Experiment #1

Fire Detection and Ground Assistance using Drones [University of Patras]

Overview and Objectives

Fire Detection and Ground Assistance using Drones (FIDEGAD) is a cloud-native Network Application, which is part of the 5GASP project ecosystem, developed and maintained by the **University of Patras**. It is aimed at the timely detection and provision of a first assessment of structures and forests on fire. A FIDEGAD/5G-EPICENTRE demonstration was included in the ICT-41 Plugfest, co-organized by the 5GASP and 5G-EPICENTRE projects, which took place in Aveiro, Portugal, on 23-24/10/2023[1]. This demonstration included the deployment of the FIDEGAD application (server component) on the 5G-EPICENTRE Aveiro testbed, followed by the execution of the FIDEGAD use case.

FIDEGAD is composed of two main components (Figure 1). The FIDEGAD client is usually installed in a drone, and is able to provide conventional video, telemetry data, as well as information from infrared sensors and thermal vision, which are then transmitted through 5G connectivity to the FIDEGAD server, for inspection and monitoring by the PPDR operating teams. The FIDEGAD server, hosted at the Edge to ensure low latency, is further divided in two sub-components – the Application Server (AS), which processes the stream received from the drone to detect specific events that may require any type of action, such as fire; and the Application Function (AF), in charge of communicating with the 5G core if/when needed through the Network Exposure Function (NEF) or Policy Control Function (PCF), to ensure the mission-critical functionality of the application.

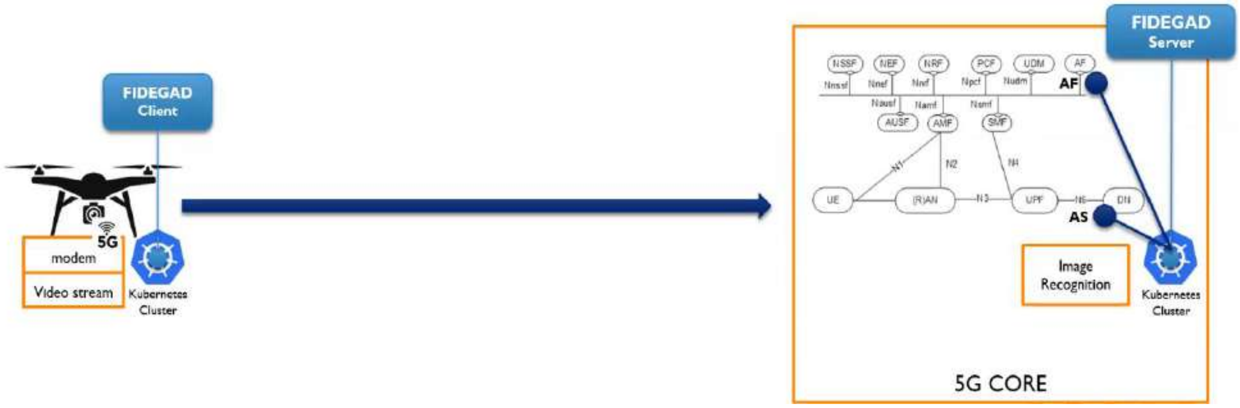


Figure 1: FIDEGAD main components

5G plays a key role in this use case, by leveraging the policy control capabilities enabled by the 5G core PCF component, which proves to be especially valuable in PPDR scenarios that require network services, to promptly adjust to specific requirements, as a result of an unexpected event.

The use case is divided in two phases, before and after the event detection. The initial application traffic stream is conveyed through a network slice that is unable to properly cope with network saturation, which is reflected by the low frame rate (measured in frames-per-second, FPS), not adequate for a mission-critical service. A command is

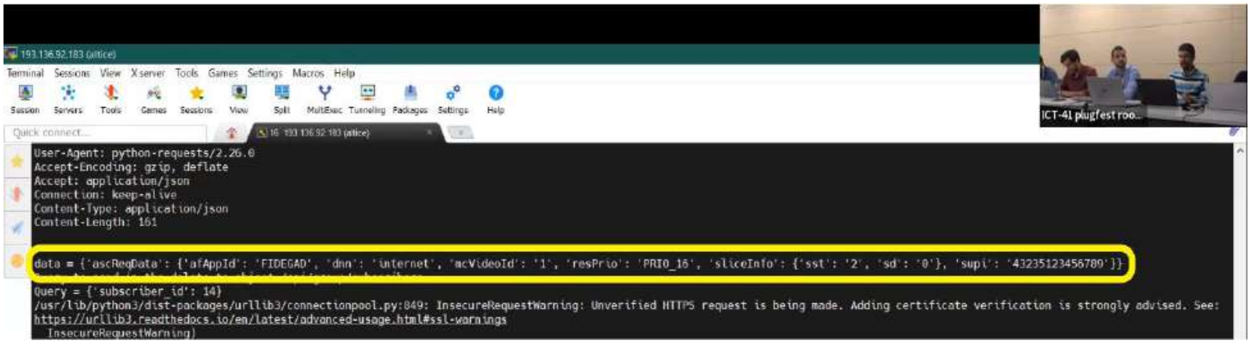
[1] IEEE 9th World Forum on Internet of Things, "Demonstration and PlugFest Program," [Online]. Available: https://wfiot2023.iot.ieee.org/ict-41-plugfest. [Accessed June 2024].

