLab server functionality extends to support CI/CD pipelines, which are integral to the platform's integration, testing, and deployment lifecycle. This CI/CD process automates several key stages:

- Build Software Artifacts: Upon code commits, the CI pipeline can automatically compile source code and build executable components. For the platform's microservices and nApps, this typically involves building Docker container images.
- Publish to Registry: The newly built Docker images are then published to a designated container registry. While the project GitLab may have its own registry, the system allows for nApps to reference images in various accessible registries.
- Deploy to Test Environments: Updated components or nApps can be automatically deployed to the specified test environments within the Nokia and Telekom Slovenije infrastructures. This automated deployment is crucial for maintaining consistency and enabling rapid iteration.
- Trigger Test Automation: Following a successful deployment, the CI/CD pipeline can also initiate automated test suites to validate the functionality and

performance of the deployed components or nApps. Deliverable D5.2 outlines how the 5G-IANA Test Automation framework can be triggered by the DevOps pipeline after an nApp is built, published, and deployed.

The establishment of this structured CI/CD process, supported by a flexible repository approach and centralized version control, is of paramount importance for both nApp developers and the vertical end-users who will ultimately consume the automotive services. For nApp developers, including SMEs and third-party experimenters, it provides a clear, modular, and significantly simplified pathway to onboard, integrate, and iterate on their innovative applications. The automation reduces manual overhead, speeds up development cycles, and ensures that nApps can be reliably tested and deployed. For vertical end-users, this translates to faster availability of new and updated automotive services, improved service reliability due to standardized testing, and the agility for service providers to quickly adapt to evolving requirements or introduce new features. Ultimately, this easily updatable and manageable process lowers the barrier to entry for nApp creation and fosters a more dynamic and responsive ecosystem for advanced automotive applications.

3 5G-IANA Use Cases & Validation

This section presents a structured overview of the use cases explored within the 5G-IANA project [2], together with the adopted corresponding validation methodologies. Full technical details and methodological foundations such as the performed Test Cases (TCs) can be found in project deliverables [D512023, D522024]. The use cases have been categorised into five thematic section areas, to support a more focused examination of objectives, implementation contexts, and observed outcomes:

- Connected and Automated Mobility;
- Smart Infrastructure and Safety;
- In-Vehicle Infotainment and Personalization;
- and a general category for the Third-Party Experiments by external experimenters.

Additionally, this section outlines *(i) Cross-use case and (ii) Third-Party validation*.

While the exact performance results and detailed KPIs remain confidential and are not disclosed in this public whitepaper, the methodological principles, evaluation approach, and deployment configurations are outlined to offer insight into the project's overarching validation strategy. Specific artefacts and results may be made available by the partners involved[2] per UC upon request.

3.1 Connected and Automated Mobility

3.1.1 UC1 – Remote Driving

3.1.1.1 Use Case Description – Step-by-Step Overview

UC1 focuses on enabling advanced remote driving capabilities through the 5G-IANA experimentation platform. The goal is to achieve real-time, low-latency teleoperation of

[2] Details available online: https://www.5g-iana.eu/about/use-cases/

an Automated Guided Vehicle (AGV) connected via 5G to a remote cockpit interface. Implementation has been broken into two major cycles. In **Cycle A**, the AGV is outfitted with an OBU, front and rear video cameras, and a LiDAR sensor. The video feed is transmitted to the 5G edge where AI/ML algorithms process and overlay object detection information (e.g., pedestrians, vehicles, signage). This data, combined with LiDAR input, feeds into a tracking algorithm visualizing the AGV's surroundings in 2D.

FIGURE 7. CYCLE A HUMAN MACHINE INTERFACE (HMI).

Cycle B builds on this setup by adding two lateral cameras to provide a full 360° view, rendered into a virtual reality (VR) environment for the operator in the cockpit. The resulting configuration delivers a fully immersive remote driving interface.

The integration and validation approach were designed to be incremental. First, we checked pure connectivity between the vehicle and the 5G network. Next, we validated video and sensor data transfer to the edge, and finally, we tested end-to-end connectivity with the remote driving interface. Each component was tested individually and then integrated into the full-service chain for complete validation.

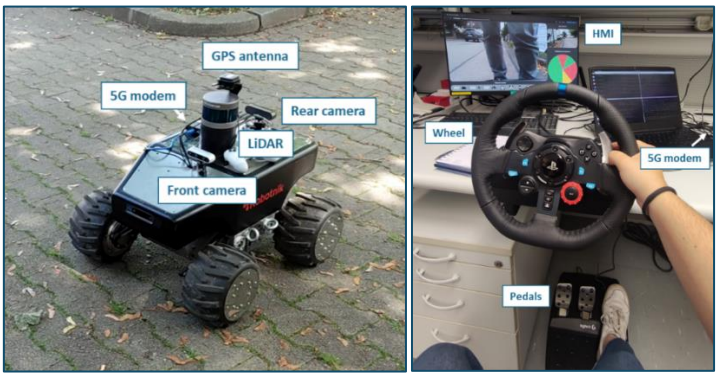


FIGURE 8. CYCLE B SETUP AND DEMONSTRATION: VEHICLE WITH OBU (LEFT) AND DRIVER'S COCKPIT (RIGHT).

3.1.1.2 Platform Requirements

The successful execution of UC1 hinged on several platform-level needs. These include:

- **Low-latency 5G connectivity** to support real-time teleoperation and video streaming;
- **Reliable edge computing** capabilities for deploying AI-based object detection and data fusion algorithms;
- **Container orchestration** (via microk8s) for managing nApps (network applications) on both the OBU and edge nodes;
- **Secure and stable video/data transmission** pipelines, including encoding at the vehicle and decoding at the edge;
- **Scalability for future expansion**, particularly as more sensors and cameras are added (e.g., in Cycle B with VR);
- **Cross-node visibility**, ensuring all components (vehicle, cockpit, edge) are reachable with minimal delay and loss;
- **Support for VR rendering** in the cockpit for the 360° spatial awareness system;
- **Cross-UC compatibility**, demonstrated in Cycle B during joint validation with UC6's predictive QoS algorithm.

3.1.1.3 Validation

Validation was conducted in structured phases, with real-world deployments and field demonstrations.

In **Cycle A**, testing began in Valencia with local integration, leading to a successful full-stack test over a commercial 5G network, including remote driving using cockpit controls. Subsequent tests were done at Nokia's testbed in Ulm, where all critical functionalities—including video transmission, AI inference, LiDAR fusion, and control interface—were confirmed. Each test case passed, including connectivity, edge integration, and full control handover.

Cycle B added complexity with containerized video encoding and object positioning plus the 360° VR cockpit. These features were tested both in Valencia and Ulm, culminating in a successful demo with the vehicle and cockpit physically separated and operating over a constrained 5G network. Notably, the vehicle was placed in a low-coverage area to test resilience, while the operator remained in Nokia's lab.

3.1.1.4 Expected Long-Term Benefits of Using the OAEP

The OAEP proved instrumental in getting UC1 to work in both lab and field environments. In the long run, its modular, containerized approach offers multiple strategic advantages:

- **Flexibility to scale and adapt**—future iterations can easily include new sensors, upgraded AI models, or enhanced UIs.
- **Streamlined onboarding**—nApps were deployed across different locations and 5G setups without reworking the core system.
- **Cross-project synergy**—shared components between UC1 and other use cases (e.g., UC6) helped unify development efforts and reduce redundancy.
- **Operational robustness**—the platform's orchestration tools and monitoring capabilities helped ensure stability across varied network conditions.
- **Accelerated innovation**—by abstracting infrastructure complexity, teams can focus more on feature development and less on integration overhead.

Ultimately, the platform's capabilities are paving the way for next-gen remote operation services in smart mobility, where edge intelligence, 5G connectivity, and AI-driven perception come together seamlessly.

3.1.2 UC2 - Manoeuvres Coordination for Autonomous Driving

3.1.2.1 Use Case Description – Step-by-Step Overview

UC2 centers on the coordination of manoeuvres between connected autonomous and manually-driven vehicles in potentially hazardous or congested scenarios—such as intersections—using a distributed network of rover and edge applications (nApps). This use case was validated over two development cycles within the 5G-IANA project and was demonstrated during two major review meetings.

In **Cycle A**, validation began by deploying the service chain (SC) across two development machines. The initial test verified the message flow integrity between a simulated vehicle and the Manoeuvre Planner. The vehicle, navigating a virtual replica of the NOKIA testbed, successfully triggered a response from the planner, demonstrating proper interaction.

FIGURE 9. 3D MODELLED PARKING AREA FROM A TOP-DOWN VIEW, INSIDE OF CARLA SIMULATOR.

Subsequent testing introduced a second virtual vehicle. With collision detection disabled via CARLA's percentage_ignore_vehicles parameter and safety distance reduced to 2m, the planner was stress-tested. These runs confirmed that increasing the action radius to 20m prevented erratic vehicle behavior. Upon confirming functional performance in simulation, the MCAD-rover and MCAD-edge nApps were deployed across the NOKIA testbed—first in virtual, then on physical OBUs. After resolving minor message format and ITS communication issues, all functional KPIs were validated, aside from those related to the HMI.

In **Cycle B**, validation moved to the TS testbed, which required multiple iterations due to changing 5G coverage. Tests involved a hybrid setup—one physical vehicle with HMI and one virtual vehicle, both governed by the MCAD-rover nApp. With the Manoeuvre Planner active, vehicles responded as expected to intersection scenarios: for example, the real car received a slowdown instruction when another vehicle had priority. Speed commands were later fine-tuned to better match the constraints of the parking environment.

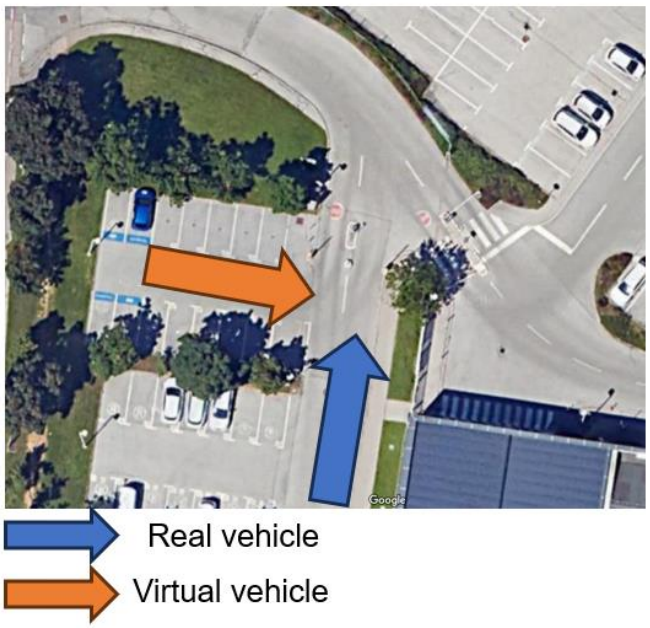


FIGURE 10. UC2 CYCLE B TEST SETUP.

3.1.2.2 Platform Requirements

Successful deployment and operation of UC2 required a reliable and flexible edge-cloud infrastructure capable of hosting distributed nApps, including real-time data handling between rover (vehicle-side) and edge (infrastructure-side) nodes. Key requirements included:

- **Low-latency communication** between OBUs and edge servers.
- **Modular deployment** of dockerized components across physical and virtual machines.
- **Accurate vehicle simulation environments**, notably CARLA, tightly integrated with control logic.
- **Robust HMI integration** for real-time user feedback during hybrid testing.
- **ITS-G5 and long-distance 5G communication support** to accommodate mobility and scalability.

3.1.2.3 Validation

By the end of Cycle B, all major functional KPIs had been met. Key validation highlights included:

- Real-time coordination between physical and virtual vehicles.

- Safe negotiation of intersection right-of-way, with speed modulation based on planner output.
- Successful deployment of SC components on real hardware (OBUs, edge servers).
- Live demonstrations showing measurable benefits of UC2 in preventing collisions.

Final validation results of the various KPIs imposed for this UC showcased a service deployment time (from nApp deployment to the first transaction with the intersection management software) of 149 ms, which allows for seamless onboarding of any vehicle that is approaching an intersection managed by MCAD. Furthermore, the lack of packet loss and end-to-end latency of 37ms demonstrate that the platform and developed nApps can be used in safety-critical scenarios.

The January 2024 demonstration highlighted the efficacy of the SC: virtual vehicles collided when disconnected from UC2 but avoided collision when connected. The July 2024 demonstration was more advanced, involving hybrid runs with physical and virtual vehicles both executing planner instructions while reviewers observed behavior live from within the connected vehicle.

