H2020 DAEMON Project

Grant Agreement No. 101017109

Deliverable 2.2

Initial DAEMON Network Intelligence framework and toolsets

Abstract

This document is the first deliverable of the second iteration of the DAEMON project, which builds on the results of the first iteration presented in D2.1, D3.1, D4.1 and D5.1. towards (i) improving the design of the Network Intelligence (NI) models developed by the project so as to overcome limitations identified in the initial algorithms, and (ii) starting the definition of a framework for the coordinated operation of the multiple NI-assisted functionalities. The present deliverable acts as the connecting document between the first and second iterations of the project, using the conclusions and results of the first iteration to pave the way for the activities of the second iteration. Therefore, D2.2 tackles (i) the updating of NI-assisted functionality requirements, (ii) the creation of a set of guidelines for a pragmatic NI design, (iii) the drafting of new requirements for the NI orchestration, and (iv) the provisioning of a NI plane blueprint, i.e., a high-level view of the organization of a novel network plane dedicated to end-to-end NI coordination.

Document properties

Document number	D2.2
Document title	Initial DAEMON Network Intelligence framework and toolsets
Document responsible	IMDEA
Document editor	Antonio Bazco Nogueras (IMDEA) Marco Fiore (IMDEA)

Document Contributors

Partner	Name	Surname	Section
i2CAT	Ginés	García-Avilés	2.1.4, 4.2.7, A.1.4, B
IMDEA	Antonio	Bazco Nogueras	1, 2, 3, 4, 5, A, B 1.1, 1.2, 1.3, 2.1.3, 2.1.7, 2.1.8, 2.2, 4.1.2, 4.1.3, 4.2.3, 4.2.4, 4.2.5,
	Marco	Fiore	
	Michele	Gucciardo	
IMEC	Miguel	Camelo	2.1.2, 3.3.2, 4.1.5, 4.2.1, 4.2.2, 5
	Paola	Soto-Arenas	
NBL	Chia-Yu	Chang	2.1.6, 4.1.1, 4.2.3
	Danny	De Vleeschauwer	
NEC	Josep Xavier	Salvat Lozano	2.1.1, 2.1.4, 3.1, 3.2, 4.1.6, 4.2.7, A.1.1, A.1.5, A.1.4, B
	Andrés	García Saavedra	
	Xi	Li	
OTE	Alexandros	Kostopoulos	5.1.1, 5.1.3
TID	Andra	Lutu	2.1.8
	Carlos	Segura	
UC3M	Marco	Gramaglia	2.1.4, 3.1, 3.2, 4.2.7, A.1.4
UMA	Joaquin	Ballesteros	0, 2.1.5, A
	Lidia	Fuentes	
WINGS	Vangelis	Kosmatos	4.1.7, 4.2.6
ZSC	Ivan	Paez	4.1.4

Target dissemination level	Public
Status of the document	Final
Version	1.0

Production properties

Reviewers Evangelos Kosmatos – (WINGS)

Document history

Revision	Date	Issued by	Description
0.1	06/05/2022	IMDEA	First draft with document structure
0.7	01/07/2022	IMDEA	First complete draft with full content
0.8	08/07/2022	IMDEA	Revised complete draft with full content
0.9	20/07/2022	IMDEA	Revised version based on review
1.0	31/07/2022	IMDEA	Final version

Disclaimer

This document has been produced in the context of the DAEMON Project. The research leading to these results has received funding from the European Union Horizon 2020 research and innovation programme under grant agreement no.101017109.

All information in this document is provided "as is" and no guarantee or warranty is given that the information is fit for any particular purpose. The user thereof uses the information at its sole risk and liability.

For the avoidance of all doubts, the European Commission has no liability in respect of this document, which is merely representing the authors view.

