Action full title:

Universal, mobile-centric and opportunistic communications architecture

Action acronym:

UMOBILE

Deliverable:

D2.4 “System and Network Deployability Design”

Project Information:

Project Full Title	Universal, mobile-centric and opportunistic communications architecture
Project Acronym	UMOBILE

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

Grant agreement number	645124
Call identifier	H2020-ICT-2014-1
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 Research and Innovation Center

Deliverable Information:

Deliverable Number-Title	D2.4 System and Network Deployability Design
WP Number	WP2
WP Leader	UCAM
Task Leader (s)	Tekever
Authors	**Tekever:** Tiago Ramalho, Ricardo Faria, Luis Deprez **FON:** Alberto Pineda, Pablo Salvador **AFA:** Francesco Amorosa, Gianmichele Russi **Senception:** Rute Sofia **Tecnalia:** Susana Pérez Sánchez **UCAM:** Adisorn Lertsinsrubtavee, Carlos Molina-Jiménez **Athena:** Sotiris Diamantopoulos, Christos-Alexandros Sarros **UCL:** Sergi Rene, Ioannis Psaras **Copelabs:** Paulo Mendes
Editors	Alberto Pineda Pablo Salvador
Contact	Alberto Pineda: alberto.pineda@fon.com

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

Due date	30/09/2017
Actual date of submission	17/07/2018

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	01/07/17	Initial template with Table of Contents
Version 0.2	16/07/17	Revised table of contents
Version 0.3	31/07/17	Added content to some sections
Version 0.4	10/09/17	Updated version after some feedback
Version 0.5	30/09/17	Finalized preliminary version of the deliverable, circulated for internal review
Version 1.0	06/10/17	Final version
Version 2.0	17/07/18	Revised Version based on Final Review 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

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

List of Figures

List of Tables

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

List of definitions

Term	Meaning
DTN	Delay Tolerant Network (DTN) is an emerging technology that supports interoperability of other networks by accommodating long disruptions and delays between and within those networks. DTN operates in a store-and-forward fashion where intermediate nodes can temporary keep the messages and opportunistically forward them to the next hop. This inherently deals with temporary disruptions and allows connecting nodes that would be disconnected in space at any point in time by exploiting time-space paths.
ICN	Information-Centric Network (ICN) has emerged as a promising solution for the future Internet's architecture that aims to provide better support for efficient information delivery. ICN approach uniquely identifies information by name at the network layer, deploys in-network caching architecture (store information at the network node) and supports multicast mechanisms. These key mechanisms facilitate the efficient and timely information (contents and services) delivery to the end-users.
Content	Content refers to a piece of digital information that is disseminated and consumed by the end-user equipment.
User	An entity (individual or collective) that is both a consumer and a relay of user services.
User Service	Context-aware services are considered as a set of mechanisms that assist incorporating information about the current surrounding of mobile users to provide more relevant of services.
User Interest	A parameter capable of providing a measure (cost) of the "attention" of a user towards a specific (piece of) information in a specific time instant. Particularly, users can cooperate and share their personal and individual interests that enable the social interactions and data sharing across multiple users.
User Requirement	User requirement corresponds to the specifications that users expect from the application.
Upstream	Upstream traffic refers to outgoing data such as short message, photo or uploading video clips that are sent from user equipment.

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

Downstream	Downstream traffic refers to data is obtained by use equipment from network. This includes downloading files, web page, receiving messages, etc.
Gateway	Gateway typically means an equipment installed at the edge of a network. It connects the local network to larger network or Internet. In addition, gateway also has a capability to store services and contents in its cache to subsequently provide local access communication.
Customer Premises	Customer Premises relate to residential households and enterprise market and are, as of today, controlled by the end-user.
User-centric	User-centric refers to a new paradigm leveraging user information at large to deliver novel content or services by users towards other users.
UMOBILE System	UMOBILE System refers to an open system that provides communication access to users through wired or wireless connectivity. This system exploits the benefit of local communication to minimize upstream and downstream traffic. The services or contents can be exchanged and stored in several devices such as gateways; user equipment; customer premises equipment such as WiFi Access Points to efficiently delivery the desired contents or services to end users.
UMOBILE Architecture	A mobile-centric service-oriented architecture that efficiently delivers contents and services to the end-users. The UMOBILE architecture integrates the principles of Delay Tolerant Networks (DTN) and Information-Centric Networks (ICN).
User-equipment	User-equipment (UE) corresponds to a generic user terminal (for example smart phone or notebook). In terms of UE and for operating systems we consider mainly smartphones equipped with Android; notebooks with UNIX, Windows, Mac OS.
Application	Computer software design to perform a single or several specific tasks, e.g. a calendar and map services. In UMOBILE, context-aware applications are considered.
Incentive	A factor (e.g., economic or sociological) that motivates an action or a preference for a specific choice.

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

Service	Service refers to a computational operation or application running on the network which can fulfil an end-user's request. The services can be hosted and computed in some specific nodes such servers or gateways. Specifically, services are normally provided for remuneration, at a distance, by electronic means and at the individual request of a recipient of services. For the purposes of this definition; *"at a distance"* means that the service is provided without the parties being simultaneously present; *"by electronic means"* means that the service is sent initially and received at its destination by means of electronic equipment for the processing (including digital compression) and storage of data, and entirely transmitted, conveyed and received by wire, by radio, by optical means or by other electromagnetic means; *"at the individual request of a recipient of services"* means that the service is provided through the transmission of data on individual request. Refer to D2.2 for further details.
UMOBILE gateway	Role (software functionality) which reflects an operational behaviour making a UMOBILE device capable of acting as a mediator between UMOBILE systems and non-UMOBILE systems – the outside world.

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

List of Acronyms

A_NPA	Advance Notice of Protocol Amendment
ADT	Android Development Tools
AP	Access Point
ATM	Air Traffic Management
CM	Contextual Manager
CP	Customer Premises
DTN	Delay-Tolerant Networking
EASA	European Aviation Safety Agency
EUS	End-User Service
FIB	Forwarding Information Base
HD	High Definition
HS	Hotspot
ICN	Information-Centric Networking
INRPP	In-Network Resource Pooling Protocol
ISP	Internet Service Provider
ISR	Intelligence, Surveillance, and Reconnaissance
KEBAPP	Keyword-based Application Sharing Framework
L2TP	Layer 2 Tunneling Protocol
NAA	National Aviation Authority
NDN	Named-Data Networking
NFD	NDN Forwarding Daemon
NLSR	Named Data Link State Routing
NREP	Name-based Replication Priorities Protocol

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

ONE	Opportunistic Networking Environment
OOCD	Opportunistic Off-path Content Discovery
OS	Operating System
OSPF	Open Shortest Path First Routing Protocol
PML	PerSense Mobile Light
POC	Proof-of-Concept
QoS	Quality of Service
RIB	Routing Information Base
S	Service
SEG	Service Execution Gateway
UAV	Unmanned Aerial Vehicle
UE	User Equipment
UNC	UMOBILE Network Controller
VPN	Virtual Private Network
VTOL	Vertical Take-off Landing

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

Executive Summary

This document corresponds to the output of Task 2.3 and covers the full specification for the system, including refined assumptions, requirements, as well as design choices. It shall include technology to be applied from previous/related work, as well as clearly indicate the contributions to be provided by the UMOBILE system.

The purpose of this document is to ensure that the system and network specification [1] as well as the end-user requirements [2] can meet operational expectations within the UMOBILE time-frame. Focus has been given in analysing how the proposed concepts can be easily deployable based on current and future technology.

Backward compatibility with current IP infrastructures is a central aspect to the deployment of UMOBILE.

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

1 Introduction

This deliverable describes the UMOBILE system deployability design, covering the following aspects: an end-to-end Internet perspective for UMOBILE deployment; main operating systems as well as main devices in the different Internet regions; tools and experimentation procedural guidelines, as well as guidelines to further detail the deployment of the UMOBILE system specifications [1]. This document shall steer the work to be developed in WP3 and WP4, in terms of assumptions and requirements, thus assisting in reaching an optimal UMOBILE proof-of-concept.

The deliverable has been split into four main parts. On the first part the deliverable maps the UMOBILE node architecture and system requirements into an Internet end-to-end perspective, having as underlying design the operational multi-access heterogeneous networks of today. On the second part the deliverable provides input and guidelines concerning operating systems and devices expected to be used by partners, to ensure a smooth deployment for each service described in UMOBILE. The third part is focused in experimental tools used in UMOBILE. Finally, the fourth part provides a guideline to deploy UAVs as a tool that completes UMOBILE functionalities.

The document is organized as follows. Section 2 describes how the UMOBILE architectural design fits, from an operational perspective, the Internet end-to-end architecture. Section 3 describes the main UMOBILE networking services [1] and user services [2], from an operational perspective, to assist in the understanding of the deployment restrictions that may arise and that need to be addressed in WP5. Section 4 covers observations and guidelines concerning experimental tools to be used in UMOBILE, such as the type of simulators or emulators; different versions; programming language(s) of choice; IDE to be used by all partners. This integration shall assist in a smoother deployability in WP5. Section 5 concludes the document by providing guidelines for UAVs deployability. Finally, Section 6 shows the conclusions of the document.

