EXPERIMENTATION AND VALIDATION OPENNESS FOR LONGTERM EVOLUTION OF VERTICAL INDUSTRIES IN 5G ERA AND BEYOND

[H2020 - Grant Agreement No.101016608]

Deliverable D4.1

5G Exposure Capabilities for Vertical Applications (Intermediate)

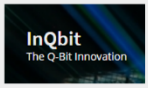
Editor G. Makropoulos (NCSRD)

Contributors (TID), (NCSRD), (MAG), (ATOS), (INTRA), (LNV), (IMM), (GMI), (INF), (CAF), (ININ), (UMA), (ZORT), (CSIC), (8BELLS), (FOGUS), (IQBT), (PAL), (UML)

Version 2.0

Date 28th February, 2021

Distribution PUBLIC (PU)



DISCLAIMER

REVISION HISTORY

Revision	*Date*	*Responsible*	*Comment*
0.2	*Month, Day, Year*	*G. Makropoulos*	*Edit ToC*
0.3	*Month, Day, Year*	*G. Makropoulos*	*Initial Contributions*
0.4	*January, 23rd, 2022*	*G. Makropoulos*	*Full draft*
0.6	*January 27th, 2022*	*G. Makropoulos*	*New structure*
0.7	*February 2nd, 2022*	*G. Makropoulos*	*Revised content*
0.9	*February 5th, 2022*	*G. Makropoulos*	*1st draft*
1.2	*February 10th, 2022*	*G. Makropoulos*	*2nd draft*
1.4	*February 18th, 2022*	*G. Makropoulos*	*Internal review*
1.5	*February 22nd, 2022*	*G. Makropoulos*	*SC review, TM review*
1.8	*February 26th, 2022*	*G. Makropoulos*	*Final review*
2.0	*February 28th, 2022*	*G. Makropoulos*	*Final version*

LIST OF AUTHORS

Partner ACRONYM	Partner FULL NAME	Name & Surname
TID	TELEFONICA INVESTIGACION Y DESARROLLO SA	J. Garcia D. Artuñedo
NCSRD	NATIONAL CENTER FOR SCIENTIFIC RESEARCH "DEMOKRITOS"	H. Koumaras, G. Makropoulos, D. Fragkos, A. Gogos
ATOS	ATOS IT SOLUTIONS AND SERVICES IBERIA SL	R. Marco P. Encinar
INTRA	NETCOMPANY INTRASOFT SA	A. Dimitriou
MAG	MAGGIOLI SPA	Y. Karadimas
LNV	Lenovo (Deutschland) GmbH	A. Salkintzis, D. Dimopoulos
IMM	IMMERSION	C. Bailey
GMI	GMI AERO	G. Kanterakis M.O. Sauer
INF	INFOLYSIS P.C.	T. Dounia A. Varkas, C. Sakkas, G. Theodoropoulos
CAF	CAFA TECH OU	T. Jarvet M. Hiiemma
ININ	INTERNET INSTITUTE, COMMUNICATIONS SOLUTIONS AND CONSULTING LTD	L. Koršič, J. Cijan J. Sterle R. Susnik
ZORTENET	ZORTENET P.C.	A. Kourtis, A. Oikonomakis, G. Xilouris
UMA	UNIVERSIDAD DE MALAGA	B.Garcia, R. Lopez, J. Canca
8BELLS	EIGHT BELLS LTD	I.Margaritis
FOGUS	FOGUS INNOVATIONS & SERVICES P.C.	D. Tsolkas, S. Charismiadis
IQBT	INQBIT INNOVATIONS SRL	J. Stylianou
PAL	PAL ROBOTICS SL	T. Peyrucain
UMS	UM AUTONOMOUS SYSTEMS LIMITED	D. Cupello A. Marino

GLOSSARY

Abbreviations/Acronym	*Description*
3GPP	*3rd Generation Partnership Project*
5GC	*5G Core*
5GS	*5G System*
5QI	*5G Quality of Service Identifier*
AF	*Application Function*
AKMA	Authentication and Key Agreement for Applications
API	*Application Programming Interface*
APN	*Access Point Name*
AR	*Augmented Reality*
CAPIF	*Common API Framework*
CI/CD	*Continuous Integration / Continuous Development*
CLI	*Command Line Interface*
DNN	*Data Network Name*
ECRC	*Enhanced Coverage Restriction Control*
eDRX	Extended idle-mode Discontinuous Reception
eMBB	Enhanced Mobile Broadband
EPS	*Evolved Packet System*
FoF	*Factory of the Future*
FIM	*File Integrity Monitoring*
GBR	Guaranteed Bit Rate
GW	*Gateway*
HTTP	*Hypertext Transfer Protocol*
IDE	*Integrated Development Environment*
IIoT	*Industrial Internet of Things*
IMEI	International Mobile Equipment Identity
IMS	IP Multimedia Subsystem
IMSI	*International Mobile Subscriber Identity*
IoT	*Internet of Things*
IPv4	*Internet Protocol version 4*
IPv6	*Internet Protocol version 6*
IP	*Internet Protocol*
KPI	*Key Performance Indicator*
LAN	*Local Area Network*
LCS	*Location Services*
M2M	*Machine to Machine*
MAC	*Media Access Control*
MANO	*Management and Orchestration*
MEC	*Multi Access Edge Computing*

mMTC	*massive Machine Type Communications*
MO-LR	*Mobile Originated Location Requests*
MRO	*Maintenance Repair Operation*
MT-LR	*Mobile Terminated Location Requests*
NAS	*Non-Access Stratum*
NFVI	*Network Function Virtualization Infrastructure*
NIDD	*Non-IP Data Delivery*
NI-LR	*Network Induced Location Requests*
NON-GBR	*Non-Guaranteed Bit Rate*
NPN	*Non-Public Network*
NSA	*Non-Standalone*
NWDAF	*Network Data Analytics Function*
PCC	*Policy Control Charging*
PCF	*Policy Control Function*
PDN	*Packet Data Network*
PDU	*Protocol Data Unit*
PFD	*Packet Flow Descriptor*
PFDF	*Packet Flow Descriptor Function*
PSM	*Power Saving Mode*
QoE	*Quality of Experience*
QoS	*Quality of Service*
RAN	*Radio Access Network*
REST	*REpresentational State Transfer*
SA	*Stand Alone*
SCEF	*Service Capabilities Exposure Function*
SDK	*Software Development Kit*
SFC	*Service Function Chaining*
SIEM	*Security Information and Event Management*
SLS	*Service Level Specification*
SMF	*Session Management Function*
SMS	*Short Message Service*
S-NSSAI	*Single Network Slice Selection Assistance Information*
SQL	*Structured Query Language*
SSH	*Secure Shell*
TSN	*Time Sensitive Networking*
UE	*User Equipment*
UGV	*Unmanned Ground Vehicle*
UMTS	Universal Mobile Telecommunications System
URLLC	Ultra-Reliable Low Latency Communication
USB	Universal Serial Bus
USIM	UMTS Subscriber Identify Module

V2X	*Vehicle to Everything*
VAPP	*Vertical Application*
VPN	*Virtual Private Network*
VR	*Virtual Reality*
WebRTC	*Web Real Time Communication*
XSS	*Cross site scripting*

EXECUTIVE SUMMARY

The purpose of this document is to serve as a reference technical report towards the design and implementation of the EVOLVED-5G NetApps and their related Vertical Applications (vAPPs). The NetApps i) fall under the four Industry 4.0 pillars defined in the project ii) share the main principle of EVOLVED-5G definition for the NetApps, i.e., the utilization of the northbound APIs of the infrastructure. The deliverable is a result of work carried out in Tasks 4.1 *"5G Exposure Capabilities for Vertical Applications Development"* and also in Tasks T4.2-T4.5 (NetApp development tasks). As such, key contribution of the deliverable is the description of the work that has been performed towards the definition and design of the northbound APIs, as well as the development and the initial release of the EVOLVED-5G NetApps.

At project level, the work presented in this deliverable is meant to guide developers towards the next versions of the NetApps, since it provides an initial implementation, design and realization methodology that each NetApp should undergo in order to utilize the provided APIs. The structure of the document will serve as a placeholder for the future deliverables of WP4 (especially, the D4.2 "EVOLVED-5G FoF NetApps", due at M17) in order to best capture the progress during the project lifetime.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

1 INTRODUCTION

1.1 PURPOSE OF THE DOCUMENT

Two of the main objectives of EVOLVED-5G project is the design, development and testing of the NetApps, which fall under the four Industry 4.0 pillars, as well as the design of the business APIs exposed by the non-standalone NetApps that interface with the 5G infrastructure. Thus, the current document *"5G Exposure Capabilities for Vertical Applications"* provides details on the use of the EVOLVED-5G tools, introduced in D3.1 [1], towards the development and testing of the first release (v1.0) of the NetApps supporting the vertical applications. Moreover, the document contributes to the description of the initial version of NetApps following the requirements and specifications introduced in D2.1 [2] as well as the 3rd Generation Partnership Project (3GPP) Application Programmable Interfaces (APIs) that are utilised by the Network Exposure Function (NEF) emulator. The entire implementation work taking place in WP4, and reported in this document, provides valuable input to WP5 (Overall Evaluation process, NetApp Validation, Certification and Release) where NetApp lifecycle phases are actually executed.

1.2 STRUCTURE OF THE DOCUMENT

The core part of the document is divided into the following sections:

- Section 2 "5G EXPOSURE CAPABILITIES" outlines the 5G Exposure capabilities and describes in detail the available services comprising NEF.
- Section 3 "DEVELOPMENT OF THE NETAPPS AND BUSINESS APIs" focuses on presenting the development and verfication tools within the Workspace, which will act as host for the Development and Verification phases, and are utilised by the NetApp developers, leading to a mature version of the NetApps. Moreover, the development workflow for the current version of the NetApps is presented along with the exposure of the business APIs from the vApps.
- Section 4 "NETAPPS AND USE CASES" comprises a dedicated description of the preliminary version (v1.0) of the NetApps within the four Industry 4.0 Pillars EVOLVED-5G brings.

1.3 TARGET AUDIENCE

The release of the deliverable is public, intending to expose the overall EVOLVED-5G ecosystem and NetApps Lifecycle design to a wide variety of research individuals and communities.

From specific to broader, different target audiences for D4.1 are identified as detailed below:

- **Project Consortium**: To validate that all objectives and proposed technological advancements have been analysed and to ensure that, through the proposed NetApp Lifecycle phases and the various Environments, further work can be concretely derived. Furthermore, the deliverable sets to establish a common understanding among the consortium with regards to:
    - The CI/CD approach for the software using the open workspace, developed in WP3, leading to the development of the NetApps
    - The actual development of the NetApps as categorized in four different Industry 4.0 domains following the requirements and specifications introduced in Task 2.2 and Task 3.1.

- **Industry 4.0 and FoF (factories of the future) vertical groups:** To crystallise a common understanding of technologies, and design principles that underline the development of the NetApps, and to understand the utilisation of the network APIs exposed by the 5G Infrastructure. A non-exhaustive list of Industry 4.0-related groups is as follows:
    - Manufacturing industries (including both large and SMEs) and IIoT (Industrial Internet of Things) technology providers.
    - European, national, and regional manufacturing initiatives, including funding programs, 5G-related research projects, public bodies and policy makers.
    - Technology transfer organizations and market-uptake experts, researchers, and individuals.
    - Standardisation Bodies and Open-Source Communities.
    - Industry 4.0 professionals and researchers with technical knowledge and expertise, who have an industrial professional background and work on industry 4.0-related areas.
    - Industry 4.0 Investors and business angels.
- **Telecom Service Providers**: to engage with verticals and also to simplify the way 5G services can be offered to a potential customer or 3rd party service provider.
- **Other vertical industries and groups**: To seek impact on other 5G-enabled vertical industries and groups in the long run. Indeed, all the architectural components of the facility are designed to secure interoperability beyond vendor specific implementation and across multiple domains. The same categorization as the above but beyond Industry 4.0 can be of application.
- **The scientific audience, general public and the funding EC Organisation**: To document the work performed and justify the effort reported for the relevant activities. The scientific audience can also get an insight of the design of the business APIs as well as the NetApps' development process and functionalities.

2 5G EXPOSURE CAPABILITIES

2.1 5G CORE NETWORK CAPABILITIES

Service-Based Architecture (SBA) [3] provides a modular framework for a new 5G Core network model that differentiates from its predecessors in the sense that all the core network functions (NFs) have been virtualized, each with authorization to access each other's services. Such approach enables a more resilient core network (CN) and it tackles the potential inefficiency of infrastructure's resources and consequently prevents the performance declension, allowing the creation of NFs leveraging virtualization technology (virtual machines and hypervisors), thereby improving the network in terms of flexibility, speed and scaling.
By utilizing the aforementioned characteristics, service providers are able to build a web-scale core with greater degrees of orchestration and automation so as to bring new and dynamic services to the market. The new emerging ecosystem also allows third-party applications to be integrated into the 5G Core via Application Functions (AFs) relying on the Network Exposure Function (NEF) to expose a variety of the network functions, and thus, to facilitate the development of third-party applications.

2.2 ANALYSIS AND IMPLEMENTATION OF NORTHBOUND APIs

2.2.1 NEF APIs

NEF Northbound APIs enable the 5G network to be more accessible, controllable and programmable, since service providers utilize those APIs to enhance their applications based on the capabilities that 5G systems offer.

Within the framework of 3GPP, the exposure of network capabilities was introduced in Rel. 13 of the specifications as Service Capabilities Exposure Function (SCEF) [4] and was designed to offer means to securely expose the services provided by the 3GPP standardized network interfaces. Nevertheless, until Rel. 15 3GPP's standardization efforts focused only on the southbound interfaces (i.e., interaction between SCEF and Evolved Packet Core (EPC). In Rel. 15[1], the first attempt of standardization of the Northbound APIs took place and the new T8 reference point was introduced for northbound Application Programming between the SCEF and Services Capability Server (SCS)/ Application Server (AS).

The NEF has a similar role as the SCEF in Evolved Packet System (EPS), as it exploits some of the exposure capabilities defined in SCEF and adjusts them so as to be compliant with the new Service Based Architecture (SBA). Compared to 4G (which is mainly providing throughput-based services), 5G is more focused on the vertical industries. Therefore, additional APIs have been defined and developed for NEF, besides the enhancements that have been performed to the inherited 4G capabilities. In order to effectively expose the 5G services, flexibility and compliance between NEF and the other 5GC (5G Core) NFs is essential, since NEF interacts with many of them (i.e., through southbound interfaces) that the Service Based Architecture realizes. Some examples include PCF (Policy Control Function) for provision of dynamic policy enforcement, NWDAF (Network Data Analytics Function) providing network analytics to assist an external application to make efficient decisions and SMF (Session Management Function) influencing the traffic by steering a connection to an Edge Server. Moreover, the capabilities

[1] 3GPP TS 29.122 version 15.0.0 Release 15

that the 5GC network offers, need to be exposed securely by hiding all the underlying network topology. The high-level architecture of NEF is depicted in Figure 1.

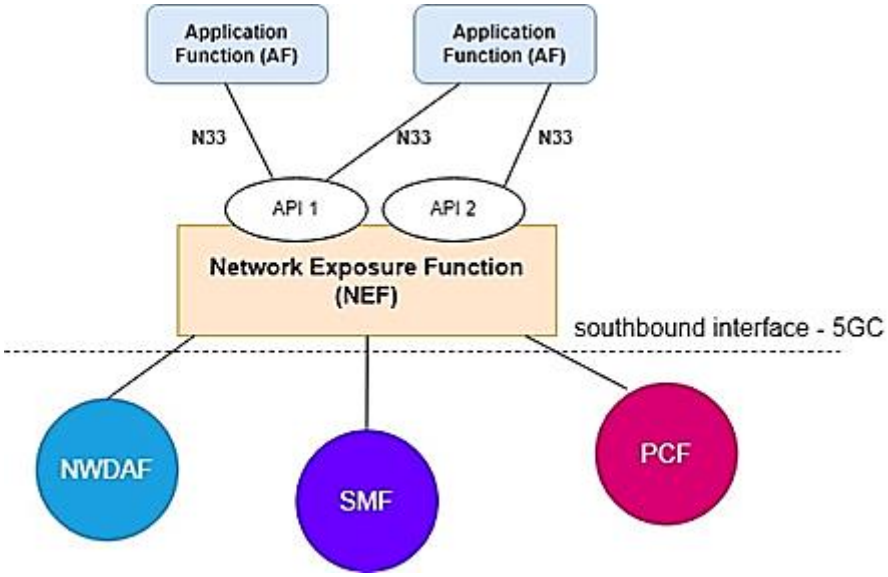


Figure 1 High level architecture of NEF

The NEF comprises several services that can be described as Monitoring services, Policy and Charging services, Application Provisioning services, Analytics services, Industry 4.0 / IoT (Internet of Things) specific services and security services [5]. The classification of the services, the correspondence with SCEF and NEF and the reference documents are presented in *Table 1*.

Table 1 Available Northbound APIs for the NEF services

API	Service	Exposure Function	Related 3GPP Document
Monitoring Event	**Monitoring**	SCEF	TS 29.122
MoLcsNotify	**Location**	NEF	TS 29.522
AsSessionWithQoS	**Policy/Charging**	SCEF	TS 29.122
ChargeableParty			
ResourceManagementOfBdt			
PfdManagement			
ApplyingBdtPolicy		NEF	TS 29.522
AnalyticsExposure	**Analytics**	NEF	TS 29.522
CpProvisioning	**Application Provisioning**	SCEF	TS 29.122
NpConfiguration			
ECRControl			
RacsParameterProvisioning			
ServiceParameter		NEF	TS 29.522
5GLANParameterProvision			
LpiParameterProvision			
TrafficInfluence			
DeviceTriggering	**Industry 4.0 / IoT**	SCEF	TS 29.122
MsisdnLessMoSMS		SCEF	
NIDD		SCEF	

NiddConfigurationTrigger		NEF	TS 29.522
AKMA	**Security**	NEF	TS 29.522

2.2.1.1 Monitoring Services

Within the 5G network, a UE (User Equipment) undergoes different stages in terms of connectivity, availability and location awareness. Sometimes, this kind of information may be valuable for an external 5G-enabled application. For example, an application needs to be aware of the location of its users within an industrial zone, so as to make authorization decisions for the personnel. The monitoring of UE's lifecycle by an external application within a 5GS (5G System) can be realized through the *MonitoringEvent* API. The several events provided by the monitoring API are described below:

- **Location Reporting**: This event detects the location of a UE. The request can include the Current Location or the Last Known Location of a UE.
- **UE reachability**: This option is mainly used when applications need to detect when a UE becomes reachable after being in a long power saving sleep cycle. The event is detected by the 5G when a UE becomes reachable (active). Then the NEF informs the application that the UE is now available and then the application can send the data, intended for downlink transmission. Applications can also provide based on a request, the maximum latency that is acceptable for the downlink data transfers, maximum response time for which the UE stays reachable to perform reliable transmission of the data; and the suggested number of downlink packets whether it is aware of. If the application can estimate the number of packets, it could help the User Plane Function (UPF) to buffer the packets in case that the UE is not reachable.
- **Loss of connectivity**: An application can request notification if the UE loses connectivity, for example, if the UE detaches from the 3GPP Network or if the UE has not communicated with the 3GPP Network after a pre-defined time. The network detects that the UE is no longer reachable for either signalling (i.e., control plane) or user plane communication. The application may provide a maximum detection time to identify the maximum interval after which the UE is considered unreachable.
- **Change of IMSI-IMEI association**: An application can request notification if the UE's International Mobile Subscriber Identity (IMSI) is suddenly associated with a different device, possibly indicating that a Subscriber Identification Module (SIM) card was moved to a separate device. (i.e., IMSI is a unique identifier allocated to each mobile subscriber in 3GPP networks – International Mobile Equipment Identity (IMEI) is the international mobile equipment identity allocated to each mobile equipment)
- **Number of UEs present in a geographic area**: This event indicates the number of UEs that are in the geographic area. An application can request a report of how many UEs that are associated with a specific group are in a geographical area. For example, a fleet tracking application could use this information to check how many of its delivery trucks are in a geographical area [5]
- **Packet Data Network (PDN) connectivity status**: This event is detected when Protocol Data Unit (PDU) session is established or released. (i.e., PDU Session in 5G networks is the end-to-end "tunnel" that connects a UE to external data networks)
- **Downlink data delivery status**: It indicates the downlink data delivery status in the core network. Events are reported at the first occurrence of packets being buffered,

transmitted or discarded, including first data packet buffered, estimated buffering time, first downlink data transmitted and first downlink data discarded.

