MONROE

Measuring Mobile Broadband Networks in Europe

H2020-ICT-11-2014
Project number: 644399

Deliverable D4.2
Maintenance Activities

Editor(s): Giacomo Bernini
Contributor(s): Pietro G. Giardina, Thomas Hirsch, Audun Fosselie Hansen, Ozgu Alay, Vincenzo Mancuso, Jonas Karlsson, Mohamed Moulay

Work Package: 4 / Maintenance
Revision: 1.0
Date: January 31, 2018
Deliverable type: R (Report)
Dissemination level: Public

European Commission
Horizon 2020
European Union funding
for Research & Innovation

Abstract

This report presents an overview of the maintenance activities carried out on the MONROE platform in the context of WP4. In particular the document focuses on how the platform has evolved and how troubleshooting activities have been performed, at both hardware and software level. This has translated into dedicated maintenance activities applied to the MONROE node software and hardware components, as well as to all those components building the platform backend.

Participant organisation name	**Short name**
SIMULA RESEARCH LABORATORY AS *(Coordinator)*	SRL
CELERWAY COMMUNICATION AS	CWY
TELENOR ASA	Telenor
NEXTWORKS	NXW
FUNDACION IMDEA NETWORKS	IMDEA
KARLSTADS UNIVERSITET	KaU
POLITECNICO DI TORINO	POLITO

Contents

1 Introduction

The MONROE project aims to build and operate a novel platform to perform measurements and experiments in the operational mobile broadband networks (MBBs). The main target of the project is deploy an pan-European open testbed for independent, multi-homed, large-scale monitoring and assessment of performance of MBBs networks in heterogeneous environments and conditions. The scale and Europe-wide deployment of the MONROE platform make maintenance a challenging task, where many different software and hardware components can potentially contribute to failures and platform malfunctioning. More than that, maintenance is key to allow experimenters and external users have full access and availability of measurement nodes and backend services.

This document lays its foundations on the maintenance procedures and routines defined in deliverable D4.1, considering that some of them required some improvement and enhancement due to the evolution of the MONROE platform. The goal of this deliverable is to provide a summary of those maintenance activities carried out in the context of WP4 with the aim of keeping MONROE nodes and backend services (including data) up and running and available for experiments. In addition to the regular platform maintenance, it is worth to mention that most of the MONROE platform upgrade to new design node has been performed in WP4 as part of special maintenance duties. In particular these activities have partially involved the engineering of the new design and the complete assembly of the new mobile and stationary nodes.

In summary, this document provides an overview of those maintenance activities carried out so far in WP4 at both software and hardware level. In particular, two main maintenance categories are reported in the deliverable: i) node maintenance, including those interventions, updates, fixes and upgrades implemented for the node itself (i.e. from an hardware perspective), as well as on the node software mostly in terms of SW releases and incremental features; ii) backend and data maintenance, including all those maintenance duties applied for the proper operation of all the MONROE platform backend services (from database, to scheduling, user access and visualization).

2 MONROE Platform Maintenance Activities: Overview and Goals

Maintenance of the MONROE platform is a crucial task to offer a stable and available MBB networks measurement facility to experimenters. Since the beginning of the project, the consortium has identified maintenance as one of the key challenges and focus for the project. Indeed, availability and usability of the platform is a pre-requisite to attract more and more external users and actors, and also an enabler for sustainability opportunities. Given the complexity and scale of the MONROE platform, from its wide distribution of measurement nodes to the composition and integration of several backend services, a set of well-defined maintenance routines have been defined in deliverable D4.1 as the ground for keeping the platform stable The main maintenance objectives and goals can be summarized as follows:

- **Keep alive**: Maintain the nodes available and maximize the operating conditions, that is guaranteeing access and usability for experiments.
- **React fast**: Automate the maintenance procedures as much as possible to quickly react to known anomalies and failures. This translates into periodic checks of individual nodes status targeting remote troubleshooting within 5 days from issues occurring. When remote interventions does not solve the anomaly, replacement of measurement nodes is planned.
- **Data first**: No measurement data should be lost once the data is transferred from the MONROE nodes

to the backend services. This is key to have efficient and stable experiments, maximizing the usage of the scheduling service thus enabling more and more experiments.

These goals have been translated into a wide set of capillar maintenance activities that the consortium has carried out periodically (from daily to weekly basis) to first check the status of the different platform building blocks (individual nodes, servers and services), and where needed take proper actions to promptly solve issues and bugs. The following chapters provide an overview of the maintenance activities the consortium has implemented so far to keep the platform stable and evolve it according to the emerging needs. The full list of maintenance interventions performed, at both hardware and software level, is reported in the project's github repository (https://github.com/MONROE-PROJECT), that has been used as tracking tool to manage platform issues.