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

2 The UMOBILE Elements from an End-to-End Perspective

This section goes over networking terminology that we consider essential to assist in the synchronization of goals and of results in UMOBILE. The UMOBILE communities are expected to be integrated into Internet fringes. Still, from an operational perspective there is the need to consider an internet end-to-end perspective as illustrated in Figure 1. As explained in [3], the UMOBILE platform follows a modular software architecture. It includes the following main components:

- **UMOBILE GATEWAY (1):** an element that provides interconnectivity between the UMOBILE domain and the Internet domain.
- **UMOBILE SERVICE MANAGER (2)**: a piece of software that is responsible for making informative decisions on the deployment of services. It is launched by Service Provider (a human being) when he or she wishes to deploy services with QoS awareness. It can be run in an ordinary computer with computation, storage and communication facilities.
- **UMOBILE HOTSPOT (4):** computers with computation, storage and wireless communication facilities that makes them able to i) collect information (e.g., alert messages, instructions from emergency authorities) from users and transmit it to the UMOBILE network and ii) host instantiated services and iii) store collected data, check its validity and perform computational functions (e.g. data fusion) to increase the value of the information delivered to the civil authorities.
- **UMOBILE End-User Service (3):** the entry point for the UMOBILE solution, which allows a user to seamlessly profit from UMOBILE services and native apps, without having to independently install specific software pieces. The EUS gives access to native applications as well as to the automatic activation of UMOBILE services required to run specific applications and services, as explained next.

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

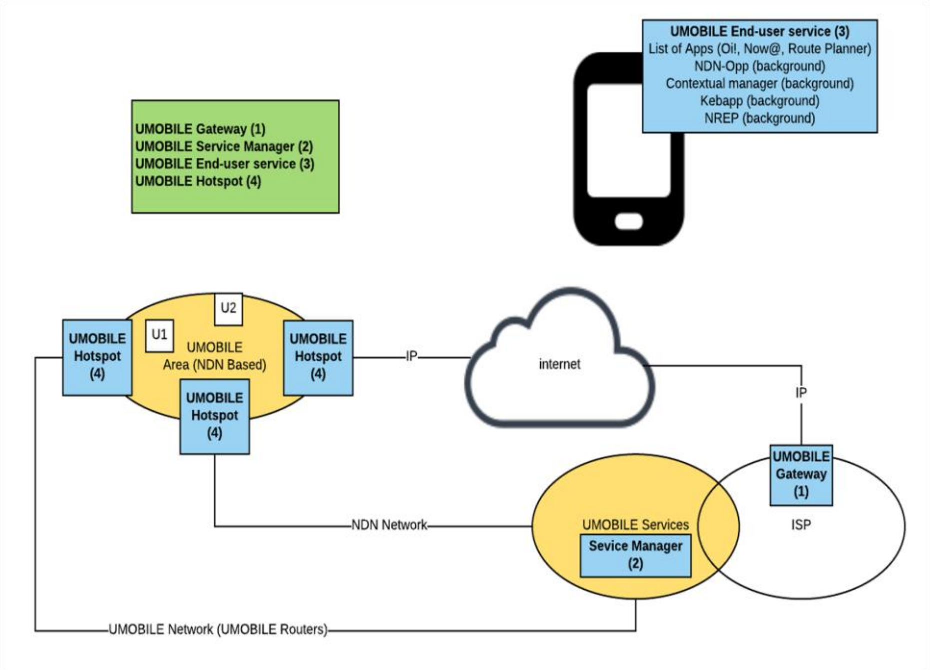

Figure 1: UMOBILE elements from an end-to-end perspective

UMOBILE elements are therefore developed to reside in multiple locations, being most of networking services placed closer to the user - in the Customer Premises (CP), which are physically associated with residential households, public spaces, and enterprise market and are, as of today, controlled by the end-user.

The **UMOBILE gateway (1)** is an element that performs translation between IP and NDN/ICN. From an end-to-end perspective and despite the fact it can be placed anywhere, it is associated to Service Providers (IP backbone). Central to the issue of backward compatibility with the Internet is the UMOBILE gateway, it deserves further discussion. As shown in the figure, the UMOBILE gateway provides connectivity between the UMOBILE domain and the Internet domain. In the UMOBILE platform, we use a conventional computer with computation, storage and communication facilities to implement the UMOBILE gateway. The reason for the storage and computational resources is that such a computer can be used by service and content providers as a repository of images of services (for example, Docker images). The dual connection of the UMOBILE gateway and the computation and storage resources enable it to retrieve data and service images, store them locally and upon request, make them available to the UMOBILE domain. Though there are no technical difficulties in making the UMOBILE gateway to move data in the opposite direction, we focus on service and content sharing over the UMOBILE network, presumably retrieved from the ISP network. UMOBILE Gateways can be part of the infrastructure of service/content providers or civil authorities. Providers can regard and use the UMOBILE gateway as an "entry point" to the UMOBILE network, that is, as a means of connecting their legacy IP-based infrastructure with UMOBILE nodes belonging to the UMOBILE domain. This entry point is normally physically deployed within the provider's premises. However, depending on the deployment model any UMOBILE hotspot can act as gateway, if it is equipped with an IP interface and has sufficient storage and processing facilities. The gateway provides universal access to content or services

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

located in both the host-centric and information-centric domain. For example, a service in the IP domain can be "fetched" by the UMOBILE gateway – with the help of the Service Manager – and stored in the UMOBILE part of the network. In this manner, the service is available to the UMOBILE users from the gateway repository. The users can choose to download the service and deploy it, as they like. The UMOBILE gateway is normally physically deployed within the provider's premises. However, depending on the deployment model any UMOBILE hotspot can act as gateway, if it is equipped with an IP interface and has sufficient storage and processing facilities. As a UMOBILE node itself, the UMOBILE gateway is deployed with the full UMOBILE platform which allows it to forward data over the DTN interface if required. In summary, the gateway provides a common pool of shared services and information between the two domains, it other words, it provides backward compatibility with the ISP Internet.

The **UMOBILE Service Manager (2)** is being developed to integrate a Service Provider backbone, as it assists the support of current services in an NDN/ICN environment.

The **UMOBILE End-User Service (EUS, 3)** resides in the Customer Premises, currently being devised to be deployed in end-user devices (*User Equipment, UE*). While the fourth UMOBILE element, the **UMOBILE hotspot (4)** is placed in an Access Point and therefore located in the CP.

3 UMOBILE User and Networking Services

3.1 UMOBILE User Services

The UMOBILE End-User Service is a software element that is being devised to assist in a smoother deployment of UMOBILE. It provides the end-user with a unique entry point for UMOBILE native applications, as well as to call automatically UMOBILE services that run in

Figure 2: End-User Service frontend

background. The bootstrapping is as follows. A person in a challenged environment (such as the environments described in [2]) walks around and its personal device automatically connects to an existing UMOBILE hotspot, holding a captive portal mechanism which provides the user with access to download the UMOBILE EUS[1]. UMOBILE services can also be used in

[1] https://gitlab.com/UMOBILESenception/UES

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

isolation. The UMOBILE EUS simply serves the purpose of integration and user-friendliness. The EUS frontend is depicted in Figure 2 and gives access to both UMOBILE native applications as well as to UMOBILE networking services. The UMOBILE networking services run in background and do not need to be activated by the user (even though the user is also allowed to do so). In fact, they become active depending on the service or application that is selected in UMOBILE. For instance, if the user selects an Instant Messaging application such as Oi! which requires direct communication, the UMOBILE routing module will be activated. Currently, the EUS gives access to the following native and independent apps:

- Oi! instant messenger for NDN/ICN opportunistic environments [4] and [5].
- Now@, data sharing application for NDN/ICN [4] and [5].
- PerSense Mobile Light (PML), tool for capturing wireless roaming data [4], [5] and [2].
- RoutePlanner, route calculator app that requests route to users nearby when no Internet connectivity

These tools require the support of UMOBILE User services, which are:

- Contextualization: Contextual manager (network and local usage data)
- UMOBILE Device-to-device communication: NDN-Opp (opportunistic communication)
- Application sharing service: KEBAPP
- Service migration

The native application and services requirements and guidelines in terms of deployability are presented in the next sections.

3.1.1 UMOBILE Native Applications Deployability Guidelines

3.1.1.1 Oi!

Oi! is an instant messenger for NDN/ICN opportunistic environments. In order to be run, Oi! requires the activation of the direct communication (supported by NDN-Opp). Table 1 summarizes guidelines to deploy Oi!.