- **Availability after Downlink Data notification failure**: This event is detected when the UE becomes reachable again after downlink data delivery failure. For example, if the application attempts to send data to a sleeping IoT device, the network may discard the data and notify the application when it wakes up and is available to communicate. To that end the application will be aware when its IoT devices (UEs) will be available again.

2.2.1.2 Location Services

5G networks categorize the location services into three types based on which component initiates the location reporting [6]. Network Induced Location Requests (NI-LR) are initiated by an Access and Mobility Management Function (AMF) to determine a UE's location on behalf of a regulatory agency (e.g., to determine location during an IP Multimedia Subsystem (IMS) emergency call). Mobile Terminated Location Requests (MT-LR) are initiated by internal or external Location Service (LCS) client which can be a third-party application accessed by the UE, regulatory agencies or mobile network operators. Last but not least, Mobile Originated Location Requests (MO-LR) are initiated by the UE to determine its own location or to provide its location to an external client. The latter is realized through the *MoLcsNotify* API that NEF provides. Therefore, the main difference with the location event offered by the *MonitoringEvent* API is that it fulfils the MT-LR service. Specifically, an external application subscribes to the *MonitoringEvent* API to receive notifications related to location, thus the external application initiates the request.

2.2.1.3 Policy and Charging Services

The NEF also exposes services that are related to policy and charging capabilities that the 5GC offers. Policy Control and Charging (PCC) system is responsible for providing these functionalities within 5GS. PCC enables mobile operators to ensure better service-awareness Quality of Service (QoS) and charging control. Typically, in wireless networks where the bandwidth and capacity are of major importance, it is essential to utilize the radio resources efficiently. Furthermore, not only the network but different services themselves require divergent QoS levels, which is necessary to transport the data. The *AsSessionWithQoS* API allows applications to set up a session (i.e., connectivity), indicating the desired level of QoS (e.g., latency and priority) for a given Internet Protocol (IP) traffic flow. Moreover, since the QoS level can be affected by several sudden changes or anomalies within the network, this API can inform an application whether the desired QoS will be guaranteed or not in the near future. The *ChargeableParty* API offers to applications the opportunity to inform the network that it will start or stop sponsoring a given traffic flow. Another API that is related with policy and charging rules is the *PfdManagement* that can be used by an application to provide Packet Flow Descriptors (PFDs) to the Packet Flow Description Function (PFDF) within the 3GPP network so they can be used to detect certain types of traffic and apply specific PCC rules to the detected flows [5].

In general, some applications may know the amount of data they need to exchange with a number of devices (UEs) in a geographical area. For example, an IoT agriculture application may need to exchange some weather forecast information with one hundred sensors during the night. The application can utilize the *ApplyingBdtPolicy* and *ResourceManagementOfBdt* APIs to provide the 5G network with all the essential requirements of the data transfer (e.g., number of UEs, data per UE, and time constraints). Consequently, the PCC system can handle the information retrieved from the application and generate a set of policies that indicate the best time to perform the data transmission.

2.2.1.4 Analytics Services

For some vertical use cases in the area of industrial automation, where EVOLVED-5G also focuses on, 3GPP has defined communication for factories of the future, including application areas and mapped applications (e.g., motion control, massive wireless sensor networks, augmented reality, process automation, connectivity for the factory floor, and inbound logistics for manufacturing). Data analytics can be useful for such use cases, ensuring network availability or for providing predictive maintenance features. Other remarkable operations that can be captured through analytics framework are efficient QoS management, traffic steering and mobility management. Regarding the QoS management, in Rel. 16 3GPP introduced the notion of network prediction [7], by enabling 5GS to notify a Vehicle to Everything (V2X) application that the QoS of a UE's ongoing communication might need to be downgraded, e.g., due to predicted bad network conditions, change of radio technology and radio congestion. Therefore, analytics contribute significantly to the harmonization of the 5GC network assisting applications to reach higher service availability and quality. NWDAF is a network-aware function that interfaces with different 5GC network functions and collects data or events that could be beneficial for analysis. On the one hand NWDAF realizes the functionality needed to perform such tasks. On the other hand, the *AnalyticsExposure* API is responsible for exposing them to the application through NEF. The application can be provided with, network performance analytics to perform network optimization. In addition, UE communication analytics can be utilized to predict some UE communication patterns that will allow for optimizing some operations (e.g., traffic routing handling or QoS improvements). Moreover, UE mobility analytics can help with location predictions and anomaly detection to automatically alert applications for changing data that behaves unexpectedly. Finally, QoS sustainability analytics can provide information regarding the QoS change statistics.

2.2.1.5 Industry 4.0 / IoT Services

IoT applications are often characterized by the fact that they rarely need to receive data but periodically send a heartbeat message to a server to indicate that the device is active and able to receive. Rather than sending heartbeat messages to the server, the UE can listen on a Short Message Service (SMS) Port or Non-IP connection for a short trigger, or wake up message, requesting that the UE application should contact the server. To achieve this type of communication between the UE and the application in the most optimal way, NEF exposes APIs that allow the external application to exchange data with the UE via two different non-IP data delivery methods: SMS and Non-Access Stratum (NAS) based Non-IP Data Delivery (NIDD). This kind of data delivery takes place over control plane since the data packets to be exchanged are so small that a potential use of the IP protocol would impose an unmanageable overhead. Since SMS messages are stored in the network when the recipient UE is sleeping, SMS triggering is particularly useful for initiating contact with UEs that sleep for long stretches of time. For SMS delivery NEF exposes the *MsisdnLessMoSms* API that allows an application server to exchange data packets with a UE-hosted application client via SMS. For data delivery over NAS, NEF exposes the *NIDD* and *NiddConfigurationTrigger* APIs. 3GPP has also standardized a reliable data service protocol that can be used when exchanging data packets between the UE and NEF, enabling the NIDD feature. In brief, this protocol uses 1-to-3-byte header non-IP data packets to perform the data transmission [8]. In order to initiate this type of communication, the network needs to reach the sleeping devices through the NEF. The *DeviceTriggering* API supports this action since it is able to recall the pending devices and awake them from the discontinuous reception cycles.

2.2.1.6 Application Provisioning Services

Besides the fact that the 5G network exposes its capabilities through the NEF, sometimes the applications can assist the network by providing application related information. For example, in V2X communications, a vehicle may provide its own destination and the path to the 5G network in order to handle more efficiently the handovers. The 3GPP core network offers several services that are particularly useful to UEs hosting applications. N33 interface exposes APIs that enable applications to update parameters that are already configured in 5GC. Applications may consume these APIs and provoke the 5GC to take specific actions. By communicating with NEF, an application can influence how traffic is routed through the network. This capability is an important auxiliary means to support Multi Access Edge Computing (MEC) to achieve traffic offload and minimize delay. A use case where this API finds practicality, could be video delivery towards MEC hosts [9]. This aspect can be realized through the *TrafficInfluence* API.

As mentioned in a previous paragraph, devices that are used in IoT applications are usually characterized by their low energy consumption due to the fact that they are limited in terms of battery capacity. The 5G network is capable of addressing these issues, since it can configure UEs to use different saving features such as Power Saving Mode (PSM) and Extended idle-mode Discontinuous Reception (eDRX), provide information to the Radio Access Network (RAN) in order to minimize UE state transitions (i.e., IDLE – CONNECTED) and buffer downlink packets that are sent towards UEs that cannot be paged because they are in deep sleep. These features can be configured based on parameters that are statically created in the UEs initial subscription to the 5G network. However, by configuring these services in a relatively static manner, the usability can become somewhat limited. For example, some sensor applications that are hosted in a UE may typically be able to sleep for long stretches of time, but the application server may demand that the UE should be available more quickly during an emergency situation.

CpProvision API enables applications to influence and configure these features in a more dynamic manner. They can use this API to tell the network when a device is expected to communicate, thus the 5G network can benefit from this kind of information and minimize, e.g., the UE state transitions. Moreover, applications can use the *NpConfiguration* API to inform the network about the maximum acceptable delay between the UE's reachability occasions and long periods that the UE needs to be available for mobile terminated data after it becomes reachable. The network can utilize this information to configure UE's DRX cycles, PSM and tracking area update timer. Also, applications can indicate the amount of downlink packets the network should buffer when the UE is sleeping.

Other APIs are related to radio capabilities that the 5G RAN offers such as *ECRControl* API and *RacsParameterProvisioning* API. The former can be used by an application to check if the Enhanced Coverage Restriction Control is enabled for a UE and it can enable or disable this feature. Service providers may have an agreement with mobile operators for coverage enhancements that may be charged extra. Therefore, this API can be used for devices that need to be reached immediately (i.e., emergency situations where the device is unreachable). The latter can be used by an application to inform the 5GC with the UE Radio Capability ID, whether it is assigned by the UE manufacturer. Also, it may provide a list of associated IMEI codes. In general, UE radio capability contains information on Radio Access Technologies that the UE supports (e.g., power class, frequency bands, etc.).

LpiParameterProvision API refers to the Location Privacy Indication [6] where an application can choose if the location of a UE or a group of UEs can be shared with other external applications

(i.e., Location Service Clients). *ServiceParameter* API allows applications to provide service parameters to the 5G system related with proximity-based services (i.e., device to device communications). This aspect of the 3GPP was initially introduced for V2X communications and it's currently out of scope.

3GPP has extensively studied new use cases and potential requirements applicable to the 5G system for a 3GPP network operator to support 5G Local Area Network (LAN)-type services over the 5G system (i.e., UE, RAN, Core Network, and potential application to manage the LAN-style service). The goal of this study on 5G LAN-type service provisioning is to provide distributed LAN-based connectivity between two or more terminals or User Equipment entities (UEs) connected to the 5G network. One potential scenario where an application could benefit from the exposure of *5GLANParameterProvision* API (via NEF), is described below:

A mid-size enterprise decides to replace their existing wired and wireless LANs in the office with a 5G private network using NR radio in unlicensed spectrum. The enterprise wants to be able to control and manage the equipment that is able to access the 5G Virtual Private Network (VPN), e.g., printers, scanners, company database, phones, computers, to allow only specified equipment to have access and block access from non-company equipment (e.g., a visitor's phone). While in the office, employees will be able to use their cell phones and computers to communicate with other office equipment such printers, scanners, and video conference displays as well as to access company files and databases that are only available to employees.

2.2.1.7 Security Services

Authentication and Key Agreement for Applications (AKMA) is a mobile network service intended to support authentication and key management based on 3GPP credential in 5G system, for third-Party applications [10]. AKMA can provide authentication and session key negotiation services for third-party applications based on the access authentication system of the USIM (Universal Mobile Telecommunications System (UMTS) Subscriber Identify Module) card and carrier network in order to establish secure transmission channels from terminals to applications. AKMA API tries to realize this security aspect and allow service provides to define the aspects for secure communication between the UE and the application.

2.2.2 Collection of Required APIs by the SMEs and their realization by the NEF Emulator

The exploitation of the NEF capabilities from industry has begun already[2]; however prior taking the full advantage of a NEF-based network exposure many challenges are to be addressed. Indeed topics regarding the exposure capabilities of the network are still to be considered in Rel. 18, while telecommunication vendors are working to adapt the service-based architecture and implement the already specified exposure functionalities. As cornerstone to the above-mentioned process is the implementation of NEF functionality since currently there are no open commercial solutions implementing the entire service-based architecture and the southbound interfaces that NEF requires in order to expose the standardized APIs. **Nevertheless, EVOLVED-5G addresses the problem through the development of a NEF emulator** [11], which aimed at surpassing this challenge by creating simulated and emulated events. The implementation details of the NEF emulator can be found in D3.1 [1].

The implementation process for the various APIs of the NEF emulator, has been prioritized based on the needed expressed by the EVOLVED-5G SMEs for the four Industry 4.0 pillars of the project.

[2] https://mediabank.ericsson.net/deployedFiles/ericsson.com/Ericsson%20Cloud%20Core%20Exposure%20Server.pdf

The following list presents the set of SMEs partitioned within the four pillars:

- Interaction of Employees and Machines (IEM) pillar (IMM, INF, GMI-Aero).
- Efficiency in FoF Operations (FoF) pillar (CAF, ININ, QUCOMM, UMA/CSIC).
- Security Guarantees and Risk Analysis (SEC) pillar (8BELLS, IQB, FOGUS).
- Production Line Infrastructure (PLI) pillar (PAL, UML).

The joint analysis of the needs for the four pillars, has resulted in a set of target APIs for the first version of the NEF emulator (Table 2 Selected APIs for implementation"). As it is depicted in Table 2, the commonly identified APIs for implementation at this stage of the project, are the AsSessionWithQoS and the MonitoringEvent API. The early realization of those APIs allowed the familiarization of the SMEs/developers with the development process of their NetApps and facilitated the build of a kind of consistency among the several NetApps, targeted by EVOLVED-5G. Next versions of the NEF emulator will incorporate additional APIs with the aim to leverage additional functionalities required for each use case.

Table 2 Selected APIs for implementation

Pillar	SME	API Selection
Innovation in the interaction of employees and machines	IMM	AsSessionWithQoS API MoLcsNotify API MonitoringEvent API AnalyticsExposure API
	GMI	MoLcsNotify API MonitoringEvent API LpiParameterProvision API
	INF	MoLcsNotify API MonitoringEvent API LpiParameterProvision API
Efficiency in FoF operations	CAF	AnalyticsExposure API MonitoringEvent API ServiceParameter API
	ININ	AnalyticsExposure API MonitoringEvent API AsSessionWithQoS API
	UMA	MonitoringEvent API AsSessionWithQoS API
	ZORTENET	AnalyticsExposure API

		5GLANParameterProvision API ServiceParameter API
Security guarantees and risk analysis	8BELLS	AnalyticsExposure API MonitoringEvent API
	IQB	AnalyticsExposure API AKMA API LpiParameterProvision API
	FOG	AnalyticsExposure API
Agility in the production line infrastructure	PAL	MonitoringEvent API MoLcsNotify API AsSessionWithQoS API
	UMN	MonitoringEvent API MoLcsNotify API AsSessionWithQoS API

2.2.2.1 MonitoringEvent API

As described in section 2.2.1 a NetApp can monitor different events that can be triggered by the 5GC. **At the timing of writing this deliverable, the MonitoringEvent API supports the location reporting event**. Specifically, when a UE handover takes place to a neighbour cell, NEF informs the NetApp for this event. At this point, the location accuracy equals to cell level, meaning the NetApp will be receiving the new id of the cell where the UE moved to. In order for the NetApp to be capable of collecting this information, initially it has to make a subscription to the NEF and include the type of the event (i.e., LOCATION_REPORTING). The request body of the POST request is depicted in Figure 2.

Request body required

```
{
  "externalId": "10002@domain.com",
  "notificationDestination": "http://localhost:80/api/v1/utils/monitoring/callback",
  "monitoringType": "LOCATION_REPORTING",
  "maximumNumberOfReports": 15,
  "monitorExpireTime": "2022-01-28T19:42:54.868Z"
}
```

Figure 2 The request body of the POST for the MonitoringEvent API

The first field of the request is the external identifier (External ID) which is a global unique identifier. External ID identifies a subscription associated to an International Mobile Subscriber

Identity – IMSI (or Subscription Permanent Identifier usually in 5G), a permanent identifier of the UE within the 3GPP network. This ID can be used by the mobile network operators to hide the permanent network identifier of the UE from the external application (i.e., NetApp). The monitoring type indicates the type of the event that the NetApp is interested in (e.g., LOCATION_REPORTING). In order to control the time of an active subscription, maximum number of reports and expiration time monitoring fields are indicated.

At this point, it should be mentioned that 3GPP adapts the client-server model implied by the REST (REpresentational State Transfer) paradigm. The main difference is that the client and server are named as service consumers and service producers, respectively [12]. Besides the direct HTTP (Hypertext Transfer Protocol) request – response sequence between these two actors, asynchronous callback notifications are supported. This mechanism could not be omitted since the mobile networks are inherently stochastic. Therefore, the service consumers are notified any time an event occurs. **When NetApps subscribe to the NEF emulator they play the role of the service consumer and NEF emulator represents the service producer** as depicted in Figure 3. In the asynchronous notification NetApp and NEF Emulator change roles. The notification destination field enables the asynchronous communication.

Figure 3 Sequence diagram for the NetApp and NEF Emulator interaction

After the NEF emulator receives the request successfully, it can answer with two possible responses. If the maximum number of reports value is greater than one then the NetApp has an active subscription, and the NEF emulator responds with a 201 created message as depicted in Figure 4. The response body includes all the necessary information from the subscription and also there is also a reference resource returned to indicate the active subscription, as a “link” field and an HTTP response header (i.e., location). This enables the NetApp to also perform GET, PUT and DELETE HTTP requests to the referenced resource, in order to receive, update and delete the active subscription, respectively. On the other hand, the NetApp may need to request the location information at a specific time frame. Thus, it can initiate a one-time request to the NEF emulator by using the value “one” in the maximum number of reports field. Then the NEF Emulator, subsequently, answers with a direct response (i.e., 200 OK), indicating the current location of the chosen UE as shown in Figure 5.

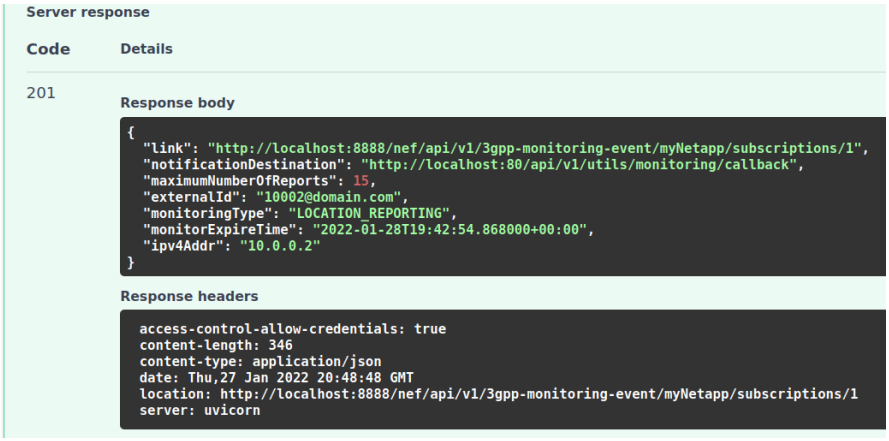


Figure 4 201 response of the NEF emulator- Active subscription message

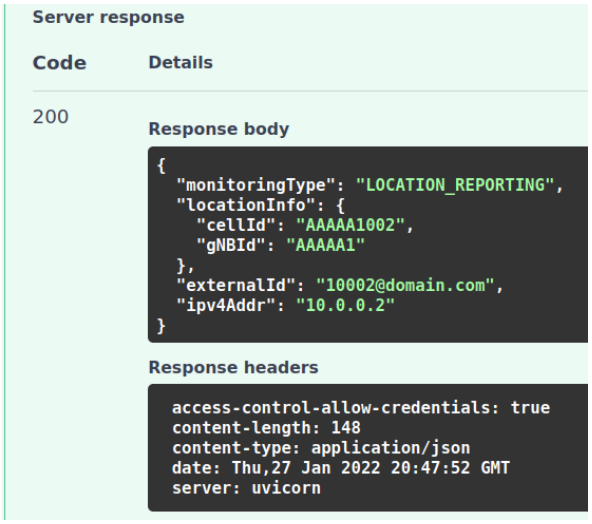


Figure 5 200 response of the NEF emulator

2.2.2.2 AsSessionWithQoS API

Applications can utilize **AsSessionWithQoS API to ensure better service experience and avoid service interruption** which may be provoked due to unexpected QoS downgrades. 5GS comprises a complex QoS model where different network functions from the 5GC and the RAN interchange information to make the necessary adaptations for the best possible end-to-end connectivity (i.e., 5GS provides connectivity by establishing a session between UE and the data network). In the EVOLVED-5G architecture NetApps are the external applications that communicate with NEF. The 5GC supports notification control mechanisms to inform the applications (i.e., NetApps) about potential changes that may occur between the different QoS Flows. Subsequently, NetApps will perform any necessary actions based on the notification messages received by the 5GC, to adjust application's behavior.

