Action full title:

Universal, mobile-centric and opportunistic communications architecture

Action acronym:

UMOBILE

Deliverable:

D5.5 “Report on the validation of the deployment trial”

Project Information:

Project Full Title	Universal, mobile-centric and opportunistic communications architecture
Project Acronym	UMOBILE
Grant agreement number	645124
Call identifier	H2020-ICT-2014-1

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 645124

Topic	ICT-05-2014 Smart Networks and novel Internet Architectures
Programme	EU Framework Programme for Research and Innovation HORIZON 2020
Project Coordinator	Prof. Vassilis Tsaoussidis, Athena R.C.

Deliverable Information:

This deliverable reports the results of the less-than-best-effort Internet access scenario, discusses lessons learned, and described the experiments that took place during the validation.

Deliverable Number-Title	D5.5 Report on the validation of the deployment trial
WP Number	WP5
WP Leader	FON
Task Leader (s)	Alberto Pineda (FON)
Authors	**COPELABS**: Paulo Mendes **ATHENA**: Sotiris Diamantopoulos, Christo-Alexandros Sarros, Vassilis Tsaoussidis **UCL**: Ioannis Psaras, Sergi Rene **UCAM**: Adisorn Lertsinsrubtavee **SENCEPTION:** Rute Sofia, José Soares **TECNALIA**: Susana Pérez **TEKEVER**: Luis Deprez, Ricardo Faria **AFA**: Francesco Amorosa, Giammichele Russi **FON:** Alberto Pineda, Pablo Salvador
Reviewer	Alberto Pineda, Pablo Salvador
Contact	alberto.pineda@fon.com
Due date	M36: 31/01/2018
Actual date of	M39: 30/04/2018

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 645124

submission	

Dissemination Level:

PU	Public	X
CO	Confidential, only for members of the consortium (including the Commission Services)	
CI	Classified, as referred to in Commission Decision 2001/844/EC	

Document History:

Version	**Date**	**Description**
Version 0.1	09/04/18	Initial template with partners' input
Version 0.2	13/04/18	Circulate among partners first draft proposal and ask for extra input
Version 0.3	23/04/18	Revised first version based on feedback
Version 0.4	25/04/18	Added some partners' contributions
Version 0.5	27/04/18	Finalized preliminary version of the deliverable and circulated for internal review
Version 1.0	31/05/17	Final version after reviewers' comments

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 645124

Table of Contents

Table of Contents 4
List of Figures 5
List of Tables 6
Executive Summary 7
Background 7
Objectives 7
1. Introduction 8
2. Deployment Trial for POC1 9
2.1. Overview 9
2.2. Integration 12
3.1. Tests & Validation 14
3. Deployment Trial for POC2 25
3.1. Overview 25
3.2. Integration 26
3.3. Tests & Validation 33
4. Additional demo: Deployment Trial for reduced set of POC1 with UAV assistance 35
4.1. Overview 35
4.2. Integration 35
4.3. Tests & Validation 35
5. Lessons Learned 41
6. Conclusions 44
References 45

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 645124

List of Figures

Figure 1: Emergency situation during final demonstration ____ 10
Figure 2: Scenario for PoC1 ____ 10
Figure 3: POC1 communication scheme ____ 11
Figure 4: PoC2 in the final demo at Umbria ____ 12
Figure 5: Service provisioning in challenging environments ____ 13
Figure 6: joint integration of KEBAPP, UMOBILE hotspot and Service Manager ____ 14
Figure 7: Overview of service migration platform ____ 16
Figure 8: Nodes deployment of QMP network in Barcelona area ____ 17
Figure 9: The topology of service migration deployment in qMp ____ 18
Figure 10: Monitoring manager retrieves the monitoring data from UMOBILE hotspot by using pull-based communication model ____ 19
Figure 11: The dash board of monitoring system of service migration platform ____ 19
Figure 12: Decision Engine delivers the service to SEG_1 by using push-based communication model ____ 20
Figure 13: Service Migration benefits from UMOBILE architecture as SEG_2 can retrieve the service from the cache ____ 20
Figure 14: Inspecting the delivery cost of each SEG ____ 22
Figure 15: Data traffic distributed over qMp network ____ 23
Figure 16: UMOBILE End-User Services interface ____ 25
Figure 17: Contextual Manager running ____ 27
Figure 18: NDN-Opp service ____ 28
Figure 19: Oi! service ____ 30
Figure 20: Now@ service ____ 32
Figure 21: PoC2 overview ____ 33
Figure 22: Static tests - Obtained bandwidth (in Mbit/s) over distance (in meters). ____ 36
Figure 23: Static tests - Jitter in milliseconds over distance in meters ____ 37
Figure 24: Static tests - Lost data packets over distance ____ 37
Figure 25: Obtained results for lost datagrams in function of the bandwidth and the distance between user and server ____ 38
Figure 26: Raspberry PI 3 on the fixed wing TEKEVER's AR3, identified by the red box ____ 39
Figure 27: TEKEVER's VR3.5 multicopter with long range antenna ____ 40

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 645124

List of Tables

Table 1: Sizes of Dockerized images used in the experiments _____ 21

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 645124

Executive Summary

Background

Work Package 5 "**Overall platform integration and validation"** of UMOBILE project evaluates the solutions developed in the project. This Report is written in the framework of Tasks 5.1 "**Definition of the validation of the setup**" and 5.2: "**Evaluation through Simulation and Emulation**" of UMOBILE project. These two tasks aim to validate the architecture and services, derived from results developed in WP3 and WP4.

The ultimate objective of UMOBILE is to advance networking technologies and architectures towards the conception and realization of Future Internet. In particular, UMOBILE extends Internet (i) functionally – by combining ICN and DTN technologies within a new architecture -, (ii) geographically – by allowing for internetworking on demand over remote and isolated areas – and (iii) socially – by allowing low-cost access to users but also free user-to-user networking.

Objectives

This report describes the UMOBILE system integration in a real deployment trial, reporting the results in a less-than-best effort Internet access scenario, discussing the integration as well as the experiments that took place during the validation. This way we can test the limits of our system and its operational capabilities. This validation in a real life scenario will also help to identify limits of our system proposing lessons learned.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 645124

1. Introduction

The UMOBILE project aims to introduce new paradigms in the provision and consumption of the Internet connectivity by developing a mobile-centric service-oriented architecture that efficiently delivers content to the end-users. To this end, UMOBILE provides an architecture that merges information-centric networking (ICN) with delay tolerant networking (DTN) in order to efficiently operate in different networks situations, reaching disconnected areas and users and providing new types of services. This architecture allows decoupling services from their origin locations and shifting the host-centric paradigm to a new paradigm that incorporates aspects from both information-centric and opportunistic networking.

The solutions defined and prototyped in WP3 and WP4 work packages give a glimpse of the specific performance and/or functionality improvements developed within the project. In order to demonstrate the performance and efficiency of UMOBILE architecture within a testbed, we have selected some specific use-cases identified in the WP2 for the purpose of feature demonstration. More specifically, we focus on two demonstrations: the first one covers both the emergency and the civil protection scenarios, and the second demonstration targets the service announcement and social routine scenarios, which both are presented in detail in the deliverables D5.3 [2] and D5.4 [2]. The aim of this document is to present the validation of the system in a real-life scenario, discussing also the lessons learned.

This document is organized as follows: in Section 2 we describe the deployment trial for PoC1 platform used to showcase the use case of an emergency scenario in terms of integration and validation. Section 3 presents the integration validation of PoC2, which corresponds to the social routine scenario. In Section 4 we present the integration and validation of a reduced set of functionalities PoC1 with UAV assistance. The lessons learned are discussed along Section 5. Finally, the conclusions are drawn in Section 6.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 645124

2. Deployment Trial for POC1

2.1. Overview

The *Centro Regionale of Protezione Civile Umbria* (CRPC) has hosted the deployment trial for POC1, in April 2018. Thanks to its location, spread over a large country campus in the Foligno area (Umbria), the UMOBILE Partners have been capable to clearly and exhaustively demonstrate the outcomes of their common work along the past three years in a realistic Civil Protection environment.

Moreover, the attendance of Civil and Military Authorities has allowed the Consortium to properly exploit the dissemination tasks within a large and distinguished audience of experts.

The demo has been set with the following goals:

- Validate in a realistic scenario the UMOBILE solution
- Disseminate the results of the project
- Focus on a different target audience more related with the practical results of UMOBILE and on the application of these results: industry, municipalities, and emergency services

The Final Demo has been split into two sessions:

- **Day 1 (April 17th)**: Technical Workshop, dedicated to the presentation of the UMOBILE project and its achievements within a Civil Protection use case.

  All Partners have described their activity in the Project, as well as their specific contribution to the Final Demo. Some key Speakers, from the CRPC and other Regional Civil Protection Bodies, have pointed out the importance of access and service continuity during catastrophic events, this confirming the social relevance, besides the technical value of the UMOBILE applications.

- **Day 2 (April 18th):** Final Demo with field simulation of the emergency scenario, due to the occurrence of a disaster event (earthquake, fire, flooding, …)

Figure 1 summarizes the way the various UMOBILE components have been activated, while Figure 2 provides a perspective of the realistic location of PoC1. In Figure 3, we provide the communication scheme that has been followed in the demo. The main areas depicted in Figure 2 are:

- Accident Area (yellow circle, simulated by an injured person and surrounding smoke) where the catastrophic event has caused either failure or overload of the standard mobile communications and, therefore, rescue cannot be provided unless by means of alternative networks.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 645124

Figure 1: Emergency situation during final demonstration

- Alert activation route (red path), where human 'carriers' (passing-by runners, or bikers, …) are able to receive, even accidentally, the emergency signal transmitted from the Accident Area (e.g. the injured person or other people) and, thanks to the UMOBILE peer-to-peer communication, forward it till the Authority Base location, where the signal has been finally triggered, recognized and elaborated, thus activating the rescue plan.

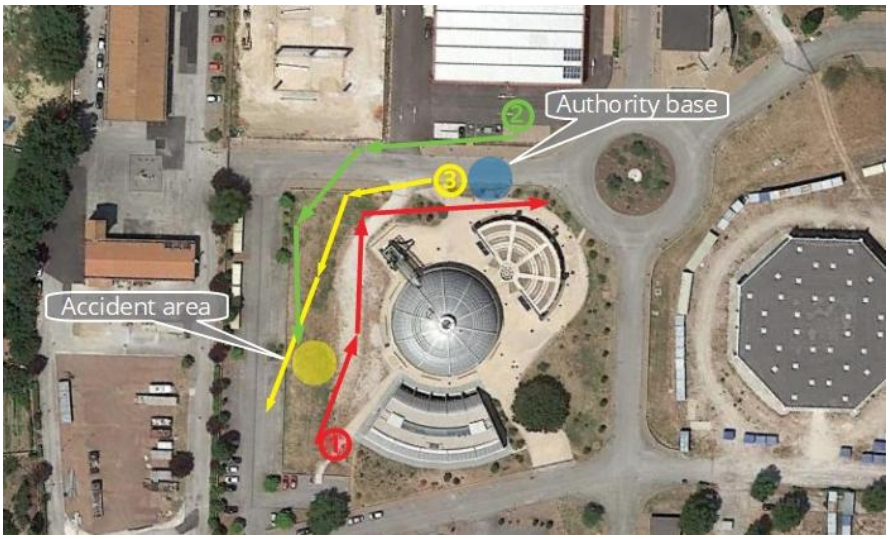


Figure 2: Scenario for PoC1

- Rescue Teams immediate intervention (yellow path) to the Accident Area, after due engagement from the Authority, in accordance with the emergency procedures. During the Final Demo, Firemen and Civil Protection Brigades have realistically been engaged.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 645124

- Creation, transport and delivery, through the UMOBILE infrastructure, of safety instructions (audio, video, text) issued by the Authority to the people living in the Accident Area (green path), thus allowing them to quickly move to safe meeting points. In the real UMOBILE scenario, this is achieved by UAVs, which are able to overcome both natural and artificial obstacles before reaching the Accident Area. Section 4 will present a detailed validation in this kind scenario.