Table 1: Oi! deployability guideline

Hardware Requirements	• End-user device
Software/hardware Requirements (OS, dependencies)	• OS Independent (java) • Requires NDN-Opp (extension of the NDN forwarding daemon for opportunistic environments)
Availability of Open-source code	• https://gitlab.com/umobile_copelabs/Oi_NDN
Availability of	• https://play.google.com/store/apps/details?id=com.copelabs.android.oi

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

binaries	

Environments it has been validated on (hardware + software)	• Hardware
	  ○ Samsung Galaxy
	• Software
	  ○ Android

Modules for simulators	• No

3.1.1.2 Now@

Now@ is an opportunistic data sharing tool. It relies on the user's interests to share data. It allows users to share data directly, as well as across Named-data Networking (NDN) environments. The data sharing relies on ChronoSync. In NDN, Now@ requires NDN-Opp: an NFD extension for opportunistic environments. Table 2 summarizes guidelines to deploy Now@.

Table 2: Now@ deployability guideline

Hardware Requirements	• End-user device
Software/hardware Requirements (OS, dependencies)	• OS Independent (java) • Requires ChronoSync • Requires NDN Android OR NDN-Opp (extension of the NDN forwarding daemon for opportunistic environments)
Availability of Open-source code	• https://gitlab.com/umobile_copelabs/NowAt
Availability of binaries	• https://play.google.com/store/apps/details?id=pt.ulusofona.copelabs.now
Environments it has been validated on (hardware + software)	• Hardware: Samsung Galaxy • Software: Android
Modules for simulators	• No

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

3.1.1.3 PerSense Mobile Light (PML)

PML is a light version tool of the PerSense ™ product line of Senception. This product line is a personal platform for interaction and communication with two main features: i) stimulation of interaction via learning and inference of daily routine context in an opportunistic way via wireless and mobile networks ("How was your day?") and notifications to circles about such context ("let's share!"); ii) secure communication anywhere, anytime (instant messaging, video calls).

This light version aims solely at assisting researchers in capturing the natural networking footprint left around by devices, to assist in inference of roaming habits, and thus to assist in the network operation. Released in the project in May 2015 under LGPLv3.0, PML captures information concerning a user's affinity network (contacts derived from WiFi Direct and Bluetooth) as well as concerning roaming habits, over time and space (WiFi).

The tool has been developed to assist the research community in gathering meaningful traces and develop scientific studies, by reusing the collected traces. Freely available for research purposes, PML can be extended upon request, and is one of the tools that shall be available via the UMOBILE Lab. Table 3 summarizes guidelines to deploy PML.

Table 3: PML deployability guideline

Hardware Requirements	• End-user device • Can be extended to integrate APs
Software Requirements (OS, dependencies)	• Requires access to wireless and Bluetooth • Location service enabled (any mode)
Availability of Open-source code	• No
Availability of binaries	• https://play.google.com/store/apps/details?id=com.senception.persenselight
Environments it has been validated on (hardware + software)	• **Hardware:** Samsung S5, S6, S7, Huawei mobile phones, Samsung tablets, Other mobile phones • **Software:** Android 4.2 to 7
Modules for simulators	• No

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

3.1.1.4 Route Planner

Route-Planner is a KEBAPP-enabled application which allow users request route calculations to other users nearby that are using the same application without connecting to the Internet or without having the maps locally. Table 4 summarizes guidelines to deploy Route Planner.

Table 4: Route Planner deployability guideline

Hardware Requirements	• Android devices, embedded devices
Software Requirements (OS, dependencies)	• Android 5 to 7 (Android 8 not tested yet) • WiFi-Direct enabled device • Google Play Services (version 11.5.09) • Location service enabled mode high accuracy • Internet connectivity in the route-provider user • Raspberry Pi with Raspbian kernel version 4.9 • Java 7
Availability of Open-source code	• https://github.com/umobileproject/KEBAPP_routefinder
Availability of binaries	• No
Environments it has been validated on (hardware + software)	• Android 6 • Raspberry Pi with Raspbian kernel version 4.9
Modules for simulators	• No

3.1.2 UMOBILE User Services' Deployability Guidelines

3.1.2.1 Direct device to device communication (NDN-Opp)

The NDN framework for Opportunistic Networks (NDN-Opp) is being developed aiming to support opportunistic data forwarding based on users' interests and their dynamic social behaviour. The NDN-Opp framework includes some changes in relation to NDN Forward Daemon (NFD) to enable social-based information-centric routing over dynamic wireless

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

networks. The first implementation makes use of WiFi direct to exploit wireless transmission opportunities, and holds the following NFD changes:

- **Forwarding Daemon.** Performs the normal forwarding decisions of NDN packets based on their names (UCLA implementation).
- **Connection Manager.** Maintains low-level communication channels used by the Opportunistic Faces to effectively transmit packets between UMOBILE devices.
- **Router.** Fulfils two functions within NDN-Opp:
  - **Face management:** As UMOBILE nodes appear and disappear in the vicinity, the routing oversees maintaining the state of the corresponding Opportunistic Faces installed in the Forwarding Daemon.
  - **Route computation:** To improve the forwarding paths used when transmitting packets, the Router runs an algorithm which recomputes the contents of its RIB and FIB based on a variety of indicators and weights provided by the Contextual Manager [2].
- **Opportunistic Face.** Implements the queuing and de-queueing of packets based on the availability of the next-hop.

Table 5 summarizes guidelines to deploy NDN-Opp.

Table 5: NDN-Opp deployability guideline

Hardware Requirements	• End-user devices
Software Requirements (OS, dependencies)	• Android • NDN Forwarding Daemon
Availability of Open-source code	• https://gitlab.com/umobile_copelabs/ndn-opp • https://gitlab.com/umobile_copelabs/ndn-fwd
Availability of binaries	• https://play.google.com/store/apps/details?id=pt.ulusofona.copelabs.ndn
Environments it has been validated on (hardware +	• Hardware ○ Samsung Galaxy

[2] Novel routing mechanisms for NDN opportunistic environments are currently under development in WP5 and shall be integrated into future version of NDN-Opp.

software)	• Software ○ Android
Modules for simulators	• No

3.1.2.2 Contextualization (Contextual Manager)

The UMOBILE Contextual Manager [6] is a service that runs in background on end-user devices and that can be easily adapted to access points. Its ultimate purpose is to provide other UMOBILE modules with contextual awareness derived from: i) internal device usage; ii) external applications; iii) available network sensors. As described in deliverable [4], [5] as well as [7] and [3], the CM performs contextualization derived from data that is either directly captured via multiple sensors (currently, Bluetooth and WiFi interfaces) as well as via external sensing applications, such as PML. Table 6 summarizes guidelines to deploy the Contextual Manager.

Table 6: Contextual Manager deployability guideline

Hardware Requirements	• End-user devices, APs
Software Requirements (OS, dependencies)	• Being developed in Java for Android - can be easily ported • Requires access to networking interfaces • Location service enabled (any mode)
Availability of Open-source code	• https://gitlab.com/UMOBILESenception/ContextualManager
Availability of binaries	• https://play.google.com/store/apps/details?id=com.senception.contextualmanager
Environments it has been validated on (hardware + software)	• Android 6[3]
Modules for simulators	• No

[3] The CM is still under implementation.

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

3.1.2.3 Kebapp

KEBAPP enables users to share their own applications with nearby users. In a sense, the client application instance can also act as a server instance to serve requests from nearby users. **KEBAPP** explicitly enables access to the desired processed and non-personalised information through the concept of application sharing, effectively leveraging on a pool of application resources. Table 7 summarizes guidelines to deploy KEBAPP.

Table 7: KEBAPP deployability guideline

Hardware Requirements	• Android devices, embedded devices
Software Requirements (OS, dependencies)	• Android 5 to 7 (Android 8 not tested yet) • WiFi-Direct enabled device • Raspberry Pi with Raspbian kernel version 4.9 • Java 7
Availability of Open-source code	• https://github.com/umobileproject/KEBAPP_routefinder
Availability of binaries	• No
Environments it has been validated on (hardware + software)	• Android 6 • Raspberry Pi with Raspbian kernel version 4.9
Modules for simulators	• No

3.1.2.4 NREP

NREP service is a name-based replication priority mechanism, which considers prioritization rules to spread emergency information, along with indicators and contextual measures provided by the CM to assist in a more efficient data dissemination. Table 8 summarizes guidelines to deploy NREP.

Table 8: NREP deployability guideline

Hardware	• Android devices

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

Requirements	
Software Requirements (OS, dependencies)	• Android 5 to 7 (Android 8 not tested yet) • WiFi-Direct enabled device • Google Play Services (version 11.5.09) • Location service enabled mode high accuracy
Availability of Open-source code	• No
Availability of binaries	• No
Environments it has been validated on (hardware + software)	• No
Modules for simulators	• ONE simulator module

3.2 Networking Services

Network services described in this section correspond to the set of network functionality that a UMOBILE architecture must support in order to provide the applications described in the previous section.