3 Node Maintenance Summary

Following D4.1, the maintenance activities of the node cover both the software and hardware maintenance. As described in D2.2, the Monroe platform has gone through a HW update which has led to a complete redesign of the node itself. The transition from a single APU to a DUAL APU architecture forced the consortium to face new challenges both in terms of development of additional SW to handle the new HW configuration and effort spent in platform rebuilding and redeployment.

3.1 Hardware Maintenance Activities

This section summarizes the maintenance activities performed from the hardware perspective.

3.1.1 Monroe Platform rebuild

As discussed before, a big effort has been put around the redesign and rebuilding of Monroe nodes, both stationary and mobile. From an hardware point of view, this effort has been focused on the transition from a single APU to a DUAL APU system, through the modification of the already existing single APU nodes. The node modification process involved the nodes deployed in each country of the consortium. When necessary, new nodes have been built. In particular, at end of rebuilding process, we have the following number of nodes:

Country	Type	Number	Assembler	When
Norway	Stationary	10	SRL	Q9
	Mobile	25	SRL	Q10
	Testing	5	SRL	Q9
Sweden	Stationary	15	KAU	Q10
	Mobile	30	NXW	Q9, Q10
	Testing	6	KAU	Q11
Italy	Stationary	5	NXW	Q11
	Mobile	35	NXW	Q9, Q10
	Testing	5	NXW	Q12
Spain	Stationary	10	NXW	Q10
	Testing	5	NXW	Q11

Table 1: Dual APU nodes produced and deployed after the rebuilding process

3.1.2 Periodic node maintenance

At the time of writing, there was only one case of complete node hardware failure, happened on Q11. The normal procedure in such cases would be to return the node to the partner which is in charge of the assembly procedures (NXW), which should analyse the device and, in the worst case, build a new node to replace the broken one. In the case mentioned above, the failure was caused by malfunctioning of the two APUs inside the node. Since the partner handles that node, KaU, was able to replace the APUs by itself, the only action performed by NXW was to ship the new APUs for replacement. A part from this isolated case, the main maintenance activity consist of periodic weekly check of the platform status.

CWY releases weekly (usually on Monday) a report contains useful information on the current status of the nodes, retrieved from the Inventory and the Scheduler. A sample report from Week 4 2018 is illustrated in Table 2. Based on the information contained on the report, each partner checks its own project and compares the status of the nodes he manages with the current information provided by the Inventory and the Scheduler, in order to figure out if issues reported (if any) are persistent or not. In case the issue is confirmed, each partner will adopt the maintenance procedures described in D4.1. This automated maintenance report generator has been running for the entire since Q11 and providing valuable help to supervise manual maintenance activities. On top of regular maintenance, experimenter and node issues are constantly being evaluated, and mitigated through support actions, maintenance and small updates.

3.1.3 Upgrade of maintenance routines and procedures

Some differences exist between the currently maintenance procedures and the ones described in D4.1. Such differences are due to a couple of different factors. The first is certainly the Monroe platform rebuild, which made some procedures, related to the old node architecture, no longer required: MiFis and USB Hub maintenance procedures do not have to be performed anymore. The second factor is the natural experience gained by the consortium once the platform started working at full speed. Indeed, for mobile nodes, the 3-day not working limit, which should trigger the proper maintenance, appear now too tight, especially with respect the normal working cycle of transport medium (bus, truck or train). To give an example, a bus could be stopped at the depot for weeks or even months. In such context, a 3-day check could give a lot a "false positives", making the maintenance procedure overhead grow. For that reason, the consortium "normalized" the periodic maintenance activities on weekly cycles.

3.2 Software Maintenance Activities

Software development on the MONROE platform has been frozen at certain intervals to provide a stable baseline for experimentation. This ensures that the basic features of the network and the software components used remain unchanged throughout long running experiments in order to avoid any artifacts on the results of the experiments. Nevertheless, continuous support was provided to eliminate issues, and add features after considering their impact on the existing testbed. Furthermore, the running software components - scheduler, importer, database - were regularly monitored to eliminate malfunctions and to provide better support for unexpected use cases. All feature updates are documented in the project's github repository: https://github.com/MONROE-PROJECT.

Next we describe the two main releases of the SW, next upcoming release and the supported features in these releases.