The most ambitious of the demonstrations took place at the final public event in late 2024, when the same scenario of the July demonstration was performed by BYL's autonomous vehicle and a manually-driven one. The former would adjust its speed based on the messages received from the maneuver planner, whilst the latter would receive notifications on the HMI screen. The scenario was successfully demonstrated during the event, with all the nApps involved being the same as the ones used in the previous demonstration.

3.1.2.4 Long-Term Benefits of OAEP

The OAEP underpins UC2 by enabling scalable, flexible deployment of intelligent coordination services across edge and vehicle hardware. In the long term, its benefits include:

- **Scalability**: As more vehicles connect to the system, nApps can be dynamically deployed or relocated across the infrastructure.
- **Safety Enhancement**: Early and precise manoeuvre planning reduces the likelihood of accidents in mixed traffic environments.
- **Infrastructure Efficiency**: With coordination handled at the edge, latency is reduced and network resources are optimized.
- **Interoperability**: The modular architecture supports integration with third-party services, paving the way for complex multi-use-case deployments.

Ultimately, UC2 serves as a strong example of how real-time coordination powered by edge-native applications can directly improve safety and efficiency in future mobility systems.

3.1.3 UC6 - Network Status Monitoring

3.1.3.1 Use Case Description & Validation Steps

UC6 performs proactive monitoring and fault detection for vehicle connectivity in low-signal environments. It explores how Federated Learning (FL) can be applied at the edge to develop and maintain predictive models for network QoS without centralizing sensitive data. It aims to create a privacy-preserving, distributed architecture for continuous QoS monitoring and model refinement using data collected from onboard units (OBUs) in connected vehicles.

The validation of UC6 was structured across two development cycles. **Cycle A** focused on validating individual service components and standalone FL functionality. This included testing the preprocessing and training modules, verifying data collection pipelines, evaluating QoS prediction models, and ensuring component interoperability with the 5G-IANA platform and EEO (Extreme Edge Orchestration) environment.

In **Cycle B**, the focus shifted to full end-to-end validation. The complete UC6 service chain was deployed and managed through the EEO, using a set of four OBUs—two in lab and two in vehicles. These tests involved extensive scenario-based evaluations including hyperparameter tuning, model benchmarking, and lifecycle orchestration through the EEO. Key scenarios explored the effects of varying batch sizes, learning rates, and training epochs, as well as the impact of these variables on accuracy and convergence speed.

The final evaluation confirmed UC6's ability to train predictive models in a distributed manner using real-world edge devices, dynamically adapting to network conditions and training configurations while maintaining synchronization across multiple OBUs.

3.1.3.2 Platform Requirements

To enable distributed machine learning at the edge, UC6 required specific capabilities from the 5G-IANA platform:

- **Federated Learning orchestration** to coordinate model training across multiple edge devices (OBUs).

- **Reliable low-latency communication** between training agents and aggregation nodes to prevent synchronization issues.
- **Edge-computing resources** to support lightweight AI/ML models with minimal network overhead.
- **Dynamic device selection** through geolocation and policy-based criteria (e.g., Area of Interest).
- **Lifecycle management** for real-time activation and deactivation of containerized VNFs on OBUs.

3.1.3.3 Validation

The validation process successfully demonstrated the effectiveness of UC6's federated learning-based QoS monitoring system. Across various test scenarios, the system consistently achieved its objectives:

- FL configurations (batch size, learning rate, epochs) were successfully tested and tuned.
- Model convergence and accuracy targets were met under controlled training rounds.
- Synchronization between distributed OBUs and the Aggregation Node was validated.
- Training sessions dynamically adapted to participating OBUs based on real-time conditions and orchestration policies.

The system remained stable under multiple training rounds and supported seamless edge-level deployment and operation.

UC6 was demonstrated twice—initially in a standalone configuration during Cycle A, and later in a full end-to-end scenario during Cycle B. Both demonstrations took place at the Nokia testbed in Ulm, with the final event incorporating four OBUs: two static and two in moving vehicles.

Using the GUI provided by the Extreme Edge Orchestrator (EEO), an FL session was initiated with real-time criteria such as area restrictions and device participation limits. As vehicles entered and exited the predefined Area of Interest, their status changed dynamically within the FL process. The EEO effectively managed these transitions, demonstrating lifecycle events like client selection, activation, and deactivation.

The demonstration showed that the platform could execute distributed training, maintain model accuracy, and adjust in real time based on geographic and network conditions—all without interrupting service continuity.

Throughout validation and demonstration, several key KPIs were measured:

- **5G Round Trip Time (RTT)** remained within acceptable thresholds, ensuring that latency did not interfere with training synchronization.
- **Service-level KPIs**, such as model training and aggregation time, were successfully met due to the use of compact models optimized for time series prediction.
- **End-to-End Reliability** was achieved, with no significant packet loss reported, meeting the system's stringent requirements for dependable training data exchange.

These results confirm the robustness and practical feasibility of deploying FL-based QoS prediction in real edge environments using the 5G-IANA infrastructure.

3.1.3.4 Long-Term Benefits of the OAEP

UC6 illustrates how the OAEP can support advanced AI-driven services at the extreme edge. With its orchestration capabilities, resource management, and lifecycle tools, the platform provides a solid foundation for federated learning and other distributed intelligence use cases. Long-term, this enables more adaptive, privacy-aware, and scalable network monitoring solutions that respond to real-world conditions and support continuous learning without centralized data collection.

3.2 Smart Infrastructure and Safety

3.2.1 UC5 - Real-Time Road Network Risk Assessment

3.2.1.1 Use Case Description & Validation Steps

The UC5 use case is focused on enhancing road safety by enabling real-time detection, analysis, and communication of hazardous driving behaviors and potential risks on the road network. It leverages multiple application and network functions working in concert to assess risks and deliver timely alerts to connected vehicles.

The implementation and validation of UC5 were structured over two development cycles. **Cycle A** centered on establishing the service chain's core capabilities, including detecting hazardous driving behaviors, predicting vehicle trajectories, and generating road segment-level risk assessments. All relevant components were integrated

manually during this phase, and initial tests used a mix of virtual and physical OBUs. The virtual OBUs provided simulated position and event data, which were then validated through controlled tests using a real vehicle equipped with a physical OBU.

```
INFO:root:--- Training Started ---
INFO:root:Clusters: []
INFO:root:--- Training Ended ---
INFO:root:--- Training Started ---
INFO:root:Clusters: []
INFO:root:--- Training Ended ---
INFO:root:--- Training Started ---
INFO:root:Clusters: []
INFO:root:--- Training Ended ---
INFO:root:--- Event received with timestamp 2023-11-22 09:48:20.087000---
INFO:root:--- Event Added ---
INFO:     172.30.0.4:38490 - "POST /process_harsh_event HTTP/1.1" 200 OK
INFO:root:--- Event received with timestamp 2023-11-22 09:48:20.087000---
INFO:     172.30.0.4:38500 - "POST /process_harsh_event HTTP/1.1" 200 OK
INFO:root:--- Event received with timestamp 2023-11-22 09:48:20.087000---
INFO:     172.30.0.4:38502 - "POST /process_harsh_event HTTP/1.1" 200 OK
INFO:root:--- Event received with timestamp 2023-11-22 09:48:20.087000---
INFO:     172.30.0.4:38514 - "POST /process_harsh_event HTTP/1.1" 200 OK
INFO:root:--- Event received with timestamp 2023-11-22 09:48:20.087000---
INFO:     172.30.0.4:38516 - "POST /process_harsh_event HTTP/1.1" 200 OK
INFO:root:--- Event received with timestamp 2023-11-22 09:48:21---
INFO:     172.30.0.4:38520 - "POST /process_harsh_event HTTP/1.1" 200 OK
INFO:root:--- Training Started ---
INFO:root:Clusters: [Cluster(latitude=48.4261782, longitude=9.9610919, radius=20.0, risk_level_index=2)]
INFO:root:--- Training Ended ---
```

FIGURE 11. CREATION OF HAZARDOUS DRIVING EVENTS CLUSTERS.

Multiple testing scenarios were designed to simulate various risky driving behaviors, such as harsh braking, harsh acceleration, speeding, mobile phone usage, and the aggregation of these behaviors into higher-level road risk assessments. In these simulations, alert notifications were successfully triggered based on the vehicles' relative positioning, heading, and the type of event detected. The system's ability to differentiate between scenarios with genuine risk and those with no collision potential was also validated, ensuring that alerts were context-aware and accurate.

Cycle B expanded on this work by refining the algorithms for driving behavior detection and risk assessment, and by integrating the system into the 5G-IANA platform for automated deployment. The Human-Machine Interface (HMI) from UC2 was also incorporated, offering a unified alert display mechanism. This phase involved two real vehicles, both equipped with OBUs, enabling more realistic end-to-end validations across the full service chain.

During Cycle B validation, all previously defined scenarios were executed again under live conditions. This included dynamic interactions between the two vehicles to validate risk prediction, event clustering, and real-time notification delivery. The alerts were displayed directly on the HMI, with validation confirming the system's responsiveness, accuracy, and overall performance in real-world conditions.

3.2.1.2 Platform Requirements

UC5 required several foundational elements of the 5G-IANA platform to perform effectively:

- **Low-latency communication** between OBUs and the Edge to ensure timely risk notification.
- **Reliable deployment and orchestration** of multiple VNFs and AFs, including modules for event detection, risk aggregation, and HMI integration.
- **Edge intelligence** to process sensor data locally and trigger alerts without relying on centralized cloud systems.
- **Context-awareness** to accurately distinguish between genuine and non-genuine risks based on vehicle trajectories and behavior patterns.

3.2.1.3 Validation

The validation process successfully demonstrated UC5's ability to detect and respond to a variety of hazardous driving events under controlled and real-world conditions. Key outcomes included:

- Real-time delivery of context-specific alerts to in-vehicle HMIs.
- Accurate risk assessments based on event clustering and vehicle trajectory predictions.
- Stable performance across different test environments, including scenarios with two physical vehicles.
- Reliable edge-to-vehicle communication supporting the delivery of warnings with minimal latency.

These results confirmed that UC5 can provide timely and relevant warnings to drivers, enhancing situational awareness and contributing to proactive road safety.

During validation, key performance indicators were assessed to evaluate network and service reliability:

- **Round Trip Time (RTT)** and **E2E latency** were well within acceptable limits, enabling responsive communication between vehicles and the edge.
- **E2E reliability** was demonstrated with zero packet loss during testing, affirming the robustness of the communication channel.
- **Service availability** remained consistent, with minor network limitations only in peripheral coverage areas, unrelated to the UC5 system itself.

- **Deployment time** was minimal, demonstrating the platform's efficiency in bringing services online quickly and consistently.

These KPI results reinforce the practical viability of UC5 for deployment in real-world road safety scenarios.

3.2.1.4 Long-Term Benefits of the OAEP

The OAEP was instrumental in supporting UC5 through streamlined service deployment, modular integration of intelligent detection components, and reliable orchestration across edge infrastructure. Its flexible architecture enables continued scaling and evolution of UC5 into broader smart mobility ecosystems, enabling dynamic adaptation to diverse road conditions, user behaviors, and policy requirements. Long-term, this positions the platform as a key enabler of proactive vehicle safety solutions.

3.2.2 UC7: Situational Awareness in Cross-Border Road Tunnel Accidents

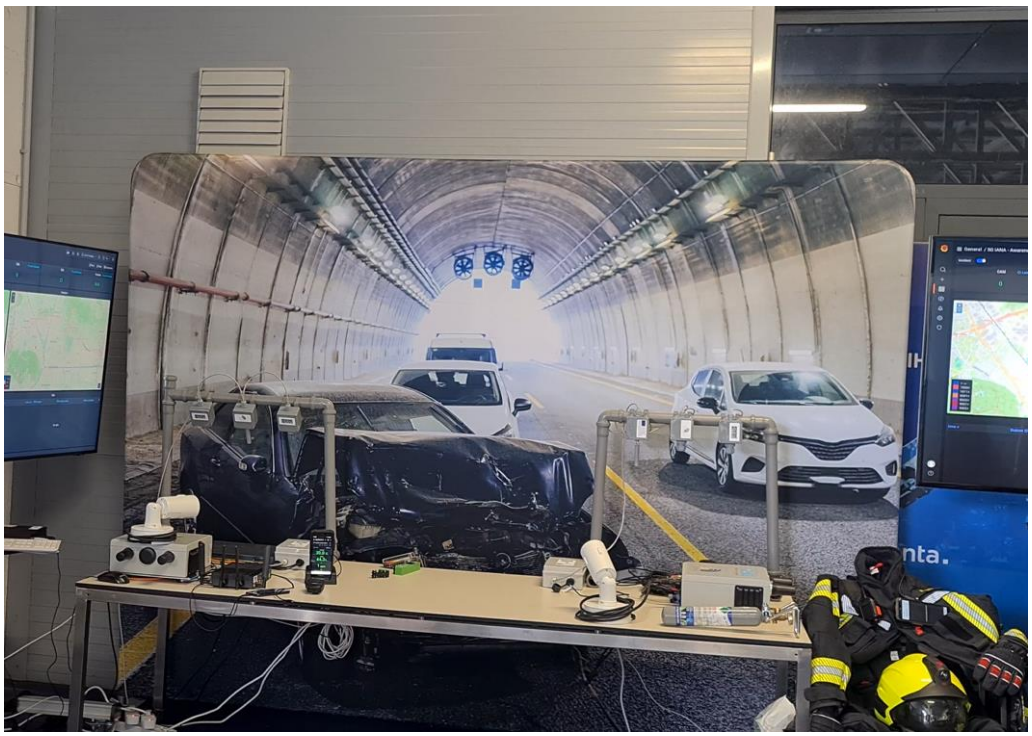


FIGURE 12.. SENSORS AND CAMERAS CONNECTED TO RSU PROVIDE RELEVANT DATA FOR THE SITUATIONAL AWARENESS ENHANCEMENTS.