3.2.1 NDN-DTN Integration

A central aim of the UMOBILE project is to enhance the original NDN implementation with mechanisms that enable it to operate over communications infrastructures with different levels of QoS, including networks with low throughput and networks that suffer from intermittent disconnections. These impairments of the underlying communication infrastructure should be concealed from the end-user's concerns. In pursuit of this aim, we have enhanced the original NDN architecture with mechanisms that allow it to meet a range of QoS requirements including less than best effort, best effort and premium. These mechanisms are thoroughly discussed in [8]. In this section we will provide only a summary the mechanism that provides disruption-tolerance. The mechanisms are based on delay-tolerance with opportunistic communication support. Figure 3 shows the main components of the solution. The modules that we have developed are shown in blue.

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

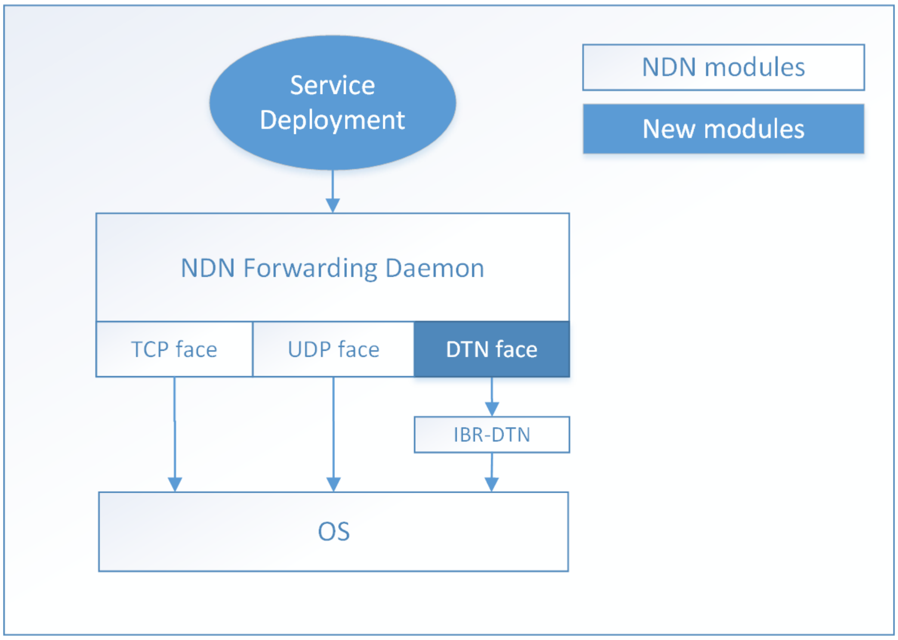

Figure 3: Architecture of the ICN-DTN integration

As shown in Figure 3, we have enhanced the Forwarding plane of the original NDN with a Delay Tolerant Network (DTN) face that is responsible for data ferrying and routing when the underlying communication infrastructure exhibits signs of disruption. This new face communicates with an underlying DTN implementation that handles intermittence by encapsulating Interest and Data packets into DTN bundles. Store-and-forward techniques are used by the underlying DTN implementation to deliver the bundles reliably with the help of standard DTN routing protocols. Upon reaching their destination, the bundles are decapsulated and the original Interest/Data messages are forwarded, resuming standard NDN communications.

We can take advantage of the DTN-face in different situations. For instance, in a demo presented at ICN'17 [9], we demonstrate how the DTN-face can be used in combination with caching and name-based routing to deploy and instantiate a service in a subnet isolated from the main network due to network partition. Such a service is instantiated from a Dockerised image that is retrieved from the main network and transported through DTN-enabled nodes.

3.2.2 NDN/ICN Routing Support for Opportunistic Environments

The adaptive and intelligent forwarding plane of NDN-/ICN implies that the routing plane only needs to disseminate **long-term** topological and policy changes. Due to this flexibility, the first approach to routing in NDN was focused on fixed networks and hence, the first attempt was derived from IP routing link-state approaches. In more dynamic wireless networking scenarios, such as ad-hoc networks, the lack of an adaptive forwarding plan placed a big burden on the routing process, which is responsible for distributing computation of best paths among any pair of nodes to react to short-term variations in topology derived from node mobility.

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

The NDN adaptive forwarding plane allows the development of routing protocols with more relaxed requirements on timely detection of failures and convergence delay. Consequently, such routing schemes may support the extension of NDN operation to more dynamic networking scenarios, such as the ones created by opportunistic wireless networks. On the other hand, there is a similarity between the operation of opportunistic networks and the initial deployment of NDN: both fully rely on an adaptive forwarding plane, which in the case of opportunistic networks is mostly based on the usage of social proximity metrics. However, to the best of our knowledge, there is no proposal to extend NDN operation beyond the fix network. This is currently under development in UMOBILE, by first extending the current link-state approach (NLSR) with a routing interface ranking scheme based on context similarity

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

4 UMOBILE Experimental Tools and Guidelines

This section provides an overview on the tools that the UMOBILE consortium is expected to rely upon. The section concludes with a set of guidelines for partners to take, which have as purpose to ensure an adequate integration of results derived from the different tasks carried out in UMOBILE.

4.1 Public repositories

The UMOBILE project relies on the GitHub [10] public repository for both: i) sharing code in progress among members of the UMOBILE consortium and 2) exposing implementation results to the general public under a free software licence. For instance, the source code of the latest version of the Service Migration Platform is freely available from [11]. We thoroughly documented the code on the hope that this will encourage other researchers to download (fork or clone the project repository) it, replicate our experiments and double-check our results. We believe that this is an effective approach to disseminate and generate social impact among the research and industry communities.

4.2 Simulators

This section presents the simulators that are relied upon in UMOBILE, during an early stage of concept validation.

4.2.1 Observations and Guidelines

The simulation tools in UMOBILE serve for testing initial concepts and ideas by different partners, being done in cooperation or independently. Furthermore, the simulation code will be made available at specific stages of the project lifetime to the global R&D community. To ensure an adequate cooperation, the following guidelines are to be followed in UMOBILE:

- All modules are to be written in Java and C++.
- To ensure results' liability, all of the simulation environments should generate results with at least a 95% Confidence Interval. For the simulators (such as ns3) that do not integrate an automatic generation of statistically sound results we advise the application of Akaroa v2 [12].
- The ONE simulator [13] version to consider is 1.6.0.
- The ndnSIM [14] version to consider is version 2.3.

All of the modules in UMOBILE are to be provided to the consortium, along with the respective source code, well documented, and a small document describing:

- Basic specification and parameters that may be required to run simulations are included in each of the UMOBILE github repositories in a README file.
- Simulation environment, including details such as simulation duration, traces, etc. are also included in the corresponding github repository.
- Evaluation parameters considered and scenarios are also specified in the corresponding README file to be able to reproduce the results.

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

4.2.2 Simulators

Following we present the simulators used within UMOBILE:

- ONE simulator: The Opportunistic Networking Environment (ONE) simulator [13] is a Java-based tool for opportunistic networks, offering a broad set of DTN protocol simulation capabilities in a single framework. It was designed targeting simulation and analysis of DTN routing and application protocols. The ONE simulator offers an extensible simulation framework itself supporting mobility and event generation, message exchange, DTN routing and application protocols, a basic notion of energy consumption, visualization and analysis, interfaces for importing and exporting mobility traces, events, and entire messages. In order to evaluate KEBAPP and NREP with mobility traces and patterns that can not be evaluated in the UMOBILE lab, we use ONE for its simplicity, flexibility and mobility traces provided. In this case, we do not require a full stack network simulator, like ns3, to evaluate NDN networks, but a simple simulator to evaluate our proposals with different traces for opportunistic networks. The ONE simulator is designed in a modular fashion, enabling extensions of virtually all functions to be implemented using well-defined interfaces, which allowed us to extend it to develop NREP and KEBAPP approaches.
- *ndnSIM* [14]: it is an extension to the ns3 simulator, implementing the basic components of the Named Data Networking architecture in a modular way. ns3 is a discrete event network simulator primarily used for research and education, one of the most extensively used in the research community to test new network protocols. The design of ndnSIM follows the philosophy of network simulations in ns3, which devises maximum abstraction for all modelled components. This way, diverse and flexible experimentation over NDN is achieved allowing for different deployment needs which would be hard to test on current testbeds. In our case, we used ndnSIM to validate and evaluate UMOBILE mechanisms that require scalability analysis and cannot be evaluated using the UMOBILE lab. These mechanisms are the Opportunistic Off-path Content Discovery (OOCD) and the In-Network Resource Pooling Protocol (INRPP). In ndnSIM, all NDN forwarding and management is implemented directly using source code of Named Data Networking Forwarding Daemon (NFD), making it straightforward to reuse the simulation code in the lab and prototype implementation.

4.3 Experimental Tools & Sites

In this section we present the set of experimental tools and sites used within UMOBILE.

4.3.1 Apache Benchmarking tool

This tool is used by UCAM to assess the workload capabilities of the Raspberry Pi computers used in the UMOBILE platform to realise the Hotspots. The interest is to learn how many Docker containers hosting web servers can be instantiated in a Pi computer before it shows signs of exhaustion; and how many concurrent http requests can a web server handle before exhibiting exhaustion (unacceptably long response time). To answer these questions in a laboratory setting, we used the ab tool to artificially generate and place concurrent http

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

requests against web services running as Docker containers, in the following manner. Linux shells are created in a generic computer and the ab tool configured to simulate different numbers of concurrent users is executed. Details about these experiments that show how to use the ab tool for this purpose are discussed in D4.4 [8].