Table of Contents

List of Figures

List of Tables

List of Acronyms

AARES - Automated Anomaly RESponse

AI – Artificial Intelligence

B5G – Beyond 5G

CAWRS – Compute-aware radio scheduling

CFORE - Capacity FOREcasting

DL - Deep Learning

DNN - Deep Neural Networks

DRL - Deep Reinforcement Learning

DU – Distributed Unit

EAWVNF - Energy-aware VNF control & orchestration

FR – Functional requirement

GPU – Graphs Processor Unit

IBN – Intent-based Networking

IBSSI - In-Backhaul Support for Service Intelligence

LP - Linear Programming

LSTM – Long-Short Memory

MAUs – Match-Action Units

MANO – MANagement and Orchestration

MTERM – Multi-Timescale Edge Resource Management

ML – Machine Learning

MLOps – Machine Learning Operations

NFR – Non-functional requirement

NI – Network Intelligence

NIF – Network Intelligence Function

NIO – Network Intelligence Orchestrator

NIP – Network Intelligence Plane

NIS – Network Intelligence Service

NSaaS – Network Slice as a Service

NWDAF – Network Data Analytics Function

PHV – Packet Header Vector

PNF – Physical Network Function

QoE - Quality of Experience

QoS - Quality of Service

RAN – Radio Access Network

RIS - Reconfigurable Intelligent Surfaces

RISC – RIS Control

RL - Reinforcement Learning

RNN – Recurrent Neural Network

SDO – Standard-Defining Organization

SL - Supervised Learning

SLA - Service Level Agreement

SLMANO - Self-Learning MANO

TC - Traffic classification

TP - Traffic Prediction

VNF – Virtual Network Function

WLAN - Wireless Local Area Network

Executive Summary

This is the second public deliverable of WP2 of the DAEMON project. It builds upon the material of the previous deliverable of WP2, i.e., D2.1, and on activities and results achieved during the first iteration of the project in WP3, WP4 and WP5. As a result, the document describes the following content.

First, it provides significant updates to the functional and non-functional requirements of the eight NI-assisted functionalities tackled by DAEMON. These updates are mainly driven by the feedback provided by the implementation and evaluation activities of the project to date, which unveiled the need for adjustments to the original requirements or the addition of completely new ones.

Second, it defines novel and equivalent functional and non-functional requirements for a NI plane, which the project proposes to complement existing network planes (e.g., user, control, management), and which lay the foundations to the modelling of the NI architectural framework targeted by the project.

Third, based on the aforementioned requirements, it draft a preliminary yet complete design of the DAEMON's NI plane, detailing its organization, its interaction with complementary blocks (e.g., traditional networking planes and machine learning operations), and its functioning in terms of NI instance coordination.

Fourth, set forth a first set of project's guidelines for the pragmatic design of NI. These span along two directions. First, a NI design tailored to the needs of B5G network management, orchestration and control. Second, a NI design that does not stretch the adoption of complex data-hungry black-box models for all possible any NI use cases; instead, it considers using more traditional, simpler, or interpretable models across the functionalities addresses in the project.

Fifth, it summarizes the current status of a continuous review of the literature on NI and machine learning for mobile network management. This is a live effort that aims at keeping the project activities up to date with the current state of the art in scientific research, as well as to ultimately produce a complete survey and taxonomy of solutions for NI design and integration.

As a result, the content of this document will be the basis on which the rest of the second iteration of the project activities will build upon. We expect that the updated design implementation of NI-assisted functionalities throughout the second iteration will be driven by the requirements set forth in this deliverable. Similarly, the alignment of the NI algorithm for orchestration via the NI plane will be carried out respect to the representation models presented in this deliverable, Finally, all evaluations will be aimed at verifying that the proposed solutions meet the project KPIs in terms of performance, but also the functional and non-functional requirements and NI design guidelines listed in this deliverable.

1 Introduction

The DAEMON project's Work Package 2 (WP2) lays the technical foundations for all technical activities in the action. It targets a twofold objective: (i) design an overall architecture for the harmonized integration of Network Intelligence (NI) in B5G systems; and (ii) develop a knowledge-base and rigorous methodology for the development of AI tools assisting NI functionalities.

By doing so, WP2 contributes to pursue the following objectives within the DAEMON project.

- ***Objective 1.1***. To enable and drive the coordination and cross-compatibility across NI deployed in different network domains that are operating at different timescales.
- ***Objective 1.2***. To enable NI deep into the network infrastructure.
- ***Objective 3.1***. To adjust AI techniques to the specific necessities of the network environment and operations, and to develop novel AI hybrid approaches.
- ***Objective 3.2***. To introduce appropriate and tailored cost functions for the networking context that can be used for training AI techniques.
- ***Objective 3.3***. To develop novel AI techniques that can dynamically adapt to available network resources, by trading-off accuracy with, e.g., inference latency or computational complexity.

Deliverable 2.1 (D2.1) [1] of the project laid down the very first contributions of WP2 towards the multiple objectives above, by reporting on the following elements.