3.2.2.1 Use Case Description & Validation Steps

UC7 addresses the need for enhanced situational awareness in road tunnel environments, with particular focus on cross-border ones. The goal is to enable reliable monitoring, incident detection, and timely response support for public protection and disaster relief (PPDR) services using a distributed 5G-enabled architecture.

Validation for UC7 was carried out through a continuous process, beginning with the testing of individual service chain components and progressing to full end-to-end system validation. Each step was essential to ensure that the components operated reliably in isolation before they were integrated into the complete pipeline.

Key elements of the validation included confirming connectivity across geographically distributed sites using a secure VPN tunnel, ensuring sensor and camera integration with roadside units (RSUs), and verifying communication between edge-deployed software components and user equipment (UE). These tests validated the real-time streaming of environmental data and live video, accurate sensor data updates, and the overall robustness of the software components that power the nApp responsible for situational awareness.

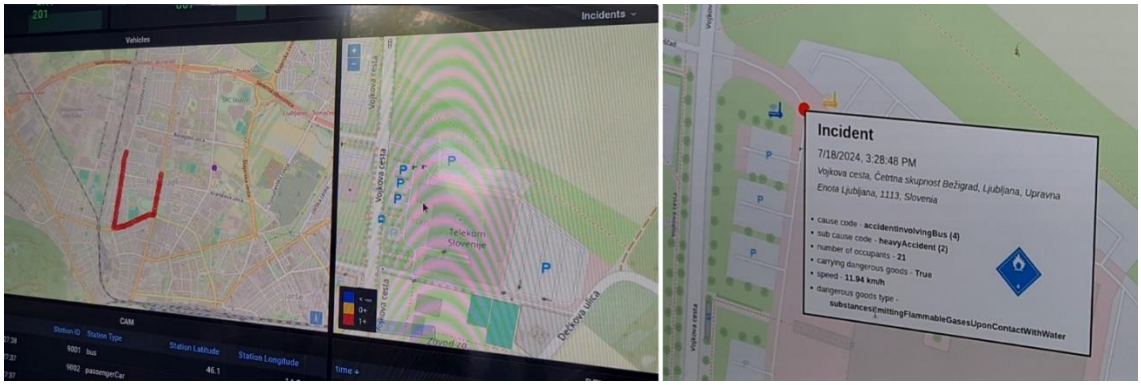


FIGURE 13. MONITORING A VEHICLE'S MOVEMENT (FOR THE DEMONSTRATIONAL PURPOSES ONLY; SEE LEFT-HAND PART OF THE FIGURE), AND DETECTING AN ACCIDENT ALREADY PROVIDING SOME RELEVANT DATA ON VEHICLES INVOLVED IN THE ACCIDENT (RIGHT-HAND PART OF THE FIGURE).

A dedicated simulator was also used to emulate Cooperative Awareness Messages (CAM), simulating the movement and behavior of vehicles involved in tunnel traffic. This allowed validation of accident detection logic, data flow consistency, and the system's ability to interpret and respond to simulated hazardous situations. These validations ensured that the entire UC7 service chain—from sensor input to user interface—could

reliably operate across testbeds, even when elements were distributed between different countries.

UC7 was demonstrated at the Telekom Slovenije (TS) testbed, where the full use case scenario was executed under controlled conditions. The demonstration included individual validation of hardware and software components followed by an integrated, end-to-end showcase of tunnel monitoring and accident response support.

The scenario simulated a traffic accident involving multiple vehicles within a tunnel environment. Once the incident was detected, relevant alerts and contextual data—including vehicle positioning, environmental sensor readings, and live video from the tunnel—were made available to tunnel control operators and PPDR users through the graphical user interface (GUI). This simulated real-time decision-making and communication capabilities required for effective emergency response.

The demonstration effectively illustrated the system's ability to detect incidents, communicate risk, and provide multiple layers of situational context through a distributed, 5G-connected platform.

3.2.2.2 Platform Requirements

UC7 required a robust set of platform capabilities to function effectively:

- **Cross-site connectivity and orchestration**: Reliable VPN and edge coordination between testbeds located in different countries (Slovenia/TS, Germany/Nokia).
- **Low-latency video and data streaming**: For real-time access to sensor data and live camera feeds.
- **Device and edge integration**: Smooth operation between RSUs, OBUs, and Edge servers, with validated communication channels.
- **Multi-interface access for PPDR tools**: Supporting standard protocols (e.g., HTTPS, RTSP) for user access to sensor and video streams.
- **Simulation tools**: For validating emergency response workflows without needing live tunnel deployment.

3.2.2.3 Validation

The validation process successfully confirmed the functionality of the complete UC7 service chain. Key achievements included:

- Verified end-to-end communication between geographically separated testbeds.

- Real-time data acquisition from sensors and cameras, streamed to user interfaces.
- Successful operation of accident detection simulations and accurate alert generation.
- Consistent system behavior across multiple test scenarios involving different data sources and hardware configurations.

This multi-phase validation approach ensured the reliability of UC7's components in both standalone and integrated forms, reinforcing the feasibility of deploying such a system in real cross-border tunnel environments.

Performance testing and KPI evaluation were conducted throughout the final validation stages, particularly during activities at the TS testbed in mid-2024. While formal KPI definitions were set earlier in the project, data was collected over an extended period to observe system stability and responsiveness under varying conditions.

KPI assessments focused on:

- **Latency and responsiveness** of data transmission from RSUs to edge servers and onward to user devices.
- **Availability and stability** of the service chain during real-time simulations.
- **Connectivity performance** across VPN tunnels linking testbeds in different geographic locations.
- **Accuracy and timeliness** of alert generation and sensor/video synchronization.

These performance observations validated the system's readiness for real-world use cases where quick decision-making, robust connectivity, and data accuracy are critical for emergency tunnel management and operator support.

3.2.2.4 Long-Term Benefits of the OAEP

UC7 highlights the strength of the OAEP in deploying complex, multi-component services across geographically distributed infrastructure. With its modular design and support for secure cross-site orchestration, the platform enables scalable deployment of intelligent monitoring systems for safety-critical environments. Its ability to unify data from diverse sources into actionable situational awareness makes it especially valuable for PPDR services, where time-sensitive and context-rich information is essential.

3.3 In-Vehicle Infotainment and Personalization

3.3.1 UC3: Virtual Bus Tour

3.3.2 Use Case Description & Validation Steps

The Virtual Bus Tour (UC3) use case aimed to demonstrate high-quality, low-latency immersive video streaming over 5G infrastructure, leveraging edge computing and network-aware adaptive media pipelines. It was validated in two development cycles, Cycle A and Cycle B, both carried out on Nokia's 5G testbed in Ulm, Germany.

Cycle A focused on setting up the foundation. The goal was to get a baseline adaptive bitrate video streaming service up and running from the extreme edge to the end user. The service chain was deployed step-by-step to the AOEP (Advanced Orchestration & Edge Platform), starting with individual components and eventually bringing the full chain online. An ASUS laptop with an NVIDIA GPU served as the deployment node and was shipped to the Nokia site. Remote validation over VPN ensured everything was functioning before on-site testing took place in November 2023.

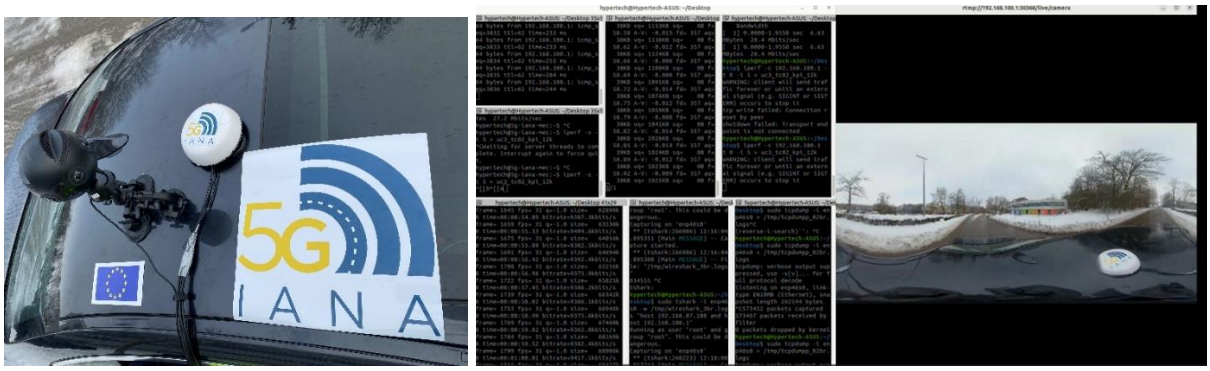


FIGURE 14. CYCLE A VALIDATION ACTIVITIES FROM THE SETUP IN THE NOKIA PARKING LOT INCLUDING SCREENCAPTURE FROM VALIDATION ACTIVITIES IN HELMHOLTZSTRAßE/DRK ULM SITE.

Due to weather conditions, the Nokia bus wasn't available, so a private vehicle was used instead. Nevertheless, core video streaming functionality was confirmed, and all major service chain components performed as expected, including adaptive streaming capabilities.

Cycle B built on that foundation with significant upgrades. We integrated a foveated rendering pipeline and added hardware acceleration across key modules (like the Privacy Masking Module, Video Slicer, and Live Stream Encoder). The Video Stream Cache was also improved by integrating the Active Network Monitoring module to enhance responsiveness to network conditions.

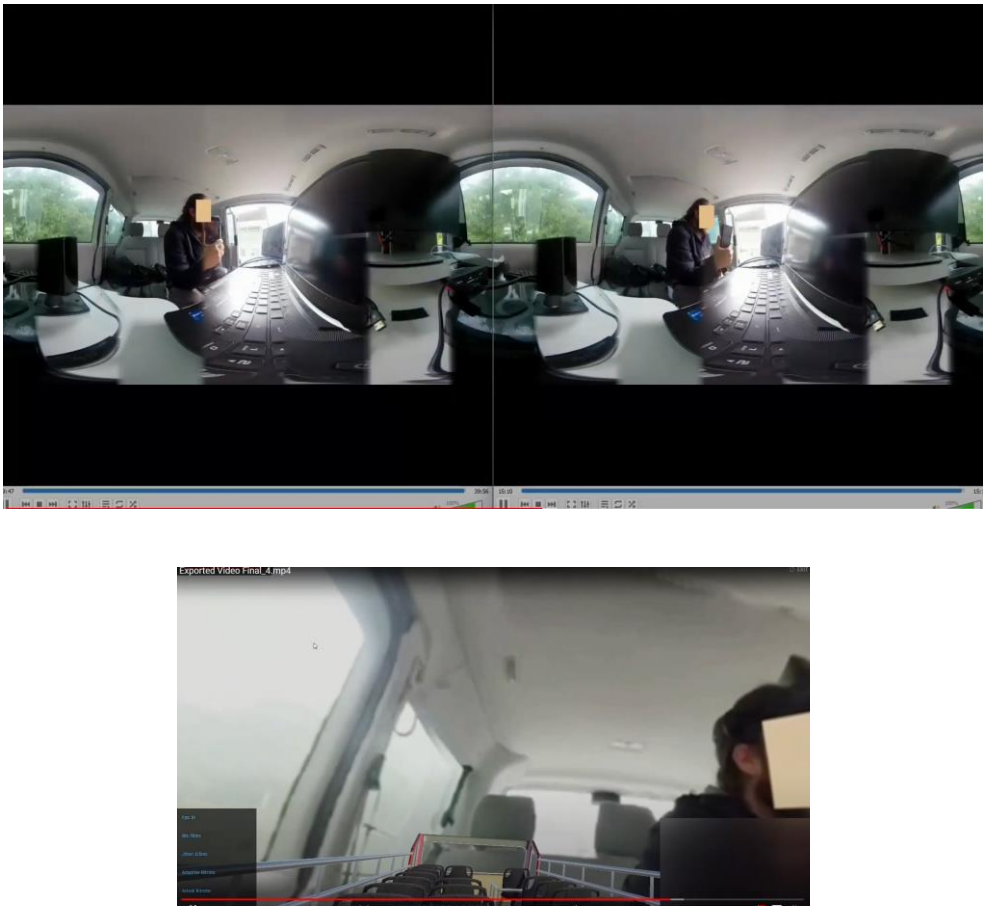


FIGURE 15. CYCLE B DEMONSTRATION ACTIVITIES.