As a first step, NetApp will send an HTTP POST request to the AsSessionWithQoS API to make a valid subscription as can be seen from Figure 6. Upon successful creation of the subscription the NEF emulator responds with an HTTP 201 Created message as depicted in Figure 7, including the

reference resource, so NetApp can easily update, delete or retrieve the active subscription. The NetApp needs to choose a valid address (e.g., Internet Protocol version 4 (ipv4), Internet Protocol version 6 (ipv6) or Media Access Control (MAC) address) to identify on which UE the QoS Flow establishment will occur. Only one of the addresses described needs to be included. For the callback notification mechanism, same applies here as in the case of Monitoring Event API.

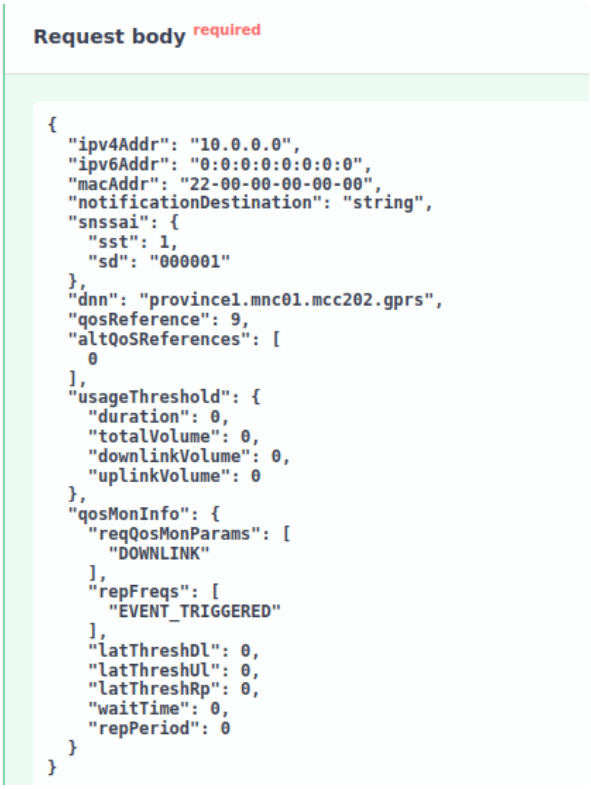


Figure 6 AsSessionWithQoS POST Request

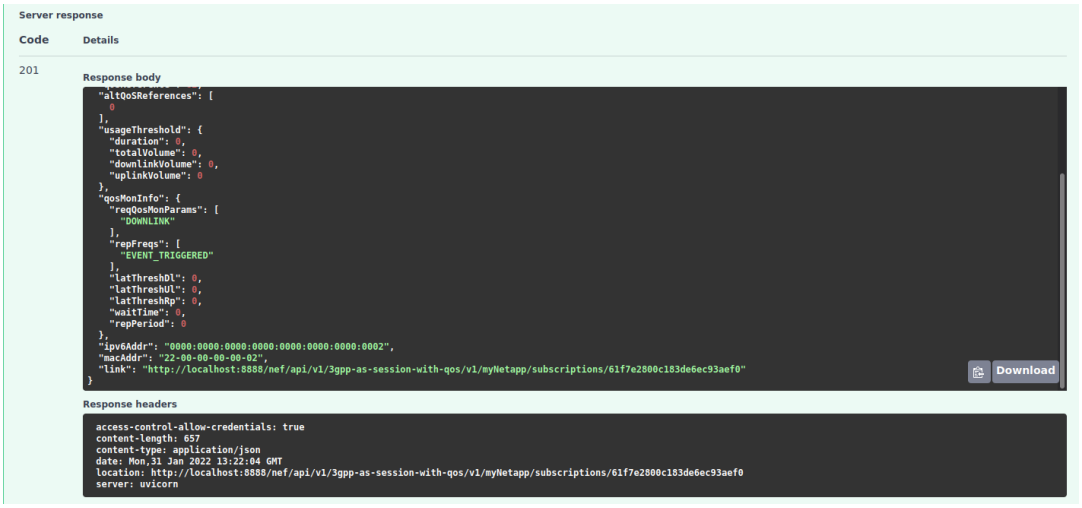


Figure 7 Successful creation of the subscription

The Single Network Slice Selection Assistance Information (S-NSSAI) indicates in which network slice the UE (vertical application) chooses to establish or modify a QoS Flow. Currently, the available slices should be provided in the UE in the initial attachment towards the mobile network, thus there are predefined values stored in the end user. Work is in progress, in 3GPP Rel. 18, on exposing APIs that will enable the application to choose a network slice by its own and not from a predefined list. Data Network Name (DNN) associates the UE with any data network. The most common term is Access Point Name (APN) used in 4G networks, which is the name of a gateway between the mobile network and another computer network, frequently the public internet. Since, experimentation activities will not take place regarding network slices at this phase of EVOLVED-5G project and given the fact that NetApps are only related to control plane functionality, 'snssai' and 'dnn' fields are non-functional within NEF emulator.

The QoS reference corresponds to a list of standardized 5QI values [13] that define a set of QoS characteristics. QoS characteristics within 3GPP networks are usually divided into Guaranteed Bit Rate (GBR) and NON-GBR resource types. GBR types are characterized by their strict requirements in terms of packet delay and packet error rates. Some of the standardized 5QI values are presented in Table 1. Some of these values are implemented in the NEF emulator, so the application can retrieve them by invoking the `/api/v1/qosInfo/qosCharacteristics` endpoint, thus it can choose the value with the characteristics that fits application's requirements and finally fill in the subscription request.

An application can be notified about several changes and issues related to QoS, that may arise throughout the lifetime of an active connection. Firstly, NEF can notify the NetApp whether the QoS targets can no longer (or can again) be guaranteed. If the QoS levels are not acceptable by the service the NetApp can choose another GBR value that has initially provided in the request (i.e., alternative QoS references). Specifically, NEF sends a callback notification that the QoS cannot be guaranteed and provides the NetApp with a list of the alternative values that can be supported by the network at this moment. According to [14], the notification control for this type of events should be enabled only for GBR QoS Flows. Another aspect is the monitoring of the uplink, downlink and round-trip time delay. Last but not least, usage monitoring control information enables the user plane monitoring of resources for both volume (e.g., megabytes) and time usage per session basis. Typically, in mobile operator networks users or service providers are charged by the mobile operator per volume or per time for using network resources. An application may specify a usage threshold, thus while this threshold is exceeded the NEF can inform the application for such a case.

At the time of writing this deliverable, the current version of the NEF emulator supports the QoS fulfilment notification event. As described, NEF does not interface with the southbound APIs that the service-based architecture offers. Therefore, the event that triggers the notification callback is the handover. If the UE chooses a GBR flow and enters a cell that another UE is currently connected, we assume that the network has no available resources and NEF emulator notifies the NetApp that the QoS cannot be guaranteed. In such a case, the NetApp should perform any actions necessary to adapt the behaviour of the associated vertical application.

3 DEVELOPMENT OF THE NETAPPS AND BUSINESS APIs

3.1 DEVELOPMENT TOOLS

3.1.1 Tools for the Development of the NetApps

As it has been thoroughly described in D2.2 [15] as well as D3.1 [1], the NetApp development and verification are the initial phases within EVOLVED 5G workflow and both are handled by the Workspace, as depicted in Figure 8. The development of a NetApp is the first phase in the lifecycle of a NetApp, where developers make use of the different tools provided by the EVOLVED-5G Facility. Such tools are comprising the SDK tools, which can be found in the EVOLVED-5G GitHub repository [16] and were thoroughly described in D3.1.

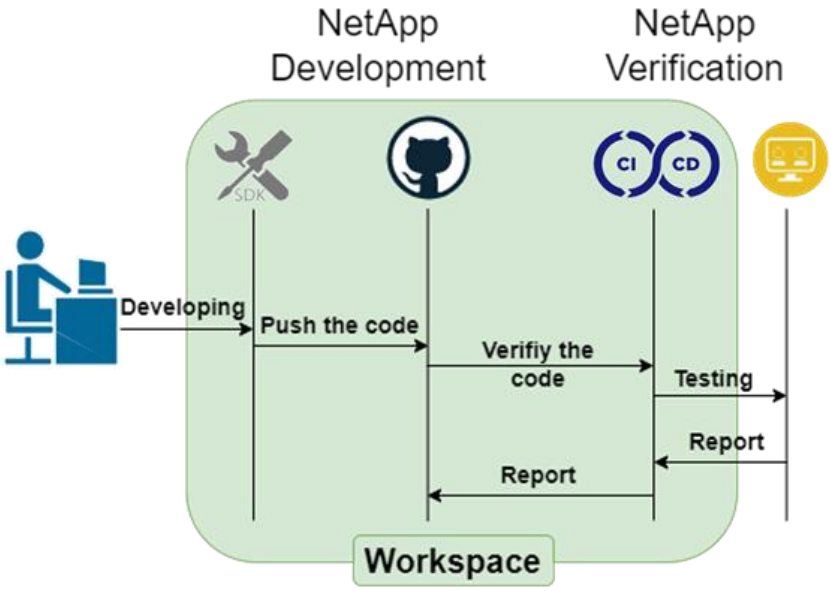


Figure 8 EVOLVED-5G Workflow diagram regarding Development and Verification phases

The SDK tools is a package where a Command Line Interface (CLI) tool (additional details regarding the CLI are provided in D3.1 [1]) as well as a template is provided to developers. In particular, the CLI is a fundamental tool, since, it is in charge of:

- Providing NetApp template from EVOLVED-5G GitHub repository.
- Implementing automated pipelines to build, deploy or destroy a NetApp
- Offering libraries that enable the interaction with 5G Core APIs, provided in that phase of the project by the NEF emulator [17]. The purpose of developing and providing such libraries for NetApp developers is to speed up and agile the development process of a NetApp.

To start building the NetApp, the developer has to make use of the already mentioned CLI tool. The developer executes a very straightforward command and a repository on GitHub is created, together with the NetApp options provided by the developer, some of them are, NetApp name, NetApp port, NetApp repository name, as well as the skeleton to start the NetApp development [18]. To develop the NetApp any Integrated Development Environment (IDE) can be used by the developer (i.e., Eclipse, Visual Studio etc), and hence EVOLVED-5G is not imposing any IDE. This decision was made in order to make the NetApp development as much easier as possible. The developing process to implement the core functionalities towards the delivery of the initial

release of NetApps (v1.0) is presented in detail in chapter 4 “NETAPPS AND USE CASES”. The GitHub repository created for the NetApp is public, therefore anyone outside the EVOLVED-5G organization can access. However, only authorized members (such as the developers) of the organization can modify it. This way of working entails a collaborative approach, bringing new ideas or functionalities to the NetApps because all the developments are publicly shared.

Once the NetApp development is finalized and is uploaded to GitHub repository, within the CLI tool, different pipelines are implemented to offer the developer functionalities such as build, deploy and destroy the NetApp. Such pipelines are integrated in the CLI tool; therefore, the developer has handy all the development from just one tool. Figure 9 presents an overall approach of the tools that are utilised for the NetApp development. Some tools have been developed and implemented specifically for the project, while some others are commercial solutions which are used to enable the lifecycle of the NetApp.

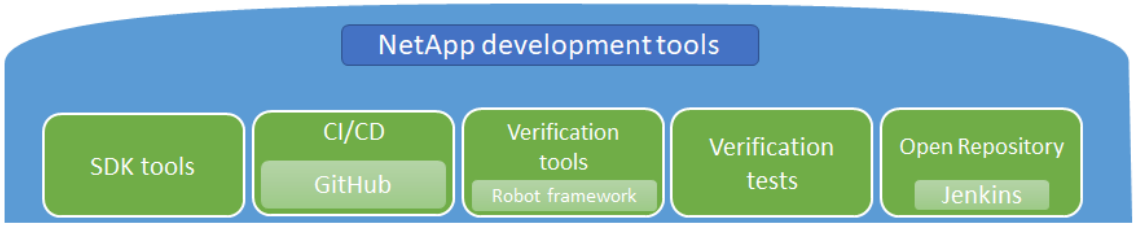


Figure 9 NetApp development tools

By making use of the CLI tool, the developer can also test the NetApp. Three simple commands are available to launch the pipeline in order to verify that the NetApp operations and status are correct. The CI/CD centralised server is in charge of carrying out such pipelines. In addition to the build, deploy and destroy operations, the NetApp has some functionalities which the developer should verify; this is done using the verification tools and the corresponding tests during the verification phase described in the next section.

In order to save time on the developers’ side trying to set up proper communication, SDK libraries encapsulate the calls to 5G Core APIs or NEF emulator and hide the low-level complexity, offering the developer more time to focus on the business need their NetApp is solving. At the submission of this deliverable, and following the state of art described in section 2.2.2, two main SDK libraries are applied to the all NetApps being implemented:

- Location subscriber, implemented for those NetApps that need to monitor the location of end-devices, based on the services that MonitoringEvent API provides, as depicted in Figure 10.

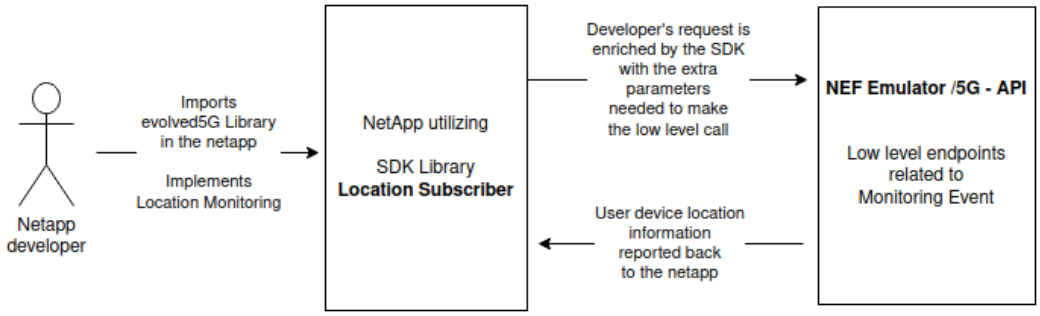


Figure 10 Usage of libraries towards the functionality of the Monitoring Event API

- QoS Awareness, implemented for those NetApps that need to adjust the application behaviour when QoS targets cannot be satisfied.

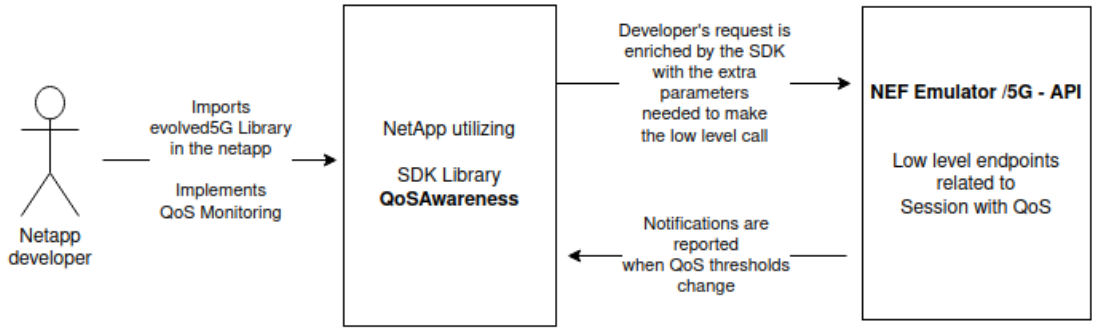


Figure 11 Usage of libraries towards the functionality of the AsSessionwithQoS API

3.1.2 Tools for Verification testing

EVOLVED-5G has defined a series of requirements that NetApps must fulfil (refer to deliverable D2.1 "*Overall Framework Design and Industry 4.0 Requirements*") [2] in order to accomplish its goal of decoupling the Vertical Application (vApp) from the underlying 5G network specific properties. The target of the verification tests, (also encapsulated in the Workspace environment), is the verification of the expected functionality of the NetApps in terms of interoperation with the 5G infrastructure. As already documented in D3.1 [1], and more specifically in sections 7.2.3 and 7.2.4 together with NEF a Common API Framework (CAPIF) core function are being developed. Both NEF and CAPIF facilitate the development of NetApps in a completely virtualized environment and hence verification tests need to be agnostic of whether the 5G functionalities are offered by actual 5G networks or NEF Emulator.

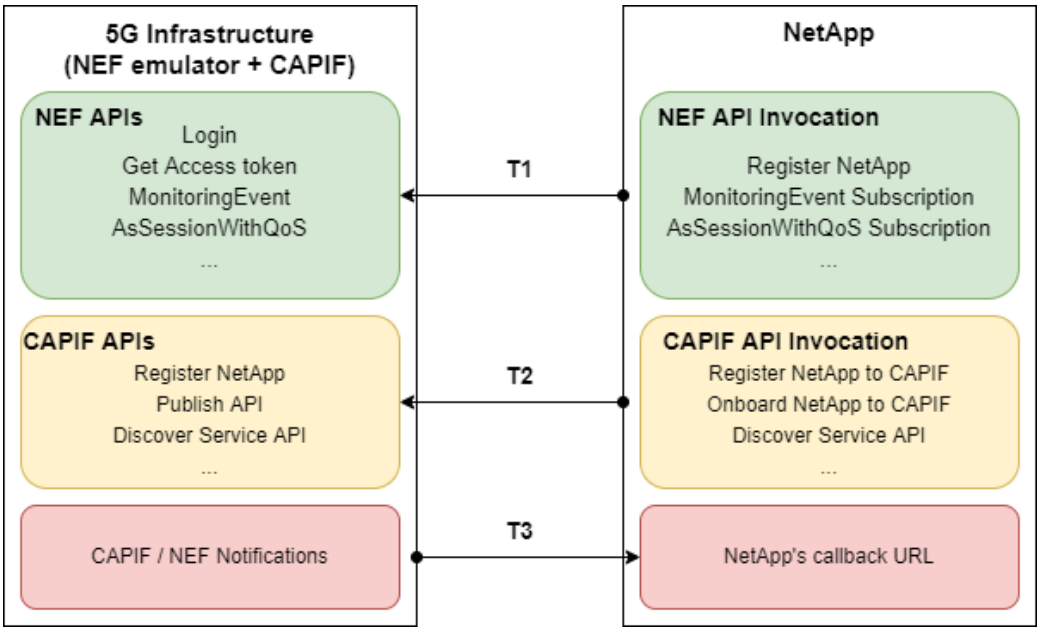


Figure 12 NetApp verification tests

By the time this deliverable is submitted, an initial set of the verification tests have been designed following the progress of the relevant artefacts involved in the testing process, i.e., the

NEF emulator, the CAPIF core function and the NetApps. The verification tests' design and implementation are based on the open-source software Robot Framework [19].

Figure 12 depicts the verification tests' approach. Three sets of tests are currently being defined. T1 and T2 refer to any communication initiated by the NetApp with synchronous responses from the 5G NPN infrastructure where the exposure of the services will be realized by the NEF emulator and CAPIF tool, leveraging NEF and CAPIF APIs respectively. T3 targets the verification of asynchronous communication between the NetApp and 5G services, which is initiated by the 5G network and is implemented through appropriate callback functions on the side of the NetApp. For T3, the relevant NetApp REST API is put into test. For T1 and T2, verification tests are defined either by letting part of the NEF/CAPIF API invocation calls being exposed as REST APIs, which allows actual integration tests to be realized between the NetApp and the 5G components, or in cases when this is not preferable, by defining the relevant unit tests of the corresponding API calls on the side of the NetApp.