- The vision of a NI architectural framework that enables a more systematic integration of NI in the B5G infrastructure, while staying fully aligned with emerging designs in standardization. See Section 2 of D2.1.
- The presentation of eight NI use cases that tackle various key functionalities of mobile network operation, mapped into the three categories of Network Planning, Network Diagnosis and insights, and Network Optimization and Control, according to the main B5G AI/ML use cases defined by 5GPPP [2]. These are the main NI case studies addressed by the DAEMON project. See Section 3 of D2.1.
- The definition and initial analysis of the functional and non-functional requirements of all NI use cases mentioned before, according the recommendation in ISO/IEC/IEEE 29148:2018 [3]. See Section 4 of D2.1.
- A review of existing multi-domain and multi-timescale frameworks and architectures, including a summary of the different aspects that characterize closed-loop NI operations such as data management, control, and training, and the associated ML workflows. This allowed pinpointing gaps in the current solutions, where the DAEMON architecture is bringing innovation. See Section 5 of D2.1.

In this second public deliverable of the project's WP2, we build on the material of D2.1 and describe the substantial progress attained by the project during the first half of its lifetime towards the objectives outlined before. Specifically, Deliverable 2.2 (D2.2) presents the following new content and contributions.

- We provide significant updates to the functional and non-functional requirements of the eight NI-assisted functionalities tackled by DAEMON. These updates are mainly driven by the feedback provided by the implementation and evaluation activities of the project to date, which unveiled the need for adjustments to the original requirements or the addition of completely new ones.
- We define novel and equivalent functional and non-functional requirements for a NI plane, which we propose to complement existing network planes (e.g., user, control, management), and which lay the foundations to the modelling of the NI architectural framework targeted by the project.
- Based on the aforementioned requirements, we draft a preliminary yet complete design of the DAEMON's NI plane, detailing its organization, its interaction with complementary blocks (e.g., traditional networking planes and machine learning operations), and its functioning in terms of NI instance coordination.
- We set forth a first set of project's guidelines for the pragmatic design of NI. These span along two directions. First, a NI design tailored to the needs of B5G network management, orchestration and control. Second, a NI design that does not stretch the adoption of complex data-hungry black-box models for all possible any NI use cases; instead, it considers using more traditional, simpler, or interpretable models across the functionalities addresses in the project.
- We summarize the current status of a continuous review of the literature on NI and machine learning for mobile network management. This is a live effort that aims at keeping the project activities up to date with the current state of the art in scientific research, as well as to ultimately produce a complete survey and taxonomy of solutions for NI design and integration.

The first three contributions above are substantial steps towards Objectives 1.1 and 1.2, whereas the last two contributions are in the direction of addressing Objectives 3.1, 3.2 and 3.3 of the project.

1.1 Connecting the first and second iterations of the DAEMON project

In order to contextualize the content of the deliverable with respect to the progress of the DAEMON project, we position D2.2 with respect to the overall work plan schedule project. Figure 1 displays the original Gantt diagram of the project, which includes three iterative phases that are highlighted in the figure. Each iteration encompasses the phases of (i) design of NI framework and models, mainly carried out in WP2, (ii) implementation of NI-assisted functionalities based on the design above, mainly carried out in WP3 and WP4, and (iii) evaluation of NI-assisted functionalities in dependable settings, mainly carried out in WP5. The concept for the iterative nature of the work plan is that each iteration paves the road and feeds the following one, creating a flexible structure that allows identifying and understanding issues emerging in the developed solutions, and addressing them in the following iteration.