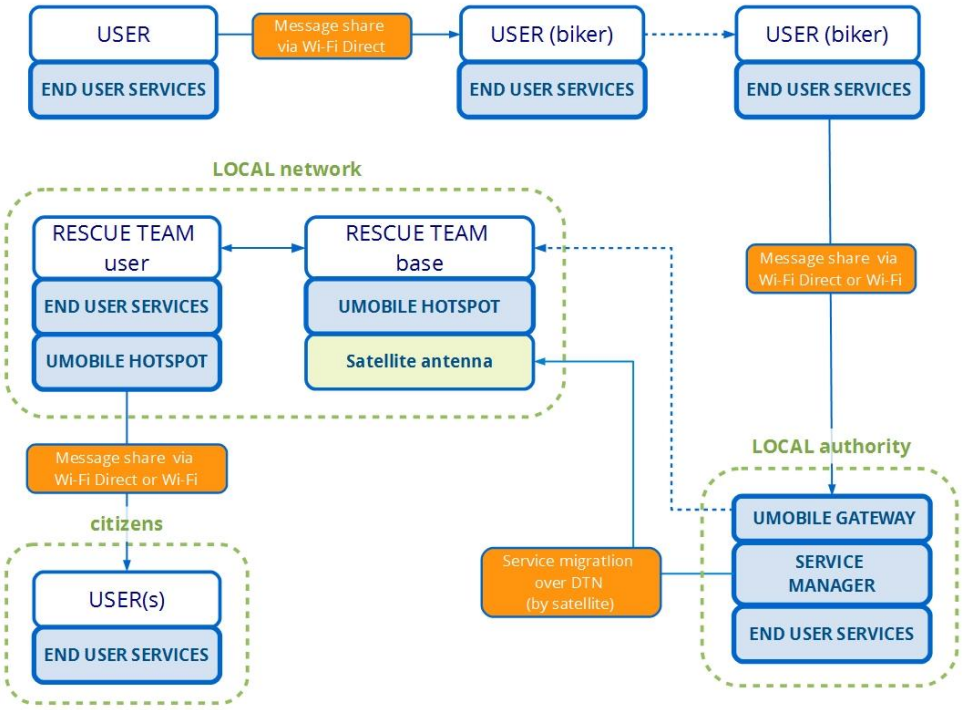


Figure 3: POC1 communication scheme

In order to provide a full demo, PoC2, focused on the support of opportunistic communication has been integrated into PoC1 as represented in Figure 4. Please note that due to legal permission the communication through UAV or satellite has been substituted for a WiFi link. However, UMOBILE consortium has also performed a reduced validation of PoC1 with UAV assistance in Portugal, and this is presented in Section 4.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 645124

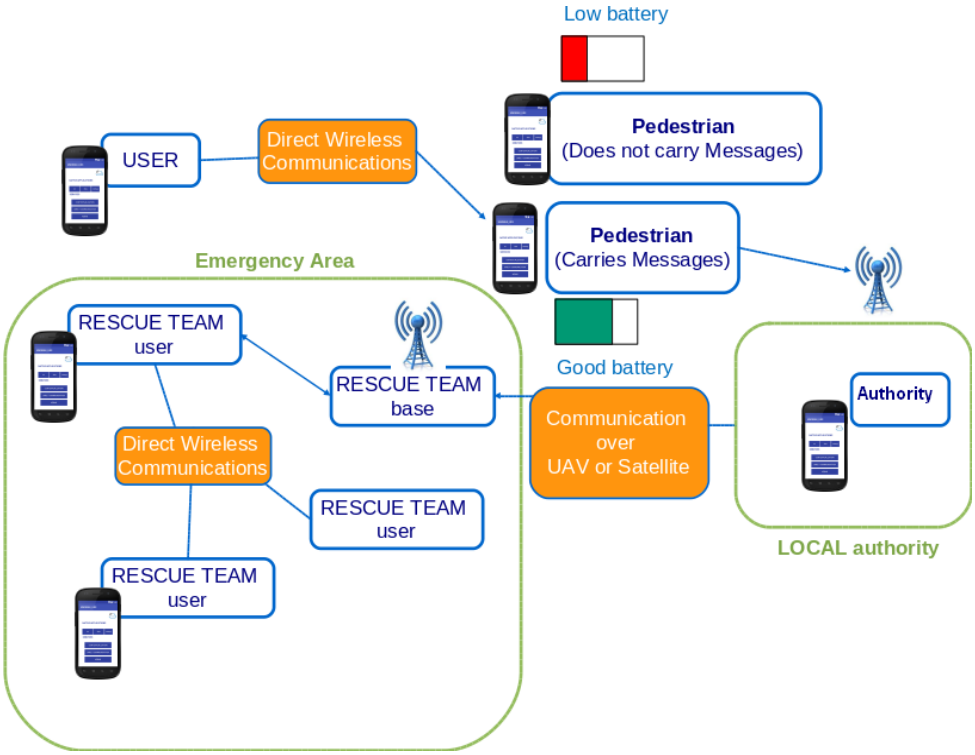


Figure 4: PoC2 in the final demo at Umbria

For a more detailed description of PoC1 please refer to D5.4 [3].

2.2. Integration

DTN Tunneling and Service Migration

Service migration allows services to be deployed and executed across all compatible devices in the network. Following the fog computation paradigm, simple information processing can take place locally, (even in remote hotspots) significantly reducing data to and from the core network. It is to be expected that this functionality will be more helpful in cases where communications are impaired, as experienced, for example, in emergency scenarios. To alleviate network disruptions under these conditions, we support the service migration functionality of the UMOBILE platform with DTN forwarding. By employing a robust network interface, the UMOBILE platform enables the reliable and efficient forwarding of services over communication links between moving vehicles, UAVs and other data mules. This functionality is exploited in POC1, where a service needs to be migrated to a remote hotspot through a UAV, or mobile user, operated by the Civil Protection Authority. In this case, the data comprising the service is first forwarded to the mobile node (the mobile user in our scenario). The latter is responsible for moving towards and, finally, transmitting the service to the remote hotspot.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 645124

Getting into more detail, the part of POC1 where services are migrated over DTN is depicted in Figure 5. We assume that HS1 is at the edge of the main network and that the disaster area network is miles away and disconnected. The aim is to retrieve a service (S) from the Main Network and deploy it in the Disaster area (HS2). We assume S to be a stateless service, e.g., a self-contained web server. Both Wi-Fi hotspots HS1 and HS2 are equipped with the UMOBILE protocol stack. Android phone is a physically mobile, DTN-enabled node that can travel backwards and forwards between HS2 and HS1. In fact, any smart device attached to a UAV or any other means of transport can replace the Android phone, as long as it supports a DTN framework (e.g., IBR-DTN). An Interest request is initially constructed by HS2, including the name of the desired service (S). The location of the affected area is embedded into the Interest name (/service/emergency/Xanthi). This Interest is forwarded to the DTN face when the phone becomes wirelessly reachable to HS2. The Interest is encapsulated in a DTN bundle and stored in the phone's persistent storage. The phone loses contact with HS2 when it travels towards HS1. When it reaches HS1, the Interest is decapsulated and delivered to the NDN layer. It is then transferred by H S 1 through the main network, towards the service producer or the nearest cache. The latter —or another intermediate NDN node in possession of S— retrieves it and prepares a service image with customized information (i.e., emergency contact of local civil protection authority or map in Xanthi). The reply includes one or more Data messages sent along a reverse path using the DTN face. Communication follows the multiple-interest protocol of Section 2.3. When HS2 receives all the chunks of S, it calls a service deployment function to execute S to provide the emergency service to local users via WiFi. They can access S from any standard web browser.

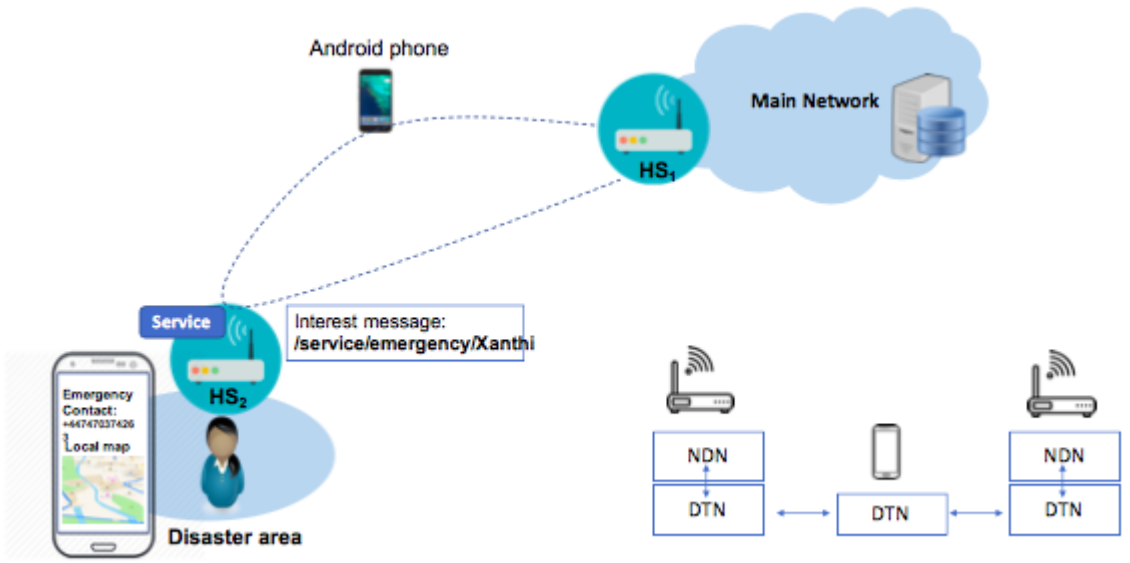


Figure 5: Service provisioning in challenging environments

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 645124

Service Migration, KEBAPP and UMOBILE Hotspot

A key integration in this PoC corresponded to the integration of the last step of PoC, where a KEBAPP emergency service is migrated and deployed in a UMOBILE hotspot placed in the rescue area (depicted in Figure 6). To overcome the absence of network communication, the service migration platform resorts to DTN tunneling provided by an Android phone equipped with the DTN framework. Service Execution Function migrates a service from the local authorities' premises to the UMOBILE hotspot using DTN. Once the service is migrated, a KEBAPP service will be activated in the Hotspot, that will disseminate an emergency server to the users nearby, totally transparent to users without requiring any user interaction. In next section we list all tests & issues found during the integration and the demo for the integration of KEBAPP + Service Manager + UMOBILE Hotspot.

Figure 6: joint integration of KEBAPP, UMOBILE hotspot and Service Manager

3.1. Tests & Validation

Following we present the validation of the service migration service in an isolated area with limited connectivity to perform stress tests and identify issues that in the final demo with a reduced set of resources might not be identified. Next we present the jointly validation of the last part of PoC1 (service migration, KEBAPP and UMOBILE Hotspot). The first part of PoC1 is more extensively validated in PoC2 (Section 3.2).

Validation of the service migration platform in Guifi.net

A central objective of the UMOBILE project is to provide network connectivity in locations with limited connectivity or no connectivity at all. To address the challenge, we are developing a service migration platform that on the basis of information collected by monitors, can strategically deploy services to ameliorate the impact of network problems

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 645124

or ideally, to prevent the materialization of threats. As explained in deliverable D5.3, to demonstrate that the service migration platform has a potential to solve the problem, we will consider a scenario where the network infrastructure is not well connected or has intermittence connectivity.

To overcome these challenges, service migration is developed based on three main aspects: lightweight virtualization, service abstraction layer over ICN and smart service orchestration. The lightweight virtualization technology such as Docker container [5] substantially reduces the size of service image as the system libraries can be customized for each particular service. This makes the service deployment process more efficient as it requires less bandwidth for delivering the service. We also implement the service abstraction layer over ICN which decouples the service from its original location. The node requesting a service image by name can dynamically choose the optimal forwarding path to retrieve a copy of service image from the nearest cache. This is very useful for the challenge network environment as the link to the service repository can be highly intermittent (e.g., link broken, limited bandwidth). Lastly, deploying service in such environment requires the smart service orchestration to select a suitable node to host the service. Given that the node availability can be vastly fluctuated, a node can become suddenly unavailable (e.g., disconnected) or it might not have enough resources to host the service. In this regard, we build the full functional monitoring system that can monitor the nodes in the network. Subsequently, the decision engine applies this monitoring data along with the smart algorithm to make the optimal decision for service deployment.