Week # 4 Overview	
Nodes 294	Nodes last week 291
Issues 89	Issues last week 88
Working 205	Working last week 203
Failed 12	Fixed 12
Per Project Overview	
Project	Number of Operational Nodes
VTAB	43
NSB	40
Sweden	35
Norway	21
Spain	20
GTT	16
Italy	15
WSYS	8
ALLBESMART	3
NORLAB	3
UMA	1
Overall Issue Overview	
Issue	Issue Count
last seen > 1 month.	45
Not all interfaces live	24
last seen > 1 week.	12
In maintenance	3
last heartbeat > 1 week.	3
not in scheduler.	2

Table 2: Weekly Maintenance Status

3.2.1 First SW Release: v1.0 'Let it fly'

The main updates included in ***v1.0: Let it fly*** of the platform is released early in Q11 and includes all updates after Q8 . The main features are as follows:

Management traffic: With the removal of dedicated management interfaces, the software was updated to adjust management traffic priorities according to quota usage on the interfaces. CWY and SRL also implemented a necessary change in the scheduling system, whereby the current interface traffic quotas are used to predict availability of nodes and resources. While in the old system, a dedicated management interface with a separate data quota could be used for all background traffic, the new system has a merged quota for system and experiment traffic. Therefore, usage has to be continuously monitored, and can no longer be predicted by the scheduling system.

DNS: Core software has been upgraded to provide multipath DNS resolution using the operators DNS servers, providing a more correct routing of requests, e.g where content delivery networks can be used.

Configuration: The new configuration provided by CWY and IMDEA includes support for separating head and tail nodes, in both the scheduling system and on the node: In the scheduling system and user interface, head and tail nodes are selectable separately and their characteristics with respect to interface count and wifi capability taken into account. On the node, head and tail nodes receive a separate network configuration, so that they are able to communicate with each other.

Wifi and GPS support: The configuration also includes support for WIFI/WPA client mode to access Wifi networks where the nodes are deployed. CWY provided an update to the network listener component to support GPS data from the new MC7455 modems.

3.2.2 Second SW Release: v1.1 'Winter is coming'

Following the main release, the consortium and the external users have identified new features that are desirable. The second main release of the platform ***v1.1: Winter is coming*** has been updated in early Q12 with the following features:

DNS lookups: The containers was using a shared file /etc/resolv.conf for DNS lookups, which by docker is configured to fall back to Google DNS. In the new configuration implemented by SRL and CWY, each container receives its own /etc/resolv.conf which is mounted writable and defaults to "resolv 127.0.0.1", i.e a dnsmasq service running in the host system. In addition, a configuration for dnsmasq is mounted into the containers at "/dns", to identify which servers will be used by the lookup. The host system is using ALL operator DNS and Google DNS in parallel, caching the fastest reply. In order to control which DNS should be used, the recommendation is to read out the operator DNS from the /dns file, and modify /etc/resolv.conf

As an example of how to configure the new DNS setup, as well as to test how CDN addresses are differentiated, KAU have created the DeNniS experiment: https://github.com/MONROE-PROJECT/Experiments/blob/master/experiments/dennis/files/experiment.py#L57

Data quotas: The unused quotas from deploying experiments are returned to the user accounts, providing more flexibility to the experiment planning.

MTU: There were issues with MC7304 modems where modems crashed when sending large, fragmented packets. This has been fixed by setting the MTU to 1500. In other words, dhclient no longer requests (broken) MTU from modems.

SCTP: In the core components, SCTP modules are enabled and firewall allows SCTP traffic.

Scheduling client: We received reports that experiments had been deployed twice if a correct status code has not been received by the server. The issue has been addressed by adding additional checks - in the process we also disabled redeployment (originally meant to be a feature) of tasks, also in order to preserve the logs of the first deployment.

First availability scheduling: In addition to the currently reserved time slot mechanism, the scheduler now supports a best-effort, first availability scheduling. Experimenters will provide most of the parameters as they do today, but their experiment will not receive a reserved slot in the scheduling calendar. Instead, they will be executed on matching nodes when they have free capacity, at any time within a provided time window. To that aim, the server will automatically assign 'first availability' tasks to nodes who match the experiment filter profile when they send their scheduling requests and have free capacity. At this point, the start/stop time of the task will be set, and the task treated like a regular experiment.

This has many advantages including:

- Tasks will only be executed when the nodes are online. This should be advantageous for mobile nodes with regular downtimes.

However, it has also disadvantages:

- Tasks in the 'first availability' will not be deployed 24h in advance on the nodes. Tasks with long deployment times will have to take this into account in their time slot.