This project has received funding from the European Union's Horizon 2020
Innovation Action programme under Grant Agreement No 101016521.

issued from the FIDEGAD Server to the 5G Core PCF (following the relevant 3GPP specifications, TS 29.513[2] and TS 29.514[3]), requesting for the highest priority among connected UEs / network slice change based on relevant 5G QoS Identifier (5QI) characteristics (e.g., 5QI 70 – Mission Critical Data). Then, the 5G Core is expected to receive this request, process it, and switch the video stream output to a different slice, capable of providing a higher, and more stable framerate.

Testbed Readiness

The deployment of the FIDEGAD application on 5G-EPICENTRE platform was relatively straightforward, especially facilitated by the common use of the Helm chart packaging format. The only major challenge faced was the integration of the N5 (AF-PCF) interface. In fact, a basic requirement of the experiment is the interoperability between FIDEGAD and 5G-EPICENTRE's 5G core through the N5 interface, essential to make the network behaviour responsive to specific events. Figure 2 shows the service information sent by FIDEGAD to install a policy, in compliance with 3GPP TS 29.514.


193.136.92.183 (altice)
Terminal Sessions View X server Tools Games Settings Macros Help
Session Servers Tools Games Sessions View Split MultiExec Tunneling Packages Settings Help
Quick connect...
16. 193.136.92.183 (altice)
ICT-41 plugfest roo...

```
User-Agent: python-requests/2.26.0
Accept-Encoding: gzip, deflate
Accept: application/json
Connection: keep-alive
Content-Type: application/json
Content-Length: 161

data = {'ascReqData': {'afAppId': 'FIDEGAD', 'dnn': 'internet', 'mcVideoId': '1', 'resPrio': 'PRIO_16', 'sliceInfo': {'sst': '2', 'sd': '0'}, 'supi': '43235123456789'}}
Query = {'subscriber_id': 14}
/usr/lib/python3/dist-packages/urllib3/connectionpool.py:849: InsecureRequestWarning: Unverified HTTPS request is being made. Adding certificate verification is strongly advised. See: https://urllib3.readthedocs.io/en/latest/advanced-usage.html#ssl-warnings
  InsecureRequestWarning)
```

Figure 2: Information passed to PCF by FIDEGAD

To overcome the lack of interoperability at the N5 interface level, a proxy component was developed to translate 3GPP standard API into Druid Raemis 5G Core[4] PCF interface. This proxy component was developed in Python Language, uses the library "Quart"[5], provides a RESTful Application Programming Interface (API), compliant with the 3GPP standard, and runs in port 185. This API abstracts the execution of proprietary Druid RESTful API [4], and in this specific case, needs to realize different calls to achieve the request present in Figure 2: change the subscriber to the slice requested; change the 5QI characteristics for that requested; force to apply immediately the new changes for the Packet Data Unit (PDU) Session.

Experiment Deployment

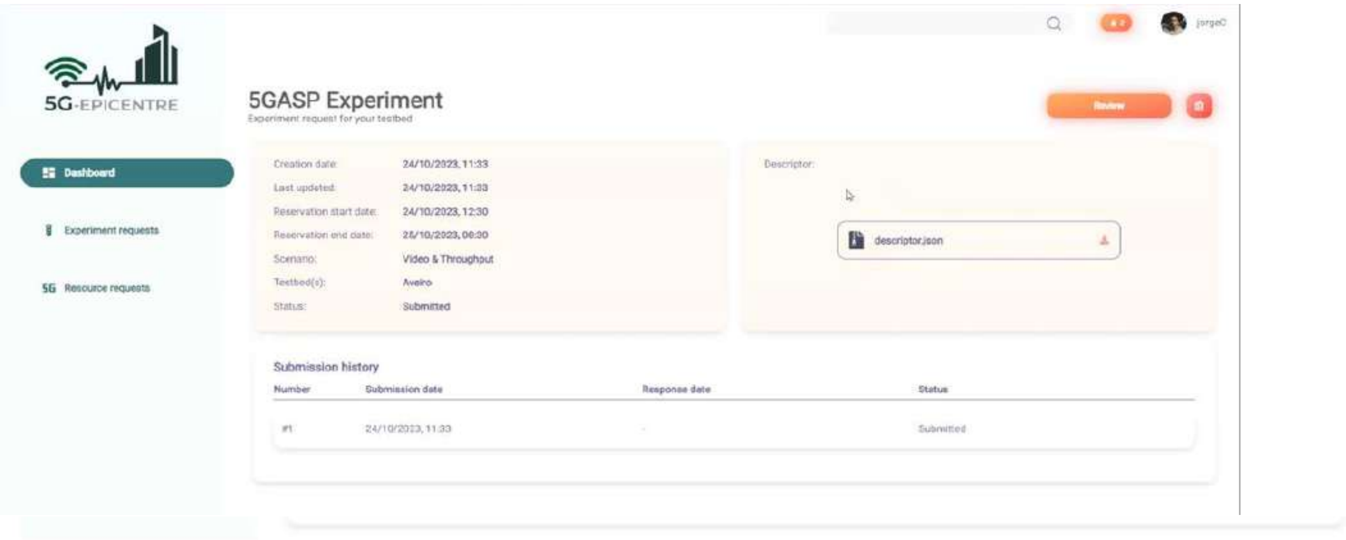
The FIDEGAD Network Application was deployed on the 5G-EPICENTRE Altice Labs 5G infrastructure. The FIDEGAD server and client components were deployed on separate Kubernetes clusters, interconnected through 5G. The FIDEGAD Client and Server components are two separate Helm charts, with a dependency on Load Balancer configuration in the deploying K8s Cluster.