The verification tests of a NetApp are in practice defined as unit tests between the NetApp and the exposed NEF and CAPIF APIs. The implementation of the EVOLVED-5G NEF and the CAPIF core function provides NetApp developers with the appropriate API calls to realize the communication between the NetApp and the 5GC. Depending on the definition of each NetApp and the relevant use cases, a subset of the API calls offered are going to be exploited, as described in section 2.2.2. To this direction, a homogeneous approach in verifying the NetApps, (needed also to allow the automation of the verification process) is required. Hence a common API enabling effective communicating through NEF and CAPIF is necessary to be implemented horizontally by all NetApps. To realize that, a dummy NetApp [20] has been developed to serve as the basis and example for all NetApp developers, with a proper implementation of NEF and CAPIF API calls, in order to let the NetApp interoperation with the 5G network be automatically tested. The aforementioned functionalities within the relevant repository of GitHub are shown in Figure 13.

dummy-netapp

Architecture

Container	Folder	Description
python_netapp	pythonnetapp	Python NetApp (communication example with CAPIF)
redis	-	DB to store info exchanged with CAPIF
web_netapp	webnetapp	HTML NetApp
nef_callback_server	nef_callback_server	Server implementing NEF callback endpoints
capif_callback_server	capif_callback_server	Server implementing CAPIF callback endpoints

Figure 13 Dummy NetApp functionalities- GitHub repository

3.2 Type Of NetApps And Development Process

3.2.1 Type of NetApps

As it has been described in detail in D2.2 "*Design of NetApps development and evaluation environments*" and more specifically in section 2.2 "*Design Principles*", EVOLVED-5G project has identified two types of NetApps, namely the Stand Alone (SA) and Non-Stand-Alone (NSA) NetApp. These two types are deriving from the way that the services are provided to the verticals and it has been decided that EVOLVED 5G will support both modes, in order to provide flexibility and engage with vApp developers.

The SMEs that are delivering an NSA mode have chosen their NetApp to operate as a wrapper of Northbound APIs, aiming to expose services through Business APIs. This means that the NetApp will act as an auxiliary software module and becomes functional when its business APIs are consumed by the vApp. The vApps in such case will be upgraded leveraging the 5G exposure capabilities by utilizing the business APIs. Those business APIs will return the requested information to the vApp once the retrieval has been completed successfully. The rest of the SMEs that have selected the SA mode are utilizing 5G exposure capabilities through the integration of the NetApp to their vApp.

In the light of the above, Table 3 below summarizes the preferences of EVOLVED-5G developers towards SA or NSA implementation, taking into account the distinct characteristics that each use case delivers.

Table 3 Type of NetApp per SME

Pillar	NetApp Name/SME	Characteristics	NetApp Mode (SA/NSA)
IEM	*Remote assistance in AR/ IMM*	The NetApp is supporting QoS features for other Augmented Reality, Virtual Reality or video streaming applications, by exposing QoS APIs.	NSA
	Digital/physical twin/ GMI	The NetApp could be used by several devices at the same time, for the same repair. Furthermore, the vApp side needs to be upgradeable independently, and does not contain the necessary components to run the NetApp.	NSA
	Chatbot assistant/ INF	The NetApp supports the integration with a vertical app through the APIs it exposes, providing additional services and enhancing the functionalities of the vertical app.	SA
FoF	*Occupational Safety Analysis/ CAF*	The NetApp uses location and QoS information for short-term path planning and action timing. Safety Analysis is implemented in vApp allowing more flexibility in the exploitation of various hardware resources and scalability.	NSA
	Industrial grade 5G connectivity/ ININ	The NetApp is integrated in industrial 5G IoT system and enhances system provisioning and operation through the integrated 5G IoT management component.	SA

	Anomaly detection/ ZORTENET	The NetApp exploits location and netflow information provided by the 5G NEF in order to identify offenders, generate alerts and protect critical assets	SA
	Smart irrigation and agricultural drones/ CSIC	The NetApp is used for geo-locating and storing the information created by a large number of sensors in a central location. In a future phase it will also act as intermediary with the 5G Network by exposing the QoS APIs.	NSA
SEC	*Traffic Management/ 8BELLS*	The NetApp is used for the accurate measurement of traffic over an interface, in order to check for "unregistered" traffic, and the lessening of the burden of a congested device in the network.	NSA
	ID Management and Access Control/ IQBT	The NetApp acts as an intermediary between third-Party NetApps and the 5G Network Exposure Function.	SA
	5G SIEM add on FOGUS	The NetApp offers interoperability between a SIEM system and a 5G network.	SA
PLI	*Teleoperation/ PAL*	The specific NetApp acts as a bridge between the 5G Network and the robot standards.	SA
	Localization UML/PAL	The specific NetApp acts as a bridge between the 5G Network and the robot standards.	SA

3.2.2 Development process and workflow

In order for the SMEs to deliver the preliminary v1.0 of the NetApps and for the project to achieve a homogeneity in terms of the maturity level of the results, a development process comprised of four steps has been defined for NetApps' developers. The overall development process can be seen in Figure 14Figure 14. The first step (Step 1) is the definition of the functionalities that will be offloaded from the vApp to the NetApp. This kind of offloading describes the interaction between the vApp and the NetApp with the aim to ensure that the exchange of data would be successful. Once the aforementioned interaction is functional, the next step (Step 2) is the interaction of the NetApp with the NEF Emulator, where the latter sends a call back notification to the NetApp asynchronously (Step 3). Moreover, in order to support the asynchronous notification, there was a need for the NetApps to implement its own RESTful API. The final step (Step 4) of the process is the notification that the vApp receives, including the requested data provided by the APIs, implemented within the NEF Emulator.

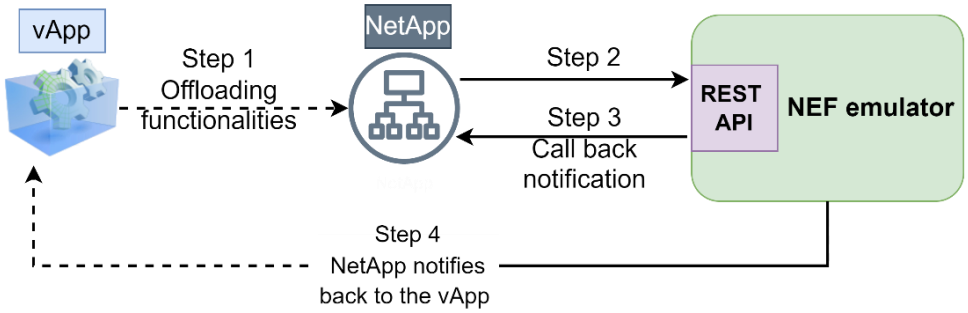


Figure 14 Workflow of the development process -First release of NetApps

All the SMEs have acted according to the aforementioned process in order to reach the delivery of version 1.0 -early prototypes- of their NetApps and the regular monitoring of the results for each step was one of the key responsibilities in the context of WP4 activities.

At the time of writing this document, the NetApp developers have used their own preferred editor, to start the process of developing the core functionalities so as to deliver v1.0 of NetApps, which are presented in detail in the next chapter. The utilisation of the SDK, leading to version 2.0 of the NetApps, is currently an ongoing process, as illustrated in Figure 15 below. All the details for the full-scale integration are planned to be described in deliverable D4.2 "EVOLVED-5G FoF NetApps" (M20), which will also include additional details towards the planned releases of NetApps based on the early prototypes provided by SMEs partners for the four pillars.

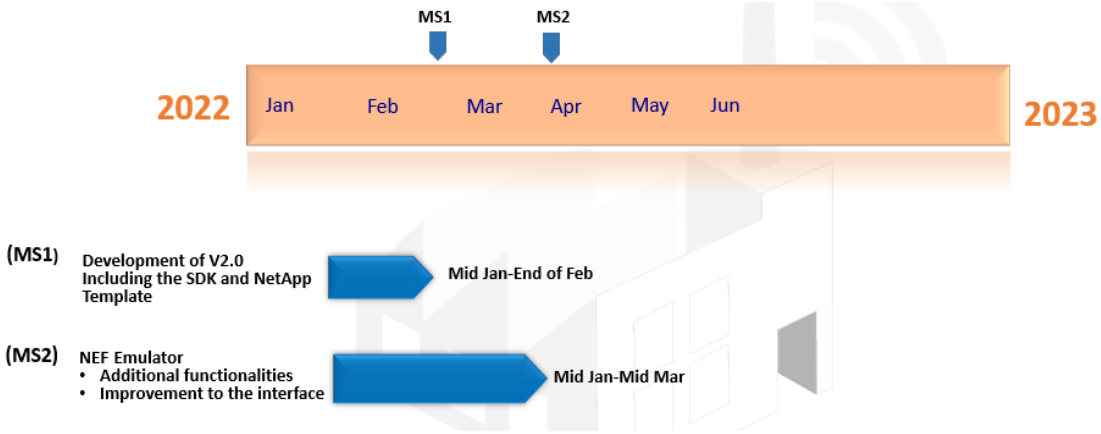


Figure 15 Time-plan describing the delivery of version 2.0 of NetApps

4 NETAPPS AND USE CASES

This section describes the development of the early prototypes (v1.0) for the NetApps within the four pillars in the EVOLVED-5G context: Interaction of Employees and Machines (IEM), Factory of the Future (FoF), Security Guarantees and risk Analysis (SEC), as well as Production Line Infrastructure (PLI). The development process for each NetApp is being based on the workflow, as described in section 3.2.2 and is composed of the following subsections: the split of functionalities between the NetApp and vApp, the use case by which the NetApp will be utilized, as well as the interaction with the NEF Emulator.

4.1 NETAPPS FOR INTERACTIONS OF EMPLOYEES AND MACHINES (IEM)

4.1.1 Remote Assistance in AR NetApp

Within the EVOLVED-5G project, the objective of Immersion (IMM) is to benefit from the new capabilities offered by the 5G network to perform remote assistance in Augmented Reality (AR). Remote assistance is a key task for industry as machines and factories become increasingly more complex. AR is a promising tool to support Industry 4.0 workers. In addition to accessing digital information on-site and in real-time, AR can facilitate remote cooperation between distant users by allowing them to create visual guidance cues to complete audio and video communications.

4.1.1.1 Split of NetApp-vApp

The IMM vApp will be dedicated to the remote assistance in AR. It will handle the communications between the two envisioned users: 1) a technician inside the factory and 2) a remote expert helping the technician. Both users will wear a HoloLens 2 AR HMD and collaborate remotely on an industrial task like machine maintenance.

The role of the IMM NetApp is to facilitate the network QoS monitoring aspects related to the 5G network. Two major roles are planned:

1. Expose business APIs to allow an Augmented Reality (AR)/ Virtual Reality (VR) vApp to easily request a given level of QoS in a high-level manner. This role is illustrated in Figure 16 below.
2. Monitor the state of the network to notify the vApp in case of network issue. In this case, the goal is to add intelligence to the NetApp to allow it to autonomously propose the best compromise between the initially requested QoS and the current state of the network. This added value of the NetApp will facilitate the internal adaptations on the vApp side to match the current network performance.

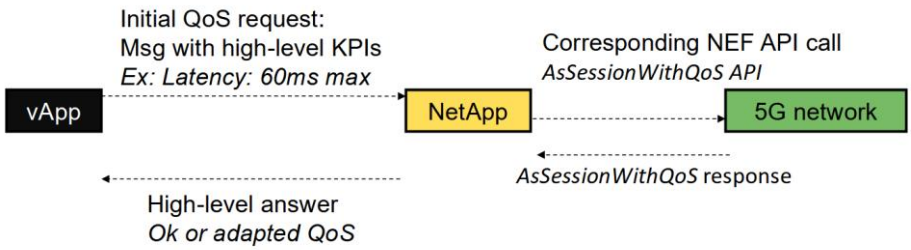


Figure 16 Overview of the first role of the IMM NetApp

4.1.1.2 Use case design-description

Two main use cases naturally emerge from the split between the vApp and the NetApp:

The first use case concerns the establishment of time-sensitive, secure communications between the two remote AR headsets. AR features requires specific network performance to make sure users can collaborate efficiently. Time Sensitive Networking (TSN) is thus necessary for real-time guidance and interaction with virtual objects. The audio and video communications are handled by the vApp using Mixed Reality Web Real Time Communication (WebRTC) technologies. The NetApp will fulfill its first role about managing the initial QoS request from the vApp.

The second IMM use case focuses on the network state monitoring and QoS adaptations performed by the NetApp. Upon receiving a monitoring request from the vApp, the NetApp will create the corresponding subscriptions on the 5G network side. In case of network issue, the NetApp will notify the vApp and propose a temporary QoS. This QoS will be the optimal compromise between the initially requested QoS and the current state of the network. The envisioned Key Performance Indicators (KPIs) are based on the rapidity of the proposed adaptations (time to be notified and time to be operational). The relevance of the adaptations (quality of the compromise) will also be an important evaluation criterion. This use case is illustrated in Figure 17.

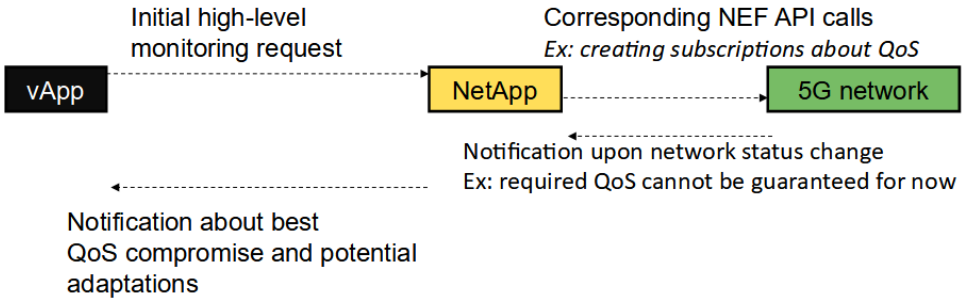


Figure 17 The second IMM use case: autonomous adaptation to network performance and user needs

4.1.1.3 Interaction with NEF Emulator

The vApp running on each AR headset is based on the Unity framework and C#. It communicates with the Python NetApp based on Transmission Control Protocol (TCP) sockets. Currently, the NEF emulator is installed locally and runs on the same machine as the NetApp (for now, an external computer). Upon receiving a high-level request from the vApp, the NetApp can make REST API requests to the NEF emulator. To do so, the NetApp uses the *requests* Python module to make http requests to the corresponding endpoints. To handle bidirectional communications with the emulator and manage callbacks, the NetApp also includes a *Flask* server. This local server allows to easily create different endpoints to receive notifications from the emulator.

So far, basic interaction between the vApp, the NetApp and the emulator have been implemented based on the currently available APIs. The vApp can send an initial QoS request to the NetApp and receive dummy data in exchange. The NetApp can request basic data from the emulator (for instance, information about UEs and cells) and create subscriptions to location monitoring events. The corresponding endpoint is created to receive data from the emulator.

With the recent progress about the QoS APIs implementation (*AsSessionWithQoS*), our current goal is to update progressively the NetApp with the aim to fulfill its first role and answer to the initial QoS request with meaningful data (instead of dummy data).

4.1.2 Digital/Physical twin NetApp

The purpose of this Digital/Physical twin NetApp is to is to take advantage of the possibilities offered by 5G network, to make more efficient twin composite repairs. By connecting the ANITA hot bonder(s) used for an aircraft structure repair [20] to the internet through 5G at the repair area, it will be possible to transmit in real-time all related data to the Engineering Centre of aircraft manufacturer / airline / Maintenance Repair Operations (MRO). This data will then be processed to build a digital twin on which certain computer calculations will be made, or to bake a physical twin that will be controlled mechanically.

4.1.2.1 Split of NetApp-vApp

The NetApp will be responsible for making the necessary requests to the 5G core and then transmitting them to the Anita, more precisely to the software enabling the human/machine interface (vApp).

The Anita's internal software will use the information obtained from the NetApp in two ways:

- To enrich the information about the repair firing, providing more details about atmospheric and altitude conditions for example. This information will be included in the firing report and will be taken into account when studying the digital twin or creating the physical twin.
- The information obtained will also allow the vApp to adapt its performance, depending on the state of the network, the quality of communication etc...

Finally, the vApp could, in case of degraded communication quality, ask to get more priority on the 5G network access, in order to be able to send this data at regular intervals as agreed, and not to risk losing information during the repair process.

To summarize, we can say that the vApp will use the information transmitted by the NetApp, but will also send requests, depending on the information obtained, to modify its behavior.

4.1.2.2 Use case design-description

Depending on the distribution between the vApp and the NetApp, we can refer to three use cases previously highlighted in D2.1 [2].

The first use case is to obtain information on the location of the composite part at the time of repair. The NetApp could provide information on the location of the aircraft location to facilitate the certification process. The vApp will query the NetApp identifying the request, the NetApp will then use the MonitoringEvent API to retrieve the cell coordinates, which it will then pass on to the vApp. This information will be included in the composite curing report or even the archiving process at the remote site.

The next use case consists of a request concerning the quality of service of the 5G communication network. Again, this will be a request made by the vApp, where upon the NetApp will query the AsSessionWithQoS API. Depending on the state of the network and the QoS returned to the vApp, its behaviour will be adapted, as described in the previous paragraph.

Finally, a third use case could be identified here. Unlike the first two, this is not a request for information, but rather a request for adaptation, since in the event of poor communication quality, and in an overloaded environment, the vApp can send a request to the NetApp, and then use the TrafficInfluence API, in order to be benefited from priority access to the data

network. This will be most useful in the vicinity of airports, which is where the majority of composite repair consoles are used.

4.1.2.3 Interaction with NEF Emulator

At this stage of development, the interaction with the NEF emulator is simulated, but not directly linked to the firing cycle. The first call will take place at the start of the repair cycle, and will retrieve the coordinates. As the UE is used on a fixed position, we don't use callback request here, it is only a simple request to the API with specific parameters. Then, several times during the firing, even at regular intervals, the quality of service is analyzed. Callback and notification are used here in accordance with the operation of the relevant API, and as this is a recurring request. Depending on the QoS result, the vApp can select the frequency to send data packets. The overall architecture of the interaction with the NEF Emulator is depicted in Figure 18 below.

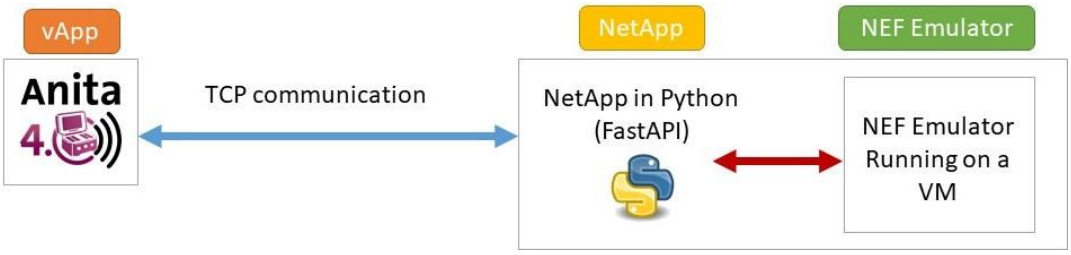


Figure 18 GMI NetApp Architecture

4.1.3 Chatbot Assistance NetApp

The main idea of developing a chatbot assistance NetApp, is to establish a dedicated series of actions that will take place in a factory environment in order to realize the handling of maintenance scenarios via the chatbot platform. This use case targets the ways that maintenance scenarios are taking place within a factory. The goal is to introduce the chatbot as a means of reporting malfunctions and facilitating the course of actions needed to address and fix the reported issue.

4.1.3.1 Split of NetApp-vApp

The NetApp resides between the chatbot/vApp and the 5GS in order to advance the functionalities that are offered by a regular chatbot application.