4.3.2 UMOBILE Lab

To test the software that we have developed, we have built a test bed lab physically located within AFA's premises in Termoli (Italy), namely UMOBILE Lab or just the Lab. It has been in operation since May 11, 2016 and it is continuously evolving. We have updated and refined it several times in accordance with the findings and needs of the project. For example, we have added a UMOBILE Network Controller (UNC1) Server to implement Service Controller operated through the Service Execution Gateway (SEG) devices. Currently, it includes many devices such as WiFi access points, Linux systems (Raspberry PI) and Android systems (Banana PI). As we will explain at large in a subsequent section, remote access to the Lab can be gained through VPN connections.

The Lab is meant to be used remotely by project partners and, upon request approved by the project PI[4], by other members of the general public such as researchers, sponsors, reviewers and industries. As such, we will keep it available beyond the completion of the project for as long as there is interest in the services that it provides or we run out of resources to maintain it.

The Lab has been discussed thoroughly in D5.3 [3]. In this section we provide only a summary to motivate and frame the discussion. A high-level view of the architecture is shown in Figure 4. It shows a UMOBILE gateway, some UMOBILE hotspots (access points), and some UMOBILE users.

[4] vtsaousi@ee.duth.gr

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

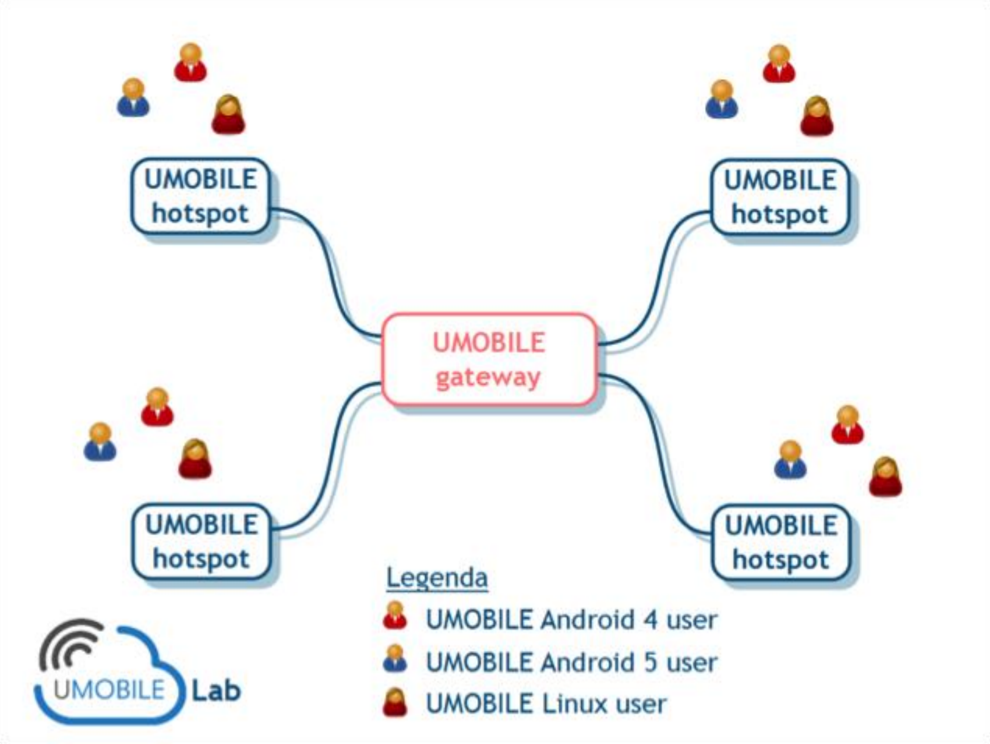

Figure 4: High level view of UMOBILE lab.

Figure 5 expands Figure 4 and shows the communication and computation components of the UMOBILE Lab.

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

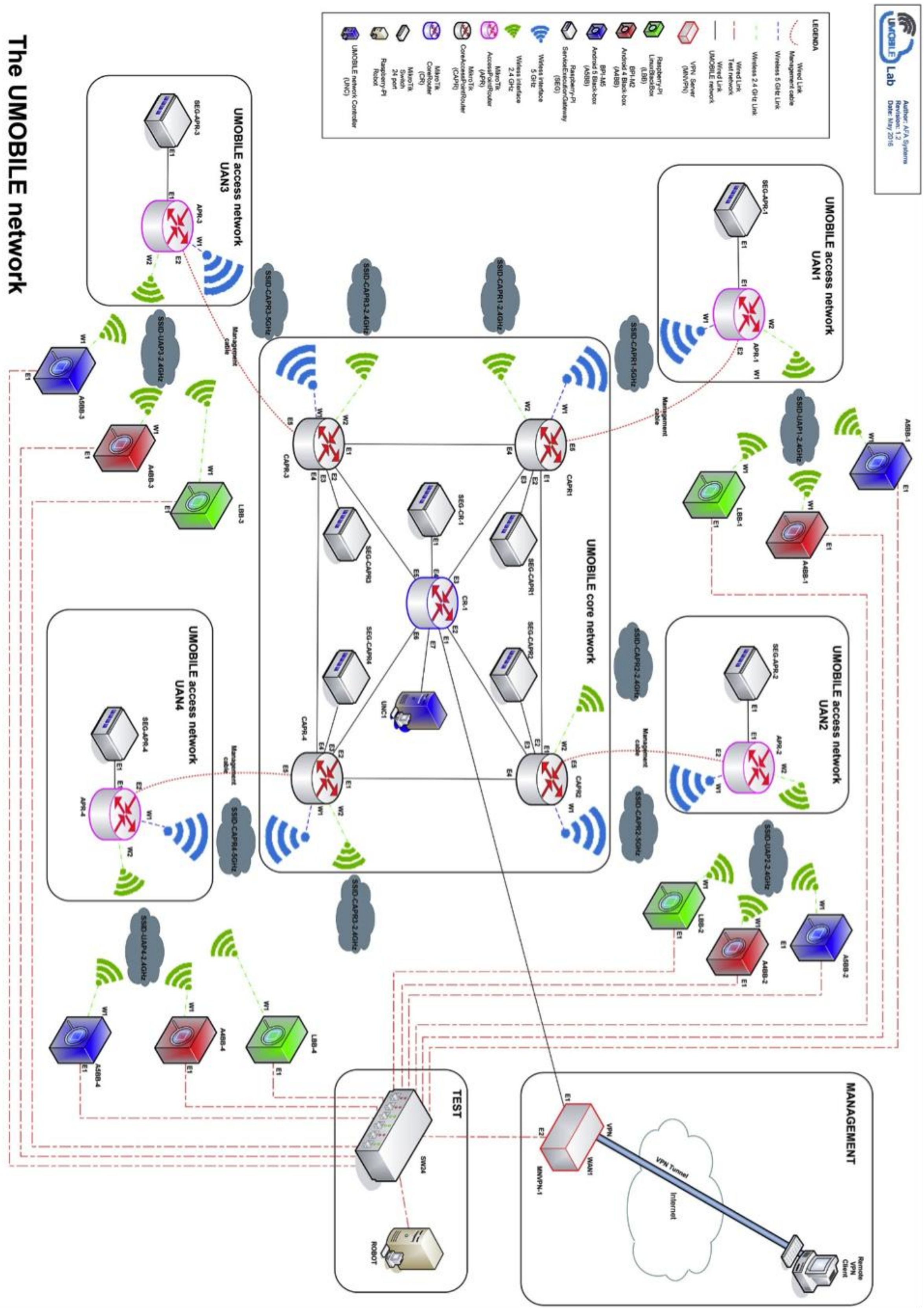

Figure 5: Detailed architecture of the UMOBILE Lab.

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

4.3.2.1 User guide

The UMOBILE Lab is composed by two networks and permits different kinds of tests. We have:

- UMOBILE network, IP routed by the OSPF protocol;
- TEST network, for devices handling (over Ethernet), in order to guarantee accessibility during the change of the UMOBILE Lab wireless network.

To gain remote access to the UMOBILE Lab, go through the following procedure:

1) Open an L2TP/IPSec connection with the following parameters:

Server: **lab1.umobile-project.eu**

VPN Type: **L2TP/IPSec (Layer 2 Tunneling Protocol with IPsec)**

Advanced properties use preshared Key: **password**

Authentication Protocol allow only: **PAP**

2) Once logged into the UMOBILE Lab, you have a list of different devices (shown in Table 9) at your disposition that can be accessed either through ssh or HTTP.