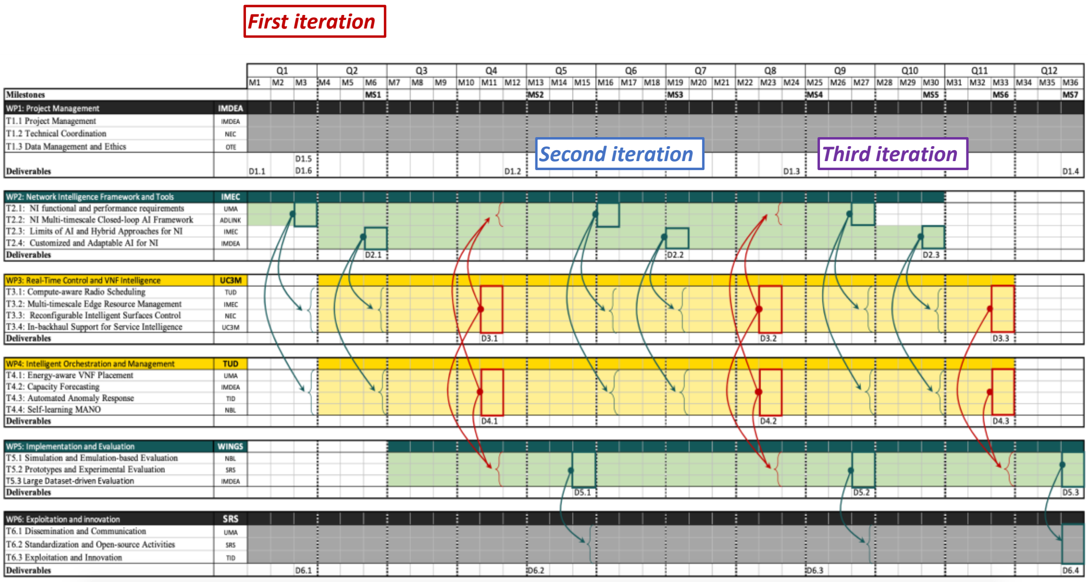


Figure 1. *Gantt diagram of the DAEMON project, with the three iterations of the work plan highlighted.*

1.1.1 Targets of the first and second iterations

Clearly, as iterations occur at different moments during the project lifetime, they also have their own specificities in terms of achievements and duration. In particular, the technological target expected at the end of each iteration is pushed forward as the project progresses. Also, the duration of the iterations is progressively reduced since the required adjustments become less foundational as the project advances.

In this sense, the first iteration of the project focused on the following foundational aspects for the activities in DAEMON.

- Drafting preliminary requirements for the NI-assisted functionalities.
- Providing early design of the NI algorithms based on the drafted requirements.
- Implementing a first version of the NI-assisted functionalities based on the design above.
- Completing tests to assess the performance and limitations of the implemented functionalities.

The work conducted to complete the tasks above ultimately led to the initial evaluations reported in D5.1 [4], but results of the first iteration are also included in D2.1 [1], D3.1 [5], and D4.1 [6] of the DAEMON project. An important observation is that ***the target of the first iteration was the development and testing of a very first version of the eight individual NI-assisted functionalities that are at the core of the DAEMON action***. The purpose was determining if and to what extent different machine learning tools are useful and relevant to tackling each functionality. However, during the first iteration, we investigated such NI-assisted functionalities in isolation, and did not yet look at their coordination or integration. Indeed, developing sound NI algorithms for each network functionality is a necessary first step before studying how to make such NI algorithms coexist and coordinate in the end-to-end mobile network infrastructure. This also means that the design of the holistic NI orchestration framework was not at the core of the first iteration activities.

The second iteration of the project builds on the results of the first to shift targets to a higher bar. More precisely, having now a first set of operational and validated NI algorithms supporting the eight target functionalities, ***the DAEMON project will focus during the second iteration on (i) improving the NI model design so as to overcome limitations identified in the initial algorithms developed during the first iteration, and (ii) start defining a framework for the coordinated operation of the multiple NI-assisted functionalities***.

The two targets above will be achieved by completing the following tasks during the second iteration, fed by results of the first one.

- Updating the requirements and design of the eight NI-assisted functionalities, based on the novel experience developed during the first iteration by the consortium about the limitations of the initial design. A new version of the NI algorithms shall be devised during the second iteration that meets the new requirements above, again building on the know-how created by the work in WP3 and WP4 in the first project iteration [5] [6].
- Producing a first set of practical guidelines for the design of NI algorithms that are tailored to mobile network environments. Again, these guidelines are derived from the expertise developed by the project partners in the first iteration of DAEMON, and stem from objective analysis of which algorithms proved to work well in supporting the target NI-assisted network functionalities.
- Drafting original requirements for a NI orchestration framework, and producing a blueprint of a NI plane that meets such requirements. Once more, the work carried out during the first iteration will be key to support these activities, as the basic operation of the NI algorithms is now clear, and it can inform the design of an end-to-end NI orchestration solution.
- Aligning the improved NI-assisted functionalities with the NI plane operation, specifically by providing suitable representations of all NI algorithms that are in line with the specifications set forth by the NI plane operation blueprint.
- Evaluating the new improved versions of the NI-assisted functionalities in dependable settings, so as to ensure that they meet the updated relevant requirements in practical network scenarios. Note that the coordinated operation of NI algorithms is not yet a target of the second iteration, although we plan to have an initial proof-of-concept evaluation on the coordination of a pair of NI algorithms.