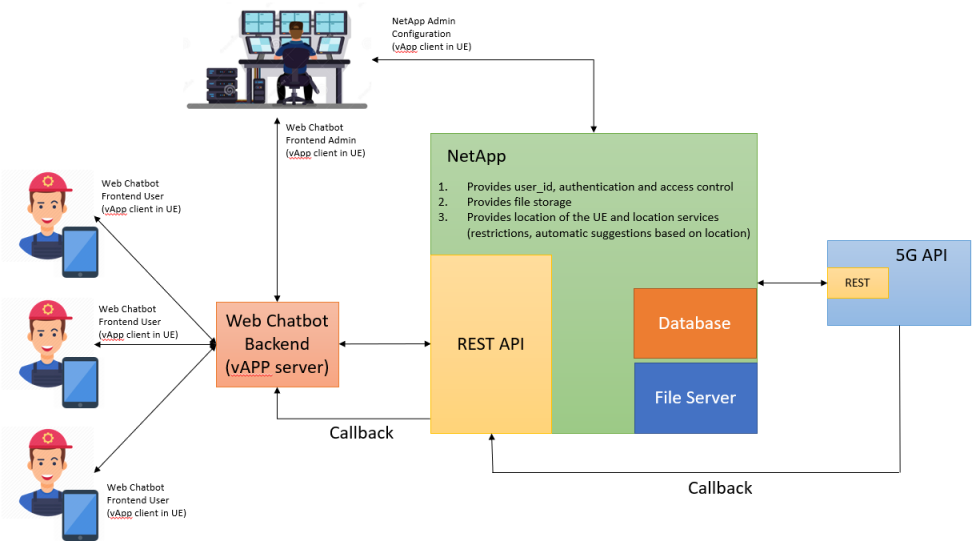


Figure 19 Architecture of vApp-NetApp-NEF APIs

More specifically, the NetApp, by making API calls to the 5G exposure interfaces and receiving data from the internal database and the embedded files' server, as illustrated in Figure 19 provides the functionalities that are listed below:

- Retrieve the UE location via the 5GS. The NetApp, based on the user id, will request from the 5G system to retrieve the location information of the user.
- Correlate the UE-location provided by the 5GS to specific Factory areas. The NetApp will have been properly configured in order to be aware of the factory areas and be able to map the retrieved 5G UE location to each of them. This functionality will allow the worker to easily choose only between the machines that are under this area while it also ensures the workers' safety since this area could be marked as dangerous.
- Prevent unauthorized personnel to perform actions that are not approved by the factory policy. Based on the factory area that has been spotted according to the UE location, the NetApp performs an authorization process of the specific worker to proceed with the maintenance at the specific area. If the worker has not granted the appropriate level of authorization, then the worker in close proximity with the appropriate authorization level should be notified to fix the problem. Otherwise clearance is given to the worker.
- Provide to authorized personnel, access to Maintenance documentation and Service manuals. The NetApp upon successful authorization and clearance of the worker to fix the issue, it will provide access to its data storage in order to store and retrieve maintenance documentation and service manuals. Considering that the use case scenario requires storing input information and displaying documentation files, combined with the local deployment of the system, the NetApp need also to provide access to data storage.

On the other hand, the chatbot application (vApp) will utilize the functionalities provided by the NetApp by making API calls to the respective NetApp endpoints. Additionally, the vApp will handle the display of information to the user/worker in a friendly and easy-to-use interface. More specifically, the vApp will have the following functionalities:

- Provide a series of built-in questions (multiple choice, true-false, free text, etc.). During the displaying of information, it is often needed to interact between the worker and the chatbot application, especially when reporting maintenance issues. It is in chatbot that this interaction is happening in question-answer format. Thus, the vApp will handle the creation and displaying of those questions.
- Create and display multimedia content. During the use of the chatbot application it is often necessary that multimedia content provided by the NetApp can be displayed on the chatbot's screen. Additionally, according to the question, type provided by the vApp, the user input can be a multimedia file.
  Handle user-to-user communication. Whenever a user needs to receive a notification that is triggered either by the control room or by another device, the chatbot backend will make the necessary calls to the NetApp and finally trigger the receiving device.

4.1.3.2 Use case design-description

Based on the split between the vApp and the NetApp that is described in Section 4.1.3.1 the following use case scenario has been identified, proposing an innovative solution by leveraging the advantages and capabilities of the chatbot concept.

Given a 5G non-public network that can be installed in a factory, the NetApp will enable the use of the chatbot (vApp) to help identify and solve possible malfunctions in a shorter time frame using a more user-friendly solution. Also, the 5G network can provide an ID for all the connected workers of the factory and their relative location. It is also important to mention that workers can connect from any device. At any point, a worker might encounter a piece of faulty equipment, and then he/she can use the chatbot app, installed and configured to their device, to report the issue. The procedure followed is described below:

Firstly, the NetApp gets the location of the worker, for example the worker is located in Area A. After checking whether the specific area is safe, and thus evacuation procedure is not innitiated, a second check takes place regarding the access status of the worker in Area A. In case the person does not have clearance to perform any actions in the specific area, he/she is prompted to leave and a worker with access is notified. If the worker is cleared, the predefined reporting procedure is initiated. The worker follows the indications that appear on the chatbot screen in order to properly report the problem. After the reporting is completed the worker can choose to make a request for the machine's manuals so to fix the damaged machinery. Finally, even if the problem is not resolved at that time, an issue is active including a documentation report on the fault.

4.1.3.3 Interaction with NEF Emulator

The current status of development supports the following use case flow which is a subcategory of the use case described in Section 4.1.3.2:

1. The worker opens the chatbot (vApp) and types "work" to instantiate the process.
2. It is being checked whether the worker, according to his id, has clearance to act in the respective area.
3. A list of machines that are malfunctioning in the area of the worker appears on chatbot screen.
4. The worker chooses the machine that he intends to repair.
5. A list of the available files for the specific machine appears on chatbot screen.
6. The worker can view the contents of the files on his smart device.

To support the identified use case, interaction between the NetApp and the 5GS needs to be established. To fulfill that need, currently, the endpoints of the NEF Emulator are used directly. For the function of the above scenario, the information needed from the 5GS is the factory cell, in which the worker is located, and so the NetApp makes a post request on the http://185.184.71.39:8888/api/v1/3gpp-monitoring-event/v1/myNetapp/subscriptions NEF Emulator endpoint to retrieve the respective cell id.

4.2 NETAPPS FOR FOF OPERATIONS (FOF)

4.2.1 Occupational safety analysis NetApp (CAFA SafeLyzer)

CAFA SafeLyzer NetApp will employ computer vision software to detect whether or not Personal Protective Equipment (PPE) such as hardhat, safety glasses are being worn by employees and provides a near real time warning signal directly to the control room safety officer when any element of PPE equipment is not being detected. The video from the factory is collected using a CAFA AMR robot, a wheeled platform that carries stereo cameras that cover a 360-degree field

of view around the robot. The robot has a 5G communication device that transmits the videos to the 5G MEC-based CAFA SafeLyzer vApp, which is used to analyze whether workers are wearing PPE.

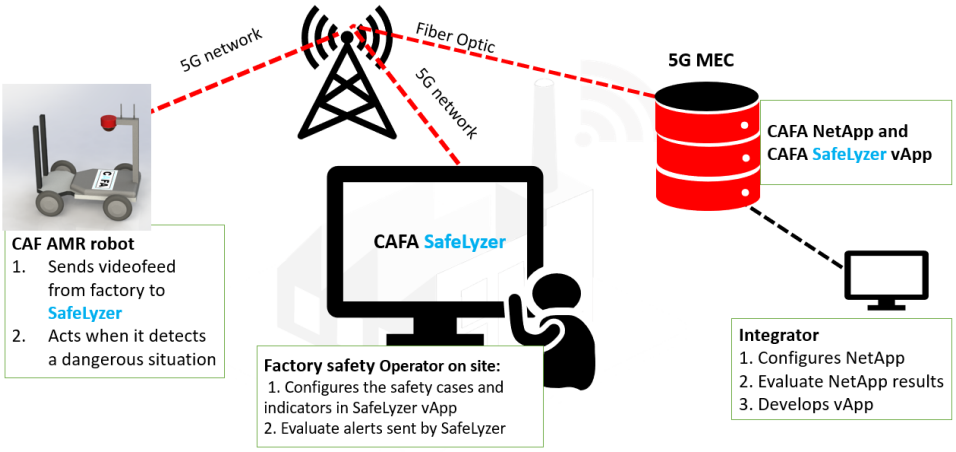


Figure 20 High level Architecture of Occupational Safety Analysis NetApp

4.2.1.1 Split of NetApp-vApp

vApp -CAFA SafeLyzer

The vApp monitors 5G related Service Level Agreeement (SLA) and notifies NetApp and end-user i.e., Factory safety office if SLA is not passing. It provides user interface (UI) for the occupational safety analysis results (if Personal Protective Equipment (PPE) helmet, glasses etc. are missing). The vApp displays (UI) 5G related KPIs (throughput, latency, MEC resources). The vApp adapts the workload if updated resources are insufficient. The CamControl (Client software for robot platform control) displays QoS info and allows remote operator to reconfigure vApp parameters such as temporarily pausing video stream among other options.

NetApp CAFA SafeLyzer

The NetApp receives 5G network information and sends instructions to vApp to switch to another mode of operation if necessary. Also, NetApp automatically gets feedback from the NEF Emulator and sends relevant network information to vApp.

4.2.1.2 Use case design-description

The use case can be comprised of three steps:

- NetApp receives information about 5G network conditions by setting up subscriptions with certain data throughput levels to be notified automatically as soon as those criteria cannot be met.
- NetApp detects that required data throughput is not passing (based on the automatic notification from NEF Emulator) and notifies end-user i.e., Factory safety officer and robot's remote operator as well as sends instructions to vApp to switch operating modes to less demanding.
- After some time, Network conditions improve. NetApp automatically receives feedback from 5G infrastructure manager about restored data throughput conditions and sends complementary instructions to vApp to resume operation at former levels.

4.2.1.3 Interaction with NEF Emulator

In order to achieve the interaction with the NEF Emulator the NetApp opens a port and listens for incoming asynchronous messages about 5G network conditions (guaranteed data throughput related messages) utilizing the AsSessionWithQoS API. This subscription-based communication prevents unnecessary back and forth messaging in most use cases. Also, subscriptions can be canceled if network conditions are poor and robot cannot perform its basic tasks anyway.

4.2.2 Industrial grade 5G connectivity with assured QoS and integrated SLA/SLS monitoring capabilities NetApp

The goal of the NetApp is to enhance the IoT monitoring platform, a distributed cloud-based monitoring solution with 5G capabilities, management system for remote gateway and sensor control (a range of IoT sensors for environmental sensing), analytics, and advanced visualization, system notification and alerting for collected data. In-build 5G capabilities enable end-to-end network performance monitoring based on automated collection of various radio, network and cloud related performance metrics. Continuous performance monitoring enables additional Industry 4.0 use cases such as continuous network and services (SLA/SLS) monitoring and application continuity check.

Envisioned 5G technological advancements such as eMBB, mMTC, URLLC, NFVI and MANO and its incorporation into the IoT/M2M-based remote monitoring platform will enrich system capabilities with the industrial-grade and mission-critical capabilities that are required to build the Factories of the Future. In particular, the following are the main technological benefits the solution will gain due to the 5G:

- Extending capabilities of the IoT/M2M system components (management, data collector, analytical modules) to support automated deployment, components scaling and life-cycle management in the industrial 5G environments.
- Reduced IoT/M2M system service deployment time.
- Extending the capabilities of the IoT/M2M Gateway to support operation with 5G NSA/SA (eMBB, URLLC and mMTC) capabilities.
- Extending the network performance monitoring capabilities of the system to support the collection of Industry 4.0 network and application specific metrics.
- Technological and operational validation, interoperability check and verification of the system operational use in the Industry 4.0 environments.

4.2.2.1 Split of NetApp-vApp

vApp, as part of the IoT monitoring solution/platform, is primarily responsible for collecting data sent by environmental sensors and certain 5G-related data/KPIs. It also provides an UI to display the aforementioned sensors' and 5G-related data. As well, vApp provides management user interface (UI/API) for the platform and IoT/M2M gateways connected to it. In terms of vApp – NetApp relations, vApp provides NetApp with SLA requirements.

NetApp's responsibility in relation to the vApp is to configure and monitor the 5G radio network QoS according to the vApp SLA requirements. In order to be able to perform this, it retrieves the service monitoring parameters, it retrieves the analytics information exposure parameters, and it retrieves the IoT/M2M gateways locations.

The overview of the FoF IoT monitoring solution and its split to vApp and NetApp is depicted in Figure 21.

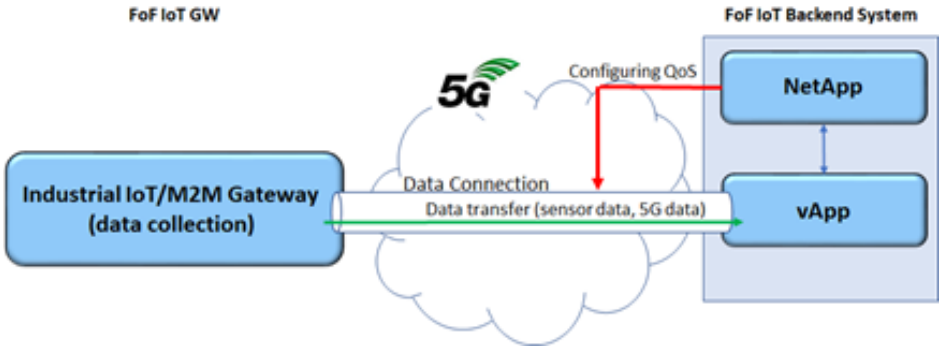


Figure 21 Overview of vApp's and NetApp's functions within the FoF IoT management solution.

4.2.2.2 Use case design-description

The use case of the FoF IoT monitoring solution/platform focuses on FoF environment with strict requirements for collecting, analyzing, and representing/visualizing data from sensors included into the industrial process. In order to achieve requirements defined by SLA (throughput, latency, error rate, etc.), vApp of FoF IoT monitoring solution also collects and visualizes certain 5G-related parameters. The user has, among others, a possibility to select the preconfigured slice to accommodate the SLA requirements according to the needs of the certain industrial process, which means each FoF IoT GW connected to the platform (FoF IoT backend system) can have its arbitrary requirements set.

Initially, the NetApp provisions the 5G radio network according to the vApp SLA requirements. Based on comparison of the SLA requirements and the actual status of the data connection, the vApp notifies the NetApp in case of an action due to the SLA not passing is required. The NetApp retrieves 5G monitoring service parameters and Analytics Information Exposure from the NEF (i.e., coverage prediction, signal strength) and executes re-configuration of the 5G radio network with new optimal parameters to achieve expected SLA. The NetApp requests network slice re-configuration via NEF northbound API (or CAPIF when ready).

The solution targets emerging sectors such as ports, Industry 4.0, critical communications, and similar. More in-depth overview of the solution and its components is depicted in Figure 22.

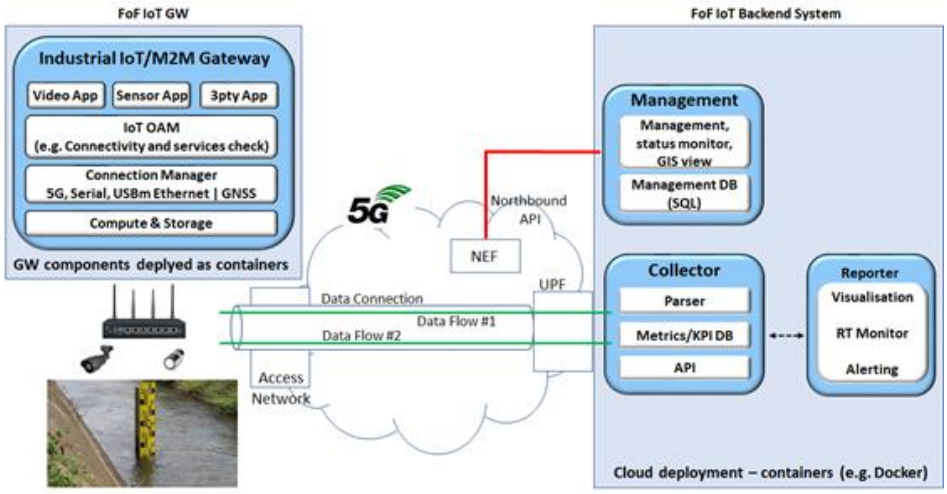


Figure 22 Detailed overview of the solution and its components

4.2.2.3 *Interaction with NEF Emulator*

During the boot up procedure, the NetApp authenticates with 5G NEF (Emulator) API, and subscribes to the NEF monitoring updates/events. At the same time, the NetApp gets the UE identification to emulate a corresponding IoT GW, followed by pushing initial configuration to it. Upon receiving an update from the NEF, the NetApp pushes received KPI to the "IoT collector" (via web hosted API) to be further parsed and stored in the database already containing 5G KPIs gathered by "IoT Gateway". This enables additional data enrichment, correlation options and analytics provided by the Grafana based "IoT Reporter" aimed for data visualization. The interactions are graphically depicted in Figure 23.

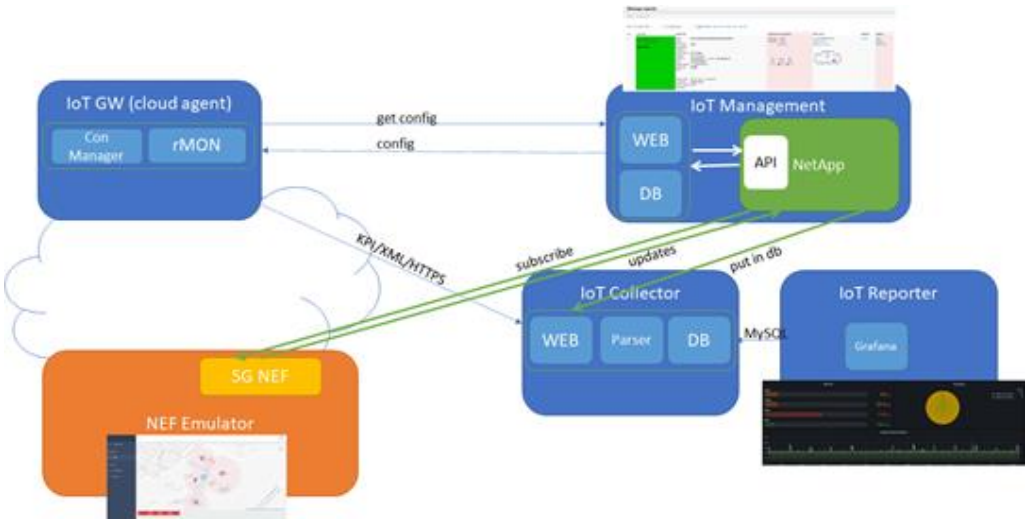


Figure 23 Interaction with NEF Emulator

4.2.3 5G network anomaly detection NetApp

Anomaly Detection in the network context is vital to the management of the network infrastructure and services health during operations. Moreover, in the industrial network context and with the emergence of 5G for Non-Public Networks, it is expected to be an indispensable component of future deployments. However, the current integration with the specific to 5G core events and monitoring is in its infancy adopting in most of the applications proprietary or intrusive methods for retrieving 5G network information in order to detect anomalies. The EVOLVED-5G network anomaly detection application is based on concepts originally applied to wireless and wired networks that demand the deployment of the application within the protected/monitored network domain. In order for the application to be able to detect and identify anomaly network attributes, device context, monitoring information and access to traffic flows is required. The architecture of such an application can be decomposed into the following basic functional components, as depicted in Figure 24.

- Monitoring components – Components that allow ingestion of monitoring information from various locations of the infrastructure and services. The components may be just consumers of APIs from available monitoring elements at the place of deployment or the application itself integrated in the network infrastructure able to collect data (i.e., bump in the wire as depicted in Figure 24).

- Policy component – The component that allows the introduction of user and network attributes as well as associated policies. For example, denote MAC/IP addresses of local network devices, operational context etc.

- Detection and mitigation components – Components that implement heuristic or ML based algorithms to allow for the detection and identification of anomalies and actions to mitigate the anomaly or provide alerting
- Visualization and Management components – Components that offer dashboard capabilities for management of the operation of the application as well as immediate monitoring of the current situation.

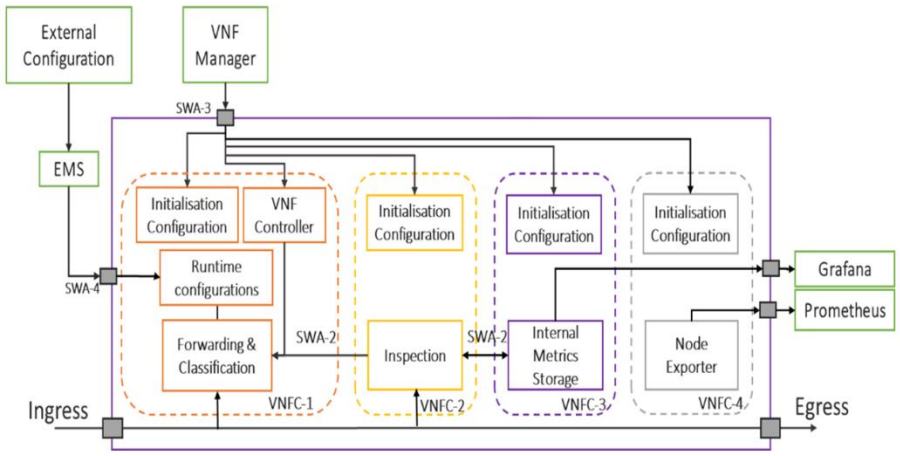


Figure 24 Architecture of Anomaly detection application

Approaching the anomaly detection for 5G deployments in Industry 4.0 setting via the EVOLVED -5G framework, the Anomaly Detection Application is split in two functional structures. The following subsection provides information for the functional split of the Anomaly Application into NetApp and vApp in order to take advantage of EVOLVED-5G capabilities.