One big takeaway from Cycle A was the importance of optimizing hardware performance, especially given the encoding/decoding load on the Hypertech ASUS system. Another was the realization that streaming resolution was limited by specific antenna sites. While moving to the TS testbed was considered, resource constraints meant UC3 remained in Nokia's testbed but was expanded to include a shared component with UC6.

Validation took place across multiple visits—June 24–26, during the July review, and on September 10. The Nokia bus was used for all tests. Two main locations were involved:

Helmholtzstraße (DRK ULM antenna) and the city center near the Theatre (SWU K1 antenna), each presenting different 5G conditions.

In each round, we began by setting up the hardware inside the bus and performing stationary tests to ensure full pipeline functionality. Once a steady stream was confirmed, we validated video quality during movement. Foveated rendering was evaluated on-site, and early tests revealed that the ML model driving it needed retraining due to visible artifacts under network strain. We also introduced a fallback mechanism to mitigate this.

In the July review rehearsal, a major issue surfaced between the video stream cache and the 360 video endpoint, related to DASH manifest compatibility during bitrate switching. This was eventually traced to a versioning mismatch between GPAC and FFMpeg components and resolved after the review.

The final round in September focused on SWU K1. Unfortunately, the pQoS model hadn't been trained for that site, so we couldn't validate UC3-6 integration. A power supply issue with the camera also caused a one-day delay. Despite that, we completed KPI measurements and recorded the UC3 demo video on the final day.

3.3.2.1 Platform Requirements

The success of UC3 relied on several critical platform capabilities:

- **Edge-Oriented Video Pipeline Support**: Real-time video processing at the extreme edge, requiring GPU acceleration and efficient data routing.
- **Dynamic Orchestration**: Onboarding and reconfiguring service chains via AOEP to adapt to hardware and network constraints.
- **Network Awareness**: Active monitoring and feedback mechanisms to allow dynamic quality adjustment based on network performance.
- **Modular Interoperability**: Seamless integration of NF/AF components (e.g., DASH clients, video encoders) across different vendors and versions.
- **ML Integration at the Edge**: For foveated rendering to function effectively with real-time input and fallback in low-quality conditions.

3.3.2.2 Validation

All major UC3 functions were validated successfully across both development cycles, with concrete performance insights and problem-solving steps documented. Highlights included:

- Stable end-to-end video delivery over 5G with adaptive bitrate handling.
- Real-time streaming through mobile edge setups inside the Nokia bus.
- Detection and resolution of foveated rendering artifacts and compatibility bugs.
- Integration of streaming optimization modules with fallback strategies for network fluctuation.

Issues encountered, like model limitations and component incompatibilities, were systematically addressed, often leading to more robust pipeline behavior in subsequent iterations.

3.3.2.3 Long-Term Benefits of OAEP

The OAEP played a pivotal role in enabling UC3 by providing the tools and structure for edge-native application deployment and lifecycle management. Long-term, the benefits are clear:

- **High Adaptability**: Services can scale up or down depending on device and network conditions, making it ideal for urban mobility applications.
- **Efficient Resource Use**: Edge computing offloads intensive tasks from end devices, improving energy efficiency and responsiveness.
- **Enhanced Quality of Experience**: Network-aware, ML-enhanced streaming pipelines help deliver a smooth, high-quality viewing experience in real-world conditions.
- **Modularity for Future UCs**: UC3's architecture supports future extensions—like pQoS-based personalization or immersive AR streaming—without major redesigns.

As 5G infrastructure matures and more edge-capable devices come online, platforms like OAEP are key for media-rich, latency-sensitive experiences at scale.

3.3.3 UC4 – AR Content Delivery for Vehicular Networks

3.3.3.1 Use Case Description & Validation Steps

The UC4 use case showcases the ability to stream real-time Augmented Reality (AR) content—such as 3D models and contextual text—directly to AR-enabled devices in moving vehicles, utilizing the 5G-IANA platform. This functionality was validated and demonstrated across two distinct development cycles, Cycle A and Cycle B.

In **Cycle A**, the focus was on getting the foundational components in place. The core goal was to prove that AR content could be requested and received by a mobile client connected to the 5G-IANA infrastructure. This involved deploying key VNFs such as Long-Distance Data Communication (NF#1), AR Content Repository, and AR Media Access Function. These components worked together to create a stable framework where an AR device could connect to the platform, request content, and display it in a navigation-like interface.

The validation process began by ensuring the AR client device could connect to the Edge server. Once connectivity was confirmed, the team validated that AR content could be streamed in real time to the client, which was often mounted in a moving vehicle. The interface clearly displayed multiple AR markers based on proximity, GPS location, and user orientation. These markers represented various POIs (e.g., banks, coffee shops, bus stops), with their scale dynamically changing depending on distance and viewing angle—effectively simulating a real-world AR navigation experience.

In **Cycle B**, the pipeline was enhanced to support more robust and equitable content delivery. A Virtualized Cache was introduced to bring content closer to the user, while a Reinforcement Learning-based scheduling algorithm was implemented within the AR Media Access Function to manage multiple simultaneous content requests. This upgrade was driven by two key insights from Cycle A: (1) the importance of resource efficiency for streaming at scale, and (2) the need to avoid quality drops due to network variability and client concurrency.

Multiple rounds of testing were conducted with real-world movement in vehicles. The AR devices were tested in various 5G zones with help from project partners, ensuring accurate GPS, low latency, and content responsiveness. The Reinforcement Learning algorithm used real-time network metrics (e.g., latency, data rate) to fairly manage content delivery between clients, even when their demands varied.

The result was a validated pipeline that could reliably stream AR content to multiple devices in parallel, while maintaining consistent quality and responsiveness across a variety of real-world scenarios and network conditions.

3.3.3.2 Platform Requirements

UC4 required several core platform features to function effectively:

- **Reliable 5G Edge Connectivity**: Low-latency communication between user equipment and the VNFs on the 5G-IANA platform.
- **Content-Oriented VNFs**: Modular VNFs for storage, access, and caching of AR data.
- **Mobility-Aware GPS Integration**: High precision geolocation to ensure accurate AR object placement.
- **Intelligent Scheduling**: Use of Reinforcement Learning to handle concurrent requests efficiently under variable network conditions.
- **Multithreading Support**: To allow parallel content streaming across devices without impacting performance.

3.3.3.3 Validation

Both validation cycles proved the viability and robustness of the UC4 pipeline. Key outcomes include:

- Verified end-to-end AR content delivery in real-time during vehicle movement.
- Stable GPS and AR alignment during travel across various testbed locations.
- Real-time responsiveness to user orientation and location changes.
- Fair and efficient handling of multiple user requests using RL-based scheduling.
- Successful use of caching and media access logic to enhance delivery speed and reduce latency.

These results demonstrated that AR content can be streamed effectively across vehicular networks with dynamic user interactions and in motion, which is critical for future AR-driven services in mobility and tourism.

Cycle A was demonstrated in January 2024 at the Nokia site in Ulm, Germany. While not live, the demonstration included a visual presentation of the route taken by the test vehicle and real AR content responses on the AR device screen. The content dynamically changed in size and position based on the vehicle's distance from relevant POIs and its orientation—visually confirming the core pipeline's operational logic.

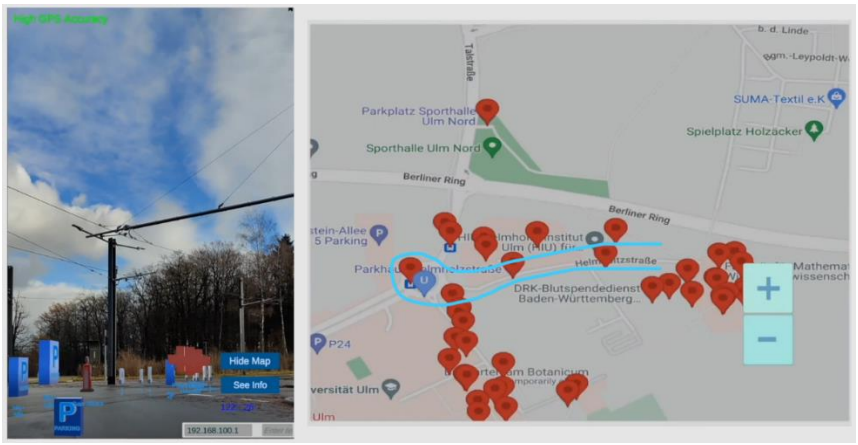


FIGURE 16. THE ROUTE USED ON CYCLE A DEMONSTRATION.

Cycle B took place in July 2024 in Ljubljana, Slovenije, at Telekom Slovenije's testbed. This time, the demonstration was live. The UC4 service chain was deployed using the 5G-IANA Composer GUI, and a vehicle equipped with multiple AR devices followed a predefined route. Devices consistently received AR content without interruption throughout the drive.

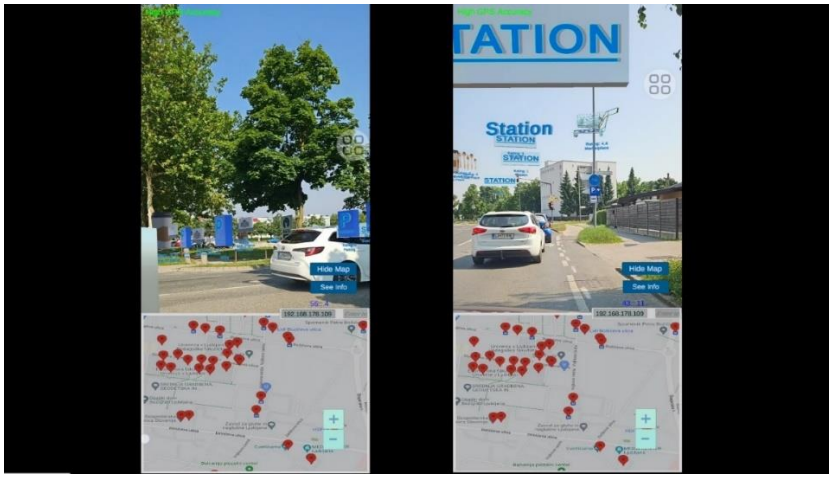


FIGURE 17. AR CONTENT ON AR DEVICES IN CYCLE B DEMONSTRATION.

This demonstration also validated the use of the Virtualized Cache and RL scheduler under real network load, proving the system could deliver content smoothly and equitably to multiple users in parallel. The AR content appeared on both devices with no service failures, even as network conditions changed during movement.

Throughout validation and demonstrations, the following KPIs were measured and recorded:

- **5G Latency**: Low latency was consistently maintained during both stationary and mobile use, enabling smooth AR rendering.
- **End-to-End (E2E) Latency**: Demonstrated low and stable E2E latency across different test environments.
- **E2E Reliability**: No content dropouts were observed in final validation tests, including under concurrent use scenarios.
- **E2E User Data Rate**: Bandwidth was sufficient to support real-time AR streaming to multiple devices.
- **Service Availability**: The AR pipeline maintained full availability during all tested trips.
- **Service Deployment Time**: Quick and repeatable deployment of the full service chain via the 5G-IANA Composer GUI.

3.3.3.4 Long-Term Benefits of OAEP

UC4 clearly demonstrates the long-term value of the OAEP in enabling modular, scalable, and intelligent service deployment across edge infrastructure. By allowing seamless integration of advanced features like virtualized caching and reinforcement learning-based scheduling, the platform supports high-performance, context-aware AR services that can evolve with user needs, network conditions, and device capabilities. This flexibility is key for scaling AR-driven applications across smart mobility, tourism, and infotainment use cases.

3.4 Cross-Use Case Integrations

3.4.1 UC1/6: Autonomous Driving Incorporates Predictive Network Monitoring

UC1 – Autonomous and Connected Vehicle Navigation integrates with UC6 – Network Status Monitoring to enhance decision-making through predictive Quality of Service (pQoS) awareness. The system uses real-time and historical 5G network performance data to forecast network conditions and adjust driving behavior accordingly.

3.4.1.1 Validation

The integrated UC is validated through a set of scenarios that simulate varying levels of network latency. The UC6 component continuously collects and analyzes network metrics from OBUs and uses a federated learning model to predict short-term QoS trends. UC1 uses these predictions to guide adaptive vehicle responses. The system behavior was evaluated across defined thresholds—ranging from normal to poor connectivity—triggering vehicle-level actions such as speed reduction or stopping, based on latency projections.