Table 9: UMOBILE Lab devices

Nr.	IP	Hostname	OS	zone	ssh access	http access
1	10.1.1.2	CAPR-1	OpenWRT CHAOS CALMER (15.05.1, r48532) – AP	core	ssh://capr-1.umobilelab.local /	http://capr-1.umobilelab.local
2	10.1.1.3	CAPR-2	OpenWRT CHAOS CALMER (15.05.1, r48532) – AP	core	ssh://capr-2.umobilelab.local	http://capr-2.umobilelab.local

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

3	10.1.1.4	CAPR-3	OpenWRT CHAOS CALMER (15.05.1, r48532) – AP	core	ssh://capr-3.umobilelab.local	http://capr-3.umobilelab.local

In addition to devices, you have at your disposition:

- UMOBILE website and UMOBILE project-wiki,
- Internet access through a captive portal (see Figure 6), using the same username/password as L2TP,
- The Drive: a common document repository.

Figure 6: UMOBILE testbed: captive portal.

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

3) Perform your test. For example, let us see how to install an App in an Android device. Observe that A5BB-x devices use Android 5.1 and, having their IP addresses; one can access them with the ADB (Android Debug Bridge) over TCP/IP.

1. Download the App of interest from the Internet, that is, its apk file, for example, **myapp.apk**
2. Use "**adb connect ipaddress:5555**"
3. Use "**adb install myapp.apk**".
4. Use "**adb shell**" to connect to the device and use the application that you have installed.

So far, we have used the UMOBILE Lab to successfully conduct the following basic tests:

1. NDN communication between the nodes of the lab connected via cable through WiFi.
2. Test automation through the command robot.
3. Migration of services between SEG nodes with services implemented as docker containers and presented as dockerised images.

In addition, we are positive that the UMOBILE Lab can serve as a test bed for conducting more ambitious experiments such as:

1. POC1: "Emergency and Civil scenario"
2. POC2: "Service Announcement and Social-Routine scenario"

4.3.2.2 Service Migration in Challenged Networks

As an example of the lab's value for testing future deployments, we have used it for testing the remote service migration mechanism that we have implemented in the UMOBILE projects. Internally, we have used the UMOBILE Lab to prepare a demo [9] presented at ACM ICN 2017 conference. In this demo, we present a NDN-based approach to deploy dockerised services closer to end-users when the network is impaired. In this manner, we further increased resiliency, employing DTN to tunnel traffic between intermittently connected NDN nodes.

The demo scenario is depicted in Figure 7. 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.

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

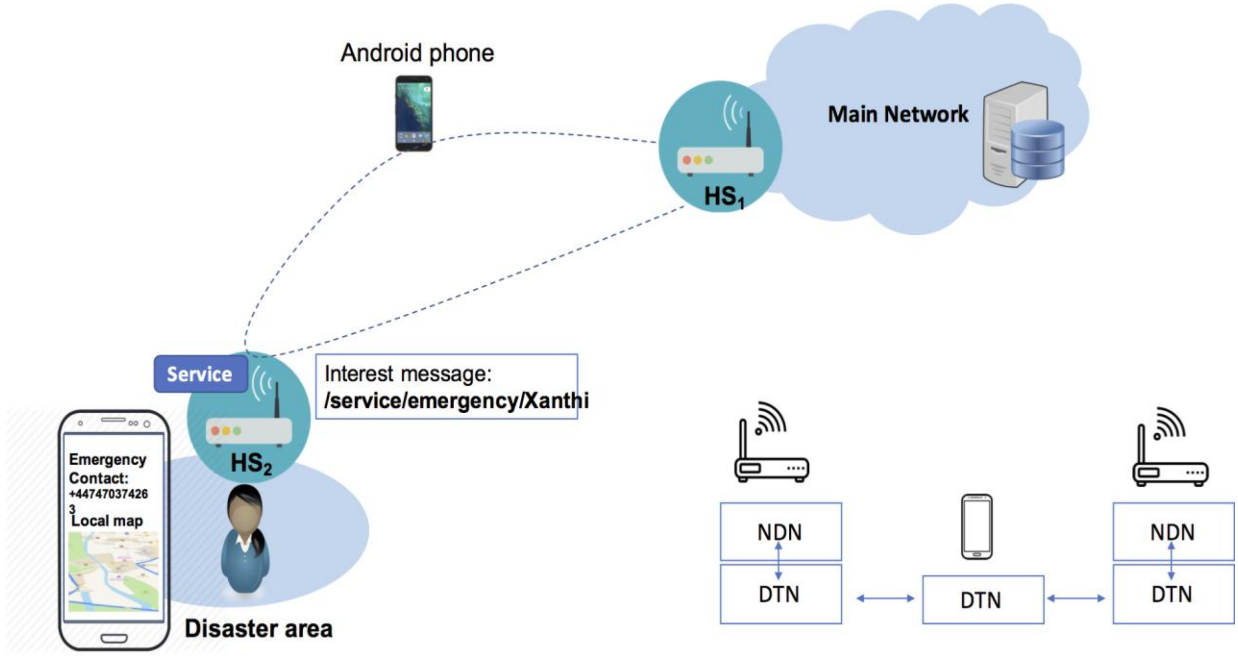

Figure 7: Demonstrating service migration scenario in challenging environments.

We setup the SEG-CAPR-1 and SEG-CAPR-2 in the UMOBILE Lab as the WiFi hotspots HS1 and HS2 respectively. Both hotspots are equipped with the UMOBILE protocol stack. The image of the service is available from [11]. Android phone is a physically mobile DTN-enabled node that can travel backwards and forwards between HS2 and HS1. In the UMOBILE lab, we setup the A4BB-1, a handheld device emulator as an Android phone (see Figure 7). In the final demonstration, we are planning to use a smart device attached to a UAV in place of the Android phone. Other means of transport can replace the Android phone, as long as they support 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 selected 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 HS1 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. In this experiment, we access (ssh) to the A4BB-1 node and manually connect to the hotspot as following sequence: HS2 - HS1 - HS2. When HS2 receives all the chunks of S, it calls a service execution function [25] to execute S to provide the emergency service.

4.4 Programming Interface Environment Observations

This section provides guidelines and consideration to observe regarding experimental environments in UMOBILE. These guidelines comprise a set of tools selected by partners to

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

ensure adequate result synchronization and a smooth integration in UMOBILE. The full definition of the programming environment is to be provided in early task 3.4. The requirements here described are intended to be applied as a starting point. They represent a mandatory sub-set of requirements which are necessary at the current project stage, as there may be architectural aspects that have to be dealt with.

- **Eclipse IDE.** The environment for programming and for integration purposes is Eclipse IDE [15]. Eclipse is an open source multi-language software development environment, which is originally developed for Java development tools (JDT). However, by means of plug-ins, many other programming languages (C, C++, Python, etc.) can be supported by Eclipse. Moreover, Eclipse can also be used to support ns2 simulations, OMNET++ simulations, etc.
- **Android Development Tools.** In addition, it is mandatory in UMOBILE to consider the Android Development Tools (ADT) plug-in to Eclipse in order to build, debug and test Android applications with an Android phone or doing emulation through emulator.
- **Android Studio** [16]: Official integrated development environment (IDE) for Google's Android operating system, built on JetBrains' IntelliJ IDEA software and designed specifically for Android development. It is available for download on Windows, macOS and Linux based operating systems. It is a replacement for the Eclipse Android Development Tools (ADT) as primary IDE for native Android application development.
- **GNU Tools**. The partners must consider GNU GPL v3.0+.
- **Python and Python libraries:** The implementation of the Service Migration Platform was conducted in the Python language (Python 2.7.12) [17]. We opted for Python against Java, C++ and other candidates because we aimed at a quickly produced research implementation (as opposed to comprehensive production software). In this regard, Python excels. It is a language with strong support from the online community which provides both libraries and samples of running codes. Particularly useful to the Service Migration Platform are the PyNDN, Docker, json and threading libraries.
- **Docker containers:** To realise the light weight virtual machines, we used Docker containers. Docker containers fall within the category of process virtual machines. As such, a Docker container is an application level process supported by a virtualisation software layer running on a host operating system [18]. A process virtual machine is capable of supporting the execution of a single user application, as opposed to a full virtual machine capable of running several processes. A docker container runs when its process runs and dies when the process does. One can instantiate several containers simultaneously within a host operating system. Though each container runs its application separately, all the containers share and (depending on the configuration) compete for the underlying hardware resources. Thanks to the virtualisation layer, application developers can pack the main file of an application along with all its dependencies in a single image that can be seamlessly instantiated on any platform running a Docker engine (the run time environment that support the image). The portability, and small size of Docker images, as well as the small time to

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

create, instantiate and discard them motivated us to use Docker [19] in the implementation of the Service Migration Platform. We have run Docker on Pi computers running 4.4.50-hypriotos-v7+ operating system, used images of Web servers of 88 Kbytes to 368 Mbytes that can be instantiated in less than 1.5 seconds. As demonstrated by results reported in D4.4, we found Docker to be a convenient technology to validate the UMOBILE ideas and implementations.