The overview of service migration platform is presented in Figure 7. The key entity is referred to *Service Manager* (SM) that periodically observes the network topology and resource consumption of potential nodes for the service deployment. In our model, we assume that the service providers upload their services to a service repository inside the SM before distributing to the network edge. To achieve the QoS and overcome the network connectivity problems, SM augments the monitoring data along with service deployment algorithms to decide where and when to place the services. We also introduce the *Service Execution Gateway* (SEG) which provides a virtualization capability to run a service instance at the network edge (e.g., users' house). We use Docker, a container-based virtualization to build lightweight services and deploy across the SEGs. Each SEG is also equipped with the access point daemon (e.g., hostapd [6]) to act as the point of attachment for the end-users to access the services via WiFi connection. A prototype of SEG has been developed on the Raspberry Pi 3 running the Hypriot OS Version 1.2.03 [7]. The *Forwarding Node (FN)* is responsible for forwarding the requests towards the original content source or nearby caches. Each FN is equipped with storage while dynamically caching the content chunks that flow through it. Notice that, FN does not necessarily need to execute the services.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 645124

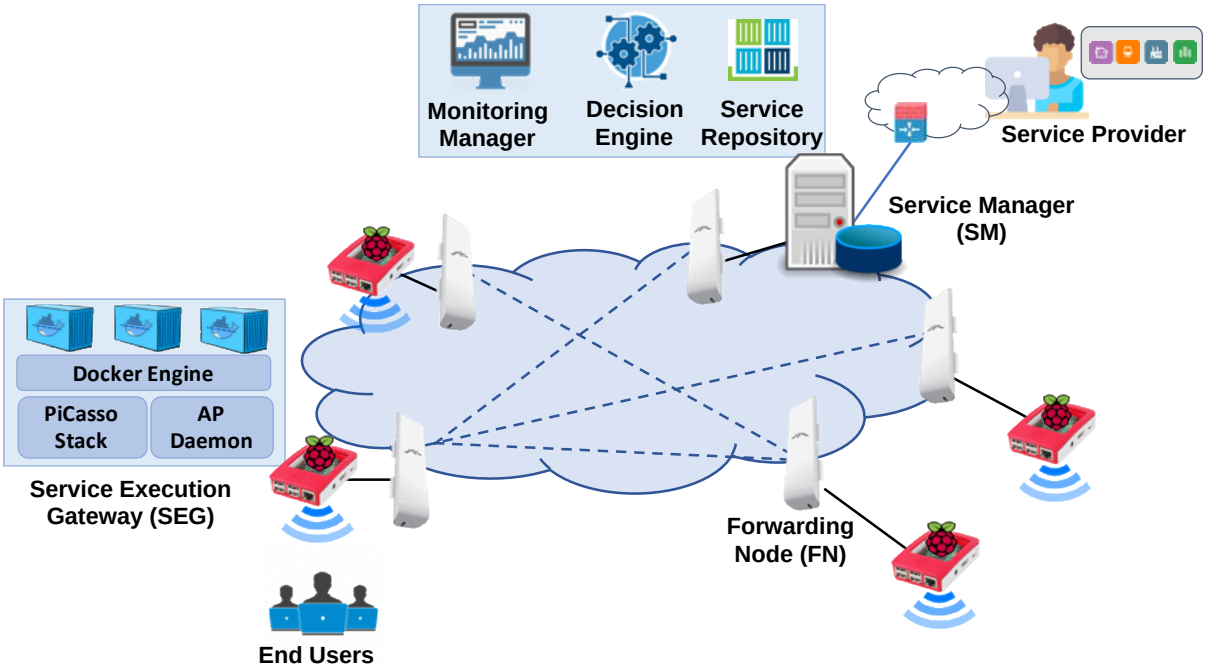


Figure 7: Overview of service migration platform

Validation Key features

In D5.1, the validations of service migration were focused on the quantification and capabilities. For instance, we evaluated the scalability of service migration on UMOBILE hotspot devices. In those validations, we were able quantify the maximum number of services that could be run on the UMOBILE hotspot as well as the maximum number of simultaneous users who can access to a service instance (i.e., Docker container) at the same time. Furthermore, we were also able to identify the critical parameters that impact the service quality. Those parameters include CPU usage, memory usage and CPU load. As a result, we applied this knowledge and understanding to develop the decision engine in order to provide the QoS in UMOBILE network, the results of this study was also reported in D4.4. However, those evaluations were conducted in the control environment like laboratory or UMOBILE testbed where the network connectivity is stable. In fact this is not sufficient to qualify the performance of service migration platform. In addition to that, one of the main objectives of UMOBILE is to provide the service in challenge network work environment such as emergency scenario, undeserved area. Therefore in this deliverable, we aim to evaluate the performance of service migration with the real network environment where network condition might not be stable as well as user traffic sometimes can be highly skewed.

This evaluation has three objectives:

- Objective 1 (O1): To measure and quantify the capability of service migration platform in the real operating network

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 645124

- Objective 2 (O2): To validate the correctness of the functionality of the service migration platform as a whole and of their main components such as its decision engine and monitoring system.
- Objective 3 (O3): To validate the effectiveness of the ICN facilities and compare the performance with existing solution

In pursuit of O1, O2 and O3, we have deployed the service execution gateway (SEG) in Guifi.net, a largest community wireless mesh network located in Catalonia, Spain. In this evaluation, we focus on the QMP network operated since 2009 in a quarter of the city of Barcelona, Spain, called Sants. QMP is an urban mesh network and it is a subset of the Guifi.net community network sometimes called GuifiSants. At the time of writing, QMP has around 80 nodes as shown in Figure 8. A detailed description of QMP can be found in [8].

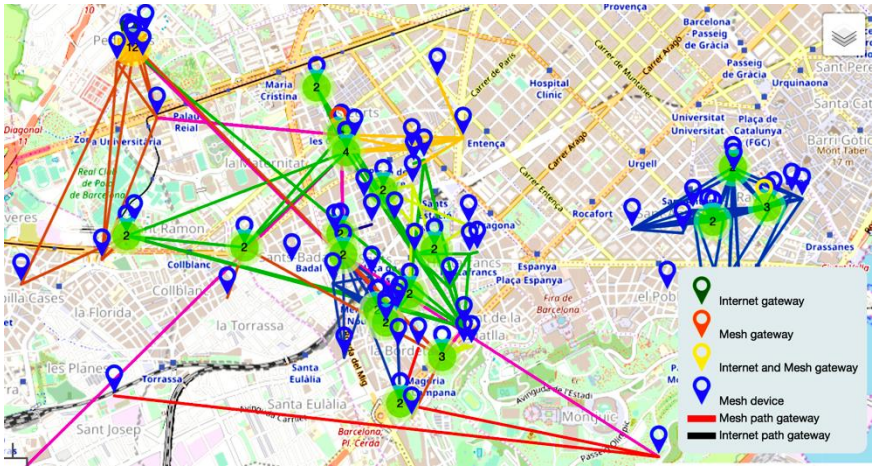


Figure 8: Nodes deployment of QMP network in Barcelona area

In terms of hardware, the qMp users have an outdoor router (OR) with a WiFi interface on the roof as shown in the Figure 8. The ORs are used to build P2P (point-to-point) links in the network. The ORs are connected through Ethernet to an indoor AP (access point) as a premises network where the edge services are running (e.g., on home-gateways, Raspberry Pi's etc.). ORs in qMp are using BMX6 as the mesh routing protocol [9]. For our experimental cases, we deploy (i.e., attach) Raspberry Pi's at the ORs in the network and use them as servers.

In order to understand the feasibility of running the service migration, we deploy our system in a real hardware connected to the nodes of the qMp network located in the city of Barcelona. We have strategically deployed 10 SEGs to cover the area of qMp network as presented in Figure 9. In our configuration, SEGs are connected to the ORs via Ethernet cable and the service controller is centrally set up inside the main campus of Universitat Politecnica de Catalunya (UPC) where the Guifi lab is located.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 645124

We follow the ICN-as-an-Overlay approach [10] to construct the ICN shim layer on top of the existing qMp's routing protocol (i.e., BMX6/7). The NFD forwarding plane is responsible for managing the name based routing in this ICN layer. In this deployment trial, we use a static routing to setup the forwarding table (FIB) of each SEG and service controller based on actual information taken from the IP routing table of ORs in the qMp network.

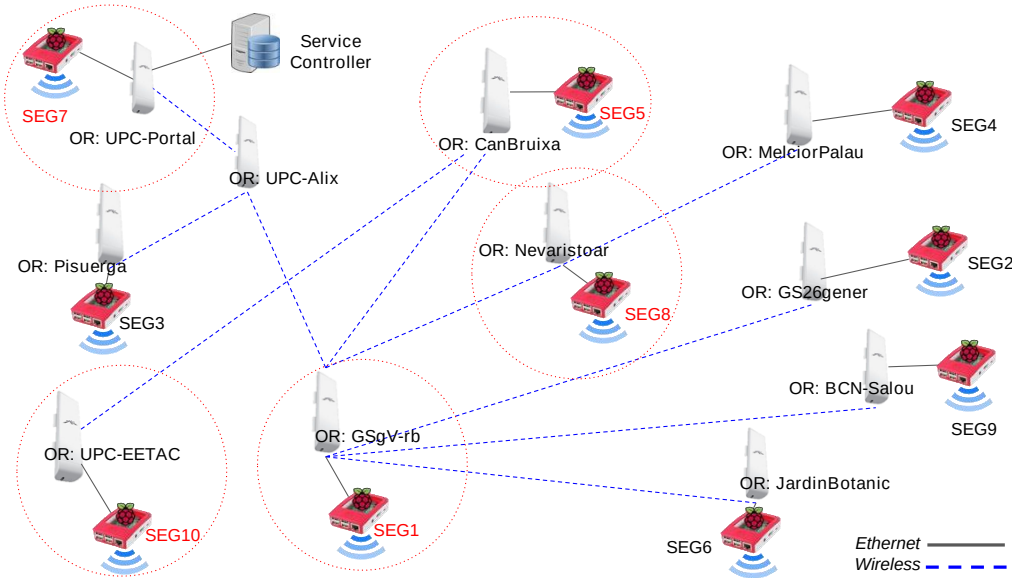


Figure 9: The topology of service migration deployment in qMp

Validate the monitoring system of service migration platform

As mentioned in D3.2, the service manager is also integrated with monitoring manger to collect the monitoring data from all the UMOBILE hotspots deployed in the network. The monitoring manager periodically places the pull requests against the monitoring agent in each SEG to collect the current status of their resources (e.g., CPU load, CPU usage, memory usage, number of running services). This operation follows the native pull-based communication model of NDN where the monitoring manager (consumer) issues the Interest message with a specific name-prefix. Consequently, the SEG (content producer) attaches the monitoring data with JSON format to the DATA message and responds by means of forwarding the requested content towards the Interface following information in the PIT table. As shown in Figure 10, the monitoring manager places the pull requests towards SEG1 and SEG2 while configuring name-prefixes as */picasso/monitoring/SEG1/* and */picasso/monitoring/SEG2/* respectively. When the SEG receives this pull Interest message, it attaches the current monitoring data with JSON format to the Data message and forwards to the same path that Interest message (reverse path

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 645124

forwarding) came from by using information in the PIT. To avoid receiving outdated data from the caches, we set the data freshness to a small value (e.g., 10ms). Figure 11 shows the dash board of monitoring system while retrieving the data from SEGs deployed in qMp network.

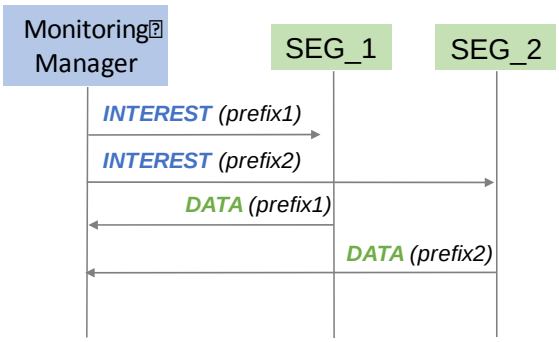
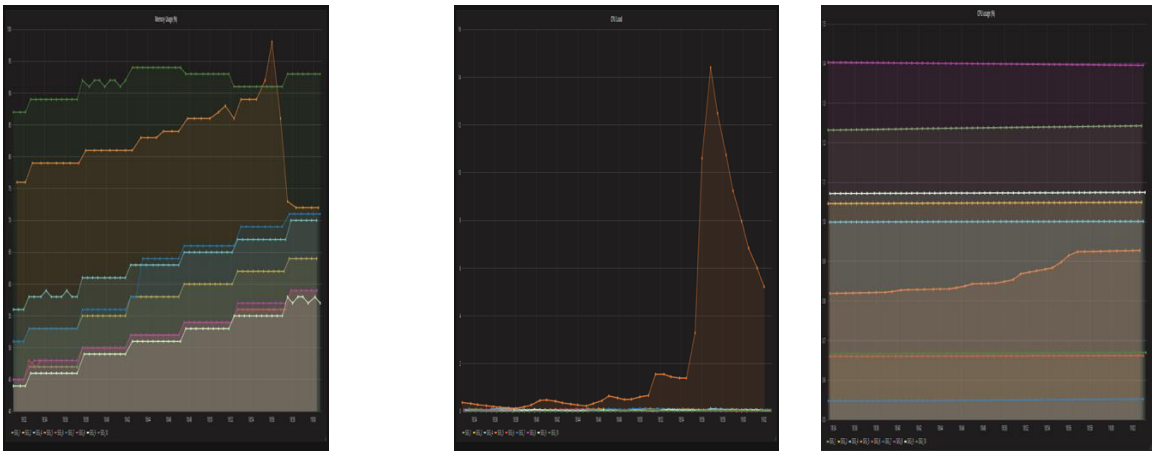


Figure 10: Monitoring manager retrieves the monitoring data from UMOBILE hotspot by using pull-based communication model



(a) Memory usage (b) CPU load (c) CPU utilization

Figure 11: The dash board of monitoring system of service migration platform

Validate the efficient of UMOBILE service migration in the real community wireless mesh network

In the design of service migration platform (see details in D3.4), the service images are kept in Service Repo which is the main repository for service provider to upload their services. To distribute the services over the network, service migration platform requires the push-based communication model to push the service from the service repo to the SEG at the edge of the network. To support this operation, we have implemented the push communication model based on Interest/Data exchange of primitive NDN. We follow the publish-subscribe model [11] where a data producer (DE) publishes contents or services via Interest message to a subscribed consumer which in turn trigger an Interest back from

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 645124

the consumer to fetch the data. Figure 12 illustrates the Interest/Data exchange of the push-based model, where the DE initially sends a push Interest message to SEG1 with a name prefix: */picasso/service deployment/push/SEG1/service name*.