4.2.3.1 Split of NetApp-vApp

The overall NetApp-vApp split of the EVOLVED-5G based Anomaly Detection is illustrated in Figure 25. As can be observed, it provides for the retrieval of 5G network information as those are exported by the NEF function, following the relevant standards. The NetApp has a southbound interface that provides APIs allowing communication for data and control with the vApp.

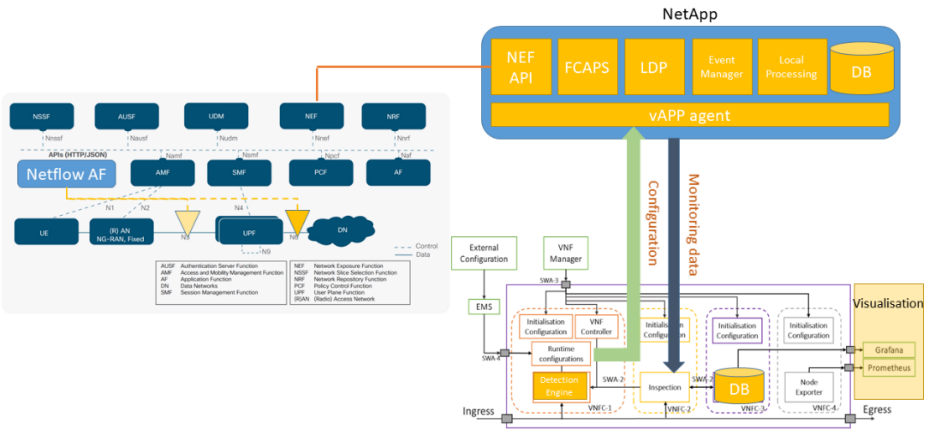


Figure 25 NetApp-vApp split

The API has open specifications so that different vApps may also take advantage of the NetApp features. Through the API the following actions are available:

- Management and Configuration Policies
- Initialization of NetApp
- Maintenance
- Update of local detection algorithms and model Monitoring and visualization
- Alerting

As Figure 25 illustrates, the functional split between the NetApp and vApp can be modular. The NetApp can operate either as a proxy conveying much of the information retrieved from the NEF to the vApp to allow for detection, alerting and mitigations, or as a standalone app offering limited detection capabilities suitable for the intended environment. In the latter case, the vApp role is mostly for management and monitoring purposes.

4.2.3.2 Use case design-description

The main objective for the proposed application is to exploit the features and information that is made available by the NEF to tighter integrate the Anomaly Detection Application with the 5G environment. This objective is achieved by the EVOLVED-5G Architecture. The envisaged use case will explore a multiplicity of anomaly scenario, tightly aligned with production process and the machinery interactions. Two main scenarios are considered, anomaly detection based on location information in the Unmanned Ground Vehicle (UGV) scenario considering restricted zones and abnormal trajectories of autonomous mobile equipment and a more network anomaly related scenario where information related to 5G the network status of the factory plant networking infrastructure through NEF.

4.2.3.3 Interaction with NEF Emulator

The objective of the NetApp is the consumption of a variety of information, depending on the capabilities of the NEF implementation and correlate them for achieving timely detection of anomalies. Currently the considered inputs as those are detailed in 3GPP standards are:

- Analytics Exposure - information that will allow identification of false positives or more focused analytics for the flows.
- Abnormal Behaviour
- Congestion analytics
- Network Performance
- QoS sustainability
- 5GLAN parameters
- Netflow information - an AF may be required to provision parameters to the Analytics Exposure
- Service Parameter – information related to the service status, UEs etc. Rich information at a Service base level using also NSSAI information

The first implementation of the NetApp – NEF API implementation supports Location and QoS information that are available by the NEF Emulator. The supported scenarios are based on this initial set of information that can be retrieved.

4.2.4 5G agriculture use case: Smart irrigation and agricultural drones

Smart irrigation consists of the use of sensors to measure humidity, soil moisture, etc. to decide with precision the water requirements of the crops. In this context water sensors, micro-tensiometers and humidity sensors will be connected to dataloggers. Dataloggers are multi-purpose measurement and control devices that are very flexible in terms of sensor compatibility, communications, supported protocols and power options. Dataloggers have a number of analog inputs for the interconnection of the sensors cited previously. Regarding the communications supported, the datalogger directly connects to Ethernet with 10/100 Ethernet RJ-45 or over USB (virtual Ethernet), which will be used to connect the datalogger to a 5G router. The datalogger includes programming capabilities for the development of a vApp that collects the measurements from the sensors, requests the location to the NetApp and sends the data to the vApp server that aggregates the data from all the datalogger, though depending on the available support in the particular models used on the field an external microcomputer may be used for extra computation capabilities.

In addition to dataloggers, agriculture drones are used to monitor crop health, scan areas, agriculture photography etc. In this case, a drone equipped with a hyperspectral camera can send images to the NetApp that can be geolocated and made available to a vApp. We plan to study this approach, as well as the usage of the Quality of Service (QoS) APIs for managing the transmission of data from and to the drone in a future phase, when a more stable version of the smart irrigation approach is established.

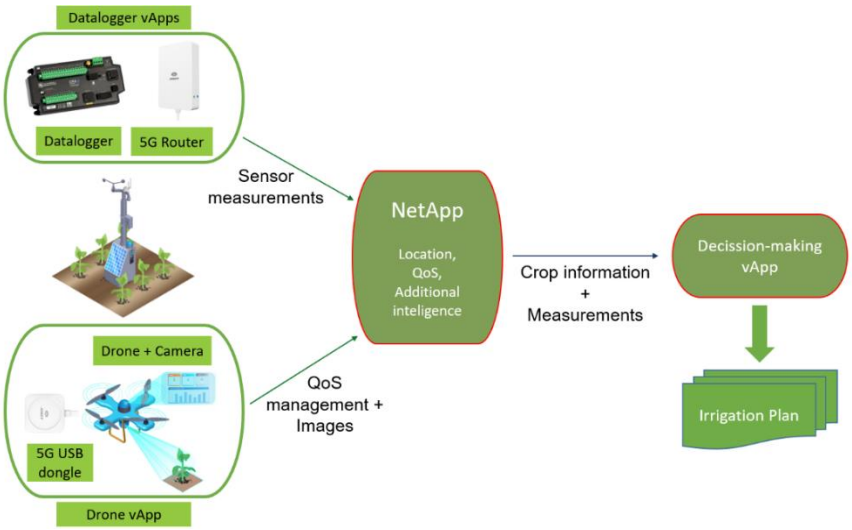


Figure 26 Smart Irrigation use case description

4.2.4.1 Split of NetApp-vApp

In the first development phase the NetApp will be used to store the data generated by a set of dataloggers distributed in the terrain. Each datalogger will send data at regular intervals, which is then geolocated using the functionalities provided by the NEF Emulator and complemented with additional details, such as the kind of crops in the vicinity.

On the other hand, the vApp retrieves the aggregated data from the NetApp, and uses a decision-making algorithm in order to generate the best irrigation plan possible, considering the levels of humidity, temperature and crops in each zone, the historic evolution, seasonal information, etc. The overall approach is depicted in Figure 26.

4.2.4.2 *Use case design-description*

The objective of this use case is the optimization of hydric resources in a plantation. For this, it is necessary to obtain at regular intervals data that describes the current status of the different zones in the covered terrain, including, for example, temperature and humidity in the soil and the nearby crops.

This data is obtained by a set of sensors distributed in the terrain and that are connected to dataloggers. These dataloggers (through the datalogger vApp) will perform an initial computation on the results generated by the sensors, which may use heterogeneous formats, and are sent via 5G connectivity to the NetApp.

When the NetApp receives information from any of the dataloggers, it makes use of the functionality provided by the 5G Network (in this case the NEF Emulator) in order to geolocate this information, which is then enriched with additional data and added to the historic information already contained in the NetApp. By using all the aggregated data in the NetApp, as well as additional information such as the available levels of water for each zone or seasonal information, the vApp makes use of a decision-making algorithm in order to generate an optimized irrigation plan for the crops.

4.2.4.3 *Interaction with NEF Emulator*

Figure 27 shows the general architecture of the use case in this first implementation phase. For the time being, the real dataloggers have been replaced by a set of small applications that send randomized data to the NetApp, in order to simulate the behavior of the real equipment without the need of an existing deployment in the plantation.

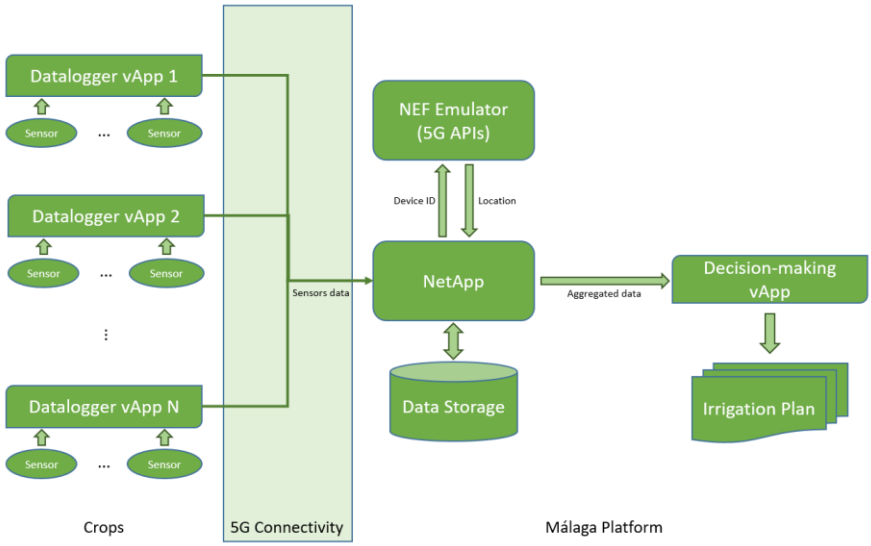


Figure 27 Smart Irrigation use case architecture

The simulated datalogger vApps send the crop information to the NetApp: for example, temperature and humidity. This is done at regular intervals, so we can have consistent information about the evolution of these values.

The NetApp takes all these measurements, and saves them as historical data. In order to accomplish this, we will also need to save the current date and time, in order to know when each record was stored. The NetApp obtains the cell information for each record by querying this information to the NEF Emulator, which allows us to properly know where the crop is

located. The NetApp also stores the crop type and the type of measurements that were used for registering this data.

For the first version the NetApp exposes the following endpoints:

Cell management:

This group of endpoints is used for the creation and management of the different areas handled by the NetApp. Areas are defined by the ID of the cell that provides coverage to the zone, and the kind of crop located in the terrain.

Endpoint	Method	Description
/api/v1/cells /api/v1/cells/[cellNumber]	**GET**	Sends all the cells data stored in the database by default or data of a specific cell if there is a cellNumber in the endpoint. Example: get("http://localhost:5000/api/v1/cells/[cellNumber]")
	POST	Creates a new cell entry in the netApp database. Example: post ("http://localhost:5000/api/v1/cells", data={'cell_num':'','crop_type':''})
	DELETE	Deletes the cell with the given cellNumber. Example: delete("http://localhost:5000/api/v1/cells/[cellNumber]")
	PUT	Updates the cell that has the cellNumber given in the endpoint. Example: put("http://localhost:5000/api/v1/cells/[cellNumber]", data={'cell_num':'' ...})

Historic Management:

Endpoints in this group are used for the creation of separate measurement records, as well as for obtaining the aggregated data contained in the NetApp.

Endpoint	Method	Description
api/v1/historics/[cellnumber]	**GET**	Sends the historics attached to a cell of a given cellNumber. Example: get("http://localhost:5000/api/v1/historics/[cellnumber]")
	POST	Creates a new historic and attaches it to a given cellNumber.

		Example: post("http://localhost:5000/api/v1/historics/[cellNumber]",data={'temperature':'','humidity':''})
	PUT	Updates the historic that has the HistoricId given in the endpoint. Example: put("http://localhost:5000/api/v1/historics/[HistoricId]", data={'temperature':'' ...})
/api/v1/historics/[HistoricId]	**DELETE**	Deletes the historic which has the HistoricId given in the endpoint Example: delete("http://localhost:5000/api/v1/historics/[HistoricId]")

4.3 NETAPPS FOR SECURITY GUARANTEES AND RISK ANALYSIS (SEC)

4.3.1 Traffic Management NetApp

Eight Bells' NetApp, being developed for security guarantees risk analysis pillar and in the scope of the EVOLVED-5G project, offers the following services:

- The accurate measurement of traffic over an interface, specific to a device and perform a simple check of "unregistered" traffic outside the 5G Core network.
- The lessening of the burden on a reportedly congested device in the network, applicable to many different aspects of a FoF.
- Real-time monitoring of events and managing the security of IP
- To offer security in 5G networks

The objectives and services mentioned above are being offered with the help and use of the NetApp and the vApp which are briefly described I the next subsections.

4.3.1.1 Split of NetApp-vApp

As it can be seen in Figure 28 Eight Bells' NetApp is accessing the NEF Emulator through specific APIs and communicates with the L7-Switch (vApp) though a secure shell (ssh) link. The 5G core is depicted in a Service-Based Architecture (SBA). NetApp gets ipv4 of all registered UEs from NEF and also performs a simple QoS check in every cell, based on the radius of the cell and the number of UE's inside. 8BELLS L7-aware White Box Switch is an intelligent traffic steering mechanism which configures chains of service functions, in sequence and number Dynamic Service Function Chaining (SFC) based on higher layer inspection (Layer 7).

4.3.1.2 Use case design-description

The NEF northbound API provides a rich set of APIs that allow third-party authorized AFs to monitor and configure the network's behavior for a number of subscribers. The 5GC network uses NEFs to create and expose standard APIs to the internal and/or external developer ecosystem.

Exposure enables:

- Hiding the complexity of the underlying network
- Secure/controlled access of the network to external AFs
- Monetization of the network capabilities

The NEF can expose network APIs required by specific AFs.

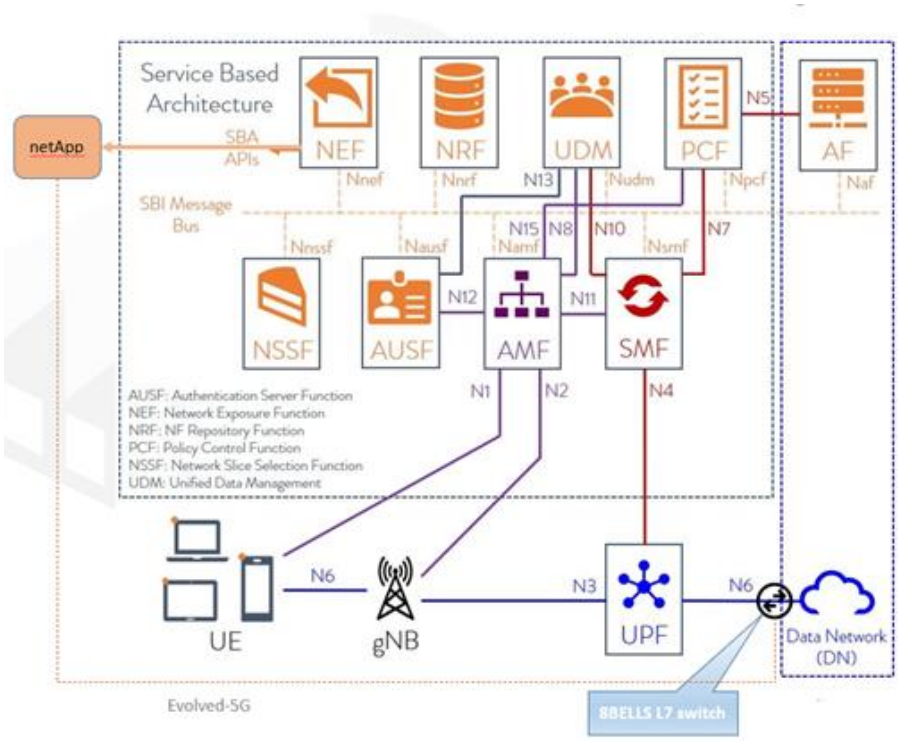


Figure 28 Separation of vApp and NetApp

The main operation and mechanism (for detecting congestion and limits or redirects) being performed by Eight Bells proposed and developed system, is the following:

- L7 Switch requests for congestion statistics from NetApp which is subscribed to AnalyticsExposure API
- The processed congestion info (of NetApp) supplies the list with devices/destination IPs with high congestion stats
- L7 Switch requests for new congestion statistics from NetApp
- The processed congestion info resupplies the list with devices/destination IPs with high congestion stats
- L7 Switch requests for further processing of traffic Filters which are subscribed to AnalyticsExposure API
- The processed congestion info resupplies the list with devices/destination IPs with high congestion stats
- No further processing takes place

4.3.1.3 Interaction with NEF Emulator

The NetApp is developed using flask-python. The NEF emulator runs locally, on the same machine as the NetApp. Currently NEF supports two APIs: one for location and one for QOS monitoring. Currently, the NetApp uses post and get requests to those endpoints of the NEF emulator, in order to get notifications and data that will be used by the vApp in its current version, the NetApp utilizes AsSessionwithQoS API in order to get the IP addresses of all

subscribed UEs from the emulator, and the endpoint regarding the cell id of UEs, in order to detect congestions.

4.3.2 ID Management and Access Control NetApp

The objective of IQB in the scope of the EVOLVED-5G project is to ensure that the new capabilities offered by the 5G network are consumed in a secure fashion. The fact that a variety of industries benefit heavily from the new 5G capabilities, entails the risk of disregarding security, in order to capitalize on them expeditiously. In addition to offering authentication and authorization services, this NetApp can be extended to support monitoring capabilities such as logging API responses, as well as event generation in case of unusual behavior based on aforementioned logs.

4.3.2.1 Split of NetApp-vApp

The ID Management and Access Control NetApp does not extend to a vApp. It rather acts as an intermediary NetApp, operating as a point of Authentication and Authorization of third-Party NetApps to consume the 5G NEF APIs.

4.3.2.2 Use case design-description (based on the split)

The IQB NetApp has the purpose of authenticating other NetApps using OpenID Connect (a simple identity layer on top of the OAuth 2.0 protocol [22]), as well as authorizing the consumption of NEF APIs under the event of successful authentication. The NetApp then acts as an intermediary NetApp, carrying the requests of other NetApps to the respective NEF APIs, as well as returning the responses of the APIs to the corresponding NetApps. Those responses can be monitored and logged, which allows to expand the security of the NetApp security by examining the logs and generating events in case of unusual behavior.