The validation confirms that the predictive QoS mechanism is both responsive and accurate enough to support time-sensitive adjustments in autonomous driving logic. Real-time alerts and automated behavior changes are correctly triggered in response to forecasted connectivity issues.

3.4.1.2 Long-Term Benefits of the OAEP

The integration of UC1 and UC6 demonstrates how the OAEP supports intelligent service chaining across domains. The ability to link vehicle control with real-time network analytics enables *future autonomous systems to become more resilient* to *variable coverage* areas and network degradation. This cross-domain interaction, made seamless by the nApp orchestration layer, sets a foundation for broader applications in connected mobility and predictive infrastructure-aware services.

3.4.2 UC5/2: Hazard Detection Informs Cooperative Vehicle Manoeuvres

UC5 – Real-Time Road Network Risk Assessment enhances UC2 – Manoeuvres Coordination for Autonomous Driving by providing behavioral risk insights. UC5 detects hazardous driving behaviors—like mobile device use—and updates vehicle safety scores in real time. These scores are then used by UC2 to inform and adjust manoeuvre decisions at intersections.

3.4.2.1 Validation

The UC is validated through coordinated interactions between physical and simulated vehicles. UC5 uses sensor data and behavioral models to detect risky events, which are relayed to UC2 via a decentralized environmental notification service. UC2 subscribes to these notifications and updates its manoeuvre planner accordingly.

The system correctly identifies driver risk, adjusts internal safety scores, and alters vehicle commands based on updated risk profiles. This proves that behavior-aware manoeuvre planning can be achieved in real time and that the integrated safety logic improves traffic decision-making under dynamic conditions.

3.4.2.2 Long-Term Benefits of the OAEP

This integration highlights how the OAEP enables interoperability between safety analytics and coordination systems. By combining behavior detection with automated planning, the platform supports a shift toward context-aware mobility services. The modular architecture allows for rapid integration of new risk indicators or behavioural models, making the solution extensible for more complex traffic scenarios, including multi-agent coordination and urban congestion handling.

3.4.3 General Long-Term Benefits of the OAEP

Across all UCs and cross-UC integrations, the OAEP proves itself as a powerful enabler for intelligent, distributed applications. Its modular and flexible architecture allows for real-time collaboration between services, seamless data sharing across domains, and dynamic orchestration across edge and vehicle nodes. As smart mobility continues to evolve, the platform provides a robust foundation for deploying scalable, cross-functional services in connected, autonomous, and safety-critical environments.

3.5 Third-Party experiments

This section focuses on the case of external experimenters, with all information presented below stemming from the publicly available deliverable D6.5 [D652024].

3.5.1 VINS-RTK by Link Robotics[3]

3.5.1.1 Use Case Description & Validation Steps

VINS-RTK is a real-time localization system designed to deliver centimeter-level precision in vehicle positioning by fusing data from an Inertial Measurement Unit (IMU), a monocular camera, and an RTK-enabled GNSS receiver. The system outputs localization data at 10Hz with approximately 50ms latency and integrates with 5G infrastructure to ensure robust performance even in environments with degraded GNSS signals.

VINS-RTK targets multiple application domains including autonomous vehicles, fleet logistics, infrastructure inspection, and smart city systems. By combining vision-based tracking, IMU data, and GNSS corrections, it ensures accurate pose estimation critical for safe and efficient autonomous navigation.

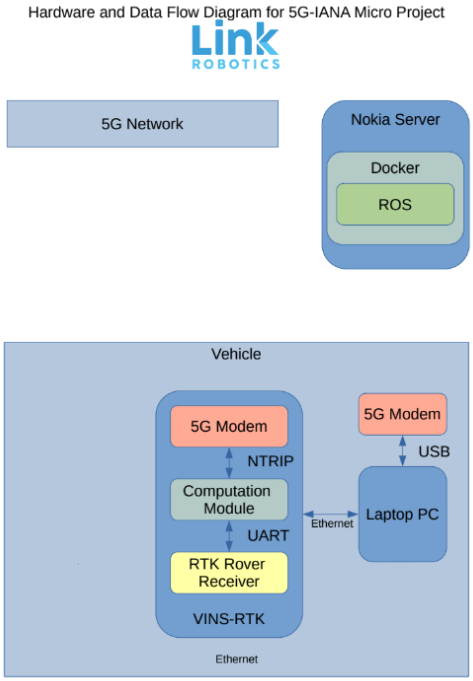


FIGURE 18. VINS-RTK HIGH-LEVEL ARCHITECTURE.

In the context of the 5G-IANA project, VINS-RTK is tested on a robotic platform in a real-world environment near the University of Ulm. The vehicle-mounted system utilizes a Jetson Xavier NX for visual-inertial processing (via ROS Noetic) and connects to Nokia's private 5G GSM network for high-speed data transmission. Testing involves initializing the RTK-GNSS system, calibrating sensor fusion modules, and evaluating accuracy during actual driving scenarios under varying network and GNSS signal conditions.

[3] Website: linkrobotik.com

3.5.1.2 Platform Requirements

To support its advanced real-time localization capabilities, VINS-RTK leverages several key platform features provided by the 5G-IANA OAEP. **High-bandwidth** 5G connectivity enables low-latency transmission of high-frequency sensor data to cloud services, while edge computing capabilities allow real-time execution of MSCKF algorithms on Jetson hardware. **Secure and resilient network functions** ensure a stable RTCM correction data flow over the NTRIP protocol, even in areas with poor signal quality. The platform supports **flexible**, container-based deployment using **Docker**, allowing for efficient cloud-side data access and management, and seamless integration with other mobility services. Compatibility with **ROS Noetic** enables modular sensor fusion and future extensibility, while **cross-layer coordination** between the edge, sensor software, and network functions supports automatic synchronization, data fusion, and recovery processes.

3.5.1.3 Validation

The VINS-RTK system is validated through field trials involving two experiments using a Nokia test vehicle. These trials assess the system's ability to maintain accurate localization under dynamic conditions, including partial or full loss of GNSS or 5G connectivity.

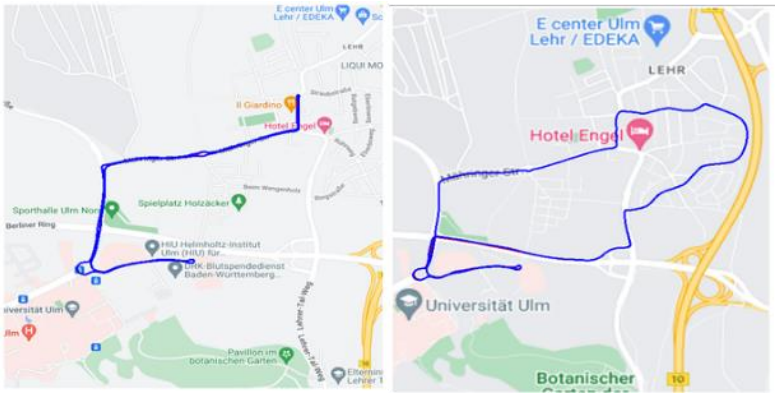


FIGURE 19. DRIVEN PATHS IN TWO EXPERIMENTS.

Key validation steps included:

- Sensor calibration using GUI interfaces with <10 cm covariance threshold.
- Real-time pose estimation under varying 5G signal strength and GNSS satellite visibility.
- Performance comparisons between RTK-Fix, RTK-Float, and GNSS-Fix operational modes.
- Continuous monitoring of localization update rate, latency, and network reconnection success.

Results show that the system consistently delivers centimeter-level accuracy in RTK-Fix mode and maintains below 1-meter accuracy in fallback modes. Despite interruptions in 5G or GNSS signals, the integrated fusion pipeline ensures continuous position and orientation output. Automatic network recovery mechanisms via the NF component reestablish RTCM correction links, preserving service continuity.

Data is logged and transmitted to Nokia cloud servers in real time, enabling further post-processing and remote evaluation.

3.5.1.4 Long-Term Benefits of the OAEP

Integration with the 5G-IANA OAEP offers several long-term advantages for VINS-RTK. Scalability is achieved through the use of Docker containers and an edge-native architecture, enabling seamless expansion of the localization service across fleets or geographic regions. Reliability is enhanced by the NF component, which ensures robust connectivity through automatic reconnection, even under fluctuating network conditions. The system's modular software stack supports interoperability, allowing future extensions such as Digital Twin generation or LIDAR integration. Real-time responsiveness is improved by deploying edge-heavy workloads—such as MSCKF algorithms—close to data sources, enhancing performance in latency-sensitive navigation scenarios. Additionally, cloud integration enables seamless access to processed localization data, supporting advanced applications like remote path monitoring, fleet tracking dashboards, and predictive analytics. Together, these capabilities position VINS-RTK as a forward-compatible solution for next-generation autonomous mobility systems, leveraging OAEP's orchestration and deployment features to ensure high performance and reliability at scale.

3.5.2 Cooperative Localization & Crossview Geolocalization by AViSense.AI[4]

3.5.2.1 Use Case Description & Validation Steps

Cooperative Localization & Crossview Geolocalization are advanced localization methods developed to enhance situational awareness in Connected and Automated Vehicles (CAVs).

1. **Cooperative Localization** enables 4D situational awareness by fusing multimodal data—position, distance, azimuth, and inclination—among vehicle swarms modeled as undirected graphs. Through real-time communication over 5G and the use of the Graph Laplacian operator, the system avoids iterative exchanges and maintains accuracy while minimizing latency and computational overhead. Designed for scalability, it extends to various platforms such as indoor robots, underwater vehicles, and aerial drones.

[4] https://avisense.ai/.

2. **Crossview Geolocalization** estimates the location of a vehicle-mounted camera by matching ground-view images with satellite maps. It uses deep learning for pose estimation, improving upon traditional image retrieval approaches by integrating geometry projection and differentiable optimization to refine accuracy.

At the Telekom Slovenije testbed, both cases were validated using simulated CARLA-based data and cloud-assisted localization frameworks. The system comprises virtual OBUs and remote cloud infrastructure, with real-time communication between sensor-equipped vehicles and processing units.

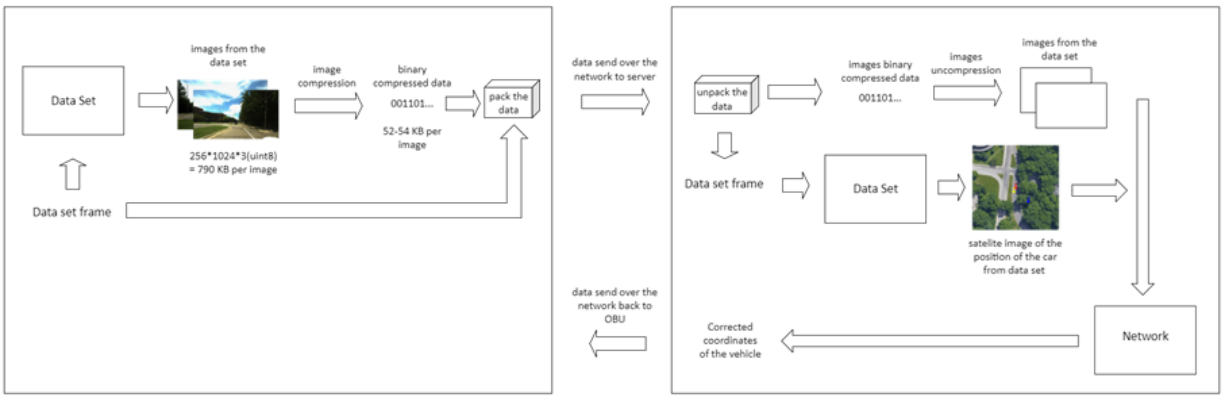


FIGURE 20. SYSTEM OVERVIEW.

3.5.2.2 Platform Requirements

The integration of these use cases into the 5G-IANA ecosystem requires ultra-low latency 5G connectivity to support real-time V2V and V2I communication. It also depends on seamless edge-cloud integration to balance real-time onboard processing with more intensive remote computation. **Dockerized** components enable deployment flexibility across both ARM-based and GPU-enabled systems, while high-throughput **eMBB** slicing is essential for the continuous transmission of high-resolution imagery and sensor data. Compatibility with ROS ensures standardized robotic middleware integration, and **MQTT-to-ROS** communication bridges are implemented to support CAM transmission.

3.5.2.3 Validation

The systems are validated at the Telekom Slovenije testbed using both edge-based deployment and remote cloud processing.