- Tasks will not be executed at a given time.

Prior to assigning a task to the 'first availability', a regular scheduling query will be executed by the server to determine if the request has a chance to be fulfilled at all. This implementation is carried out by SRL and CWY.

monroe-cli: To streamline the experiment scheduling process, one can use the MONROE command line tool under: https://github.com/MONROE-PROJECT/monroe-cli

3.2.3 Upcoming third SW Release: v1.2 'Let it snow, let it snow'

The next release is under testing and planned to be released early Q13 with the following features:

Virtualization: KAU and CWY has continued the work on extending the platform with full virtualisation capabilities, and to integrate deployment and scheduling of virtual machine in the current platform. This is with the aim to provide full support for custom kernel extensions, and other extensions requested by Open Call experimenters. Given the storage and transfer limitations, this is a relatively challenging task, which has only been resolved after the duration of Q11, but it is expected that a decision on integration with the scheduling platform will be taken in Q12.

Wifi AP subscription: Wifi Access Point subscription is enabled for eduroam where the experimenters can use eduroam with their own user name and password as parameters to use Wifi interface. This has been implemented and tested by SRL, CWY, KAU and IMDEA.

Idle network: In the default configuration the modems switch between networks, bands and technology automatically based on signal strength. If the modem is actively transmitting or receiving, the modem delays the band/technology switching to not cause outages, at the expense of a possibly better connection on another band or technogy. This "idle network" feature implements a 30 second forced firewall blocking of all traffic (including node management) before the container is started. This idle network enables the modem to automatically perform the band and technology switch.

Technology and Band Lock: In some cases it is desirable to lock the modem to a specific band and technology instead of the automatic band/technology selection usually performed by the modem itself. This is implemented and will soon be available to the experimenters so that they can lock to only 4G during their experiments. To avoid any bias to the platform, a node reboot will be triggered when container exits.

4 Backend and Data Maintenance Summary

Backend and DATA maintenance considers both HW and SW maintenance for continuously running the different backend services. As with the software running on the node, these services are crucial to ensure the measurement results are generated, transferred and stored correctly.

The complete list of services and the corresponding representatives are as follows:

- Certification Authority / User accounts (IMDEA)
- User Access (IMDEA)
- Scheduling server (SRL)
- Measurement responder (ALL)
- Container Repository (KAU)
- Result storage(KAU/IMDEA)

- Cassandra Database (IMDEA)
- Database Importer (KAU)
- Visualization (NXW)

The above assigned representatives check the status of these services weekly and in case of problems, the issues are discussed in the weekly maintenance telco and addressed as soon as possible.

Below we define the maintenance activities separately for HW, SW and data maintenance. Most of these services take the form of a web server running dedicated software, hence maintenance procedures are similar and aligned with commonplace server operations.

4.1 Hardware Maintenance Activities

Regular maintenance and follow-up for the backend servers is performed by the partner responsible for hosting the server (see above). This involves:

- Checking and ensuring that the servers are powered, running, and have resources to operate for the next month.
- Monitoring disk space, CPU load, memory, etc, and carrying out the upgrades when necessary. The servers are running in a virtualized environment which means that resource adaptation can be done flexibly.

4.2 Software Maintenance Activities

Certification Authority / User accounts: Upon arrival of new users, new accounts and certificates are created. The certificates are updated when they expire, triggered by a request from the users.

User Access: User access system is updates based on the new platform features, together with the SW releases. Consequently, the user manual is updated to detail the new features introduced to the platform.

Scheduling server: The scheduling server is run on a different virtual machine to isolate any incidents with the other services. Scheduling server is regularly rebooted to ensure healthy operation of continous experiment scheduling operation.

Measurement responder: One measurement responder per country is maintained for MONROE base experiments such as MONROE-Nettest. SRL, KAU, IMDEA and POLITO are responsible for monitoring the health and availability of these responders, in their respective countries.

Container Repository: The container repository is kept up to date with all the new certified containers both for the consortium experiments as well as the user experiments.

Result storage, Cassandra Database and Database Importer: These services are run in a single VM located in KAU. The health of these services are constantly monitored and in case of any problems, automatic restart of these services are carried out. The log files are kept to identify the root cause of the problems and to propagate any further incidents. The issues are discussed in details in the weekly maintenance calls.

Visualization: A periodic (weekly) check of Visualization App is performed in order to guarantee the proper operation (e.g. it is up and interact with Cassandra DB and Inventory). Another important activity consists in keeping up-to-date the App in terms of number variation of the node in the platform: as soon as new nodes are added or removed from the platform, an update action is triggered.