First, IQB NetApp authenticates itself towards the NEF Emulator, as described in the sequence diagram below:

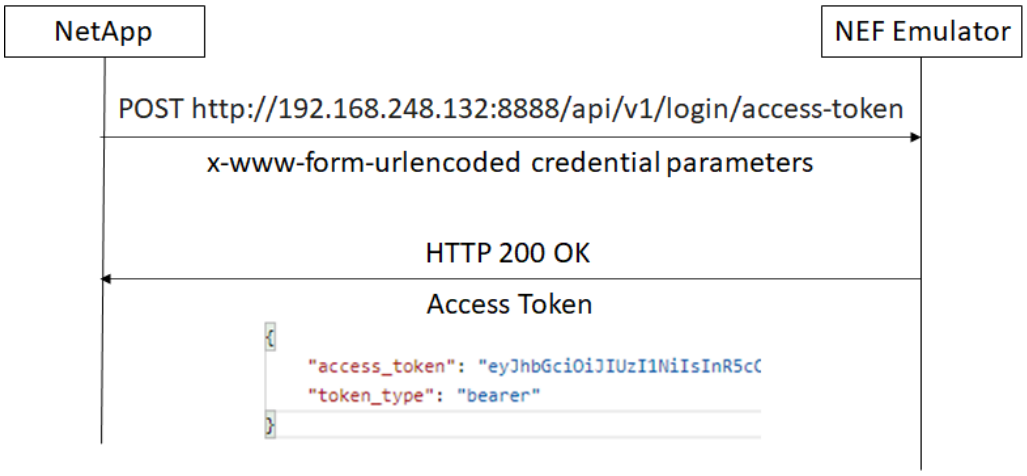


Figure 29 IQB Netapp authenticates towards the NEF Emulator

After successful authentication, the NetApp can perform requests towards the NEF Emulator and have two-way communication. The third-Party NetApp needs to authenticate to the IQB NetApp using OpenID Connect, in order to be authorized to consume NEF APIs.

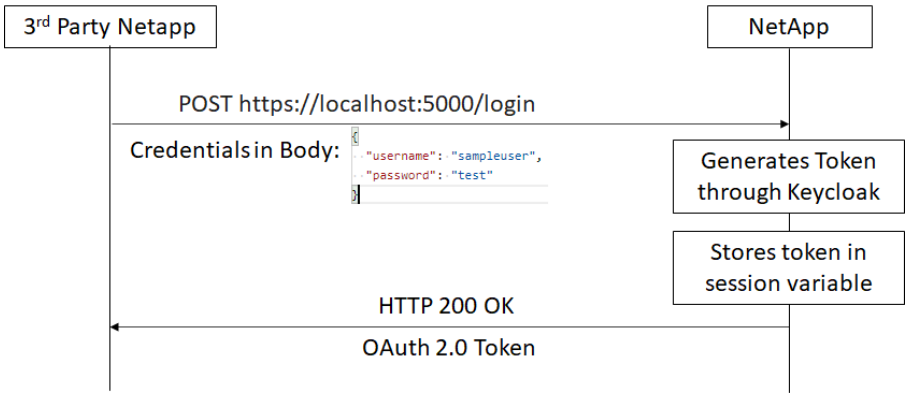


Figure 30 Third-Party NetApp authenticates towards the IQB NetApp

Now that the third-Party NetApp, the IQB NetApp, and the NEF Emulator have set up an authentication chain, the third-Party NetApp can execute requests through the IQB NetApp:

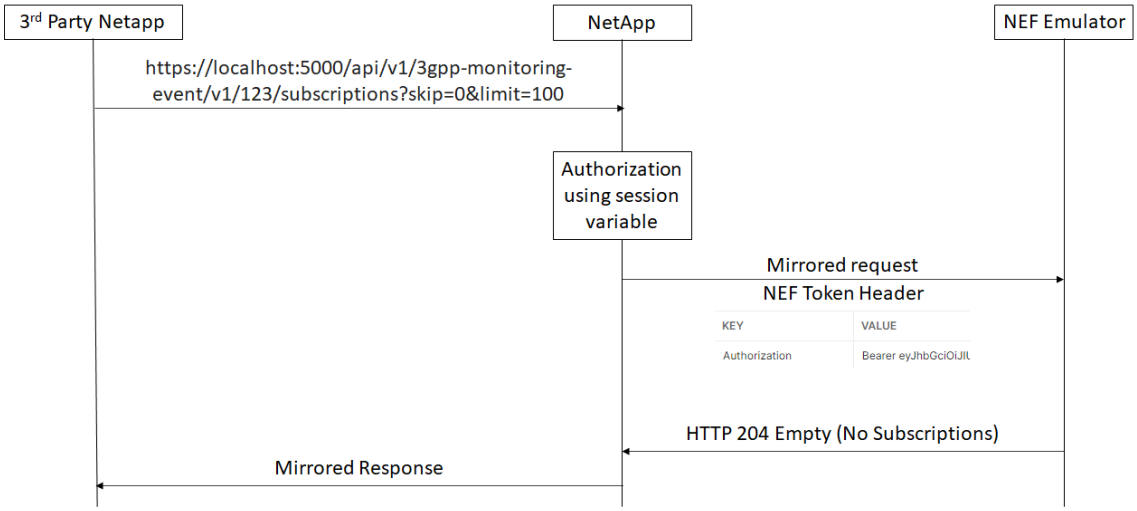


Figure 31 IQB NetApp Use Case

The NetApp is developed in Python. On the event the NetApp receives a request to login, the NetApp authenticates the request's issuer using OpenID Connect. In case of successful authentication, the Oauth2 token derived from OpenID Connect is stored inside flask's session variable. This is made possible utilizing the Python flask server to create endpoints for authentication and authorization. After successful authentication, the client can request usage for a NEF API. The NetApp confirms the issuer is authorized to access the respective endpoint using flask's session variable, and executes the request towards the NEF Emulator using the Python *requests* module. The response received from the NEF Emulator is then carried over to the client.

4.3.2.3 Interaction with NEF Emulator

The NEF emulator is currently installed locally and running on the same machine as the NetApp. An instance of Keycloak, which is an open-source Identity and Access Management solution [21], is also running as a service, inside a docker container on the same machine.

The NEF Emulator in its current state supports APIs for QoS and location monitoring. For both APIs, the NEF Emulator uses callbacks to notify clients in case of a specific event. Regarding the Monitoring API, the emulator currently performs a callback when a device connects to a different cell ID. The ID Management and Access Control NetApp can monitor and log these callbacks. It is then possible to examine the logs and generate alerts in case of unusual behavior, therefore adding an extra intelligent layer of security.

4.3.3 5G Security Information and Event Management NetApp

Security Information and Event Management (SIEM) is a software system capable of offering security protection on an underlying IP network. More specifically, SIEM can perform real-time monitoring and analysis of events as well as tracking and logging of security data for compliance or auditing purposes. SIEM's main services are the following:

- Vulnerability Assessment: Execute authenticated and (in some cases) un-authenticated scans, periodically or manually.
- Intrusion Detection: Identify network traffic-based and web traffic-based anomalies (Structured Query Language (SQL), Cross Site Scripting (XSS)), network-based and host-based anomalies as well as file integrity monitoring (FIM).
- Behavioral Monitoring: Analysis of the anomalies detected through behavioral monitoring process
- Security intelligence: Chained events are examined through correlation rules.

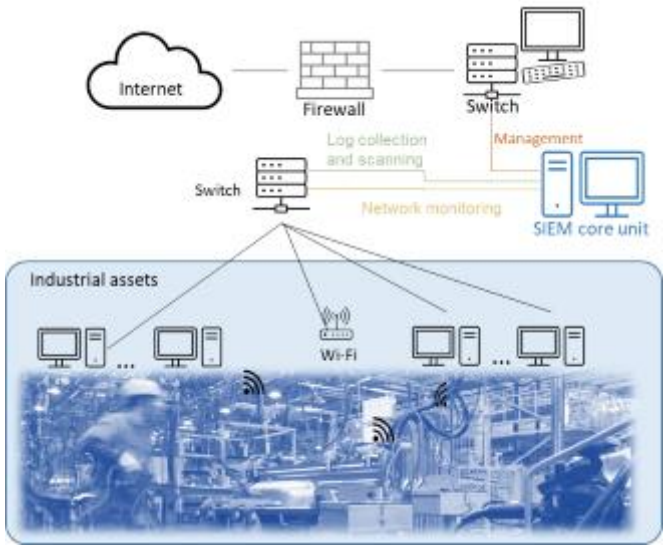


Figure 32: FOGUS Vertical Application

The SIEM software, as depicted in Figure 32, can be deployed as a node of an industrial infrastructure network and manage the security of all the rest IP devices. However, as the concept of Industry 4.0 evolves, apart from the IP network, industries will progressively establish small or large-scale 5G Non-Public Networks (NPN) to their premises, in order to exploit the advanced capabilities of 5G technology (low-latency, high throughput etc.) for a set of their equipment. As such, the implementation of enhanced mechanisms to manage and ensure security in this industrial ecosystem, is mandatory.

FOGUS, in the context of EVOLVED-5G, will develop a NetApp that bridges the communication gap between the SIEM and the 5G NPN devices in an industrial environment, thus extending SIEM capabilities to 5G.

4.3.3.1 Split of NetApp-vApp

The software that is used as vertical application by FOGUS is the open-source SIEM solution OSSIM, developed by AT&T Cybersecurity, former Alienvault. OSSIM functionalities include:

- Collecting event data from security logs
- Converting selected data to a readable format
- Analyzing data to detect possible threats
- Generating alerts for the security administrator
- Creating network security reports

OSSIM has a dedicated dashboard, in which the administrator can have access to the security condition of the network devices monitored by the software, as illustrated in Figure 33 and Figure 34.

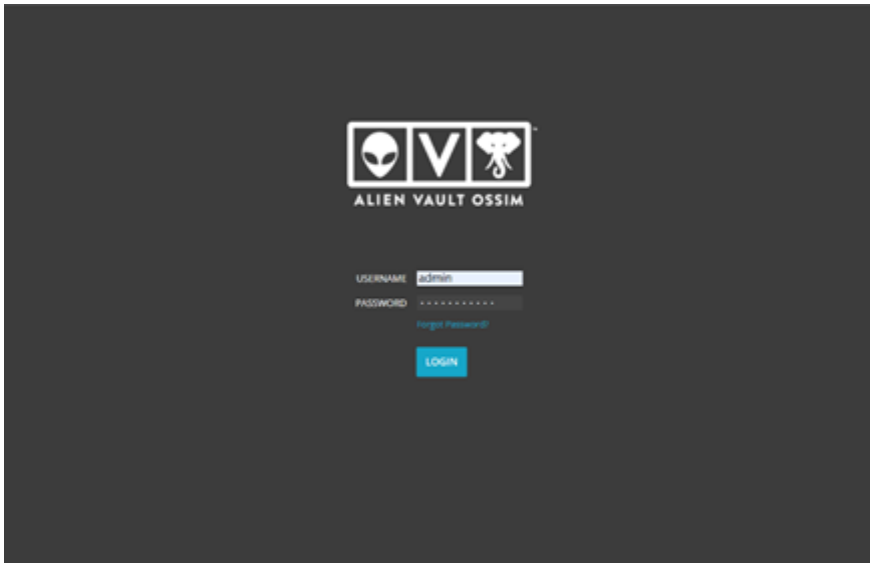


Figure 33: OSSIM login page

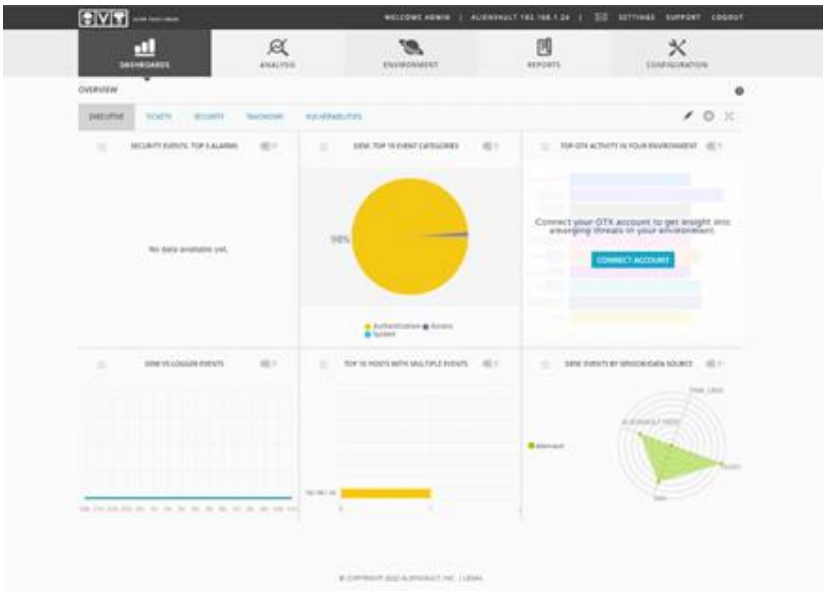


Figure 34: OSSIM dashboard

The architecture that is followed by FOGUS for the NetApp development is the standalone one. The NetApp is developed as a separate software component communicating with the vertical application, as well as 5G NPN. The NetApp resides inside the local network for the first stages of the project, as depicted in Figure 35.

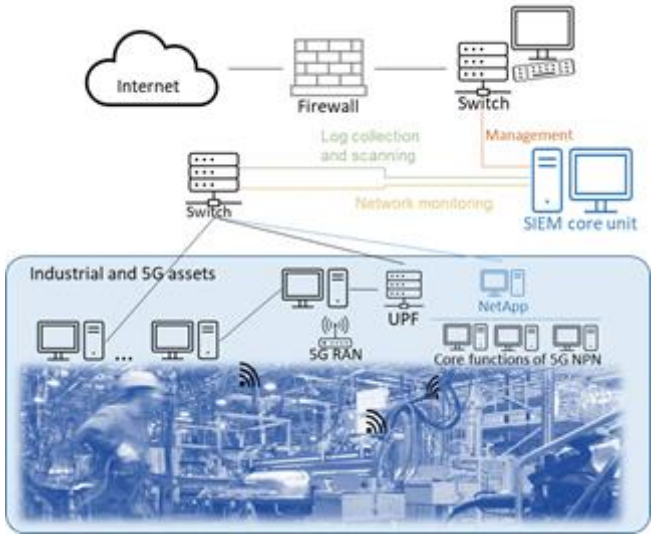


Figure 35: vApp-NetApp-5G NPN architecture

From a technical perspective, the 5G SIEM NetApp is a containerized Python application exposing a set of REST API endpoints, built on Django framework, that initiate NetApp interactions either with OSSIM (through connections to its database) or with the 5G NPN (by calling the exposed NEF APIs). Overall, the first release of 5G SIEM NetApp has the following capabilities:

- Receives data from 5G, through the exposed NEF APIs (Location Monitoring Event and Analytics Exposure API)
- Converts the collected data/logs/events, taken from 5G, to readable format for the SIEM
- Sends formatted data to SIEM

4.3.3.2 Use case design-description

The objective of 5G SIEM NetApp is the reinforcement of network security on a FoF environment by expanding SIEM capabilities to 5G networks. As a result, the use case considers a set of devices in an industrial environment, connected either with IP network or 5G NPN, whose security condition has to be monitored though the OSSIM dashboard. While the OSSIM is already able to detect the IP devices, the NetApp will assist the SIEM system to detect and monitor the devices managed by the external 5G non-public network. In addition to that, the OSSIM should also receive frequent updates on the security status of the 5G devices and generate alerts to the dashboard in case of a possible threat or anomaly.

4.3.3.3 Interaction with NEF Emulator

In the first release of 5G SIEM NetApp, 5G NPN network is emulated by the NEF Emulator component. NEF Emulator is deployed on a separate machine of the local network. The simulated 5G network is created by the developer through the graphical interface offered by NEF component. When the simulated 5G network is ready, the NetApp can make an authorization request to the NEF Emulator to get an access token. Having this access token, the NetApp is authorized to use the exposed APIs. Currently, 5G SIEM NetApp uses only the

Monitoring Event API to receive (once or multiple times) the location of the cell that a device is connected. The information received is then passed to OSSIM and depicted on the dashboard (see Figure 36 and Figure 37).

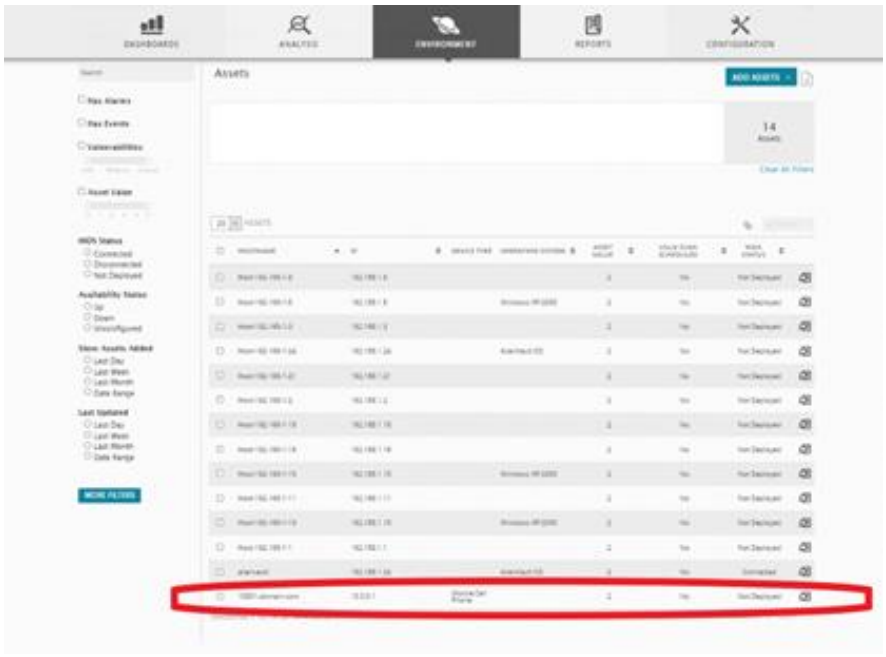


Figure 36: 5G device included in the monitoring assets

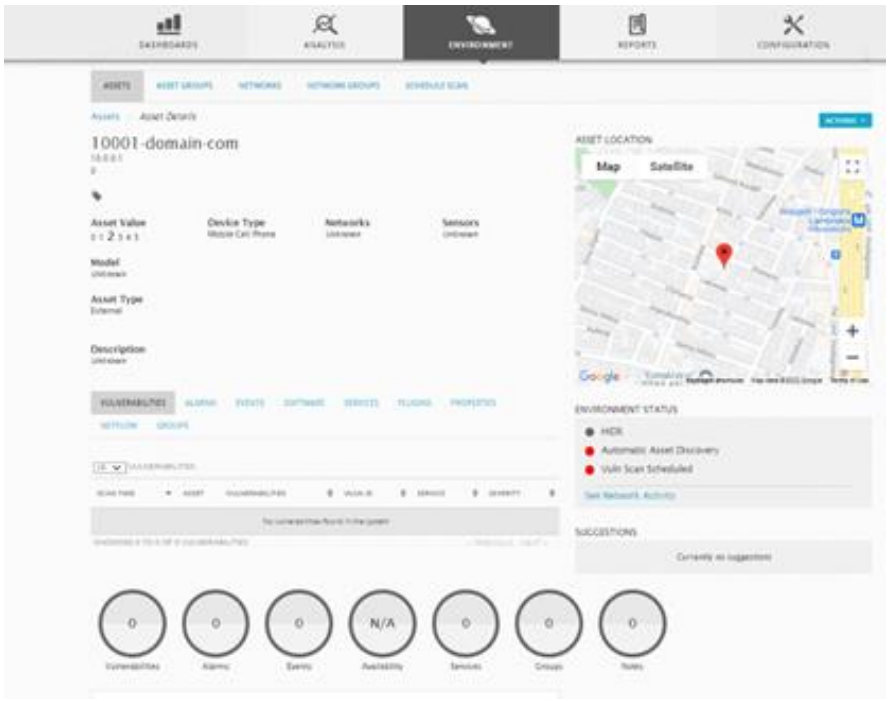


Figure 37: 5G device security details

4.4 NETAPPS FOR PRODUCTION LINE INFRASTRUCTURE (PLI)

4.4.1 5G Teleoperation NetApp

4.4.1.1 Split of NetApp-vApp

The Teleoperation NetApp serves as an additional security layer between the 5G infrastructure and the robots. Thus, it will work using the 5G connectivity with a remote PC connected to a haptic device, a motion controller and a monitor to see the video from the robot. A user will move the arm of the robot thanks to the haptic device and the base through the motion controller. The overall architecture is depicted in Figure 38.