[2] 3GPP, "TS 29.513 V18.4.0; 5G System; Policy and Charging Control signalling flows and QoS parameter mapping," 2023.

[3] 3GPP, "TS 29.514 V18.4.0; 5G System; Policy Authorization Service; Stage 3," 2023.

[4] Druid Software, "Raemis™ – Cellular Network Technology," [Online]. Available: https://druidsoftware.com/raemis-cellular-network-technology/. [Accessed June 2024].

[5] "Quart," [Online]. Available: https://pypi.org/project/Quart/.



5GASP Experiment

Experiment request for your testbed

Creation date:	24/10/2023, 11:33
Last updated:	24/10/2023, 11:33
Reservation start date:	24/10/2023, 12:30
Reservation end date:	25/10/2023, 06:30
Scenario:	Video & Throughput
Testbed(s):	Aveiro
Status:	Submitted

Descriptor:

descriptor.json

Submission history

Number	Submission date	Response date	Status
#1	24/10/2023, 11:33	-	Submitted

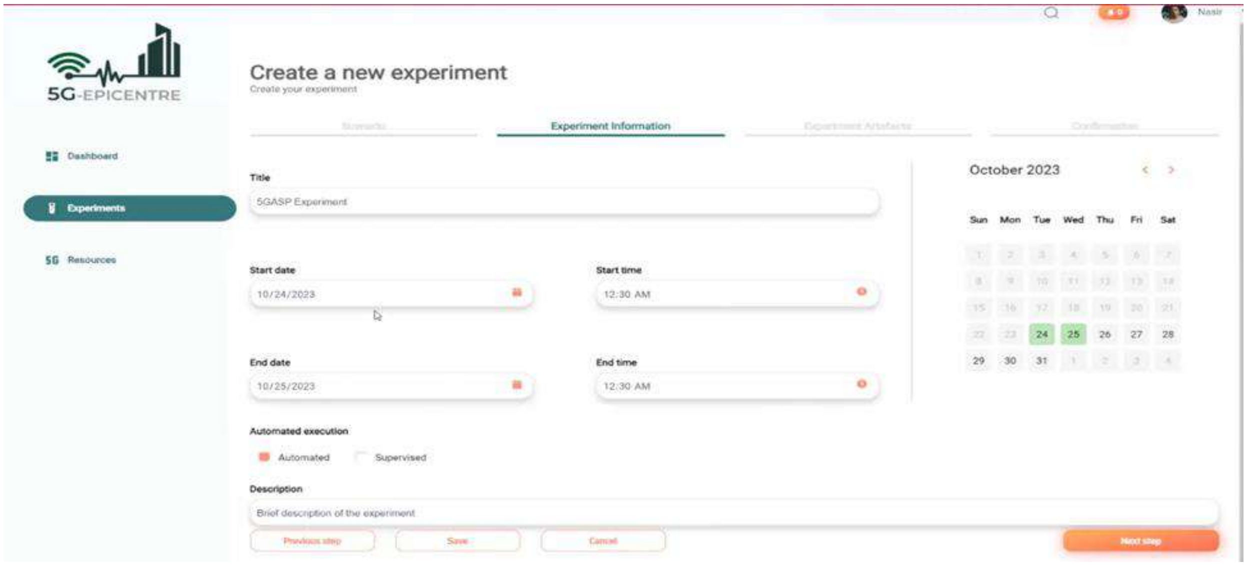
Figure 3:

The ICT-41 Plugfest demo on 24/10/2023 [1] included the onboarding of the FIDEGAD server component on Altice Labs' 5G testbed infrastructure. The creation of a new experiment and the deployment of the FIDEGAD network application on Altice Labs infrastructure strictly followed the 5G-EPICENTRE defined procedures, supported by the 5G-EPICENTRE Portal, as illustrated in **Σφάλμα! Το αρχείο προέλευσης της αναφοράς δεν βρέθηκε.**. As can be seen in the Figure, the deployment process includes 4 steps, including the scenario selection (out of the four pre-defined 5G-EPICENTRE templates, targeted at different experiment characteristics), experiment information (schedule and type of experiment automation), selection of experiment artefacts (Helm chart and optionally additional 5G-EPICENTRE network applications or traffic simulation) and confirmation. Once this 4-steps submission process has been concluded by the candidate experimenter, the testbed owner is expected to review the request, check the experiment descriptor, and finally accept the experiment, if all conditions are met, as shown in **Σφάλμα! Το αρχείο προέλευσης της αναφοράς δεν βρέθηκε.**. At this point, the experiment is ready to be executed.

The FIDEGAD server was initially deployed and was assigned a K8s external IP. This service is in charge of fire recognition on top of the incoming video stream provided by the FIDEGAD client. The server exposes two ports, i.e., port 85 (FIDEGAD UI), and port 8765 (Web Socket accepting the video stream input). The FIDEGAD client accepts a stream input (from a URL/local file/streaming server), and forwards it to the server's LoadBalancer IP, which is already known.

Figure 6: Integration of FIDEGAD in Altice Labs 5G testbed

2 illustrates the integration of the FIDEGAD client and server components in the Altice Labs 5G testbed. In order to simplify the deployment process (since this was not the primary focus of the demonstration), the client component was deployed on a standard K8s cluster and a pre-recorded video stream was used.



Create a new experiment

Create your experiment

Title: 5GASP Experiment

Start date: 10/24/2023

Start time: 12:30 AM

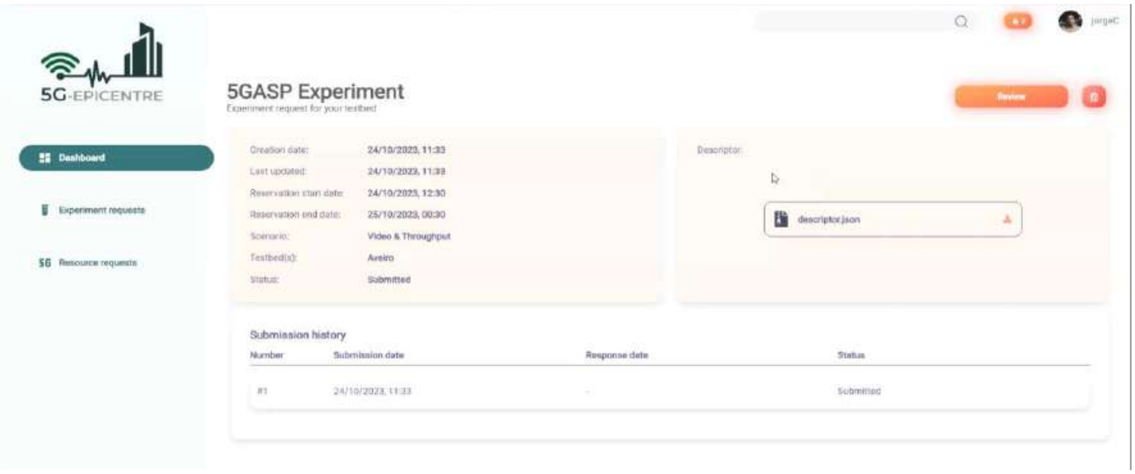
End date: 10/25/2023

End time: 12:30 AM

Automated execution: ● Automated ○ Supervised

Description: Brief description of the experiment

Figure 4: 5G-EPICENTRE experiment creation wizard environment



5GASP Experiment

Experiment request for your testbed

Creation date:	24/10/2023, 11:33
Last updated:	24/10/2023, 11:33
Reservation start date:	24/10/2023, 12:30
Reservation end date:	25/10/2023, 00:30
Scenario:	Video & Throughput
Testbed(s):	Aveiro
Status:	Submitted