4.3 Data Maintenance Activities

During the beginning of Q12, some minor issues were encountered with the Database regarding the data importer but were all resolved immediately by the corresponding personnel to guarantee smooth user experience. The data maintenance activities remain almost the same as D4.1, which now includes a bigger database backup.
The data maintenance includes backups and validation of:

- Measurement results of MONROE base experiments.
  - Source data (node JSON files): Automatically compressed and backed-up every night to a remote location (IMDEA) the size of this folder at the time of writing is 145 GB total and it is subject to daily increase.
  - Online DB replicas: The MONROE database will be replicated to two or more servers located at partners within the project (depending on SW/HW availability).
  - Additional offline DB: Kept (e.g., at IMDEA) for direct access by the external experimenters; this DB instance will be updated daily from the log files and will remain disconnected from the main copies and it is the same size as the original one (up-to-date).
- System log files from the nodes and backend servers are sent to a central SYSLOG server.
- Result files and container logs from user experiments are transferred to a central server for later retrieval by the experimenters. These files will be kept for two weeks; then, they will be deleted.

Additional processes done routinely for data maintenance include:

- Ensuring on a daily basis that the data is monitored and backed up, including the log files. The backup folder size monthly is around 16.5 GB and daily around 600 MB compressed.
- A service will run in the database server to monitor DB activity and alert the corresponding representative if no data has been inserted for any specific node in the (online) DB during the last 30 minutes so that manual procedures can be started. Additional validation procedures for the last received data may be performed by this service.
- Restarting the insertion process of metadata files per day into the database in case of any failure which was mentioned above.
- Deciding when to delete the log files (if space issues arise).

For the upcoming month, a more detailed report will be generated weekly, for the base experiments running which, will help identify success and failure rate regarding the base experiments.

5 Lessons Learned

Throughout the last year, there were many valuable lessons we learned running and maintaining a large-scale platform such as MONROE. We summarize the most critical ones below:

- Weekly maintenance reports were crucial for monitoring the node status and health of the overall platform. Especially for stationary nodes, almost all issues are identified and very shortly addressed. However, mobile node maintenance turned out to be a very challenging task. At times, we observe that some busses and trains can be in depot for long period of times, and it is not feasible to check with our hosts each time a node is offline for a week, for example. Therefore, it is important to follow a less strict maintenance routine for mobile nodes.

- While node maintenance is the first step to keep the platform alive, providing an end-to-end monitoring system was essential to make sure that all elements in the complex chain is operational. To do so, we further monitor the data produced by the base experiments to ensure the health of the overall platform, from node to the back-end to the experiments. Additionally, we take measures in order not to use any data by having a back up of the DB and all the result log files.
- While we monitor both the platform as well as the data, this has been done in a semi-manual fashion so far in the project. We believe full automation of this processes is necessary moving forward to be able to react to the issues as soon as they occur.
- For a research platform such as MONROE, we need to not only keep the platform alive, but we have to respond to the requests from the external users to provide a platform that can meet their needs. To this end, we kept a task priority list and this list is updated constantly. The major tasks has been discussed during plenary meeting while the updates has been discussed during weekly maintenance telcos.
- Last but not least, the communication among the maintenance team is very important. Tools to enable the communication was very helpful, such as the development mailing list, task priority list, monitoring tools and scripts. More importantly though, the weekly maintenance telcos ensured that everybody is involved and the issues are addressed timely.

6 Conclusions

This deliverable has presented the maintenance activities carried out on the MONROE platform, thus covering all its hardware and software building blocks, from the measurement nodes to backend services and data. On the node hardware side most of the maintenance efforts have been dedicated to the upgrade to the new dual-APU design (i.e. the assembly of the new stationary and mobile nodes), in parallel to the periodic monitoring and supervision of nodes and platform status. For what concern the node software part, two major releases have been implemented up to now and one release is planned to be implemented soon. Here, maintenance, feature upgrades and bug fixes have been included towards a more robust and stable node platform. In the context of backend and data services, most of the maintenance duties have involved periodic check and supervision of harwdare components, software services and data status, with the participation of all the consortium partners, each responsible for selected services or components. As a tracking and record tool, the MONROE maintenance interventions and tasks have been (and will still be up to the end of the project) reported in the project's github repository (https://github.com/MONROE-PROJECT).

It is important to highlight that maintenance and supervision of the whole MONROE platform, following the strategy reported in this document, will continue up to the end of the project in support of experiments and measurements.

Disclaimer