To distinguish the push Interest message from the NDN pull model, a name component, "push" is added after the operation name (i.e., "service deployment"). Consequently, when SEG1 receives the push Interest message, it discards the ("push") and ("SEG ID") prefixes while reconstructing a new Interest name: */picasso/service deployment/service_name/#00* to request the service image. In NDN, a content is divided into several chunks, the last prefix is reserved for the requesting chunk ID which is started from zero (e.g., #00).

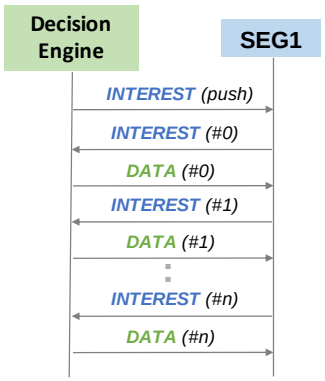


Figure 12: Decision Engine delivers the service to SEG_1 by using push-based communication model

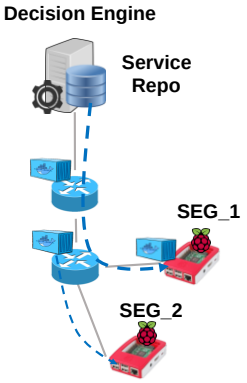


Figure 13: Service Migration benefits from UMOBILE architecture as SEG_2 can retrieve the service from the cache

Due to the efficient caching and content-oriented routing based on naming scheme of NDN, the content/service requests are not necessarily delivered from the original content source (i.e., service repo), but they are instead served by the closet node owning the matching named content. Figure 13 shows an example that highlights the efficiency of dynamic content distribution in UMOBILE network and how content/service can be kept closer to the user request at the edge. It considers a scenario where the decision engine decides to deliver the service to SEG_1 and SEG_2. At first, the decision engine initially sends the push Interest message to SEG_1 as mentioned in Figure 12. During the service deliver process, the forwarding nodes along the path between service repo and SEG_1 naturally store the content chunks in their cache (content store). As for the second stage, the decision engine subsequently sends the push Interest with the same service name to SEG_2. Thanks to the named based routing in UMOBILE and cached content chunks by the former request, SEG_2 can opportunistically fetch the content from the nearest forwarding node without unnecessary route towards master content store. This is very helpful for service delivery in the challenged network environment where connectivity between original content source and the network edge is not always stable (i.e., community network environment).

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 645124

Migrating service instance to the network edge can help to improve service performance but it also introduces some extra costs. In D4.4, we have identified two types of costs including network traffic cost and instantiation cost. The network traffic cost is the traffic generated by the transfer of the service image from the *Service Repo* to the UMOBILE Hotspot. It is mainly dependent on the size of service image and the bandwidth of network link. The instantiation cost is the time that UMOBILE hotspot takes to have a newly deployed instance ready for serving. In D4.4, we have already studied the instantiation cost of the Docker engine on the UMOBILE hotspot over the UMOBILE testbed. However, in this deliverable, we aim to investigate the network cost of the service migration over the real operating network. As a matter of fact, it would be difficult to justify the results, if we run the experiments in the control environment like the testbed. Since the network cost is mostly influenced by the real user traffic and network topology such that we decided to validate the service migration platform over the real operation network (i.e., qMp) rather than running on the UMOBILE testbed.

To evaluate the effectiveness of service migration platform, we set-up a scenario as shown in Figure 9 where the service manager in UPC campus wants to deliver the service the all UMOBILE hotspots located across the QMP network. We focus on the delivery cost which is the total time counting from when the DE makes a service deployment decision until the service is delivered to the SEG. We compare the delivery cost of our service migration solution (UMOBILE) with the classic host-centric networking approach (HCN) which is commonly used in many edge computing platforms such as Cloudy [12] and Paradrop [13]. To implement this approach, we disable in-network caching facility of UMOBILE and direct the service to be delivered from the service repo to each SEG, which is also similar to the IP unicast.

Analysis of Service Delivery Cost

In this evaluation, we select four dockerized containers which have different image sizes from the Docker hub[1] (see details in Table 1) and migrate them from the service repo to all the deployed SEGs.

Table 1: Sizes of Dockerized images used in the experiments

Image name	Image size
hypriot/rpi-nano-httpd	88 Kbytes
hypriot/rpi-busybox-httpd	2.16 Mbytes
armhf-alpine-nginx	251 Mbytes
armbuild/debian	145 Mbytes

[1] https://hub.docker.com/explore/

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 645124

Overall, the average delivery cost achieved by UMOBILE is substantially lower than the HCN approach. For instance, PiCasso can reduce the delivery cost of the *armbuild/debian* image from 154.94 to 70.74 seconds which is about 54% improvement compared to the HCN solution. To have a closer look how a service image is delivered, we focus on the Debian image and plot the delivery time across each node, as presented in Figure 14. By comparing HCN and UMOBILE, we observe that every SEG is better off through the in-network caching and named-based routing capabilities of UMOBILE. The SEGs running UMOBILE are able to retrieve the data chunks from the nearest cache. On the other hand, the HCN approach is inefficient in terms of bandwidth utilization. Given an example of SEG6, HCN acquires 295 second to deliver the service which is converted to 0.49 MBps throughput. However, from the iperf measurement, the bandwidth between SEG6 and the service repo is approximately 1.32 MBps. As previously stated, the resources in qMp network are not uniformly distributed. This indicates that the traditional HCN approach is not sufficient to support the service delivery in this dynamic environment.

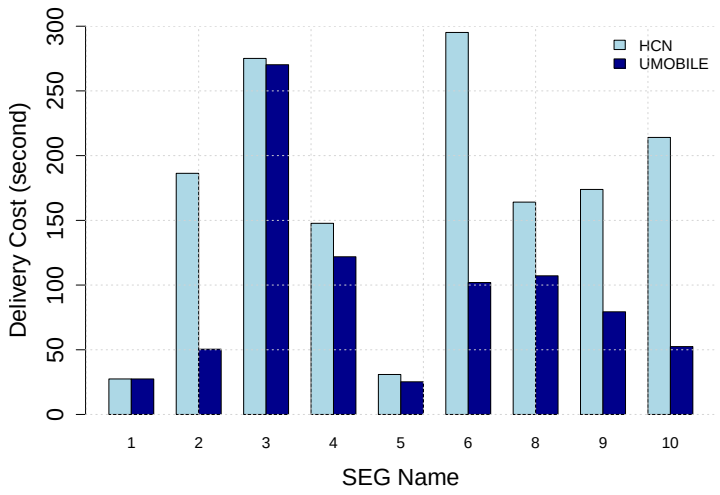


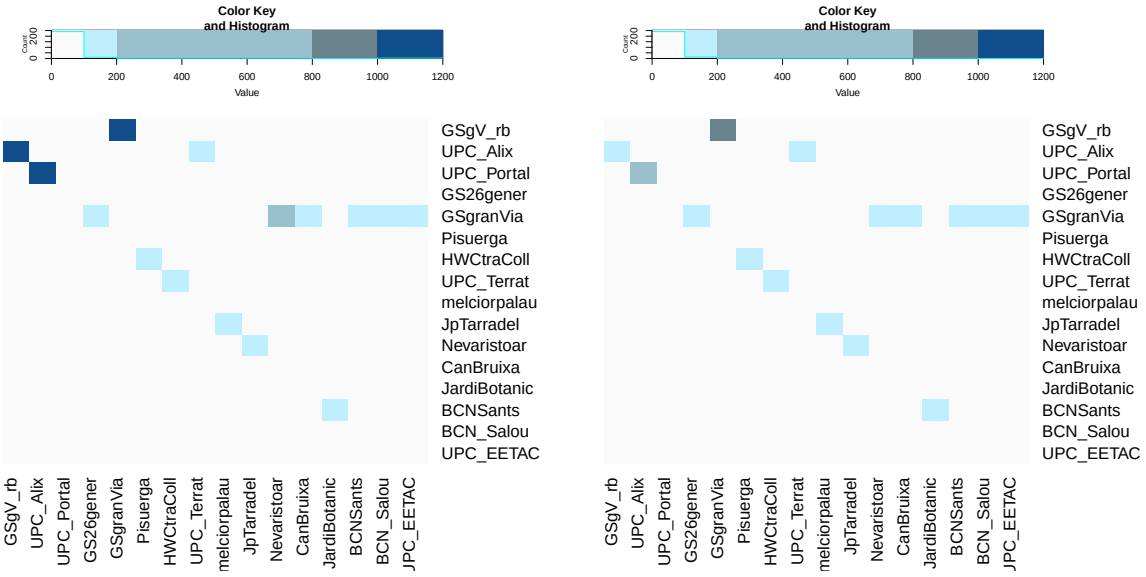
Figure 14: Inspecting the delivery cost of each SEG

Investigating Traffic Consumption of Service Delivery

Previous results demonstrated that UMOBILE efficiently improves the service delivery in the qMp network. To further investigate this, we perform sensitivity analysis on the amount of traffic that is consumed for delivering the service images to the SEGs. We inspect the amount of traffic among SEGs and the service controller from the *nfd-status* reports [14]. However, the information from these reports contains only the traffic of an overlay network. To construct the actual traffic that spread over the qMp network, we map the paths from ICN overlay with the routing tables of BMX6/7 routing protocol used in the qMp. For instance, the path between service controller and SEG5 (see Figure 9) can be mapped to *UPC-Portal - UPC-Alix - GSgV rb - GSgranVia - CanBruixa* (i.e., names denote

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 645124

as the OR nodes). Figure 15 presents the distribution of data traffic sent among the ORs to deliver a service image to all 10 SEGs. It shows in the X and Y axes the name of QMP routers while the gradient on each coordination represents the density of traffic (MBytes) over a link between two routers. Here, we solely present the results of delivering *armbuild/debian* image (the largest image size in the experiments) due to space constraints. The total amount of traffic consumed by HCN approach is approximately 5.375 GB while our UMOBILE solution achieved only 3.05 GB which is about 43.24% reduction. In case of HCN, the most dominant traffic path is a link between *GSgV rb* and *UPC-Portal* since this is a bottleneck link between nodes deployed in qMp and the service controller at UPC Campus North. In contrast, UMOBILE significantly reduces the traffic over this link. The reason is that UMOBILE takes benefits of the edge caching by allowing SEGs retrieve the service image from the closer node. As illustrated in Figure 9, we deployed SEG1 at the node *GSgV rb* which has the highest degree centrality (i.e., it is well connected by other nodes). In this manner, several nodes (e.g., SEG2, SEG5, SEG6, SEG8, SEG9) can directly retrieve the data chunks from the cache of SEG1. This is very useful as the cache is utilized closer to the network edge.