In the **Cooperative Localization** scenario, simulated two-vehicle scenarios are executed using CARLA-generated LiDAR, GNSS, and IMU data. ROS-based virtual OBUs process the sensor streams through LiDAR odometry, object detection and tracking, and cooperative localization modules. V2V communication is managed using MQTT bridged into ROS, enabling real-time message exchange.

In the **Crossview Geolocalization** scenario, ground-view images captured from OBUs are compressed and converted into binary packets to ensure efficient 5G transmission. These images are received, decompressed, and matched against satellite maps in the cloud using deep neural networks. A differentiable pose estimation pipeline refines the location and transmits corrected coordinates back to the vehicle. Performance metrics across both use cases confirm real-time localization updates at ~10Hz, pose estimation corrections with minimal latency via 5G uplink (100 Mbps), an average RTT of 15ms to the Edge server, and resilience and precision even in the absence of GPU acceleration within the testbed.These validations demonstrate successful integration with 5G infrastructure and real-world operational readiness of the localization frameworks.

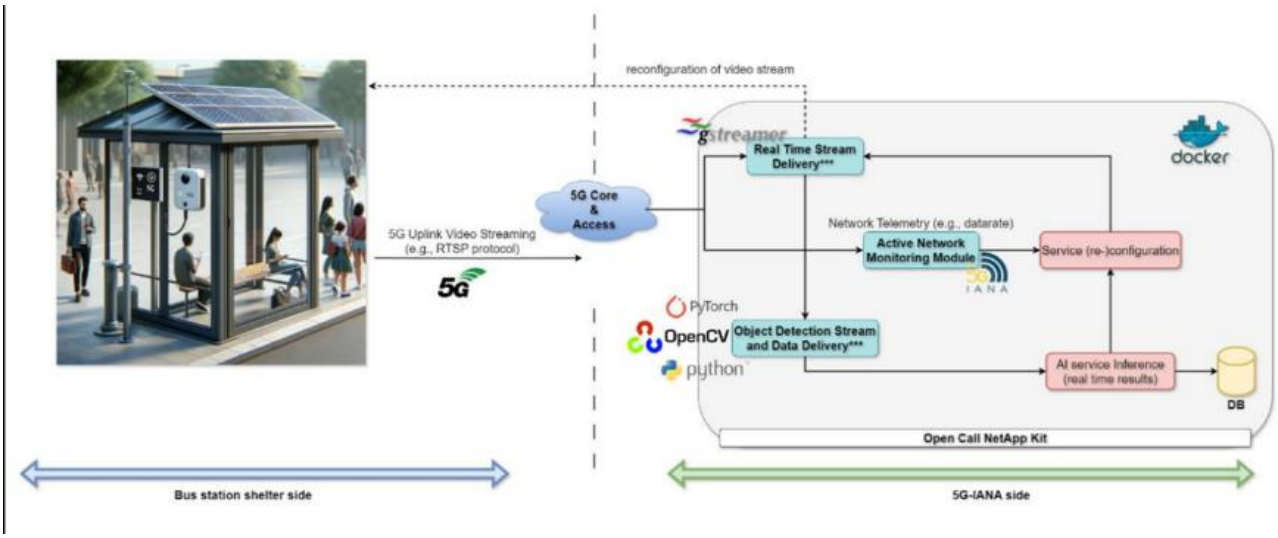
3.5.2.4 Long-Term Benefits of the OAEP

OAEP's technical and strategic guidance enables the seamless deployment and evaluation of these localization solutions, accelerating their transition from research prototypes to commercial products. By leveraging the capabilities of OAEP and 5G-IANA, AviSense achieves several key advancements. The modular, containerized components improve scalability, allowing deployment across diverse platforms and use cases. Real-time responsiveness is enhanced through high-frequency, low-latency communication, supporting mission-critical decision-making in Connected and Automated Vehicle (CAV) environments. Market readiness is also advanced, as validation within 5G-IANA testbeds elevates core technologies from TRL4 to TRL5, positioning them for real-world adoption. Furthermore, integration with 5G infrastructures and cloud services enhances commercial viability, opening new opportunities in fleet management, smart mobility, and infrastructure inspection. Collectively, these developments strengthen AviSense's business positioning as a key player in sustainable transportation and autonomous systems.

3.5.3 5G-STREAM by Diamantis[5]

3.5.3.1 Use Case Description & Validation Steps

5G-STREAM introduces a next-generation connected surveillance solution tailored for smart cities and public transportation systems. The use case centers around dual-component architecture that includes a cloud-based nApp with AI-powered video analytics and a client-side application running on a solar-powered, battery-equipped camera unit. This edge device, referred to as the SB-CAM, features integrated 5G connectivity and is designed for standalone deployment on public infrastructure such as bus shelters and lamp posts. Its core function is to capture and stream video data in real time for cloud-based analysis, initially focusing on detection of people at bus stops.



The system is built to be both sustainable and scalable, enabling installation in off-grid urban locations. Its architecture supports flexible video processing configurations: AI inference can be run locally on the SB-CAM or offloaded to the cloud via the 5G-IANA platform. Real-world testing was conducted using Nokia's 5G testbed, which supports both edge and cloud deployments and allowed performance benchmarking of both

[5] https://diamantisgroup.gr/en/home/

processing models. Particular attention was paid to power efficiency, as the project sought to evaluate the benefits of offloading heavy computation from power-constrained edge devices to the cloud.

A distinctive feature of 5G-STREAM is its cross-border implementation. The end-point device was deployed in Greece, while the platform backend operated through the Nokia testbed in another country. This separation was feasible due to the non-latency-sensitive nature of the use case, demonstrating flexibility in remote deployment scenarios.

3.5.3.2 Platform Requirements

To operate effectively, 5G-STREAM depends on several platform-level features enabled by the OAEP. These include high-throughput 5G connectivity to support video streaming, reliable container orchestration via Docker and Maestro Compose, and

FIGURE 21. HIGH-LEVEL ARCHITECTURE.

access to a cloud-based inference engine capable of processing video streams in real time. The client device, powered by a Raspberry Pi 5, must maintain secure remote communication with the 5G-IANA infrastructure, achieved via a WireGuard VPN.

The system architecture also includes GStreamer servers to manage RTP video streams, and AI models executed in containerized environments. All software components are built for compatibility with Ubuntu 22.04, and the end-to-end deployment is designed to be replicable, lightweight, and adaptable to resource-constrained devices. Processing can be toggled between CPU-only and GPU-accelerated modes depending on the hosting platform, offering further flexibility.

3.5.3.3 Validation

Validation of the 5G-STREAM system involved comparative testing between a fully edge-based processing model and a cloud-offloaded configuration. Both setups were evaluated using a 25-second sample video and assessed for power efficiency, detection accuracy, and system responsiveness. The edge-based scenario ran the full video analytics pipeline locally on Raspberry Pi 5, resulting in approximately 7 watts of power consumption. In contrast, the offloading model transmitted raw video frames over the 5G link to the remote server, where all processing took place. This reduced the power requirement on the edge device to less than 0.7 watts, demonstrating a tenfold energy efficiency improvement.

The containerized AI model used for detection was based on YOLOv8 and adjusted for deployment conditions: a lightweight version was applied on the edge, while a more accurate model was used in the cloud configuration. Both scenarios were tested with the same camera hardware, solar panel, and battery, confirming the reproducibility and robustness of the architecture.

Testing also verified the consistency of the system's operation under real-world conditions. Frame transmission over RTP was stable, the inference pipeline-maintained accuracy, and the infrastructure proved resilient during extended runs. The comparative analysis clearly demonstrated that the cloud-based configuration not only improved energy efficiency but also enhanced detection performance by enabling the use of more advanced AI models.

3.5.3.4 Long-Term Benefits of the OAEP

Through its integration with the OAEP, the 5G-STREAM project establishes a viable pathway toward commercially scalable smart city solutions. The platform enables critical features such as remote device management, containerized deployment, and low-latency cloud offloading, all of which contribute to significant reductions in hardware complexity and operational costs. The use of OAEP has accelerated the maturation of the 5G-STREAM system from a prototype to a market-ready product.

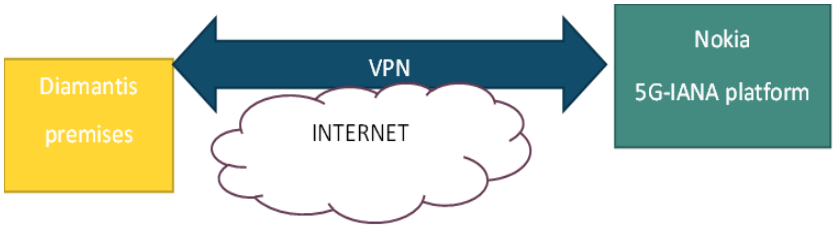


FIGURE 22. CONNECTION TO 5G-IANA BACK-END (THE OAEP)..

From a business perspective, the project has informed a revised productization plan. The current roadmap envisions bundling the solution as a one-time equipment purchase with an optional cloud service, likely to be executed on public infrastructure. The system's ability to integrate with Traffic Management Centers positions it as a valuable tool for dynamic route optimization in public transport. More broadly, it offers opportunities in areas like smart waste management, infrastructure monitoring, and security surveillance.

Despite time constraints caused by delays in the Open Call process and the summer holiday period, the OAEP ecosystem-supported by a responsive consortium—enabled smooth deployment, technical alignment, and rapid iteration. The demonstrated energy savings and reduced hardware demands by the system suggest high potential for commercial adoption in both developed and developing urban markets.

3.5.4 Surface Anomaly Detection System by FERON[6]

3.5.4.1 Use Case Description & Validation Steps

The Surface Anomaly Detection System is a network application focused on enhancing road safety and urban infrastructure management by detecting bumps and potholes in real time. The system is designed to support smart mobility by integrating an OBU equipped with a vehicle motion sensor (LPMS-UTTL2), a GNSS module for location accuracy, and a 5G modem to transmit collected data to the edge. These components work together to monitor road conditions during driving and enable immediate anomaly detection.

This application **leverages 5G-IANA's infrastructure to provide low-latency, high-speed communication necessary for real-time monitoring**. By collecting motion and location data at high frequency—100Hz for the sensor and 5Hz for GNSS—the system can detect surface irregularities as they happen. These measurements are processed at the network edge, and alerts are triggered through the 5G-IANA DENBS system when a potential hazard is detected. The design of the system ensures it can be adapted to a variety of urban contexts and road types, promoting broader adoption in smart city environments.

3.5.4.2 Platform Requirements

The system depends on several key capabilities of the 5G-IANA platform to function effectively. Real-time performance is achieved through the use of a containerized MEC architecture, which includes an MQTT broker for message exchange, Telegraf for data ingestion, InfluxDB for time-series storage, and Grafana for visualization. These services are orchestrated using 5G-IANA's open experimentation tools.

[6] https://feron-tech.com/

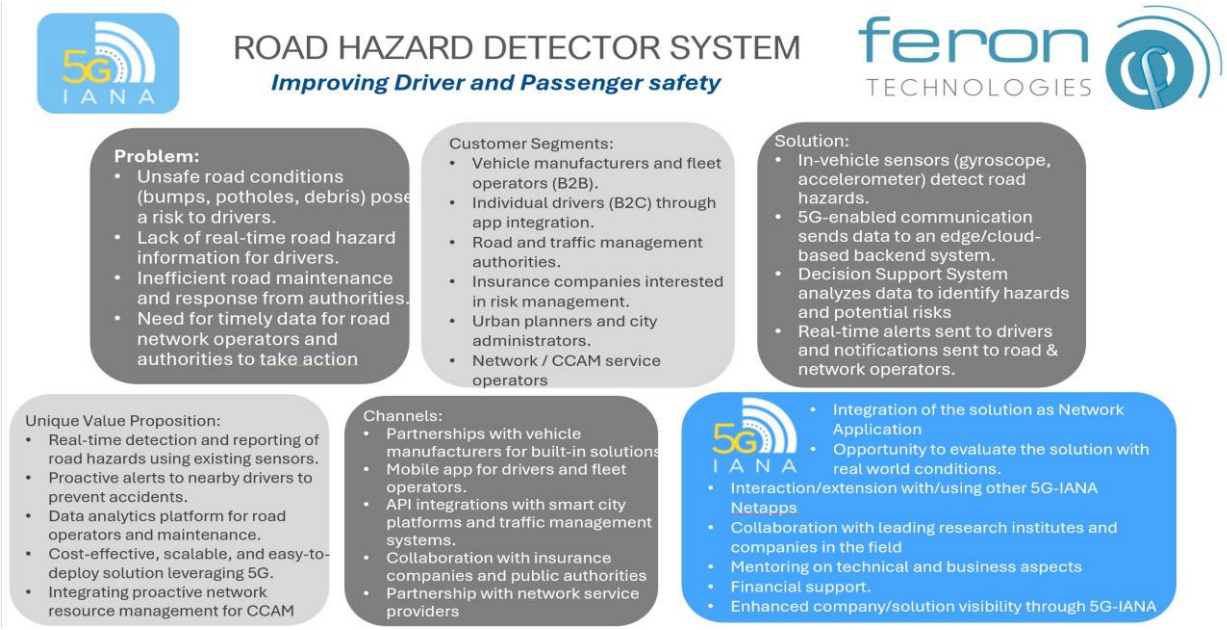


FIGURE 23. OVERVIEW: FROM PROBLEM SPECIFICATION TO OAEP INTEGRATION.