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

5 UMOBILE UAVs deployability

UAVs are equipped with Wi-Fi to create a local communication infrastructure and should be deployed to demonstrate the possibility of extending UMOBILE networks. These UAVs will be able to collect and relay relevant information. TEKEVER provides multiple aircraft in the AirRay family. The following three systems were considered as potential platforms to be applied in UMOBILE:

- VR1 is a collapsible man-portable system designed to be rapidly assembled, deployed and recovered. Intelligent software assists the operator in flight, and real-time imagery from the UAV may be displayed on multiple mobile devices. The VR family of UAVs are highly stable in flight, and sensor swapping is easily performed by the operator (HD camera, Infra-red, high-precision zoom sensors). The maximum take-off weight is 2.8 kg, the operational flight time is 30 minutes, and the range 2km, with a 250m operating ceiling.
- AR-4 Evo is a small tactical, hand-launched UAV for ISR (Intelligence, Surveillance, and Reconnaissance) targeting the defence market. With an endurance of 2 hours, a range of 20 km it is designed essentially for over-the-hill operation and maximum payload weight 1 kg. The AR-4 has a civilian counterpart, called AR-1 Blue Ray, which fulfils the surveillance needs of public safety users. It is a modular UAV with a focus on ease of use, transportability, maintainability and payload flexibility.
- AR-3 Net Ray is a higher endurance and longer range platform, which can be applied to other applications beyond ISR due to its higher payload capacity. These include maritime and coastal surveillance missions, working as an operational extender for vessels as well as serving as communications relay to other vehicles and communications range extender. The maximum payload weight is 8 kg; the operational flight time is 10 hours and the range 80 km.

Figure 8: Tekever AR3 Net Ray operation

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

The AR-4 was initially considered as the most suitable platform, due to its size, range, easy transportation and intuitive operation, while the VR-1 would operate for a short period. The AR-3, by its turn, initially could be too big to operate close enough to generate a local WiFi Hotspot. However, the equipment to be integrated revealed to be too large and heavy to fit into the AR4, so the use of other systems was analysed through a risk mitigation plan. In fact, AR3 was later then used and proved to be the most valuable solution, instead of AR4 (which did not have the appropriate payload requirements), for the integration purposes. This exchange of UAVs for the UMOBILE mission was supported by a long set of experiences and tests. The VR1 was considered for cases when the altitude and speed of the fixed wing UAVs was too high to enable the generation of a stable local network. Indeed, the VR1 proved to be able to have advantages due to its quadcopter VTOL characteristics to operate in small and strictly local areas.

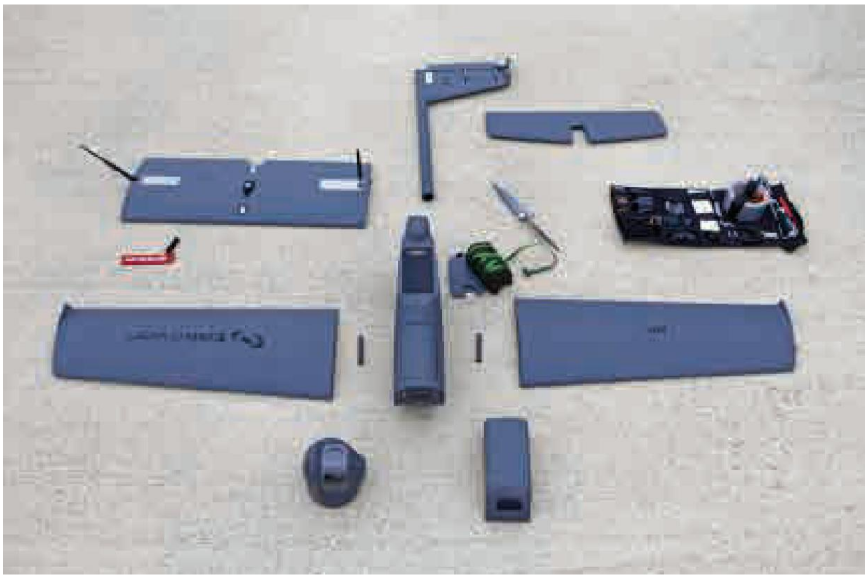

Figure 9: Disassembled Tekever AR4 EVO

Introducing the AR3 Net Ray, this is a fully autonomous tactical UAV for ISTAR (Intelligence, Surveillance, Target Acquisition and Reconnaissance). It has an endurance of more than 10 hours, range of 80 km, weight of 22kg, capable of a 5kg payload (including the Gimble camera), and designed for diverse activities, including search & rescue missions. The payload possibilities in AR3 represent a big advantage when compared to the maximum of 1kg payload that AR4 supports.

There are different sets where the UMOBILE services could be applied. In each scenario it should be used the proper UAV, suitable for the mission requirements: different scenarios differed on the UAV model to be used: AR3, (initially AR4) or/and VR1.

Figure 10 illustrates the complete rural/forest scenario.

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

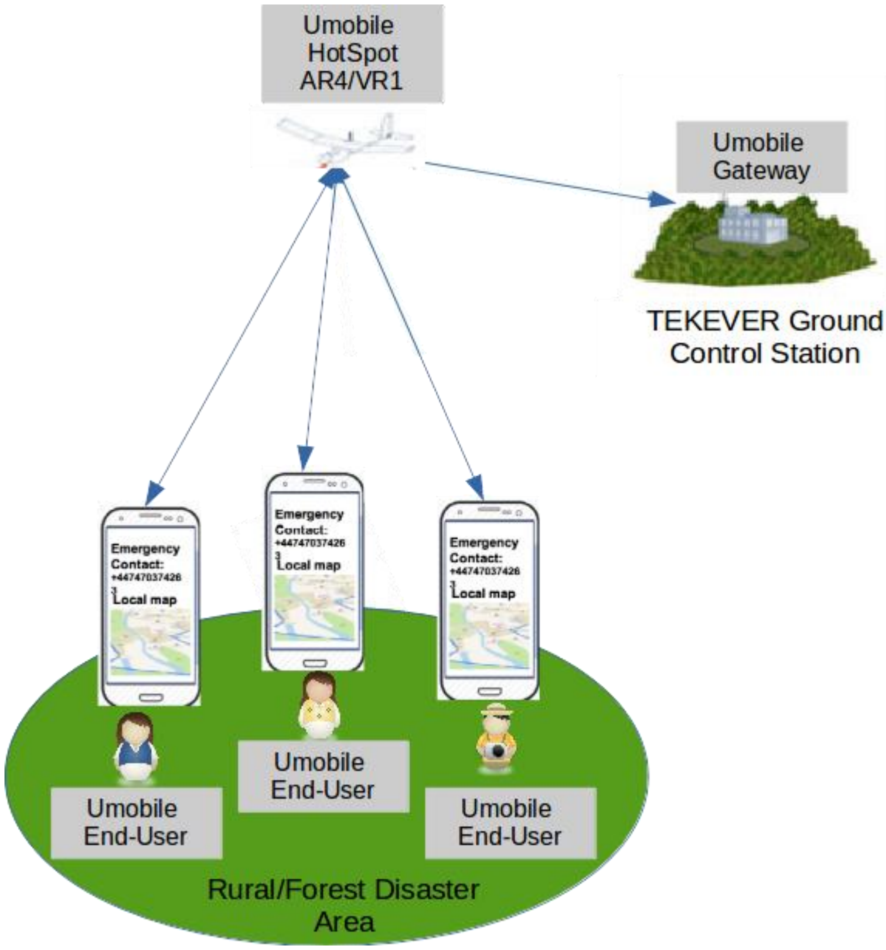

Figure 10: Rural/Forest Disaster Scenario with initially considered UAVs (still with AR4)

Nevertheless, since the use of UAVs in this mission concerns its application as an end-user to migrate data and to operate as an access point to migrate data for an operation centre, it is relevant to distinct in which of these two methodologies should be used the AR3 or the VR1. This decision relies on the area of coverage of each the UAVs – which means that, after the appropriate studies, the AR3 is capable of a loiter manoeuvre that allows the transmission of the message from the ground covering large areas, while VR1 has a propensity to messaging/data transmissions in small local areas, making use of its hover capabilities.

Figure 11 shows the integration work, in which it is possible to see the antenna attached to the VR3 and the incorporated access point identified by the red box.

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

Figure 11: AR3 with Raspberry PI 3 (Access point) and attached antennas

Since the goal of the previous experiment was to find the maximum coverage achieved by the UMOBILE hardware, the use of AR3 as a medium-range UAV had little influence on final results.

In line with these circumstances, indeed, for final demonstration purposes (supported in 2018 by the UAV Demo in Portugal under the responsibility of TEKEVER) the chosen device for validation was the VR3.5, since this UAV was previously analysed for the mission scenarios and it was proved that it would be more efficient to use a lighter UAVs multicopter that could be easily piloted. Therefore, in the UAV demo, it was decided to use the VR3.5 multicopter that is presented in Figure 12, with the appropriate incorporations of the required devices that allow the platform to accomplish its predefined data mule objectives. The VR3.5, was, then, implemented with the DTN system and the Wi-Fi antenna (RBMetal2SHPn from Mikrotik with a 6dBi/8dBi, which revealed several advantages for the communications between devices). VR3.5 offered the necessary capabilities to implement this mentioned device.