(a) Experiments with HCN (b) Experiments with UMOBILE

Figure 15: Data traffic distributed over qMp network

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 645124

Jointly Validation of Service Migration, KEBAPP and UMOBILE Hotspot

In the following we list all the operational tests done for the integration of SM, KEBAPP and UMOBILE Hotspot, and we describe all the issues we had during the integration and the demo:

- Implement the Service Execution Function running in the Raspberry Pi: Done, but having memory issues with NFD in Raspberry Pi. We had to increase swap memory allocation.
- Encapsulate any service into a Docker image to be instantiated in a UMOBILE hotspot: In order to do so, we used PyNDN[2] to build a python version of a KEBAPP emergency service that disseminates a video with instructions. We managed to build the Docker image with less than 25MB, in order to be able to migrate it with the minimum number of chunks.
- Implement and encapsulate the emergency information service into a Docker image: Done
- Integrate Service Execution Function with Fonera for the UMOBILE hotspot: We integrated both elements by automating wireless network activation only when the service migrated is up.
- Activate 802.11u information: We did not manage to activate 802.11u in the Foneras due to hardware incompatibilities. In order to announce the service though, instead of using 802.11u, we used the SSID name. We use an automatically configured hash-like name in the SSID, which by using a bloomfilter, can be detected whether the service required is activated in the announced SSID.
- Test service discovery with Android phones: Done
- Test emergency service auto discovery and instantiation: Done
- Implement user emergency app using KEBAPP: Done
- Implement service discovery using 802.11u information: Instead using 802.11u, the service is discovered by detecting the service in the SSID using a bloomfilter.
- Test network discovery and connectivity with UMOBILE hotspot: Done
- Test the whole integration Service Execution Function - hotspot – KEBAPP: We had some memory issues with the whole process, but solved after detecting we could increase the swap memory. Also, we realized most of the issues were caused by checking the NFD status during the demonstration, which was using too much memory, which we avoided.

[2] https://github.com/named-data/PyNDN2/blob/master/README.md

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 645124

3. Deployment Trial for POC2

3.1. Overview

This demonstrator aims to show the usage of UMOBILE technology on a scenario encompassing service announcements and awareness about social routines. In this scenario users generate and share an expression of interest in the form of tagged information (e.g., nearby supermarket offers or restaurant review), by means of the UMOBILE End-User Services[3] (shown in Figure 16) [17]. The message is sent to the UMOBILE system where all UMOBILE users can benefit from the local information. In this scenario, the UMOBILE system keeps track of the local event and update useful information to the users based on the new created information and its relevant hashtag.

Figure 16: UMOBILE End-User Services interface

In this scenario, users are capable of exchanging and sharing information, such as short messages, textual information, files and photos, among their contacts, although users' devices do not have to be always connected to the Internet. Data is exchanged among people passing by, based on social interaction approaches. The UMOBILE system takes advantage of all available wireless connectivity (e.g., infrastructure of provided by peers) and selected the most suitable transmission protocol depending on the network environment.

Furthermore, the UMOBILE system provides users' with additional information regarding their personal expressed interests and, e.g., estimation of the time to get to a certain suggested location based on history information. Another crucial aspect supported by UMOBILE is the capability to capture personal data of UMOBILE users (e.g. visited networks, affinity network) with the ultimate goal of improving the user's routine.

[3] https://github.com/Senception/UMOBILE_UES

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 645124

3.2. Integration

The integration of all UMOBILE components used in this scenario had a focus on the smart routing proposal of UMOBILE for allowing the operation of Named-Data Networking in networking scenarios facing intermittent connectivity. The proposal, called Data reAchaBility BasEd Routing (DABBER) protocol [18] was developed to extend the reached of Named Data Networking based routing approaches to opportunistic wireless networks. By "opportunistic wireless networks" it is meant multi-hop wireless networks where finding an end-to-end path between any pair of nodes at any moment in time may be a challenge.

The goal is to assist in better defining opportunities for the transmission of Interest packets towards the most suitable data source, based on metrics that provide information about: i) the availability of different data sources; ii) the availability and centrality of neighbor nodes; iii) the time lapse between forwarding Interest packets and receiving the corresponding data packets.

The proposed routing protocol, DABBER, is described in an IETF draft document (draft-mendes-icnrg-dabber-00), which presents an architectural overview of the proposed protocol followed by specification options related to the dissemination of name-prefix information to support the computation of next hops, and the ranking of forwarding options based on the best set of neighbors to ensure short time-to-completion.

The proposed smart routing protocol required the integration of the following components:

1) Contextual Manager:

Contextual information is obtained in a self-learning approach, by software-based agents running in wireless nodes, and not based on network wide orchestration. Contextual agents are in charge of computing node and link related costs concerning availability and centrality metrics. Contextual agents interact with DABBER via well-defined interfaces. This to say that the contextual self-learning process is not an integrating part of the routing plane, as it would add additional complexity to the simplified routing plane of NDN.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 645124

Figure 17: Contextual Manager running

The contextual agent (named Contextual Manager, CM [19]) installed in each wireless node can therefore be seen as an end-user background service that seamlessly captures wireless data to characterize the affinity network (roaming patterns and peers' context over time and space) and the usage habits and data interests (internal node information) of a node. Data is captured directly via the regular MAC Layer (e.g., WiFi, Bluetooth, LTE) as well as via native applications for which the user configures interests or other type of personal preferences. For instance, an application can request a one-time configuration of categories of data interests (e.g., music, food).

Based on the defined interface, DABBER is able of querying the local Contextual Manager about the characteristics of neighbor nodes, based on two types of information: Node availability (metric A); ii) Node centrality (metric C).

- Node Availability (A) gives an estimate of the node availability based on the usage of internal and external resources over time and space. In what concerns external resources, one SHOULD consider, for instance, indicators such as the preferred visited network and/or location of the node; in what concerns internal resources, one SHOULD consider the time spent per application category (e.g. per day), as well as the usage of physical resources (battery status; CPU status, etc).
- Node centrality (C) provides awareness about a node's affinity network neighborhood context. For instance, aspects such as the traditional contact duration between neighboring nodes and add information derived from network mining such as cluster distance, and network diameter MAY be the basis for the computation of centrality.
- Nodes Similarity (I) provides a measurement of the clustering similarity between two nodes. This parameter provides information about the link weigh, which

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 645124

should be proportional to the similarity between nodes, based upon collected data about contact duration, battery status, mostly used apps, etc.

The interface between DABBER and CM provides the former with periodic information concerning a node's centrality (C) and a node's availability (A). The interface integrates also a premise to provide a similarity weight (I) between two peers passed by DABBER to the CM. For instance, if DABBER requests similarity between node A (sender) and node B (potential successor), then the CM computes similarity for both nodes based on a specific period of time. Such analysis can assist in a better selection of peers for data transmission, for instance.

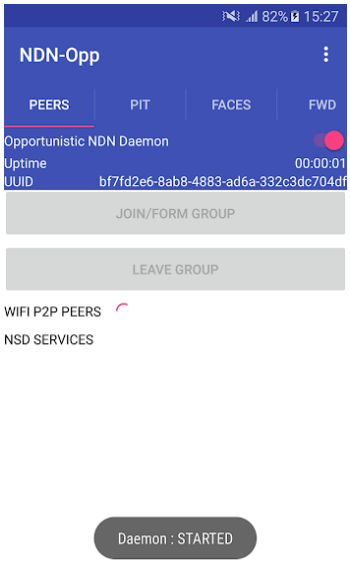


Figure 18: NDN-Opp service

From a routing perspective, the contextual plane provides weights (A, C and I) to assist the routing protocol in ranking next hops, which is an aspect highly relevant in the context of multiple path routing. We believe that contextual awareness can assist NDN routing schemes in better dealing with topological variability, by anticipating changes derived from prior learning.

2) NDN extension to Opportunistic Wireless Networks: NDN-OPP

The main functionality of the Named-Data Networking Forwarding Daemon (NFD) is to forward Interest and Data packets. This section provides a set of design considerations that need to be considered to allow the operation of NFD in opportunistic wireless networks. Such considerations have been implemented in a new branch of NDN, called NDN-OPP [20], whose code of available on GitHub[4]. NDN-OPP introduces a few modifications in the way NFD performs its forwarding, by leveraging the concept of Faces in order to adapt the operation of the NFD to the intermittent property of wireless

[4] https://github.com/COPELABS-SITI/ndn-opp

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 645124

connections. This is done by the implementation of a new type of face, called Opportunistic Face – OPPFace.

Each OPPFace is based on a system of packet queues to hide intermittent connectivity from NFD: instead of dispatching packets from the FIB, the OPPFace is able of delaying packet transmission until the wireless face is actually connected. OPPFaces are kept in the Face Table of the forwarder and their state reflects the wireless connectivity status: they can be in an Up or Down state, depending upon the wireless reachability towards neighbor nodes. Since packet queuing is concealed inside OPPFaces, no other part of the NFD or any existing forwarding strategy needs to be changed. OPPFaces can be implemented by using any direct wireless/cellular communication mode (e.g., Ad-Hoc WiFi, WiFi Direct, D2D LTE, DTN).

The current operational version of NDN-OPP (V1.0) makes usage of group communications provided by WiFi Direct. In this case there is a one-to-one correspondence between an OPPFace and a neighbor node.

In this peer-to-peer scenario, OPPFaces can be used in two transmission modes: connection-oriented, in which packets are sent to a neighbor node via a reliable TCP connection over the group owner; connection-less, in which packets are sent directly to a neighbor node during the WiFi direct service discovery phase. In the latter case data transmission is limited to the size of the TXT record (900 bytes for Android 5.1 and above). In the peer-to-peer scenario of WiFi direct, DABBER operates as follows: routing information is shared among all members of a WiFi direct group, while Interest Packets are forwarded to specific neighbors. With Dabber it is the carrier of an Interest packet that decides which of the neighbors will get a copy of the Interest packet. Hence, with the current implementation of NDN-OPP, DABBER places a copy of the Interest packet in the OPPFaces of selected neighbors. In what concerns the dissemination of routing information, it is ensured by: i) node mobility, meaning that nodes carry such information between WiFi direct groups; ii) information is passed between neighbor groups via nodes that belong to more than one group.

In a scenario where NDN-OPP would have OPPFaces implemented based on a broadcast link layer, such as adhoc WiFi, only one OPPFace would be created in each node. Such OPPFace would be used to exchange packets with any neighbor node, making use of the overhearing property of the wireless medium. Since with DABBER, it is the carrier that decides which of the neighbors are entitle to get a certain Interest packet, DABBER would need to encode in the Interest packet information about the ID of the neighbors that should process the overheard Interest packet.

This means that the operation of DABBER is the same independently of the nature of the link layer protocol, the only different being the number of transmissions that needs to be done at the link layer to forward Interest packets and to disseminate routing information.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 645124

Besides the OPPFaces towards neighbor wireless nodes, NDN-OPP makes use of the WiFi Face, already defined in NFD, and will integrate the DTN Face developed in the UMOBILE project. This means that DABBER is able of exploiting any available wireless Face (OPPFace, WiFi Face, DTN Face). Future versions of NDN-OPP will allow DAGGER to exploit interfaces to other wireless access networks, such as LTE.

3) Short Messaging in Opportunistic Wireless Named-Data Networks: Oi!

Since the communication paradigm Named-Data Networking (NDN) was proposed, a lot of research work has been carried out with the intention of providing new functions and ideas to exploit the capabilities of such infrastructure. At the same time, we have been watching to an increase of the popularity of applications such as file sharing and group text messaging.

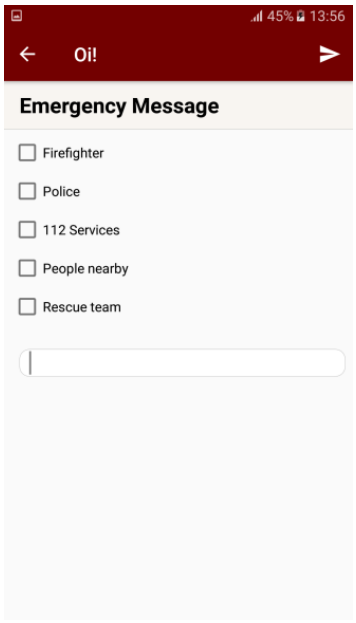


Figure 19: Oi! service

Although, continuous research has helped to create a more stable NDN network, it is important to create applications that bring value and usability from the perspective of the end user. On another hand, people want to share content more often, for that reason we consider that NDN needs similar applications to illustrate its potential.

Hence, in the UMOBILE project we have developed a novel short message application, Oi! [21]. With this application we attend to show the push model and Long Live Interest(LLI) capabilities exiting in NDN-OPP allowing data to be exchanged even in the presence of intermittent connectivity.

In the context of Oi! two different types of user are considered, End Used and Authority User. The idea of this is to show the capabilities of Oi! in different scenarios. In that sense, an End User is going to be able to establish chat sessions in other to exchange messages with others End Users, as well as, to send message to Authority Users, for instance in an

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 645124

emergency scenario after occur an accident and End User could send emergency messages to the Authorities.

In another hand, in the actual version of Oi! an Authority User is going to be able to receive messages only from End Users. The intention of this is to demonstrate the benefits of use LLI and Push Data. These mechanisms are explained in NDN-Opp technical report [20].