Seamless communication between the OBU and the MEC node relies on reliable 5G uplink throughput and low latency, ensuring immediate data transfer from the vehicle to the analysis engine. The ability to operate at the edge reduces the need for bandwidth-heavy cloud communication, supporting scalable deployments. Additionally, 5G-IANA's orchestration framework facilitates service deployment, testing, and monitoring across geographically distributed testbeds, further enhancing the solution's adaptability.

3.5.4.3 Validation

Validation of the Surface Anomaly Detection System was conducted at Nokia's 5G testbed in Ulm, Germany. The process began with the design and integration of system components, including containerized services and MEC deployment, followed by real-world testing using a specially equipped test vehicle. The OBU was configured to collect and transmit motion and GNSS data via the 5G network, simulating practical urban mobility scenarios.

FIGURE 24. IDENTIFIED ROAD HAZARDS DETECTED DURING DRIVE TESTS USING NOKIA'S 5G-IANA TEST VEHICLE.

Initial tests revealed that road surfaces in the test area were well maintained, resulting in limited natural detections. To verify the system's responsiveness, thresholds were adjusted to detect milder disturbances, and the test vehicle was intentionally driven over pavement edges. This enabled successful validation of real-time anomaly detection capabilities. All data was accurately recorded, visualized, and analyzed, confirming the effectiveness of the detection algorithm and the reliability of 5G-enabled communication.

The integrated use of edge computing further enhanced the system's responsiveness and reduced unnecessary data transmission. The anomaly detection mechanism, combining threshold-based and unsupervised learning models, proved effective in minimizing false positives. The user interface provided intuitive visualization, supporting proactive intervention by city planners and road maintenance teams.

3.5.4.4 Long-Term Benefits of the OAEP

The integration with 5G-IANA's Open Application Experimentation Platform provided FERON with a robust foundation to test, validate, and improve its road monitoring solution. Access to advanced infrastructure and dedicated support accelerated the development process and strengthened the system's potential for market scalability. Real-world validation not only proved the system's technical feasibility but also helped refine its business positioning.

The collaboration with Nokia's testbed team and OAEP orchestrators enabled high-quality data collection, seamless deployment, and real-time analysis. As a result, the system has matured into a practical tool for urban infrastructure monitoring, capable of operating reliably in demanding real-world conditions.

3.5.5 Overview of Local intelligent Traffic Orchestrator (LiTO)

3.5.5.1 Use Case Description & Validation Steps

The LiTO Real-Time Advisory System is designed to improve traffic flow and road safety by enabling connected vehicles to receive live speed advisories based on the movement of nearby buses. In this use case, the system calculates optimal vehicle speeds as a bus departs from a stop, allowing trailing vehicles to adjust accordingly. This smoothens driving behavior, minimizes abrupt braking, and increases overall traffic efficiency.

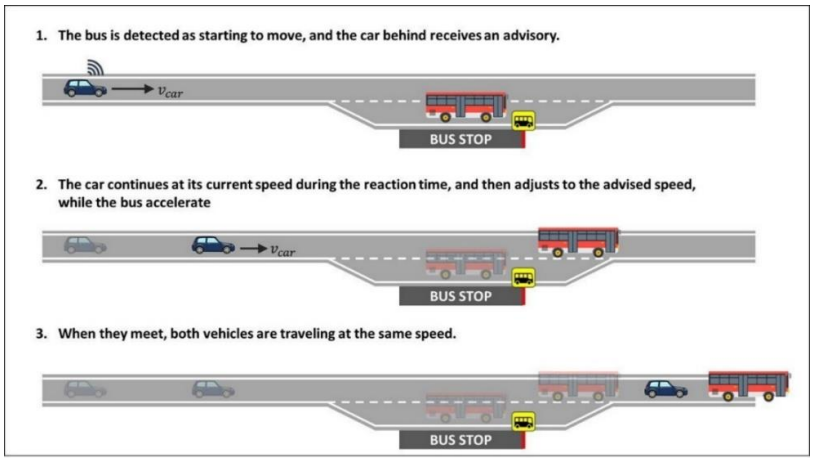


FIGURE 25. A CONNECTED CAR USES REAL-TIME SPEED ADVISORIES VIA 5G TELEMETRY TO SYNCHRONIZE WITH A BUS'S ACCELERATION, ENSURING SMOOTH TRAFFIC FLOW AND COORDINATED DRIVING.

The system operates through real-time telemetry collected from both vehicles and roadside infrastructure. Android-based devices function as Onboard and Roadside Units (OBU and RSU), transmitting data to a virtual machine hosted at the network edge. This VM runs a Python-based server that processes telemetry using the LiTO algorithm and issues speed advisories back to the vehicles. Visual and audible notifications are presented to the driver via a native Android app.

Testing took place at the Telekom Slovenije testbed, which offered consistent 5G connectivity, flexibility in scheduling, and minimal restrictions on operational setup. The end-to-end solution demonstrated how vehicle telemetry and advisory logic could be effectively processed in real time within a 5G-enabled environment.

3.5.5.2 Platform Requirements

To support its real-time functionality, the LiTO system required an edge-computing infrastructure that could process telemetry data and generate speed advisories with minimal latency. The 5G-IANA platform offered a flexible testbed environment, but certain limitations led to the use of a standalone Linux VM for demonstration purposes. This VM acted as a MEC node, hosting the core application server and managing communication with connected devices. Stable, low-latency 5G connectivity was provided by Telekom Slovenije, with all vehicles and infrastructure components equipped with dedicated SIM cards. The solution also required the integration of Android applications to function as mobile OBUs and RSUs, and a reliable software environment to process and visualize advisory data.

Although container deployment within the OAEP was attempted, platform constraints led to the use of direct VM-based hosting as a more reliable alternative during the test period.

3.5.5.3 Validation

Validation of the LiTO system centered on evaluating latency, communication reliability, and advisory effectiveness. During tests, vehicles equipped with OBUs transmitted their telemetry data to the VM-based MEC, which processed the information in real time and returned calculated speed advisories. The system successfully detected the proximity and speed of a bus, using this data to advise trailing vehicles on speed adjustments to ensure smooth traffic flow.

FIGURE 26. NATIVE APP FOR REAL-TIME TELEMETRY DATA TRANSMISSION AND ADVISORY RECEPTION. ADVISORIES WERE DISPLAYED VISUALLY AND AUDIBLY ON DRIVER INTERFACES.

One key scenario demonstrated the issuance of a speed reduction advisory to a car approaching a decelerating bus. At a precise moment, a 35 km/h advisory was transmitted and received, allowing the driver to adapt accordingly. The system consistently demonstrated low-latency operation, with average end-to-end delay measured at 59 milliseconds, and calculation time for advisories taking only a few milliseconds more.

Network performance remained stable throughout testing, with uninterrupted 5G coverage supporting continuous data exchange between OBUs, RSUs, and the edge

server. Advisories were timely and context-aware, enhancing both safety and driving comfort.

3.5.5.4 Long-Term Benefits of the OAEP

Participation in the 5G-IANA program offered valuable strategic and technical insights that have significantly shaped future development plans for the LiTO system. The program facilitated business modeling through tools like SWOT analysis and the Business Model Canvas, and provided coaching sessions that helped establish clear deliverables and market positioning strategies.

Despite some testbed limitations that required workaround solutions—such as using a VM in place of a containerized deployment—the OAEP still provided critical infrastructure and coordination support. This allowed the project to validate its core functionality under realistic network conditions and to prove its operational viability in a connected vehicle setting. Future efforts will focus on expanding LiTO's deployment and leveraging OAEP's evolving capabilities to refine container deployment, improve portability, and scale across different mobility scenarios.

4 Conclusion and Recommendations for Next Steps

4.1 Summary of the 5G IANA platform's contributions.

The 5G-IANA project resulted in the delivery and extensive validation under real conditions of a sophisticated open experimental platform, namely the AOEP, targeting the automotive sector and designed to significantly lower the barriers for developing, deploying, and managing innovative Network Applications (nApps) within the complex 5G automotive landscape. By integrating advanced orchestration, dynamic resource management, and specialized edge computing capabilities, the platform provides fertile ground for experimentation and aims to accelerate the delivery of next-generation connected mobility services.

The key technical functionalities implemented and supported by the 5G-IANA platform include:

nApp Development and Lifecycle Management:

- Simplified development is enabled by the nApp Toolkit, which streamlines nApp design, packaging, and onboarding.
- Advanced orchestration is handled by the NOD layer, offering features like fractional resource allocation, port persistency during updates, and robust policy-driven lifecycle control.

Network Slicing and Resource Orchestration:

- The SM-RO layer provides dynamic network slice management, including a critical Slice Update feature that supports Day 2 adaptability.
- A dedicated Web GUI simplifies slice and resource management for operators.

Distributed Machine Learning Orchestration (DMLO):

- This framework allows for efficient deployment of applications like Federated Learning on resource-constrained far-edge devices such as OBUs.
- It supports intelligent client selection and full lifecycle control directly at the extreme edge.

Monitoring and Analytics:

- The platform offers comprehensive, system-wide monitoring with detailed metrics from infrastructure, nApps, and network slices.

- These analytics support smarter orchestration decisions and help with performance tuning and analysis.

Virtualized Infrastructure and Edge Support:

- Support for flexible, standardized infrastructure includes on-vehicle MANO through MicroK8s.
- Essential services for localization and real-time edge resource monitoring are also included to support dynamic applications.

CI/CD and Software Management:

- A GitLab-based environment powers automated CI/CD pipelines for both platform components and nApps.
- This ensures consistent, efficient, and scalable software development and deployment across the platform.

4.2 Technical Challenges and Lessons Learned

4.2.1 Importance of Standardized Interfaces for Easing nApp Deployment

The 5G-IANA project highlighted the critical role of standardized interfaces in simplifying the deployment and fostering interoperability of Network Applications (nApps). While the platform enabled nApp deployment across different testbeds, experiences from cross-project integrations, such as the ICT-41 PlugFest, underscored that porting nApps often required adaptations to deployment descriptors due to variances in orchestration tools. The project's contribution to ETSI Technical Reports (e.g., TR 104 074 on validation methodologies) signifies a step towards broader standardization, a lesson learned being that continued efforts in defining common interfaces and descriptor translation mechanisms are essential for achieving true plug-and-play nApp deployment across diverse 5G platforms.

4.2.2 Significance of a Common Repository for nApps

The development and utilization of a common repository for nApps and their constituent atomic components (AFs/NFs) within 5G-IANA proved to be highly significant for easing design and promoting reusability. The nApp Toolkit, with its Network Application Catalogue, provided a centralized point for managing and discovering these assets.

Furthermore, the initiative to integrate and share nApps with repositories from other ICT-41 projects emphasized the value of a broader, shared ecosystem. A key lesson is that such common repositories significantly reduce development overhead, enhance visibility, and encourage collaborative innovation by allowing developers to leverage pre-validated components.

4.2.3 Significance of CI/CD Process in Platform Deployment Repeatability

The adoption of a Continuous Integration and Continuous Deployment (CI/CD) process was fundamental to achieving repeatable and reliable deployments of both the 5G-IANA platform components and the nApps themselves. By automating the build, testing (including triggering the Test Automation Framework), and deployment of containerized applications through a GitLab-based DevOps pipeline, the project ensured consistency across different environments and development iterations. This automated workflow minimized manual errors, accelerated the onboarding of new features and nApps, and provided a stable foundation for the extensive validation activities conducted in the project's diverse testbeds.

4.2.4 Work Towards Supporting Demanding nApps (e.g., Deterministic Latency)

Addressing the stringent requirements of demanding nApps, particularly those needing deterministic latency for safety-critical automotive functions, was an ongoing focus and key learning area. 5G-IANA approached this by enabling nApp/AF deployment closer to the user at the far-edge (OBUs/RSUs) and edge servers, and by utilizing specialized network slices like V2X with optimized network topologies. The work on predictive Quality of Service (pQoS) within use cases like UC1/UC6 and UC3/UC6 demonstrated network-aware service reactions to varying conditions. The lesson learned is that while significant strides were made, achieving true deterministic, ultra-low latency consistently across wide-area 5G networks remains a complex challenge requiring continued research in network-aware application design, resource orchestration, and potentially new 6G capabilities.