Submission history

Number	Submission date	Response date	Status
#1	24/10/2023, 11:33	-	Submitted

Figure 5: Experiment acceptance - Testbed owner view

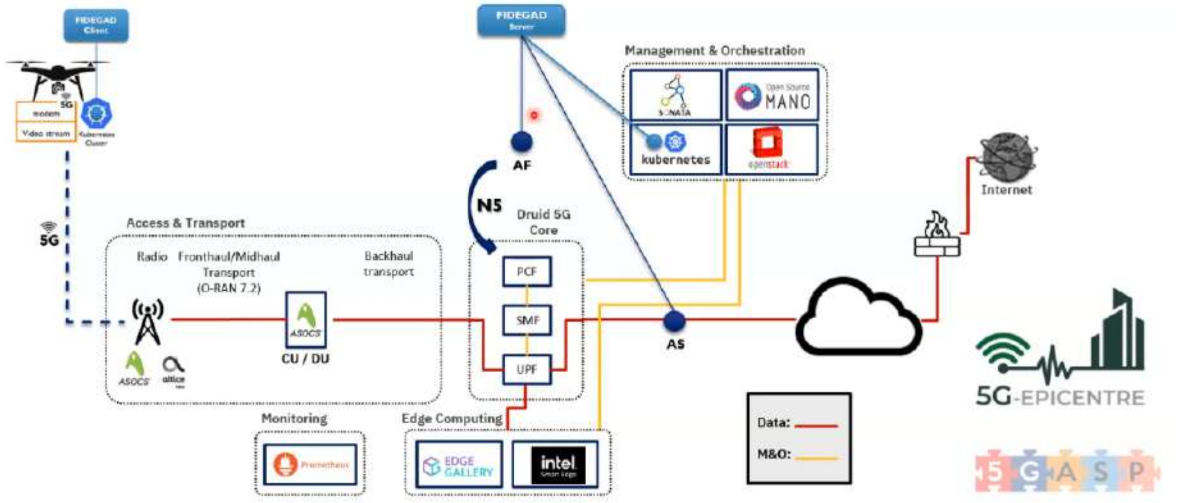
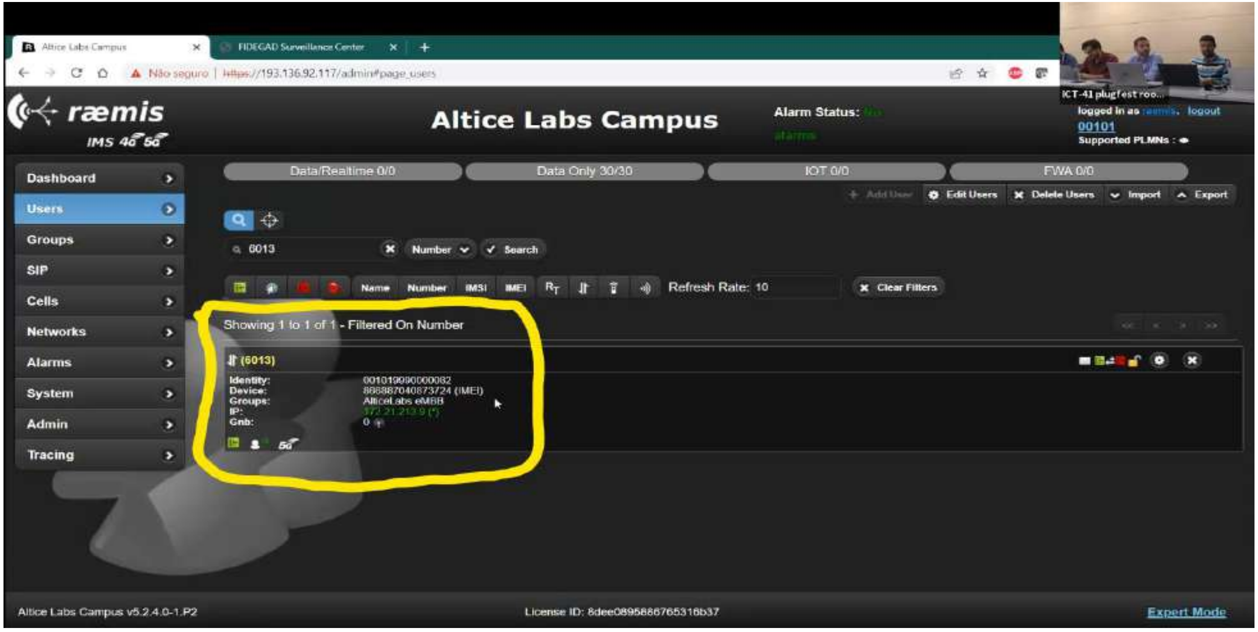


Figure 6: Integration of FIDEGAD in Altice Labs 5G testbed

Experiment Execution and Results

Following the process described in the previous section, the FIDEGAD application was deployed on the 5G-EPICENTRE platform. The FIDEGAD server also accepted a PCF endpoint as input, which would be used when prompted by the UI to ensure that the mission-critical functionality of the application is sustained. Figure 6: Integration of FIDEGAD in Altice Labs 5G testbed

shows the Druid Raemis 5G core dashboard in the initial experiment phase, before the request issued by the FIDEGAD AF to use a prioritized network slice. As can be seen, the FIDEGAD client is using the eMBB slice at this point.