For the sake of completeness, we briefly anticipate here that the targets of the third iteration of DAEMON will include a final version of all project requirements, the ultimate design of NI-assisted functionalities and a more complete coordination of NI instances of DAEMON. Full details that will necessarily consider the progress achieved by the consortium during the second iteration will be provided in Deliverable 2.3 (D2.3) of the project.

1.1.2 Role of Deliverable 2.2 across iterations

The present deliverable acts as the connecting document between the first and second iterations of the project. Namely, ***D2.2 uses the conclusions and results of the first iteration to pave the way for the activities of the second iteration***.

With respect to the targets of the second iteration of the DAEMON project listed in Section 1.1.1 above, this document covers the earliest tasks in the logical flow of activities. Indeed, by cross-checking the list above with the contributions in the present deliverable at the beginning Section 1, it is apparent that D2.2 tackles (i) the updating of NI-assisted functionality requirements, (ii) the creation of a set of guidelines for a pragmatic NI design, (iii) the drafting of new requirements for the NI orchestration, and (iv) the provisioning of a NI plane *blueprint*, i.e., a high-level view of the organization of a novel network plane dedicated to end-to-end NI coordination. These are all the targets of the second iteration that directly derive from the work carried out during the first iteration.

Also, the content of this document will be the basis on which the rest of the second iteration will build upon. More precisely, the updated design implementation of NI-assisted functionalities throughout the second iteration will be driven by the requirements set forth in this deliverable. Similarly, the alignment of the NI algorithm for orchestration via the NI plane will be carried out respect to the representation models presented in this deliverable, Finally, all evaluations will be aimed at verifying that the proposed solutions meet the project KPIs in terms of performance, but also the functional and non-functional requirements and NI design guidelines listed in this deliverable.

1.2 Relationship to the other deliverables of DAEMON

Based on the discussion in Section 1.1.2, the relationship of D2.2 with the other project deliverables of the first and second iteration is fairly clear. Specifically, this deliverable relates to the following ones.

- ***D2.1***. This document builds upon the requirements defined in D2.1 [1] for the eight NI-assisted functionalities targeted by DAEMON, extending and updating them. Besides this, it also makes the vision for a NI orchestration framework set out in D2.1 [1] more practical, by defining precise requirements and providing a first design of the DAEMON's proposal for an NI plane.
- ***D3.1 and D4.1***. This document takes into account the results of the implementations of NI-assisted functionalities presented in D3.1 [5] and D4.1 [6] to update and extend the functional and non-functional requirements of those functionalities. Also, the representation of NI algorithms in the NI plane, which is discussed as part of the NI plane proposal, is strongly influenced by the way the NI algorithms operate across network functionalities in the two deliverables.

- ***D5.1***. This document considers the outcome of the initial performance evaluations reported in D5.1 [4] as part of the information reviewed to update and broaden the requirements for all functionalities.
- ***D3.2 and D4.2***. These upcoming deliverables of the project will present journaled and improved NI algorithms that adhere to the updated requirements presented here in D2.2. Moreover, they will provide suitable representations of the NI algorithms that are compatible and can be managed by the initial NI plane blueprint introduced in D2.2.
- ***D5.2***. This upcoming document will include a more thorough performance assessment of the updated NI-assisted functionalities developed in WP3 and WP4, and will make a clear link not only to the project KPIs, but also to requirements listed in this document. Finally, we will strive to have in D5.2, i.e., already by the end of the second iteration of the project, an initial proof-of-concept evaluation on the coordination of a pair of NI algorithms.

1.3 Structure of the document

The high-level structure of this deliverable is summarized as follows.