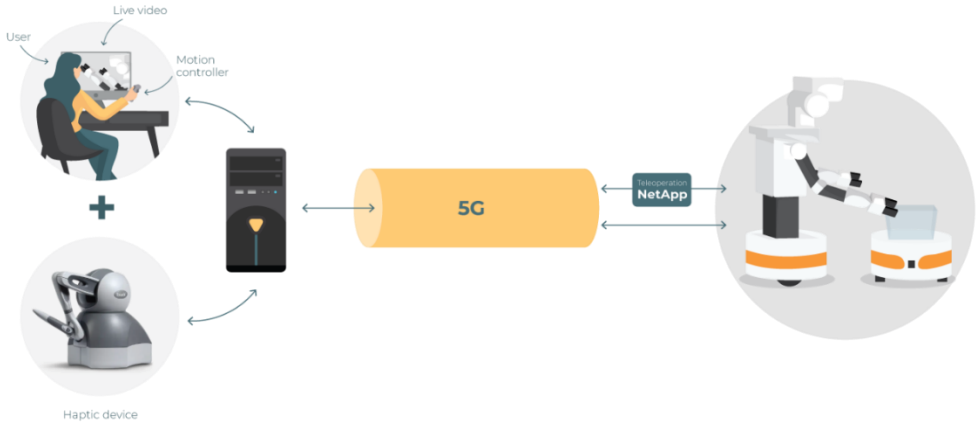


Figure 38 vApp-NetApp Architecture

4.4.1.2 Use case design-description

The use case focuses on the QoS API to make the robot teleoperation process adapt in function of the quality of the 5G Network. It will communicate with the vertical application through Robot Operating System (ROS)/ROS2 topic [24]. The NetApp will say if QoS is guaranteed or not. The robot will be teleoperated through the 5G network and once the QoS is no longer guaranteed the robot will stop completely and wait for the Network connection to be reliable again.

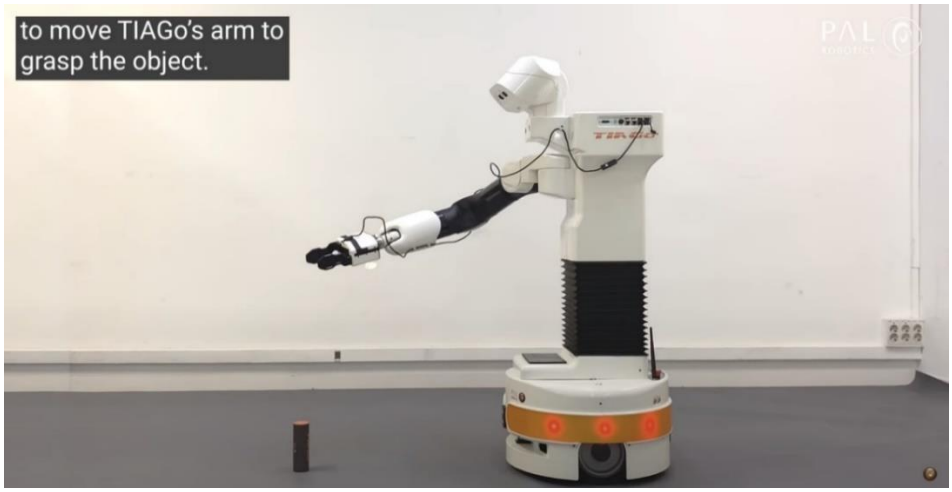


Figure 39 Robot being teleoperated

It will perform a simple task and will adapt thanks to the QoS API first from the NEF emulator and in real environment from the real 5G Network.

4.4.1.3 Interaction with NEF Emulator

In the integration with the NEF Emulator, the NetApp will use the SDK to subscribe to the QoS API. In the simulator the robot will move around and the QoS will be guaranteed if there is no other UE on the cell and not guaranteed if there is already one. This behavior will be only for the simulation since on the real 5G network will rely on the connectivity latency.

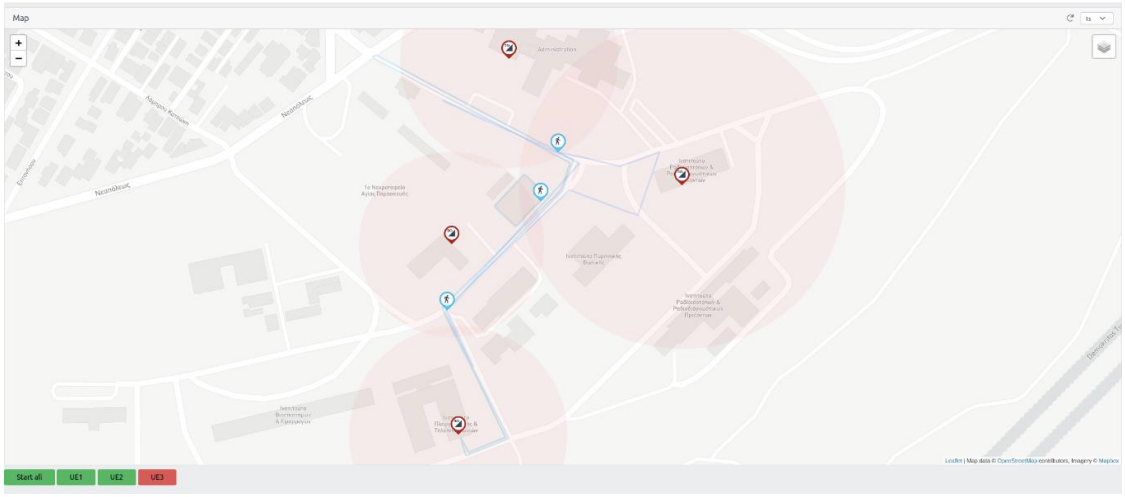


Figure 40 NetApp interaction with NEF emulator

The Notification will be retrieved thanks to the NEF Emulator and SDK endpoint.

4.4.2 Localization NetApp

UM Autonomous System (UMS) and PAL Robotics have jointly developed a NetApp allowing an autonomous mobile robot fleet to be localized in indoor environments using a centralized command center to deploy these assets over 5G infrastructures for a logistics-based use case in a FoF setting. Since no measure is implemented yet considering the final version of the NEF emulator, the localization NetApp is focused, at the moment, on receiving the Cell ID from the 5G LocationAPI and forwarded it to the robot.

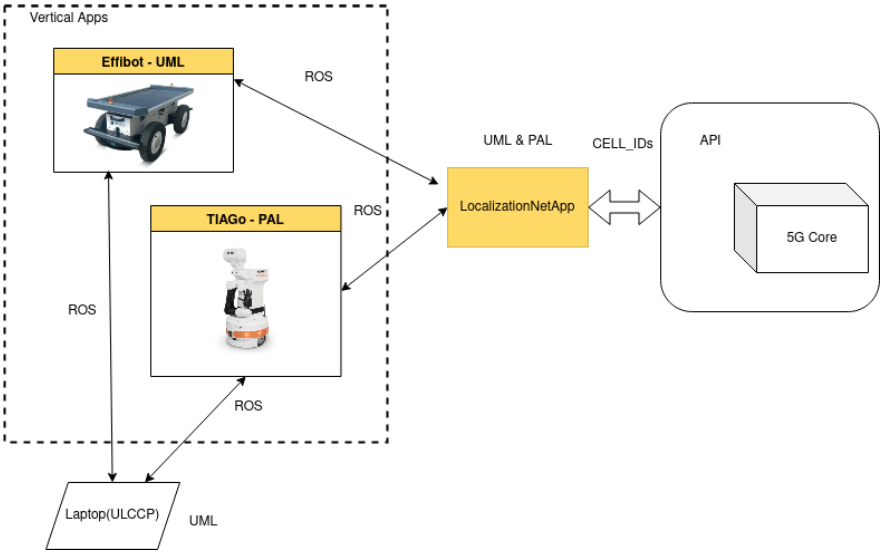


Figure 41 - Localization NetApp - High Level architecture

4.4.2.1 Split of NetApp-vApp

The Localization NetApp serves as a bridge between the 5G infrastructure and the orchestration of the robots. For this reason, NetApp is separated from one vertical or multi-vertical application.

4.4.2.2 Use case design-description

Localization NetApp communicates with the vertical applications through ROS/ROS2, famous and spread communication middleware in robotics. For this reason, the NetApp consists of a ROS and a ROS2 node. Moreover, the NetApp provides the cell IDs associated with the UEs in a ROS/ROS2 topic allowing the robot to react accordingly.

In the use case the TIAGo robot moves in different cell location, identified by a hexadecimal number, when a Cell ID event is triggered. The following picture depicts an example of usage during the testing of the Localization NetApp on ROS on TIAGo robot. The sample simulated behaviour of the robot is as follows:

- Given a simulated environment with 4 different regions (Cell 1, Cell 2, Cell 3, Cell 4), the robot starts the mission from a defined pose in the center of the simulated environment as depicted in Figure 42; A different view of the simulated environment is also presented in Figure 43.
- The Localization NetApp communicates with NEF Emulator, that provides a *Cell ID* measurement. The NEF Emulator simulates different measurements of *Cell ID*. Then those values are feed to the robot by the Localization NetApp;
- The robot will start to navigate to a defined region based on the input received from the Localization NetApp;
- While continuously checking the received *Cell ID* values, the robot will navigate to the specific region. Then, when receiving a new *Cell ID* value, it will navigate to a different location;
- If the robot is located into a specific region and it's receiving *Cell ID* measurements corresponding to the same region, it will stay on-hold;
- Once a different *Cell ID* measurement is received, then the robot will start to navigate again to the specific region.

Future developments will include a complete global localization system provided by the NetApp and associated to different robots/UEs.

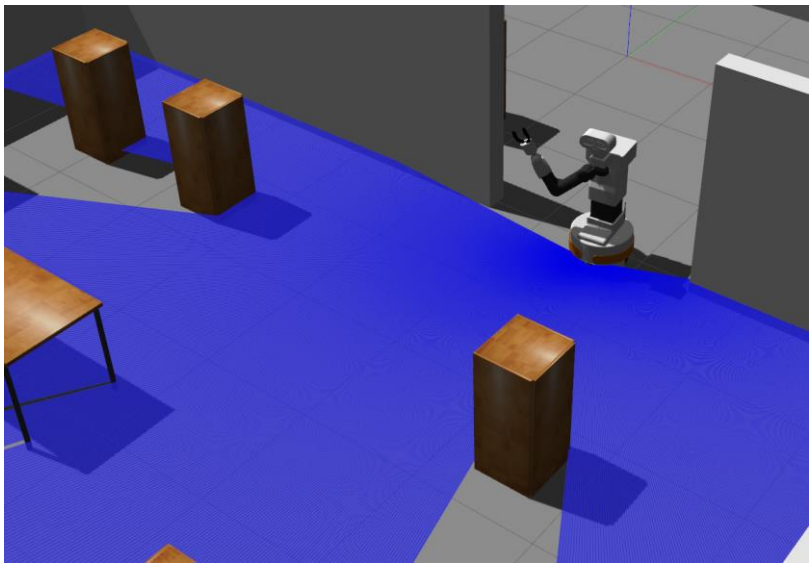


Figure 42 Localization NetApp - Robot interaction simulation view 1

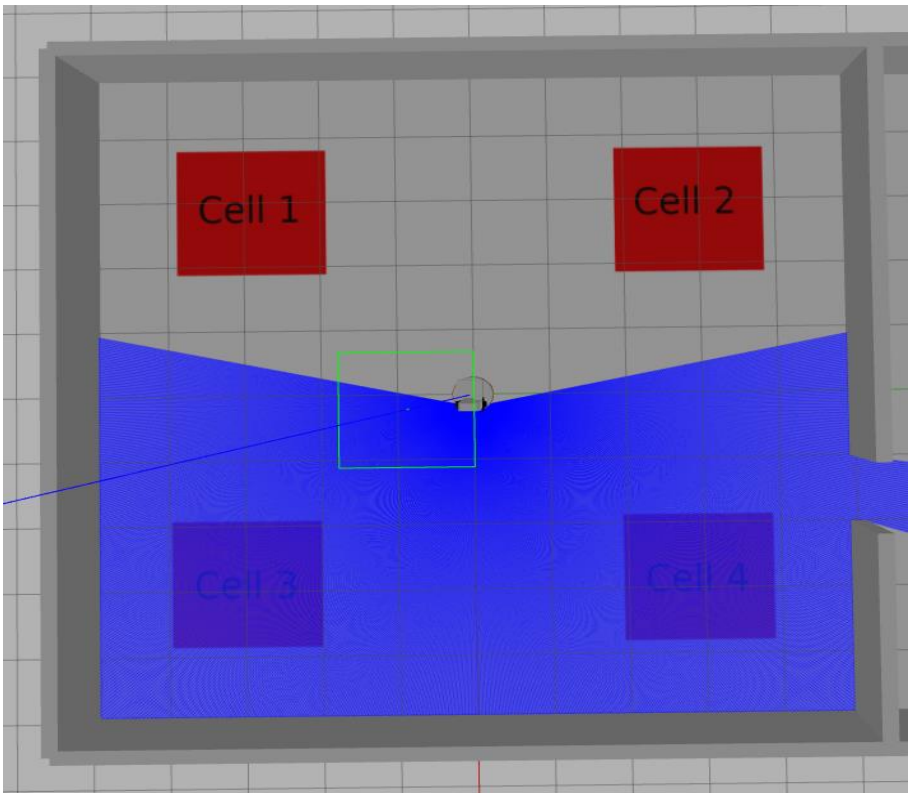


Figure 43 - Localization NetApp - Robot interaction simulation view 2

4.4.2.3 Interaction with NEF Emulator

The NEF emulator runs on the same machine as the NetApp. Currently, it allows registering for an event subscription and notifies when a new event of QoS or cell id occurs. This latter is triggered when a new device enters into a different cell area. The integration with the NEF Emulator uses the emulator REST API, sending messages to authenticate and subscribe to the cell id event.

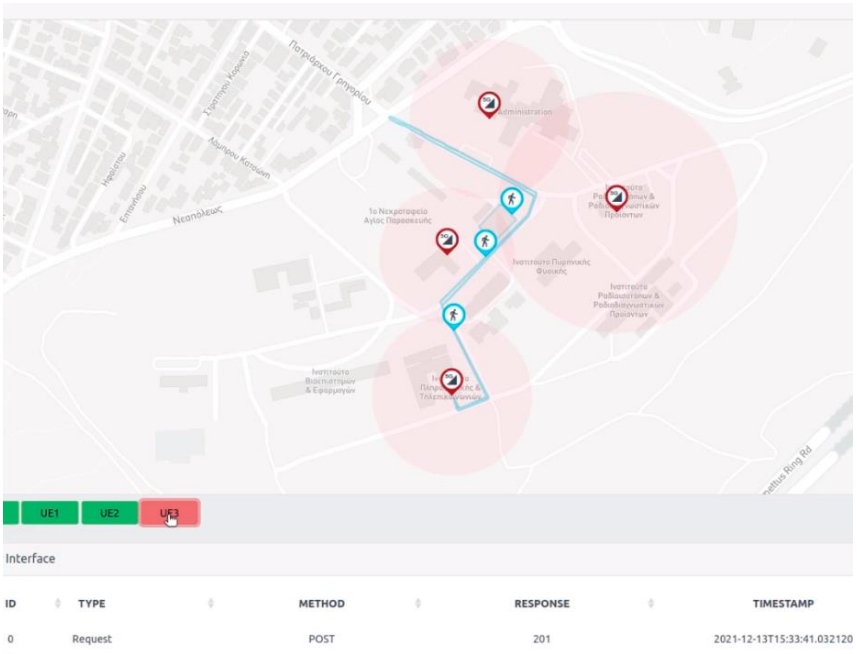


Figure 44 - Localization NetApp - NEF Emulator

5 CONCLUSION

This deliverable presents in detail the work performed in the context of WP4 during the first period of the project and more specifically from M5 to M14. Task 4.1-4.5 are driving the current deliverable and one of the primary purposes of the specific tasks, during the lifetime of the project, is to tightly interact with WP2 and WP3, in order to consistently collect the relevant architectural information from the former while the latter facilitates the development of the early prototypes of the NetApps by providing specific tools supporting the development process.

Task 4.1 focuses on the 5G Exposure Capabilities for the NetApps development and to that end section 2 presents a detailed description of the exposed services related with the NEF Northbound APIs, resulting in an accessible, controllable and programmable 5G network, which allows service providers to utilize those APIs and services to enhance their applications. Moreover, in section 2, details about the design and the implementation of specific APIs in the context of the NEF emulator that EVOLVED-5G introduces are presented. Section 3 provides details on the tools within the workspace environment that are utilised towards the development which is the primary phase in the lifecycle of the NetApps.
Tasks 4.2-4.5 focus on the development of the NetApps. All information related to the preliminary version of the NetApps, including implementation details, use cases' description as well as the interaction with the NEF emulator, in order to consume the exposed APIs, are presented in section 4.

It is important to point out that the development work of several aspects that are affecting WP4 is an ongoing process, thus this deliverable provides information about the current status of the NetApps development by the time this deliverable is being written. The SDK tool as well as the verification tools are planned to be utilised during the development of the second version of the NetApps, thus all the relative information will be covered in deliverable D4.2, to be released in M20.

6 REFERENCES

[1] EVOLVED-5G, "D3.1 Implementations and integrations towards EVOLVED-5G framework realisation," https://evolved-5g.eu/wp-content/uploads/2022/01/EVOLVED-5G-D3.1-v1.0.pdf.

[2] EVOLVED-5G, "Deliverable D2.1 "Overall Framework Design and Industry 4.0 requirements"," https://evolved-5g.eu/wp-content/uploads/2021/11/EVOLVED-5G-D2.1_v1.4.pdf.

[3] 3GPP, "TS 23.502 "Procedures for the 5G System (5GS)"," v17.0.0 , 2021.

[4] 3GPP, "TS 29.122 “T8 reference point for Northbound APIs”Release 17, v17.0.0," Dec. 2020.

[5] Michael Starsinic, Dale Seed, and Chonggang Wang., ""An Overview of 3GPP Exposed Services for IoT Service Platforms".," *GetMobile: Mobile Comp. and Comm. 22,* p. 16–21, June 2018.

[6] 3GPP, "TS 23.273, “5G System (5GS) Location Services (LCS)”," Release 16, v16.9.0, December2021.

[7] 3GPP, "TS 23.288, “Architecture enhancements for 5G System (5GS) to support network data analytics services," v16.2.0, Release 16, Dec. 2020.

[8] 3GPP, "TS 24.250 "Protocol for Reliable Data Service; Stage 3"," v16.4.0, Release 14, Dec. 2020.

[9] D. Santos, R. Silva, D. Corujo, R. L. Aguiar and B. Parreira, ""Follow the User: A Framework for Dynamically Placing Content Using 5G-Enablers,"," *IEEE Access,* vol. vol. 9, no. doi: 10.1109/ACCESS.2021.3051570, pp. pp. 14688-14709, 2021.

[10] 3GPP, "TS 33.535 “Authentication and Key Management for Applications (AKMA) based on 3GPP credentials in the 5G System (5GS)”," Release 16, v16.0.0, September 2020.

[11] 5GPP, "https://5g-ppp.eu/evolved-5g-builds-the-first-3gpp-nef-emulator-to-support-smes-on-5g-programmability/," [Online].

[12] 3GPP, "TS 29.500, "Technical Realization of Service Based Architecture"," Release 17, v17.3.0, June 2021.

[13] 3GPP, "TS 23.501 – Technical Specification Group Services and System Aspects, System Architecture for the 5G System; Stage 2"," v17.2.0, September 2021.

[14] 3GPP, "TS 23.502 “Procedures for the 5G System (5GS)”," Release 16, v16.0.0, March 2019.

[15] EVOLVED-5G, "Deliverable 2.2 “Design of the NetApps development and evaluation environments," https://evolved-5g.eu/wp-content/uploads/2021/11/EVOLVED-5G-D2.2-v1.0_final.pdf.

[16] "https://github.com/EVOLVED-5G," [Online].

[17] "https://github.com/EVOLVED-5G/NEF_emulator," [Online].

[18] EVOLVED-5G, "https://evolved5g-cli.readthedocs.io/en/latest/," [Online].

[19] "https://robotframework.org/," [Online].

[20] EVOLVED-5G, "https://github.com/EVOLVED-5G/dummy-netapp," [Online].

[21] gmi-aero, "https://www.gmi-aero.com/datas_clients/articles/anita-ez0901-hot-bonder-2-zone-3.pdf".

[22] "https://openid.net/connect/," [Online].

[23] "https://www.keycloak.org/," [Online].

[24] "https://www.ros.org," [Online].