Altice Labs Campus

Alarm Status: No alarms

logged in as raemis, logout
00101
Supported PLMNs :

6013

Refresh Rate: 10

Showing 1 to 1 of 1 - Filtered On Number

(6013)

Identity: 001019990000082
Device: 866887040873724 (IMEI)
Groups: AlticeLabs eMBB
IP: 172.21.213.9 (*)
Gnb: 0

Altice Labs Campus v5.2.4.0-1.P2

License ID: 8dee0895886765316b37

Figure 7: Druid Raemis 5G Core dashboard (FIDEGAD client connected through eMBB slice)

Figure 8 shows FIDEGAD Surveillance Center graphical user interface in the initial experiment phase. As can be seen, the frame rate of the video stream received by the FIDEGAD server at this point is in the range 3-4 FPS, which is not enough to allow the monitoring of the accident in proper conditions.

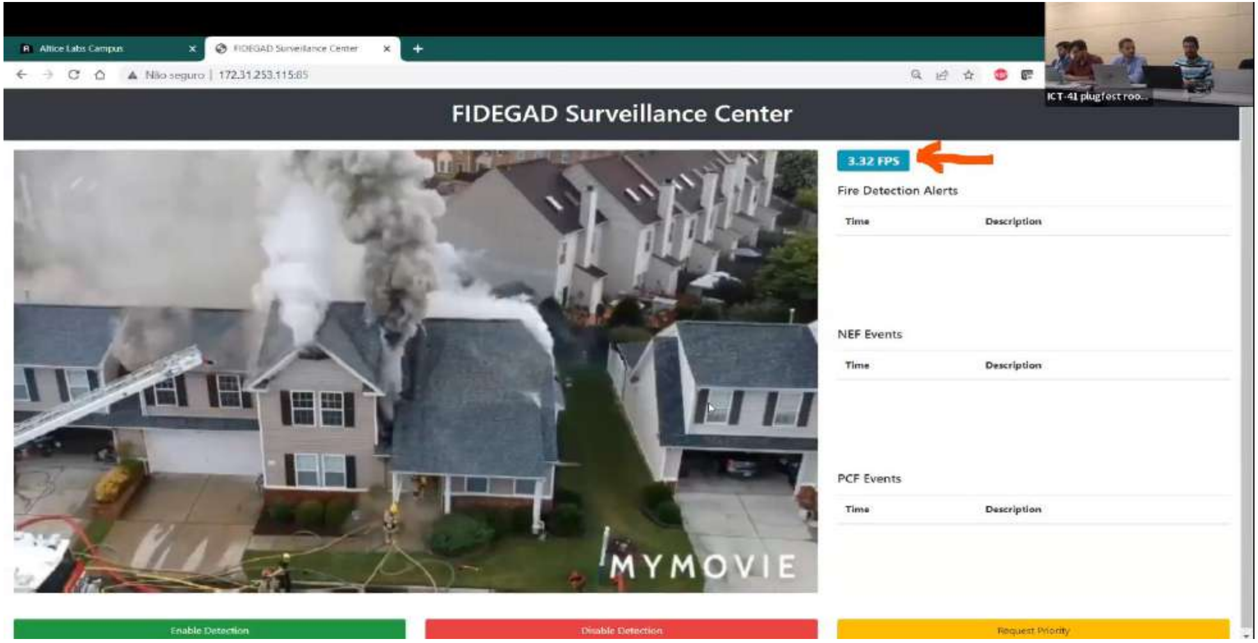


Figure 8: FIDEGAD Surveillance Center (phase 1)

The second phase of the experiment is triggered by the priority request issued by the AF component of the FIDEGAD server to use a different slice to support the video stream traffic, as illustrated in Figure 9. The Network is saturated and the Network Application requests priority.