- Section 2 details the updated requirements, both functional and non-functional ones, for the eight NI-assisted network functionalities targeted by DAEMON. Also, it introduces the original functional and non-functional requirements for the NI plane intended to orchestrate NI instances in the end-to-end mobile network architecture.
- Section 3 presents the DAEMON's initial proposal for a NI plane that meets the requirements laid out in Section 2, detailing its internal organization and its relationships with existing functional blocks like network management and orchestration (MANO) or machine learning operations (MLOps). This section also presents a model for the representation of NI algorithms and operation in the NI plane, and it includes a practical example of how such a representation can be employed to coordinate different NI instances.
- Section 4 outlines two sets of guidelines for NI design. The first set concerns recommendations on how to tailor AI models for operation in mobile network environments and how to best match the expectations of operators in terms of automated infrastructure management. The second set concerns a discussion of contexts where complex data-hungry models based on deep learning approaches are not necessarily the best choice for network automation, and the DAEMON project results actually show that more traditional or hybrid solutions provide clear advantages.
- Section 5 discusses a live survey of research in NI-assisted network functionalities, introducing the methodology adopted by DAEMON to carry out a comprehensive literature review along the lifetime of the project, and summarizes the current status of the survey of NI solutions for the eight functionalities targeted by the project.
- Section 6 draws the conclusions of the work presented in the deliverable.

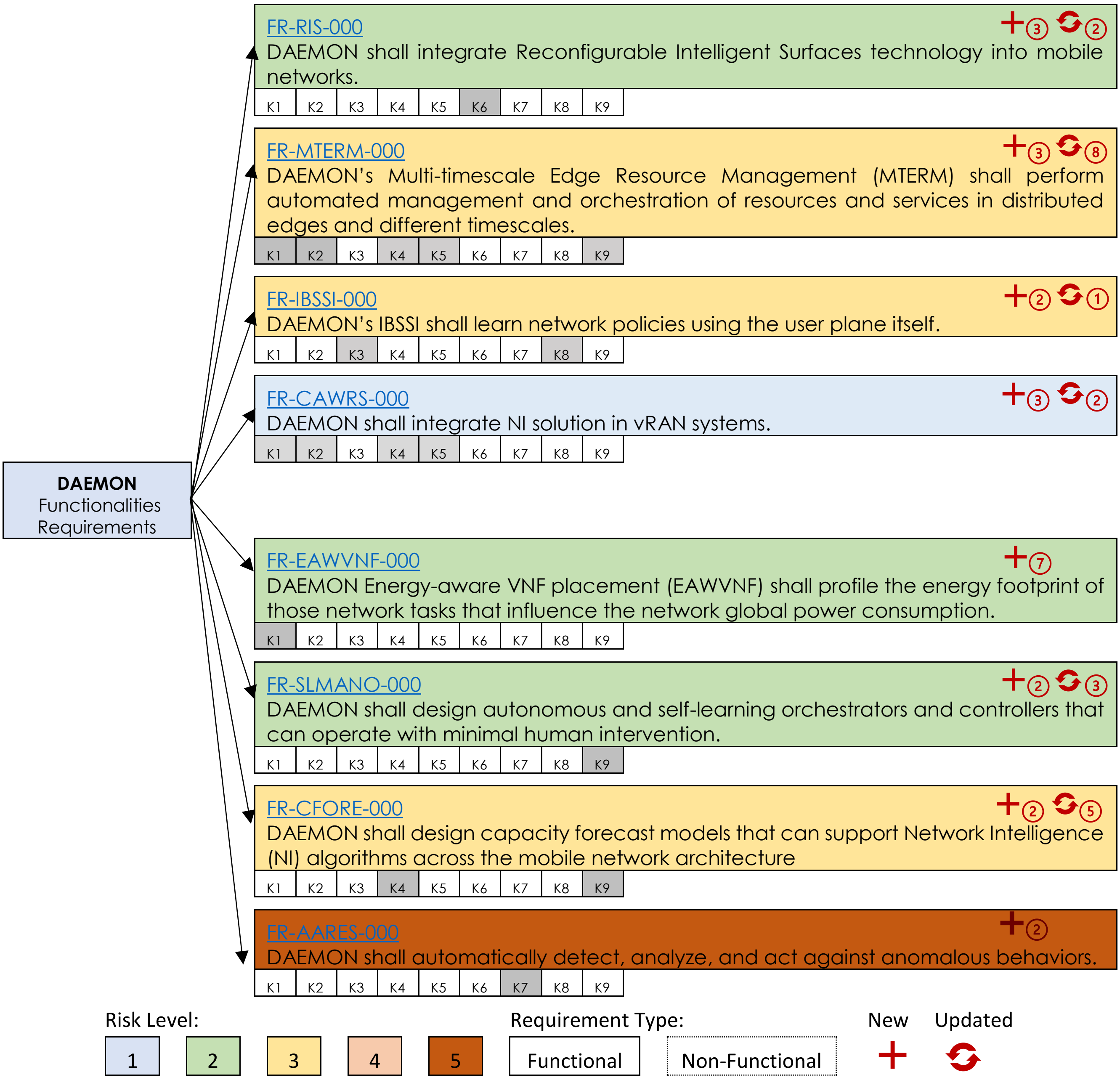
In addition, this deliverable includes two appendixes, presented next.