Oi! is an open-source application developed for Android that allows users to exchange short text message over an NDN infrastructure. Usually applications that work on top on NDN infrastructure use the pull communication paradigm in order to exchange data. However, it is possible to develop push communication applications on top of NDN, namely to establish a voice conversation over content-centric network. In the same way we aim to implement a mechanism to exchange data in Oi!, which makes use of the push communication model implemented in NDN-OPP.

With Oi!, users can maintain conversation sessions with different users at the same time. Since Oi! was developed on top of NDN-OPP, Oi! is able of exchanging messages even in condition with intermittent wireless connectivity.

Oi! is specified based on a modular approach, so it is possible to add new components and allow integration with the NDN Forwarding Daemon (NFD) which is a network forwarder that implements NDN: however, the operation of Oi! on top of NFD is only possible if NDN Android implements a push communication model, as done by NDN-OPP.

4) Data sharing in Opportunistic Wireless Named-Data Networks: Now@

Since the communication paradigm Named-Data Networking (NDN) was proposed, a lot of research work has been carried out with the intention of providing new functions and ideas to exploit the capabilities of such infrastructure. At the same time, we have been watching to an increase of the popularity of applications such as file sharing and group text messaging. As a result, to support the development of such sharing applications on the NDN framework, ChronoSync [22] provides a library developed in C++ that allows the synchronization of data in distributed systems application over NDN.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 645124

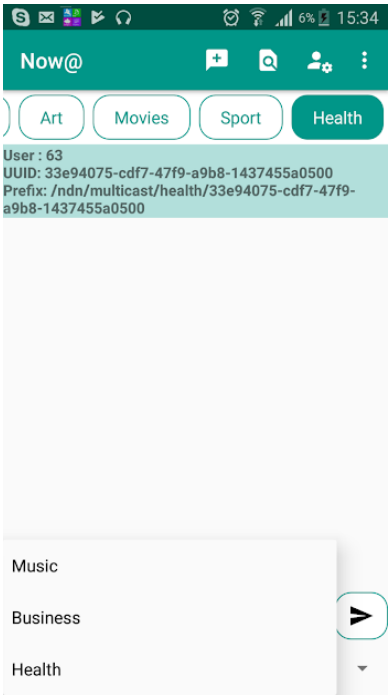


Figure 20: Now@ service

Although, continuous research has helped to create a more stable NDN network, it is important to create applications that bring value and usability from the perspective of the end user. On another hand, people want to share content more often, for that reason we consider that NDN needs similar applications to illustrate its potential.

Hence, in the UMOBILE project we have developed a data sharing application, Now@ [23, 24]. This application aims to increase the impact of NDN near the end user with an Android application that allows them to exchange data (e.g. files, photos) based on their interests. Now@ is being developed based on Chronosync for mobile devices. Now@ can operate on top of NFD Android allowing data exchange via wireless Internet and on top of NDN-OPP allowing data to be exchanged even in the presence of intermittent connectivity.

Now@ is an open-source application developed for Android that allows users to exchange information such as text, images and documents over an NDN infrastructure, using predefined categories that help users define the interests they want to communicate about.

To do this, we use ChronoSync to carry out the transfer and synchronization of data between users even in the presence of intermittent wireless connectivity. In contrast with applications like ChronoShare and others that also work on top of ChronoSync, Now@ allows users to subscribe to more than one interest at the same time.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 645124

Now@ has been specified based on a modular approach, so that it is possible to easily add new components, while allowing the integration with the NDN Forwarding Daemon (NFD), which is a network forwarder that implements NDN.

3.3. Tests & Validation

In this demonstrator, a UMOBILE user (shown in Figure 21), generates and shares an expression of interest by means of Now@ or sends a short message by means of Oi!. In either case, data is sent to NDN-OPP, which is able to disseminate such data by means of DABBER plus the Contextual Manager via direct wireless links as well as available open WiFi infrastructures. The deployed system takes advantage of all available connectivity (e.g., infrastructure wireless or directly via wireless peers) and selecting the most suitable transmission protocol depending on the network environment.

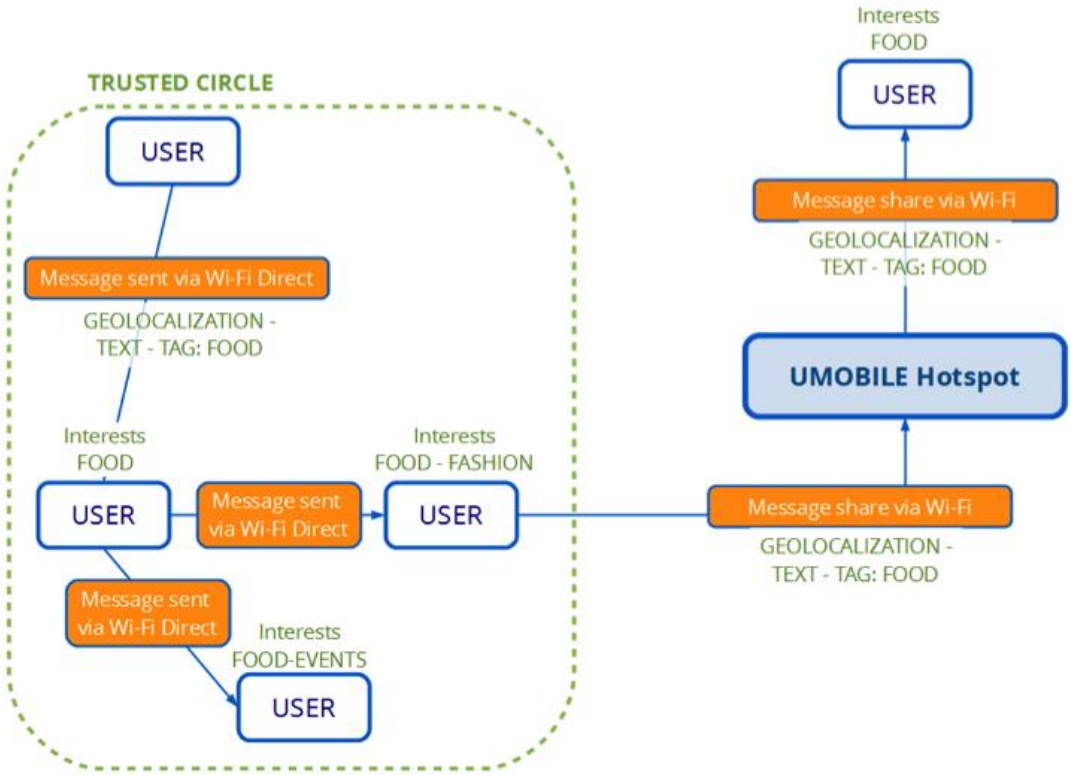


Figure 21: PoC2 overview

In a first validation scenario, we make use of Oi! with NDN-OPP, DABBER and Contextual Manager to improve the efficiency of emergency services in the presence of intermittent connectivity. This is done by allowing citizens to send messages and other digital data to emergency centers using secure, reliable wireless services, able to operate even in the presence of intermittent connectivity.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 645124

In this set of tests we leverage the Named-Data Networking (NDN) framework to develop a messaging system able to allow citizens to communicate with authorities, as well as with close-by people in emergency situations, even in the presence of intermittent connectivity.

To support emergency communications based on NDN, Oi! makes use of two push communication mechanisms implemented in NDN-OPP to allow message exchange within the same wireless network: i) push based on special data packets (pData); ii) push based on a special Interest packet (LLI - Long Lived Interest).

pData are special data packets implemented by NDN-OPP to allow emergency data to be pushed to any nearby wireless user (e.g. in a rescue situation) without the need to have them sending Interest packets. When a pData is received, NDN-OPP does not check the local content store neither the PIT, which is the regular flow of a data packet in NDN. Instead NDN-OPP sends the pData directly to the next-hop based on the information stored in the FIB. Data packets are forwarded if the pData counter is higher than zero (this counter is decreased by NDN-OPP before dispatching the packet). The pData counter aims to limit the dissemination range.

Long Lived Interests are used to allow users to send emergence messages to specific authorities within the same wireless network, independent of the topology distance. LLI are special interest packets (have a specific TLV) implemented by NDN-OPP to allow emergency data to be pushed to a well-identified authority that is willing to receive emergency messages for a certain amount of time (e.g. 3 days). Oi! passes this interest to NDN-OPP by creating an LLI via JNDN. After the reception of an LLI, NDN-OPP sends it as a regular Interest packet, the difference being that data packets do not consume a LLI entry on PIT. Such entry will only be deleted after the LLI times out. With LLI, Oi! immediately dispatches emergency data message to special authorities without the need to wait for an Interest packet. Moreover, this mechanism reduces the number of Interest packets that an authority would produce for a certain amount of time if regular Interest packets would be used.

In a second validation scenario, we make use of Now@ with NDN-OPP, DABBER and Contextual Manager to allow users to share content even in the absence of an Internet infrastructure. This validation scenario aims to show how UMOBILE contribution can increase the impact of NDN near the end user with an Android application that allows them to exchange data based on their interests.

To achieve this goal, Now@ was developed based on a mechanism able to allow the synchronization of data. Now@ can operate on top of NFD Android allowing data exchange via wireless Internet and on top of NDN-OPP allowing data to be exchanged even in the presence of intermittent connectivity.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 645124

4. Additional demo: Deployment Trial for reduced set of POC1 with UAV assistance

4.1. Overview

The use of UAVs for its incorporation in the UMOBILE project allowed the practice of the transmission of information for larger distances – which has a crucial implication in the unpredicted emergency cases in wide areas where there is limited access or it is too far away from the community centers (this means areas not typically covered, such as valleys, canyons or heavily forested areas).

To that aim, emergency scenarios were defined and evaluated the existing product line of UAVs in order to select and perform the necessary validations for this purpose.

4.2. Integration

The definition of technical and specific UAV / equipment requirements was studied, taking into account deployable, operational and regulatory aspects in order to complement the terrestrial networks with the potential of UAVs. The integration of the devices into the UAVs needed to be accomplished considering the technology capabilities for its implementation, and the constraints associated - e.g. SWAP – size, weight and power – in order to enable the UMOBILE scenarios tests at a geographic and scalability level.

Integration into the UAVs followed the reasoning behind the scenarios (with the appropriated permissions to fly, tests licenses, ...) and TEKEVER created nodes compliant with the project architecture and protocol stacks on the UAVs in order to correctly support the operations and confirm the feasibility of the developed services and date mule applications.

Through the tests, demonstrations and validations, some implementations were modified or components upgraded. However, neither specific goal points of the mission were affected; neither general objectives nor non-technical issues were compromised (such as flight safety, instructions to collaborators, or other methodologies). Integrations procedures took part in several UAVs, in order to understand the advantages and drawbacks of each aerial platform, so that the project accomplished the best possible results. Tests and validation processes are overviewed in the following subchapter.

4.3. Tests & Validation

A complete set of tests and successive demonstrations were accomplished to validate the appliance of the diverse implemented systems.

The first step with the intent of the system validation was represented by the execution of systems tests. This part, involving mainly the fundamental systems and its operability, were firstly reproduced (before the demonstrations). Tests were accomplished in order to put into practice the good functioning of the defined UMOBILE schematization that

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 645124

includes the use of various systems – and were grouped distinguishing the concepts of static and dynamic tests.

Static Tests

Concerning static tests, still without the use of UAVs, they allowed the understanding of the performance, initially with Line of Sight (LoS), of the Raspberry PI 3 (with WLAN1) with measurements up to 80m with 10m intervals. Relevant results on these tests procedures proved the functioning of an operational WiFi device, which respective results enclosed a variance of the connection in function of the distance from the WiFi hotspot – as it was supposed, it was observable the loss of bandwidth and increase of time of delay with the increase of the distances – also meaning that the bandwidth decreases with the increase of the jitter. Static tests stopped at a range of 80 meters due to a lack of reliable connection once this mark was surpassed. The following figures illustrate the obtained bandwidth and jitter over the distance, for 4 different tests with the average function.