4.3 Feedback by Stakeholders

4.3.1 Deployment infrastructures

4.3.1.1 NOKIA

Nokia operated a 5G SA testbed[7] in Ulm, Germany, including four outdoor antenna sites and, in the second half of the project, an indoor antenna site was added to streamline remote testing within Nokia's laboratory. Two network slices—eMBB and V2X—were configured according to 3GPP TS 23.501 [3GP2023] requirements. A 5G-IANA OAEP Server and a MEC server were connected directly to the 5G network. All partners were provided with VPN access and SIM cards, enabling both remote and on-site testing.

The indoor antenna site proved especially useful, *offering a controlled environment* for repeated testing and debugging. Nokia supported partners by handling tasks such as connecting different UEs for radio validation and manually resetting equipment following software issues. These capabilities were crucial, as multiple testing iterations were typically required to ensure reliable system behavior. Once stability was achieved indoors, subsequent outdoor demonstrations—while more complex and time-consuming—tended to proceed successfully.

Because sensor and video data were transmitted via the 5G infrastructure and processed on testbed servers, Nokia emphasized the need for *strict compliance with GDPR* and national data protection regulations. No use case could proceed without meeting these legal requirements.

One *key challenge involved radio frequency constraints*. Nokia does not own dedicated test frequencies and instead operates under temporary agreements with MNOs, using bands not currently deployed commercially. However, growing spectrum demand by MNOs has limited the available range. Toward the end of the project, testing was restricted to Band 38 (TDD, 2585–2615 MHz), which offers a limited cell range of about 1.5 km and reduced capacity. These limitations were seen by partners and third parties as a barrier to scaling up or extending use case tests.

[7] For technical details of the NOKIA testbed, the reader may refer to Section 2.3.1.1.

4.3.1.2 Telecom Slovenije[8]

Telekom Slovenije gained deep insights into the technical and operational requirements of third-party experimenters using its testbed. A key learning emerged around the *environment constraints* of testing scenarios. Initially, it was assumed that the indoor garage environment would be suitable for vehicle-based tests. However, this proved unfeasible due to the lack of GNSS signal indoors, which is essential for many mobility-focused use cases. This misalignment prompted a significant unplanned effort to extend 5G coverage outdoors, as the original project scope had anticipated only UC7 being executed at the TS testbed.

To support expanded testing, TS conducted a *detailed campaign of signal measurements* to precisely map the outdoor 5G coverage limits and identify zones with elevated latency. This process revealed that there were use cases requiring *unusually high latency 5G conditions*, achievable only at the very edge of the cell where coverage was weakest. Accommodating such a requirement remains a technical challenge, as the current testbed setup is not designed to maintain reliable performance beyond certain latency thresholds in those fringe areas.

Another major hurdle was *the acquisition of a 5G SA core*. As commercial options were not readily available at the time, Telekom Slovenije had to source an experimental 5G SA core from a local supplier in order to enable slicing functionality within the testbed. While this solution enabled continued progress, it highlights a broader market readiness issue for core network components critical to advanced 5G experimentation.

These experiences collectively underscored the importance of *flexibility in testbed planning and the need for early alignment* between testbed capabilities and experimenter needs—particularly when dealing with mobility, GNSS dependencies, and advanced slicing requirements.

4.3.2 Experimenters

Experimenters have shared their feedback in their individual reports after participating in two rounds of open call. While detailed descriptions can be found in D6.5 [D652024],

[8] For technical details of the Telecom Slovenije testbed, the reader may refer to Section 2.3.1.2.

here we summarise the feedback by Link Robotics, RFSAT, Level 7, Diamantis, FERON, RoadsAI, and AviSense, which engaged with the 5G-IANA platform to validate, refine, and position their respective solutions in real-world automotive and networked environments. Link Robotics used the platform for real-time localisation via VINS-RTK. RFSAT explored early-stage deployment feasibility for IoT and satcom-linked applications, while Level 7 aimed to adapt its Kaleidoscope architecture to 5G/6G network conditions. Diamantis faced integration under reduced timelines following Open Call delays, and RoadsAI attempted to leverage the AOEP but was constrained by technical limitations. FERON executed a successful demo on the Nokia testbed with full support. AviSense used the infrastructure to advance the maturity of key technologies, including cooperative localisation, cross-view geo-localisation, and multimodal data fusion.

All feedback converges towards 5G-IANA platform enabling meaningful experimentation across a spectrum of technical readiness levels and application types. From low-latency sensor fusion to cybersecurity and early IoT deployment, participants validated key aspects of their technologies. In a nutshell, FERON and AviSense reported productive testbed engagements, while Link Robotics and Diamantis demonstrated the potential of real-time data pipelines. Further feedback by RoadsAI and AviSense stressed the need for enhanced onboarding processes, better container deployment support, and increased testbed flexibility. Continued improvements in AOEP usability and self-service functionalities were also identified as essential to broadening third-party adoption and maximising platform impact.

The following subsections present a structured breakdown of this feedback, organised under three key dimensions: (i) Integration and Experimental Setup, (ii) Performance and Resource Considerations, and (iii) AI/ML Relevance and Expected Outcomes.

4.3.2.1 Integration and Experimental Setup

Integration approaches generally varied, with Link Robotics streaming VINS-RTK output to Nokia servers using Docker-based infrastructure whereas others like RFSAT deployed a basic OBU-RSU setup, and Level 7 focused on network traffic data capture. Diamantis utilised the cloud-native capabilities of the AOEP, while FERON benefited from a stable testbed and responsive orchestration. RoadsAI faced difficulties in container deployment and switched to a Linux VM to maintain project timelines. AviSense relied heavily on UBITECH for deploying and running their system in the testbed, citing the need for an expanded familiarisation period to overcome challenges even though eventually they succeeded in executing their experiments.

4.3.2.2 Performance and Resource Considerations

Stakeholders presented diverse technical needs. Link Robotics and AviSense required low-latency, high-reliability performance for real-time localisation and sensor fusion, with AviSense progressing from TRL4 to TRL5 during their trials. RFSAT's throughput demands varied depending on data type, while Level 7's infrastructure tests required stable monitoring environments. FERON and AviSense reported successful demonstrations, though the latter highlighted a steep learning curve with the platform. RoadsAI encountered constraints that prevented full AOEP integration, underlining the need for improved container support and documentation.

4.3.2.3 AI/ML Relevance and Expected Outcomes

AI and ML were critical to Link Robotics and AviSense. The latter used advanced data fusion for situational awareness and safety-critical functions. RoadsAI, while not elaborating on AI directly, addressed real-time advisory systems. RFSAT and L
evel 7 recognised the future potential of AI/ML, though their experiments remained integration-focused. AviSense, in particular, highlighted the value of rigorous testing for product reliability, safety, and commercial positioning.ll

4.4 Recommendations

Based on feedback collected from the external experimenters, internal use case validation, and technical insights from the OAEP design, development, and deployment teams, several *forward-looking recommendations* emerge for the evolution of 5G-enabled experimentation platforms based on nApps.

First and foremost, there is a need to converge to and then adherence to **open standards for nApp interfaces**. This must be prioritised to ensure *seamless interoperability across different infrastructure domains* and to facilitate *integration by third-parties*. As commonly achieved via standardisation, it shall lower any entry barrier for SMEs and independent developers. Not only that, but also standardisation enhances *portability*, enabling nApps to be reused and adapted *across verticals* and deployment environments with minimal rework.

Furthermore, the **role of platforms like OAEP in 5G-IANA** is foundational in the transition towards 6G, especially in *natively* supporting *distributed intelligence*, trusted edge services, and ultra-reliable low-latency communications. These platforms can "incubate" future-ready services via providing real-world conditions for the validation of

advanced functionalities such as automated mobility and predictive QoS-based proactive actions allowing to guarantee stringent SLAs and make networked applications resilient. As such, the creation and curation of *modular*, *reusable nApps* is essential.

In addition, establishing shared repositories and promoting open experimentation frameworks will foster innovation and *reduce redundancy* across service/application domains.

Lastly, **future iterations of the platform** should expand their validation scope by supporting *experimentation in diverse real-world automotive contexts*, including both urban and rural scenarios. This will ensure that the underlying infrastructure, orchestration mechanisms, and nApps are pragmatically tested against a broader range of performance constraints and service conditions. By combining technical excellence with strategic openness, stakeholder-driven development in platforms like 5G-IANA can enable scalable connected mobility ecosystems in 6G.

5 References

[3GP2023] 3rd Generation Partnership Project (3GPP), Technical Specification Group Services and System Aspects; System Architecture for the 5G System (5GS); Stage 2, 3GPP TS 23.501, Version 18.2.0, Dec. 2023. [Online]. Available: https://www.3gpp.org/ftp/Specs/archive/23_series/23.501/

[5GE2022] 5G-ERA Consortium, "D4.3 Reference NetApps for 5G-ERA Initial Version,2022. [Online]. Available: https://5g-era.eu/wp-content/uploads/2022/07/D4.3-Reference-NetApps-for-5G-ERA-Initial-version.pdf

[5GE2024] 5G-ERA Consortium, D4.4 Reference NetApps for 5G-ERA Final Version,2024. [Online]. Available: https://5g-era.eu/wp-content/uploads/2024/09/D4.4-5G-ERA-Reference-NetApps-for-5G-ERA-final-version.pdf

[5GP2022] 5G-PPP Software Network Working Group, Network Applications: Opening up 5G and Beyond Networks,2022. [Online]. Available: https://zenodo.org/records/7123919

[5GP2023] 5G-PPP Software Network Working Group, Network Applications: Opening up 5G and Beyond networks, V2.0, 2023. DOI: 10.5281/zenodo.8134428

[AUT2025] Automotive Technology. "The Role of 5G in Advancing Connected Vehicle Ecosystems." Automotive Technology, 2025. [Online]. Available: https://www.automotive-technology.com/articles/the-role-of-5g-in-advancing-connected-vehicle-ecosystems

[D522023] 5G-IANA various authors, 5G-IANA - D5.1 - Initial validation KPIs and metrics, Zenodo, Jul. 2023. DOI: https://doi.org/10.5281/zenodo.1003380

[D522024] FSCOM, 5G-IANA - D5.2 - Validation methodology - revision, Zenodo, Jan. 2024. DOI: https://doi.org/10.5281/zenodo.10533956

[D632024] ICCS, UBITECH and Incites, ‘D6.3 5G-IANA micro-projects integration report’, Zenodo, Jan. 2024. DOI: 10.5281/zenodo.10533884

[D652024] ICCS, UBITECH, Nextworks and Incites, ‘5G-IANA microprojects services report’, Dec. 2024.

[ERI2025] Ericsson, Three insights into how 5G innovations are transforming the enterprise today, Ericsson Online, 2025. [Online]. Available: https://www.ericsson.com/en/blog/2025/2/three-insights-into-how-5g-innovations-are-transforming-the-enterprise-today

[EVO2022] EVOLVED-5G Consortium, Overall Framework for NetApp Development and Evaluation,2022. [Online]. Available: https://evolved-5g.eu/wp-content/uploads/2022/12/EVOLVED-5G_D2.3.pdf

[GAL2022] J. Gallego-Madrid, R. Snchez-Iborra, and A. Skarmeta, From Network Functions to NetApps: The 5GASP Methodology, Computers, Materials & Continua, vol. 71, no. 2, pp. 4115-4134, 2022. DOI: https://doi.org/10.32604/cmc.2022.021754

[NOK2023] Nokia. "5G Steering the Future of Connected Cars." Nokia, 2023. [Online]. Available: https://www.nokia.com/about-us/newsroom/articles/5g-steering-the-future-of-connected-cars/

[SNS2024] Smart Networks and Services Joint Undertaking (SNS JU), Reliable Software Networks Working Group (SoftNet WG), *Network and Service Management Advancements: Key Frameworks and Interfaces towards Open, Intelligent and Reliable 6G Networks*, Version 1.0, Dec. 2024. DOI: https://doi.org/10.5281/zenodo.14234897

[TRI2025] K. Trichias, S. Col, I. Masmanidis, A. Berisha, F. Setaki, P. Demestichas, S. Papavassiliou, and N. Mitrou, 5G for connected and automated mobility Network level evaluation on real neighboring 5G networks: The Greece-Turkey cross border corridor, Computer Communications, vol. 232, p. 108047, Feb. 2025.

[WEF2025] World Economic Forum. " Why 5G mobile networks will unlock advanced digital transformation." World Economic Forum, 2025. [Online]. Available: https://www.weforum.org/stories/2025/01/5g-mobile-networks-digital-transformation/

[ZEN2022] 5G-IANA Consortium, 5G-IANA: 5G Intelligent Automotive Network Applications, Zenodo, 2022. DOI: https://doi.org/10.5281/zenodo.7304862

Acknowledgement

The 5G-IANA project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 101016427.