- Appendix A details the full list of requirement trees for each NI-assisted functionality, as well as for the NI plane. These tree structures are too long to be included in the main body of the document, but they complement the content in Section 2.
- Appendix B reports the complete taxonomy table of the related works studied by the project as part of the literature survey. Again, the table is too large to be included in the main body of the document, but it completes the discussion in Section 5 of the deliverable.

2 Updated Network Intelligence functional requirements

One of the tasks of DAEMON is to determine the functional and non-functional requirements associated to eight NI-assisted functionalities targeted by the project. These span networking problems at different timescales (i.e., real-time control versus orchestration) and scopes (i.e., management/control plane versus user plane – e.g., VNF – intelligence). The main goal of this task is therefore to produce a clear set of requirements that these network functionalities are expected to meet, so as to drive their design and close gaps in the current implementations of the target functionalities. In addition, similar requirements are laid out for the framework that is deemed to coordinate the NI instances, according to the project vision set forth in the Description of the Action (DoA) and presented in Section 2 of D2.1 [1].

In this section, we first present in Section 0 the updated requirements for each NI-assisted functionality based in the experience and results obtained in the first iteration of the project, and discuss how they close existing gaps. Then, in Section 2.2, we introduce a first version of the original requirements for the NI plane proposed by DAEMON for NI instance orchestration.



FR-RIS-000
DAEMON shall integrate Reconfigurable Intelligent Surfaces technology into mobile networks.

FR-MTERM-000
DAEMON's Multi-timescale Edge Resource Management (MTERM) shall perform automated management and orchestration of resources and services in distributed edges and different timescales.

FR-IBSSI-000
DAEMON's IBSSI shall learn network policies using the user plane itself.

FR-CAWRS-000
DAEMON shall integrate NI solution in vRAN systems.

FR-EAWVNF-000
DAEMON Energy-aware VNF placement (EAWVNF) shall profile the energy footprint of those network tasks that influence the network global power consumption.

FR-SLMANO-000
DAEMON shall design autonomous and self-learning orchestrators and controllers that can operate with minimal human intervention.

FR-CFORE-000
DAEMON shall design capacity forecast models that can support Network Intelligence (NI) algorithms across the mobile network architecture

FR-AARES-000
DAEMON shall automatically detect, analyze, and act against anomalous behaviors.

Figure 2. *First-level structure of the functionalities' requirements tree of the DAEMON project.*

2.1 Requirements for NI functionalities

The methodology followed for gathering and documenting requirements in DAEMON was presented in section 4.1 in D2.1 [1]. We split requirements by network functionalities into eight clusters: **RISC** (Reconfigurable intelligent surfaces control), **MTERM** (Multi-timescale Edge resource management), **IBSSI** (In-backhaul support for service intelligence), **CAWRS** (Compute-aware radio scheduling), **EAWVNF** (Energy-aware VNF placement), **SLMANO** (Self-learning MANO), **CFORE** (Capacity forecasting), and **AARES** (Automated anomaly response). In D2.1, each requirement was described, including the information recommended in the norm ISO/IEC/IEEE 29148:2018 [3] and assigned to one of the clusters.

During the last year, the requirements were refined based on the outcomes of the first project iteration in D3.1, D4.1, and D5.1. We have tracked the changes, which have resulted in the following overview: 18 out of 81 requirements were updated, 24 new requirements were included, and 12 were reorganized, extracting them from MTERM (cf. Section 4.1 in D2.1 [1]) to be included in a new cluster, named Network Intelligence Plane (NIP), which will be presented in Section 2.2.