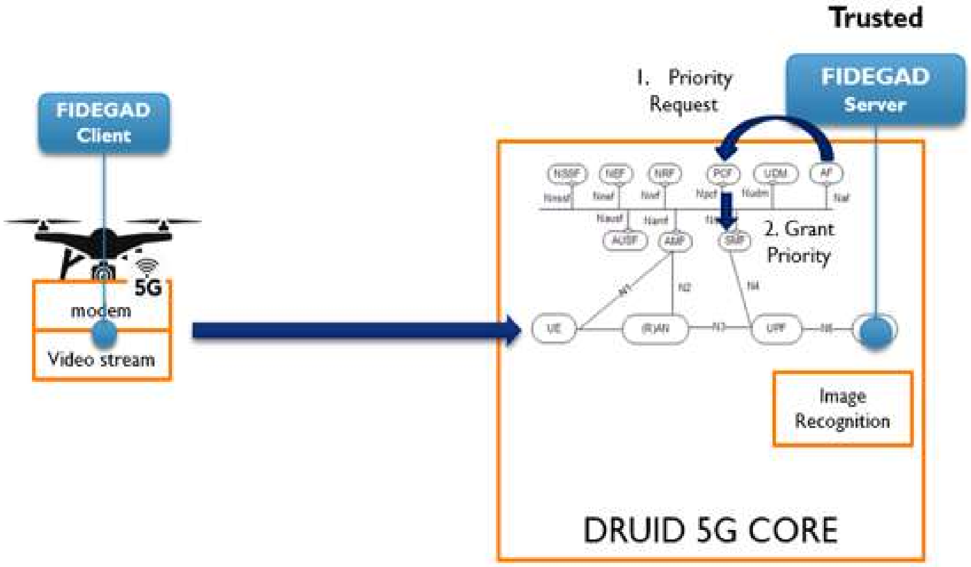
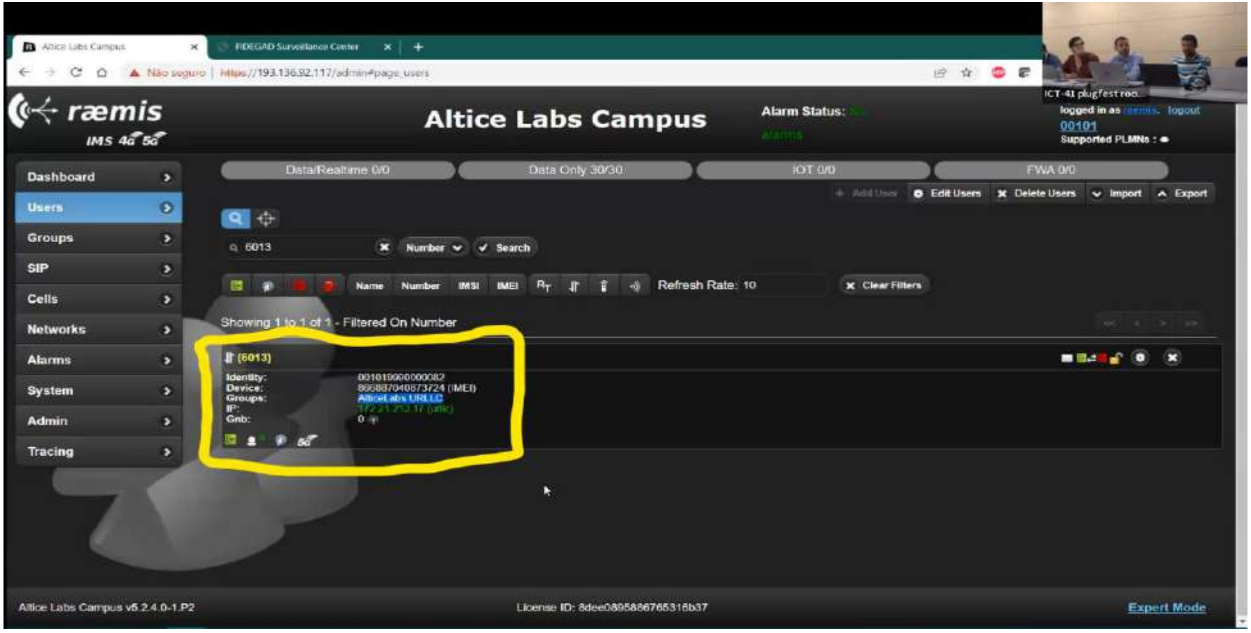


Figure 9: Start of phase 2 of the FIDEGAD experiment

Figure 10 shows the Druid Raemis 5G core dashboard after the request for a prioritized network slice has been accepted. As can be seen in the Figure, the FIDEGAD client is using the URLLC slice at this point (note that the request issued by the FIDEGAD server indicated SST=2).



6013

Refresh Rate: 10

Showing 1 to 1 of 1 - Filtered On Number

(6013)