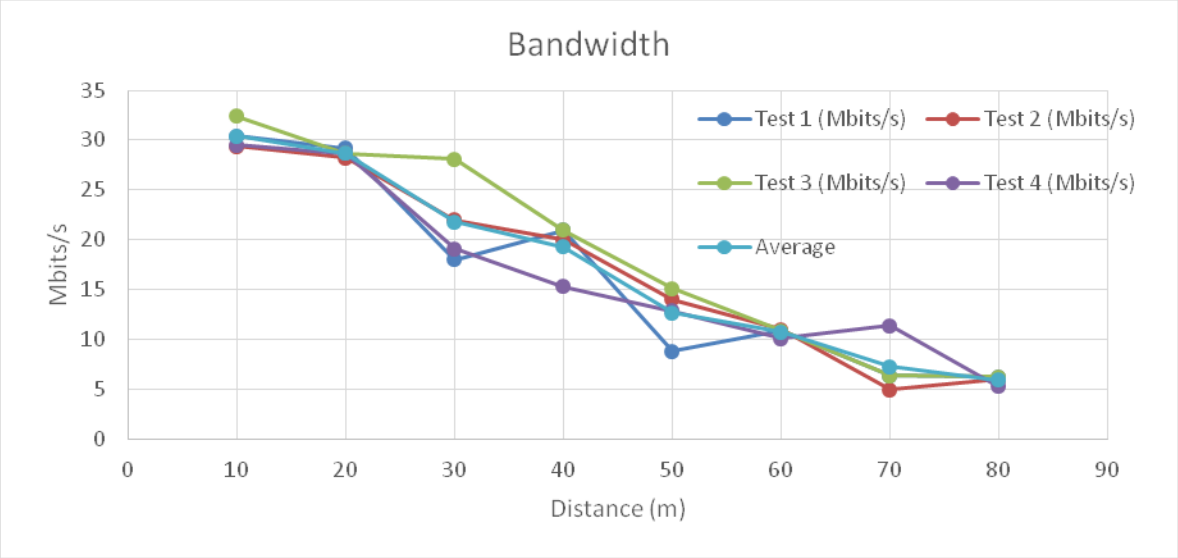


Figure 22: Static tests - Obtained bandwidth (in Mbit/s) over distance (in meters).

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 645124

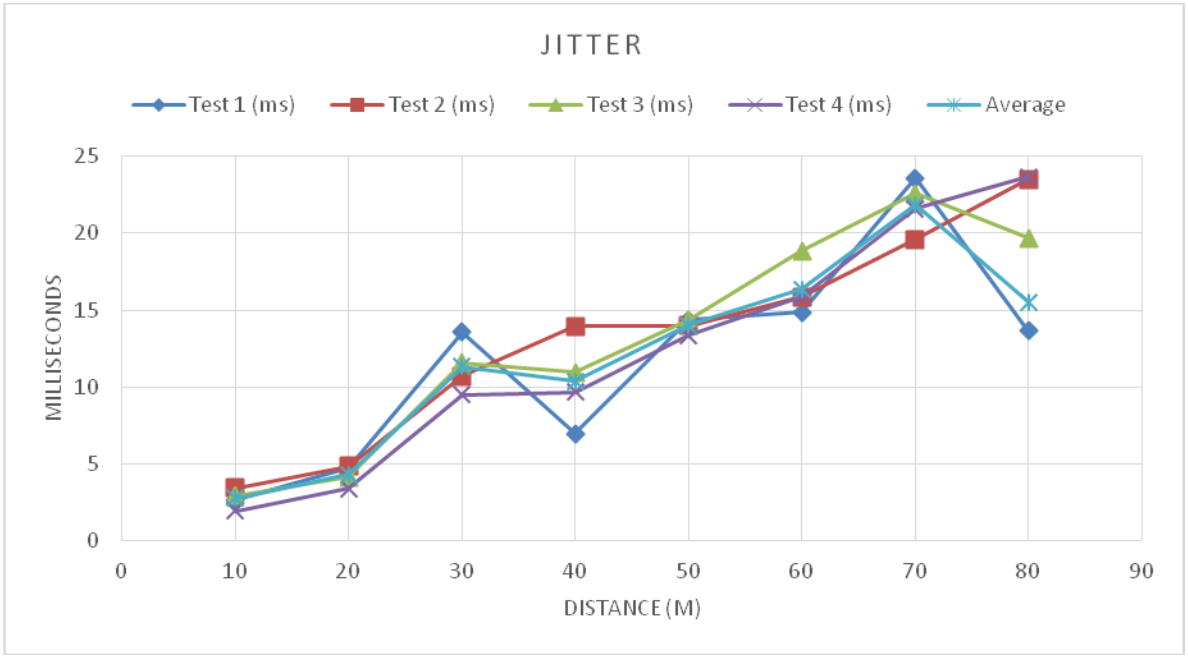


Figure 23: Static tests - Jitter in milliseconds over distance in meters

The following figure represents the lost datagrams over the distance. The same tests as the previous graphics are reported, with the average function.

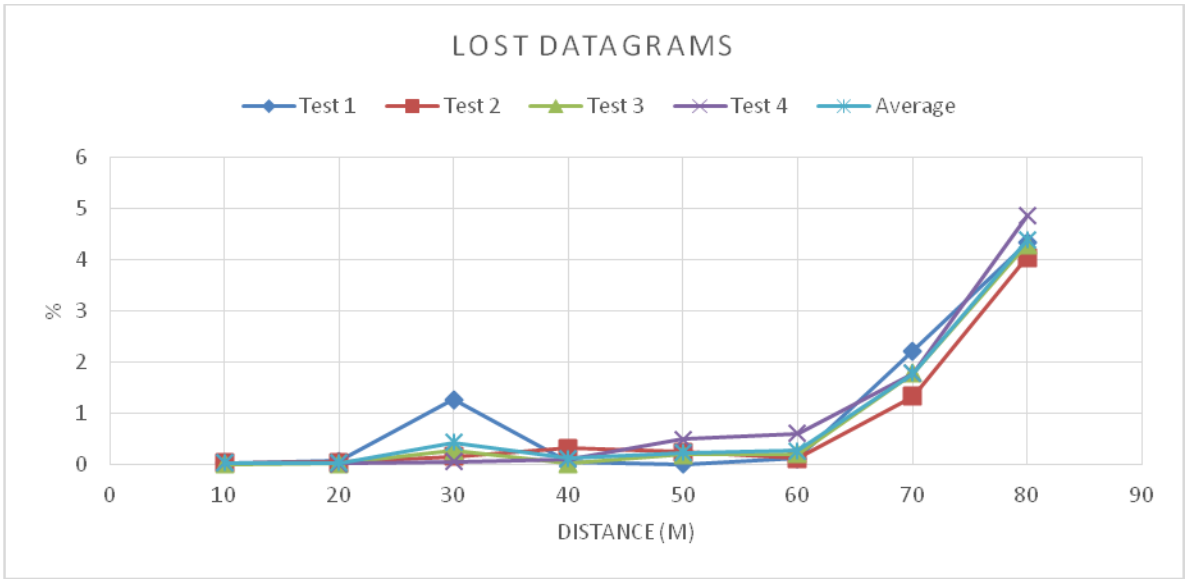


Figure 24: Static tests - Lost data packets over distance

Dynamic Test

Regarding the dynamic test, it was used the same hardware as the static one (Raspberry PI 3 + WiFi dongle). Instead, this time, components were integrated into the UAV. A

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 645124

communication link to the ground server was already established when the UAV lifted off, allowing a more specific validation concerning the installation of the Raspberry PI 3 HW in the aerial platform that circles at a 50 meter altitude, performing the loiter manoeuvre over the user. In this situation, it was concluded that the collaborator started to lose the connection at a 130 m distance to the server. The following shows the variance of the bandwidth and the lost datagrams over the distance for this dynamic test.

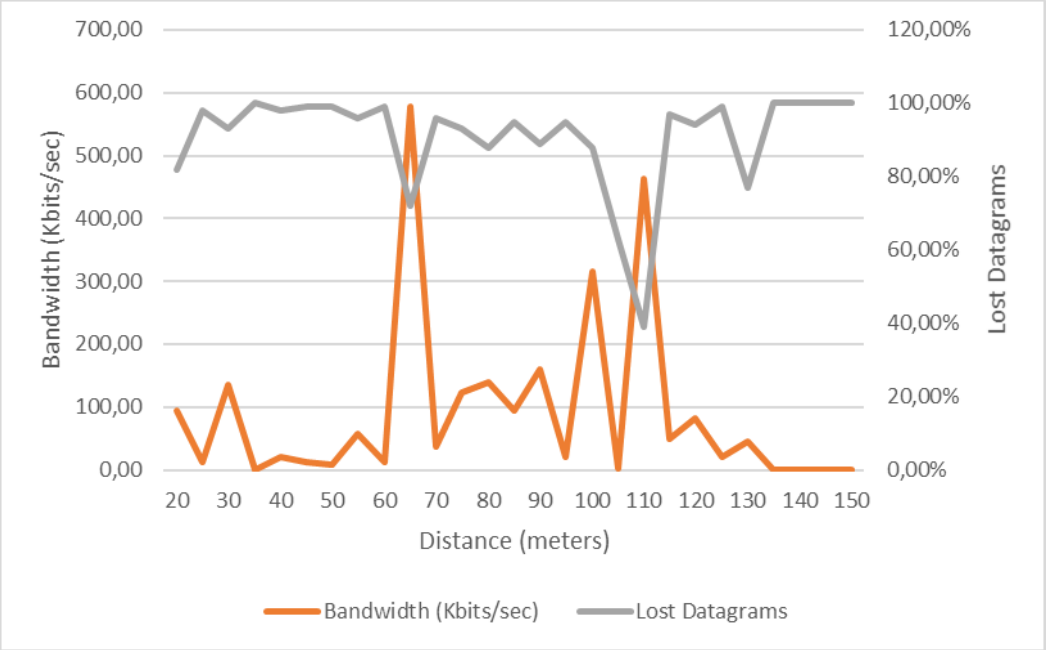


Figure 25: Obtained results for lost datagrams in function of the bandwidth and the distance between user and server

Both static and dynamic tests proved that a more reliable connection needed to be developed, since these reported distances were not sufficient to perform the realistic emergency operations and an excessive rate of data was lost through the communications. A stronger transmission power was then later obtained with the use of a more powerful (2.4 GHz long range) antenna. This was accomplished without compromising the safety of the UAV flight, in accordance with the necessary payload.

Tests also permitted to consider different UAVs to operate in UMOBILE scenarios. At first it was supposed TEKEVER'S AR4 would fulfil the mission; however AR3 was tested and showed in comparison a better performance due to its more reliable capacities.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 645124

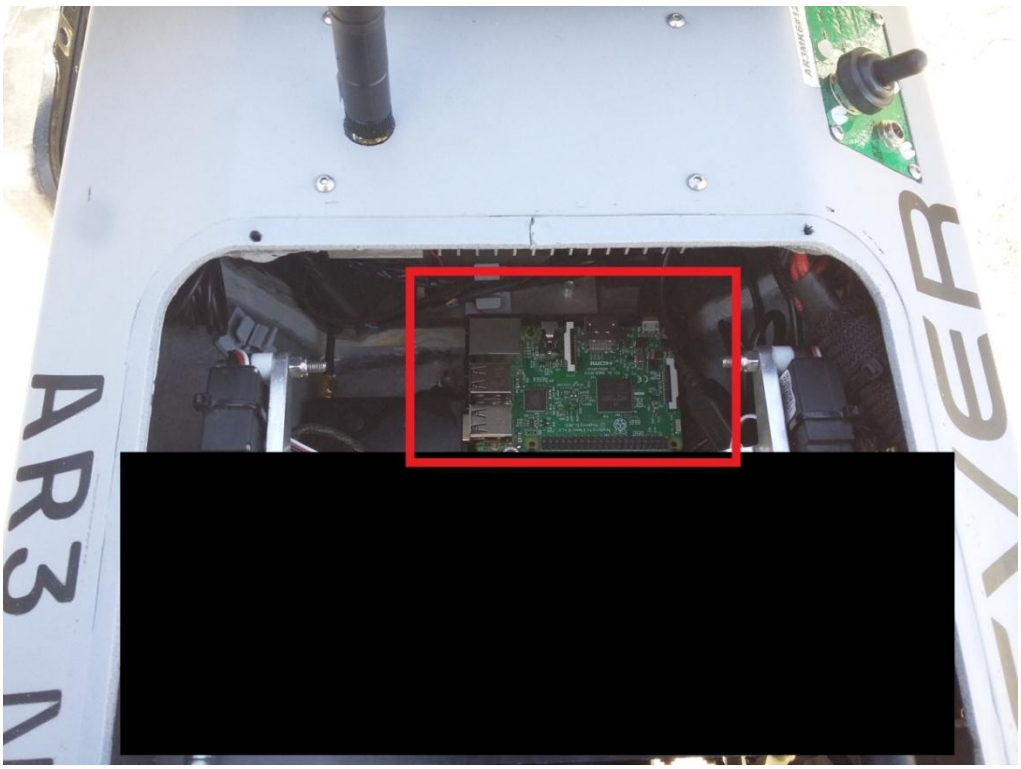


Figure 26: Raspberry PI 3 on the fixed wing TEKEVER's AR3, identified by the red box

Demonstration

This important analysis took special importance when the evolution of the project led to the step where an UAV demonstration was appropriate to perform. In the continuing attempts to plan the UAV demo, meteorological conditions were quite inadequate for UAVs to fly, due to the consecutive winter storms in Portugal (storms Felix, Giselle, and Hugo ...) that resulted in a permanent lack of climatic conditions. Simultaneously, specific licenses to operate the AR3 were not successfully obtained, which led to an UAV demonstration with a multi-copter instead of the use of the fixed wings prototype. TEKEVER VR3.5 multi-copter used for the data mule service (in substitution of the AR3), was also properly implemented to perform the exact same mission just like it was tested with the AR3. Although there was the initial disadvantage of the distance range, the multi-copter still validated the use of UAVs for distances as far as 500 meters.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 645124

Figure 27: TEKEVER's VR3.5 multicopter with long range antenna

The UAV demo was executed in order to complement the POC1 of the project, and make a maximum number of tools developed by the consortium. Once the UAV demo took place, UMOBILE components were validated in a designed scenario representative of the emergency situations without network connectivity, and where any line of sight could not be used directly.