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

Figure 12: TEKEVER Multicopter VR3.5 with the installation of the antenna

The validation of the total amount of developed systems under UMOBILE and main core of the project were then accomplished with the procedures under analysis on this UAV demo (including the TEKEVER multicopter VR3.5), without any major concerns.

5.1 Regulatory issues

The drone industry is immensely growing, but there are legitimate concerns for privacy and data protection, according to the Advance Notice of Proposed Amendment (A-NPA) 2015-10, from the European Aviation Safety Agency (EASA), which proposes the "Introduction of a regulatory framework for the operation of drones". The proposed approach in this A-NPA focuses on risk and performance, and not only on size. Three categories of operations are introduced:

Table 10: Drone Categories of Operation (according to A-NPA 2015-10, from EASA)

'Open' category **(low risk)**	Safety is ensured through operational limitations, compliance with industry standards, requirements on certain functionalities, and a minimum set of operational rules. Enforcement shall be ensured by the police.
'Specific operation' category **(medium risk)**	Authorisation by National Aviation Authorities (NAAs), possibly assisted by a Qualified Entity (QE) following a risk assessment performed by the operator. A manual of operations shall list the risk mitigation measures.
'Certified' category **(higher risk)**	Requirements comparable to manned aviation requirements. Oversight by NAAs (issue of licences and approval of maintenance, operations, training, Air Traffic Management (ATM)/Air Navigation Services (ANS) and aerodrome

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

	organisations) and by EASA (design and approval of foreign organisations).

The 'open' category poses the largest interest for the UMOBILE project because in this case, enforcement to the rules, according to the A-NPA, will be ensured by Law Enforcement Agencies that will need effective instruments and methodologies to enforce correct operation. Within the 'open' category, three subcategories are proposed:

Table 11: Subcategories in the 'open' category (according to A-NPA 2015-10, from EASA)

CAT A0 0 to 999 g	'Toys' and 'mini drones' < 1 kg A considerably high number of consumer products fall into this subcategory which are operated in all kind of operational environments. Depending on the exact definitions, this category includes tethered balloons, kites, toys as well as sophisticated devices following automatically the owner.
CAT A1 1 to 3,99 kg	'Very small drones' < 4 kg The majority of better performing consumer products fall into this subcategory. Normally equipped with navigation and automation systems, their performance is impressive and can carry payload, and start posing a more significant risk to third parties.
CAT A2 4 to 25 kg	'Small drones' < 25 kg In this subcategory there are mainly products operated commercially, e.g. carrying high-quality camera systems, or drones or models operated by enthusiasts.

Additional requirements are defined for these subcategories, namely considering restrictions in the areas where they operate, as can be observed in Figure 13. Other requirements include:

- CAT A0 shall have limited performance to assure flight below 50 m or limit it automatically;
- CAT A1 drones operating in 'Limit-drone zone' must have active identification and up-to-date geofencing capability enabled;
- CAT A1 and CAT A2 drones must automatically limit the airspace they can enter and have the means to allow automatic identification;
- Pilots operating drones over 50m must have basic aviation awareness.

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

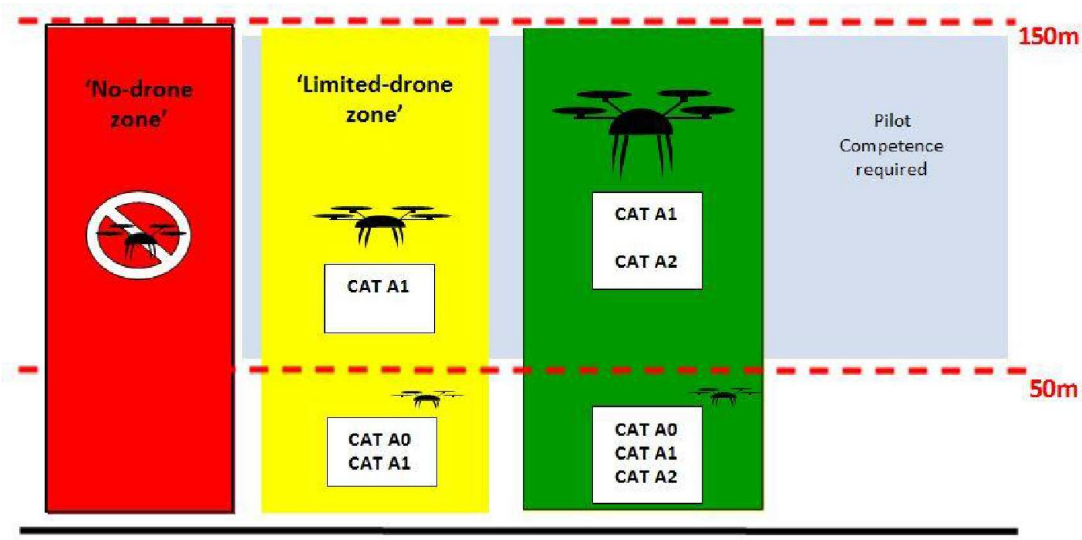

Figure 13: Zones of operation for the three subcategories of drones (according to A-NPA 2015-10, from EASA)

In the Portuguese case, regulation has come out which determines some rules for operation, such as UAV operation, in Visual Line of Sight, being limited to below 120 metres and being executed so that it minimises risks to persons goods and other aircraft.

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

6 Conclusions

In this deliverable, we have provided the description of the UMOBILE system deployability design, in terms of:

- An end-to-end Internet perspective for UMOBILE deployment, considering as underlying design the operational multi-access heterogeneous networks of today.
- Main operating systems as well as main devices in the different Internet regions, required to carry on a UMOBILE deployment.
- Tools and experimentation procedural guidelines as well as guidelines to further detail the deployment of the UMOBILE system specifications, describing briefly on the usage of the UMOBILE Lab. This part is the basis to assist in the large-scale project deployment, which will be used both for experimentation and for large-scale demos in the last year of the project.
- Design requirements for UAVs deployability in the UMOBILE ecosystem, assisting in extending the network coverage and user functionalities.

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, "D2.2 System and network requirement specifications (1)".

[2] UMOBILE Project, "D2.1 End-user requirements report".

[3] UMOBILE Project, “D5.3 Proof-of-concept”.

[4] UMOBILE project, "D3.2 UMOBILE Architecture Report (1)".

[5] UMOBILE Project, "D3.3 UMOBILE ICN layer abstranction initial specification".

[6] UMOBILE Project, "D5.1 Validation methodology and evaluation report (1)".

[7] UMOBILE Project, "D3.1 UMOBILE Architecture Report (1)".

[8] UMOBILE project, "D4.4 Set of QoS interfaces and algorithms".

[9] C.-A. Sarros, A. Lertsinsrubtavee, C. Molina-Jimenez, K. Prasopoulos, S. Diamantopoulos, D. Vardalis and A. Sathiaseelan, "ICN-based Edge Service Deployment in Challenged Networks Demo," in *ICN'17*, Berlin, Germany, 2017.

[10] "GitHub," [Online]. Available: https://github.com.

[11] "UMOBILE Raspberry Pi image (github)," [Online]. Available: https://github.com/umobileproject/raspi-image.

[12] "University of Canterbury, Project Akaroa. Available at http://www.cosc.canterbury.ac.nz/research/RG/net_sim/simulation_group/akaroa/about.chtml".

[13] "TheONE, The Opportunistic Network Environment simulator," [Online]. Available: https://akeranen.github.io/the-one/. [Accessed 13 07 2018].

[14] "ndnSIM: ns-3 based Named Data Networking (NDN) simulator," [Online]. Available: http://ndnsim.net/2.3/. [Accessed 13 07 2018].

[15] "The Eclipse foundation open-source software," Eclipse, [Online]. Available: http://www.eclipse.org/.

[16] "Android Studio 2.3.3," [Online]. Available: https://developer.android.com/studio. [Accessed 21 09 2017].

[17] "Python home page," [Online]. Available: https://www.python.org. [Accessed 13 07 2018].

[18] "Dirk Merkel, Docker: Lightweight Linux Containers for Consistent Development and Deployment, Linux Journal, Issue 239, Mar, 2014".

[19] "Docker home page," p. https://www.docker.com/.

[20] UMOBILE Project, "D4.5 Report on data collection and inference models".

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

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

[22] "Seweryn Dynerowicz, Paulo Mendes, "Named-Data Networking in Opportunistic Networks", in ACM ICN, Berlin, Germany, September 2017.".

[23] "W. Moreira, P. Mendes and E. Cerqueira, “Opportunistic Routing based on Users Daily Life Routine”, IETF Internet Draft (draft-moreira-dlife-04), May 2014".

[24] "Luis Amaral Lopes, Rute C. Sofia, Paulo Mendes, Waldir Moreira, "Oi! - Opportunistic Data Transmission Based on WiFi Direct" in Proc. of IEEE INFOCOM, San Francisco, USA, April 2016.".

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