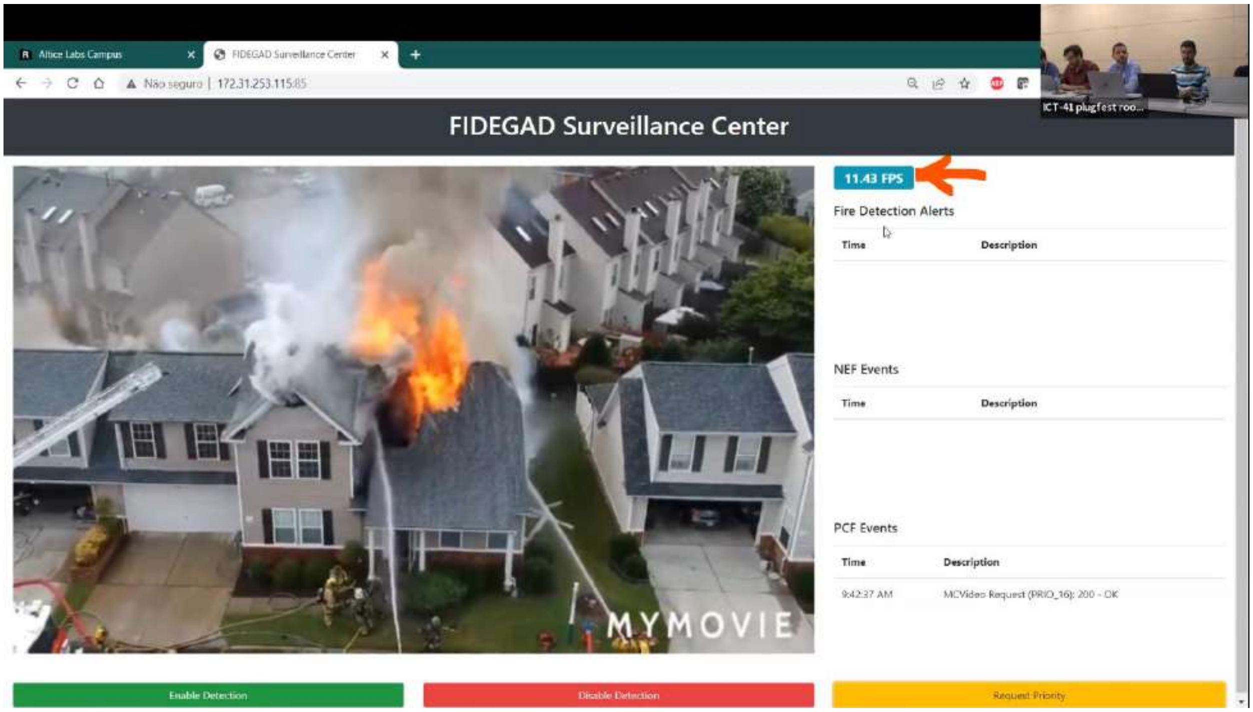
Identity: 001019990000082
Device: 866887040873724 (IMEI)
Groups: AlticeLabs URLLC
IP: 172.21.213.17 (ulic)
Gnb: 0

Altice Labs Campus v5.2.4.0-1.P2

License ID: 8dee0895886765318b37

Figure 10: Druid Raemis 5G Core dashboard (FIDEGAD client connected through URLLC slice)

Figure 11 shows FIDEGAD Surveillance Center graphical user interface after the requested traffic prioritization has been executed. As can be seen in the Figure, the frame rate is now in the order of 11-12 FPS, which represents a significant increase compared to the initial phase and shows the practical result of the 5G infrastructure reconfiguration. In the live demonstration, it was possible to confirm a significant increase in the quality of the video displayed by the FIDEGAD Surveillance Center.



FIDEGAD Surveillance Center

11.43 FPS

Fire Detection Alerts

Time	Description

NEF Events

Time	Description

PCF Events

Time	Description
9:42:37 AM	MCVideo Request (PRIO_16): 200 - OK

Figure 11: FIDEGAD Surveillance Center (phase 2)

Overall evaluation

The seamless integration of FIDEGAD on 5G-EPICENTRE platform and the Aveiro 5G testbed, hosted by Altice Labs, was showcased during the ICT-41 Plugfest on 23-24/10/2023. A special focus was placed on QoS management within the boundaries of a mission-critical scenario, with notable results in the video quality within a congested network. The fact that FIDEGAD uses Helm chart packaging format facilitated significantly the deployment of the application on the 5G-EPICENTRE platform.

To 5G-EPICENTRE, especially to Altice Labs 5G testbed facility, the integration of the FIDEGAD application was an opportunity to assess and validate the deployment of a 3GPP-compliant application function making use of the standard N5 interface to interoperate with the PCF function. The lack of compliance of the local 5G core to 3GPP was overcome through the development of a simple adaptation layer. The successful outcome and the overall integration effort of both projects' stakeholders was only made possible due to the incorporation of the aforementioned standardized interfaces.

As main lessons learned from this activity, one can identify:

- **Cross-platform standardized interfaces are a basic requirement to ensure interoperability and seamless functionality.** Indeed, the technical session's demonstration showcased this feasibility in the challenging PPDR domain, where the necessity for coherent services is even more apparent.
- **The incorporation of the 3GPP-defined PCF interface (N5) was the most significant technical challenge of this experiment.** The successful onboarding of the FIDEGAD application provided good indications that the approach followed in this experiment can be replicated in other cases.
- Last but not least, **the capability of an external application to interact directly with the 5G network through the N5 interface proved to be a valuable asset**, particularly in emergency scenarios, for which the quick adaptation of the network to handle specific may be a key requirement.

5G-EPICENTRE Experimentation Platform

ReShapinG the Future of PPDR Services

For more information, do not hesitate t visit the website https://www.5gepicentre.eu/ and/or contact the 5G-EPICENTRE team.

Follow Us on our social media for more Results