Several systems, applications and services were validated: a new NDN code was installed in the UAV (commercial DJI-Phantom) to work an access point; access point connectivity with authority base using WiFi; DTN node on board of data mule (TEKEVER VR3.5 multi-copter) to transmit data from Authority to the Rescue Team (also validating the range of the antenna); message routing apps OI! and NDN-Opp, via WiFi, assuring the communications, although the NDN Whiteboard app was used to overcome some problems with OI!. Finally, a distance of 1km was simulated between TEKEVER UAV (VR3.5 data mule) and the rescue team, leading to a final 500 meters range of transmission of data from the VR3.5 to the ground.

At the end of these validations steps, a quick communication and rescue-service responding was proved to be operational with the use of UAVs that notably contributed to achieve a long range of message exchanging, and demonstrated being a valuable element

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 645124

in the implementation of devices, fully allowing the complete operation of the systems developed under the scope of the UMOBILE project.

5. Lessons Learned

Following we present the lessons learned during tests and demonstrations, as well as the integration phase:

- The industry members of the/a consortium provided valuable expertise to get the different outcomes run together, in a meaningful environment related to the project; these outcomes are those consisting in the elements of an architecture: systems, devices, software libraries, programs, apps, etc.
- The UMOBILE laboratory and its connection to the NDN worldwide testbed has been a key element in the integration and testing phase. The lab was envisioned and started around the half of the project development. As a recommendation it would have been more helpful to start the lab since the early beginning of the project, with the existing technology components that were among the starting point of the UMOBILE project. The lab will remain an asset to be used for the rapid prototyping of any new proof of concept in further application scenarios, beyond those covered in the project.
- Tests, including the diverse effectuated validation processes, revealed that the performance was completely operational and procedures were complex but workable to its extent. It was also possible, however, to retain that a simpler communication network could allow a more efficient utilization of the systems, especially for the users or communities that are not used to deal with this type of procedures with such technical advancements. The project showed to fully understand the victims needs in emergency situations, which could be confirmed during the demonstrations; still, the connection points and nodes between the victim attempt to ask for help and the first responders reaction might be somehow extensive for someone that is not accustomed to use all the applications and services developed under UMOBILE, and possibly the efficiency of the method might lose track through a determined phase. This apparent adversarial point was mainly noticed when non-technical users or collaborators that were present in the development of the project (the Armed Forces, for example) had a certain difficulty to fully understand the concept and the methodology behind the systems, applications and connection points of the mission and the UMOBILE operational scheme.
- Clock times: If network connectivity is impaired for a long time, the clock time of network devices may differ. A large enough packet lifetime must be used on the DTN layer, to enable the successful transmission of data in these conditions, especially when delivering through NDN-IBR tunneling. More specifically, out-of-

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 645124

sync clocks can be especially problematic when the sender clock is lagging behind, if the packet lifetime is not set on a large enough value both on the DTN and the NDN layer. When launching IBR daemon on several devices, it is also important to correctly configure time synchronization options among them.

  - When the sender's clock is set on a later time that the receiver's one, we did not encounter any problems as the default IBR-DTN behavior is to accept packets arriving with a timestamp later than the receiver's current time.
  - Another solution (which can be used in addition to setting a large lifetime), is to enable synchronization between the IBR-DTN daemons. During our tests, the typical setup included setting the synchronization option on the IBR-DTN smartphone app to "slave" mode. Commonly, if a network device is involved (hotspot or router), it would play the master role for synchronization, and the UEDs (Smartphones) would be configured in slave mode. Among network devices, it runs properly without specific configuration, but in the case that only UEDs are involved and the DTN tunneling is triggered as an offloading path or alternative link, at least one UED should act as the master. That said, some deployments may demand specific tuning if singular events or communication issues are expected.
  - A third solution is to set up a local NTP server to sync the device clocks offline.

- When running NDN on Android, be aware that a TCP or UDP face can be created and used for route registration, no matter the physical interface being active for that connection. That is, the same face could be available on the NDN daemon, whether making use of the WiFi interface or the 3G connection. From a user's perspective, the logical channel would be seamlessly maintained, some applications might experience interrupted communication, though.
- During our tests, we experimented with setting up a large NDN packet size (up to 30MB). This enabled us to transfer entire services on a single NDN packet, which was fragmented and reliably transferred by the underlying DTN layer when intermittent connectivity was present. To achieve this, we followed a multiple-interest forwarding mechanism instead of traditional NDN congestion control algorithms – which were not designed for disruptive networks. This option proved viable and allowed us to reliably transfer NDN packets to remote NDN nodes.
- Large packet size limitations: As mentioned we encountered these limitations on deploying this scheme on Raspberry Pi devices. More specifically, we reached the packet size upper limit, as there was limited memory due to hardware constraints. We discovered that one of the main limitations was the fact that the NDN Content Store (i.e., the ICN cache) memory allocation is configured based on the number of Data packets to be stored (which are assumed to be numerous by default), not on

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 645124

the total content size to be allocated. This issue was causing some issues with the NDN Forwarding Daemon when transferring KEBAPP services as well. We mitigated some of the problems by:

  - Significantly decreasing the Content Store size on the NFD configuration file - to a few packets.
  - Creating a swap-file and increasing the available memory on the Raspberry Pi device.
  - Encapsulating a full KEBAPP service in a Docker image less than 30MB and we learned how to create a full working server in just a few MB.

- Hotspot 2.0 enabled AP: We had some issues enable Hotspot 2.0 protocol in the Fonera devices, due to wireless interface driver's issues. Because of that, we had to find another solution to advertise the available services in the UMOBILE hotspot. This solution was generating an automatic SSID in the UMOBILE hotspot that, utilizing bloomfilters, users can figure out which service is enabled in the hotspot in that moment. This solution will help us deploying UMOBILE in any hardware, even if is not possible to enable Hotspot 2.0.
- Transparent connectivity: In order to make KEBAPP Services totally transparent to users, without requiring any user interaction, we managed to find a way to connect our smartphones to UMOBILE hotspot, retrieve any information and disconnect. The user is not even aware of this connection and will only receive a notification of the information received.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 645124

6. Conclusions

The main objectives of this deliverable are to provide the integration and validation for the complete integrated system showcased during the final demonstrations, and consolidated in different real-scenarios. We have discussed their integration, their results as well as the lessons learned during this stage. This validation in a real life scenario has also contributed to identify limits of our system proposing lessons learned.

For the PoC1, which showcases the real-life emergency scenario, two PoCs took place. A complete integrated version has been proved in Umbria (Italy), described in Section 2 in detail the integration and validation, involving emergency services and simulating a fire and a person in danger. A reduced set of PoC1 took place in Lisbon (Portugal) but adding UAVs to the demonstration. This has been presented in Section 4 with the complete integration and validation.

The PoC2, corresponding to the social routine scenario, has been showcased in Lisbon, Portugal. A complete description of the integration and validation is found in Section 3.

In Section 5 we have identified and analyzed of the lessons learned during this demonstration phase which have arisen. The issues can be classified in multiple fields: integration issues, development, productization and usability issues, as well as hardware and software limitations. Furthermore, we have proposed and implemented mechanisms to overcome most of the identified matters.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 645124

References

[1] UMOBILE Project, "D.5.2 – Validation methodology and evaluation report (2)", January 2017

[2] UMOBILE Project, "D.5.3 - Proof-of-Concept (1)", January 2017

[3] UMOBILE Project, "D.5.4 - Proof-of-Concept (2)", November 2017

[4] UMOBILE github repository: https://github.com/umobileproject

[5] Docker technology. https://www.docker.com/what-docker. Accessed: 2018-02-10.

[6] Hostapd:Hostaccesspointdaemon.https://wiki.gentoo.org/wiki/Hostapd. Accessed: 2018-02-10.

[7] Hypriot Docker Image for Raspberry Pi. https://blog.hypriot.com/downloads/. Accessed: 2018-02-10.

[8] CERDA ALABERN, L., NEUMANN, A., AND ESCRICH, P. Experimental evaluation of a wireless community mesh network. In *Proceedings of the 16th ACM International Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems* (New York, NY, USA, 2013), MSWiM '13, ACM, pp. 23–30.

[9] NEUMANN, A., LOPEZ, E., AND NAVARRO, L. An evaluation of bmx6 for community wireless networks. In *8th IEEE International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob), 2012 I* (Oct 2012), pp. 651–658.

[10] RAHMAN, A., TROSSEN, D., KUTSCHER, D., AND RAVINDRAN, R. Deployment Considerations for Information-Centric Networking (ICN) . Internet-Draft, Jan. 2018.

[11] DE SILVA, U., LERTSINSRUBTAVEE, A., SATHIASEELAN, A., MOLINA- JIMENEZ, C., AND KANCHANASUT, K. Implementation and evaluation of an information centric-based smart lighting controller. In *Proceedings of the 12th Asian Internet Engineering Conference* (2016), AINTEC '16.

[12] BAIG, R., FREITAG, F., AND NAVARRO, L. Cloudy in guifi.net: Establishing and sustaining a community cloud as open commons. *Future Generation Computer Systems* (2018).

[13] LIU, P., WILLIS, D., AND BANERJEE, S. Paradrop: Enabling lightweight multi-tenancy at the networks extreme edge. In *2016 IEEE/ACM Symposium on Edge Computing (SEC)* (Oct. 2016), vol. 00, pp. 1–13.

[14] AFANASYEV, A. NFD Developers Guide. Tech. rep., Feb. 2018

[15] PALIT,T.,SHEN,Y.,ANDFERDMAN,M.Demystifyingcloudbenchmark- ing. In *2016 IEEE International Symposium on Performance Analysis of Systems and Software (ISPASS)* (April 2016), pp. 122–132.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 645124

[16] Introducing a powerful open source social networking engine. https://elgg.org/. Accessed: 2018-02-10.

[17] UMOBILE Software Oi:

https://play.google.com/store/apps/details?id=com.copelabs.android.oi

[18] Paulo Mendes, Rute Sofia, Vassilis Tsaoussidis, Sotiris Diamantopoulos, Alexandros Sarros, "Information-centric Routing for Opportunistic Networking Environments", IETF Draft, ICNRG, draft-mendes-icnrg-dabber-00 (work in progress), Feb 2018.

[19] Sofia, Rute C.; Santos, Igor; Soares, José; Diamantopoulos, Sotiris; Sarros, Christos-Alexandro; Vardalis, Dimitris; Tsaoussidis, Vassilis; d'Angelo, Angela. "UMOBILE D4.5 – Report on Data Collection and Inference Models". Technical Report, September 2018.

[20] Miguel Tavares, Paulo Mendes, "NDN-Opp: Named-Data Networking in Opportunistic Networks", Technical Report COPE-SITI-TR-18-01, January 2018.

[21] Omar Aponte, Paulo Mendes, "Oi!: Short Messaging in Opportunistic Wireless Named-Data Networks", COPELABS Technical Report, COPE-TR-18-03, January 2018.

[22] Zhenkai Zhu and Alexander Afanasyev. 2013. Let's ChronoSync: Decentralized Dataset State Synchronization in Named Data Networking. In Proceedings of the 21st IEEE International Conference on Network Protocols (ICNP 2013). Goettingen, Germany.

[23] Omar Aponte, Paulo Mendes, "Now@ - Content Sharing Application over NDN", in ACM ICN, Berlin, Germany, September 2017.

[24] Omar Aponte, Paulo Mendes, "Now@: Data sharing in Opportunistic Wireless Named-Data Networks", COPELABS Technical Report, COPE-TR-18-02, January 2018

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 645124