We have updated the tree-shaped representation provided in D2.1 of the set of requirements to incorporate additions and important updates. Figure 1 depicts those changes in the first level of the tree. Each requirement is colored considering the risk assessment, and the KPIs addressed by each requirement are highlighted at the bottom of each box. We represent both functional or non-functional requirements, which can be visually distinguished through a dotted (for non-functional) or continuous (for functional) line. Also, we added information about the number of updated and new requirements added to each cluster during this second year of the project. We provide at the bottom of the tree the whole semantic code used to represent the requirements tree. The detailed three with all the deeper levels is presented in Appendix A.

The following sections describe the main updates for each of the eight requirement clusters, explaining the motivation for such updates and how there are linked to the outcomes of the first-year iteration of the project and documents D3.1/D4.1/D5.1 [5] [6] [4].

2.1.1 Reconfigurable Intelligent Surfaces Control (RIS)

As a result of the activities carried out in WP3 and WP5, reported in D3.1 [5] (cf. Section 6) and D5.1 [4] (see Activity A23 in Section 4.7), we decided to adopt a more experimental approach to developing NI for Reconfigurable Intelligent Surfaces. As a result, a new testbed (Testbed T9 in D5.1) is being built for this purpose, which has motivated a substantial update in the set of requirements first reported in D2.1 for this functionality. The most important changes are the following.

A new functional requirement FR-RIS-003 has been defined to develop a RIS platform that is highly modular. This requirement will enable us to experiment with RIS deployments with different amount of reflective area, and hence, we will be able to evaluate the scalability of the NI developed within WP3 as a function of the available reflective cells in the RIS.

In addition, a better understanding of the practical limitations of nowadays low-consuming micro-control hardware motivated us to relax requirement NFR-RIS-001, which suggests a re-configuration time constraint of 100ms instead of 1ms. This modification enables us to focus on developing low-cost and low-consuming RIS-controlling NI, which otherwise would require overly complex (and energy-consuming) circuitry. Aligned with this modification, we created a new requirement NFR-RIS-002, which sets a maximum power consumption for the RIS of 100mW.

Gap analysis

In the related literature, a number of theoretical activities have analyzed the use of conventional optimization approaches and some machine learning methods to optimize the performance of RIS deployments. However, little experimental research exists in the topic, and it is the object of this project to fill this gap. This includes developing NI that respects the constraints of a real system, both in terms of scalability, power consumption, and re-configuration rate.

In the following, we highlight the most relevant requirements related to this NI functionality, particularly those that involve an innovation over the state of the art.

The following table summarizes the connection between the proposed requirements and the existing gaps for the first layer of the requirements tree. The full description of the complete tree is presented in Appendix A.

Table 1. *Summary of RIS requirements and related gap.*

Functional Requirement ID	Description	Gap
FR-RIS-000	DAEMON shall integrate Reconfigurable Intelligent Surfaces (RIS) technology into mobile networks.	Current SDO specifications do not integrate control or data plane features of Reconfigurable Intelligent Surfaces. Very recently, led by partners in DAEMON consortium, ETSI created an industry specification group (ISG) to this end.
🔄 FR-RIS-003	RIS units shall support more than one user concurrently	Most of the academic research focuses on 1-user RIS systems.
+ FR-RIS-004	RIS should be modular and enable a variable number of reflective cells.	Current RIS designs and existing solutions are monolithic planar structures and provide little flexibility to scale the amount of reflective surface.
NFR-RIS-000	RIS should aid to increase wireless capacity (bits/m2) by m2.	Most of the literature achieves these gains theoretically, but not empirically as envisioned by DAEMON.
🔄 NFR-RIS-001	Re-configuring all the components in a RIS must be achieved with 100 ms.	Most of the literature ignore this performance metric.
+ NFR-RIS-002	The (non-RF) electronic equipment required to control a RIS must consume less than 100 mW.	Electronic components devoted to control reflective cells in a RIS usually consume a non-negligible amount of energy.

2.1.2 Multi-Timescale Edge Resource Management (MTERM)

After a careful discussion regarding the architectural design of a Network Intelligence Plane (which we present in Section 3), we noticed that several of the functional and non-functional requirements written during the first iteration for the Multi-timescale Edge Resource Management (MTERM) functionality were more related to the Network Intelligence Plane (NIP) than to the functionality itself. Therefore, we shifted and refined those requirements into the newly created requirements for the NI plane. Such requirements are described in Section 2.2.

Previous requirements specific to MTERM were merged into a more general root requirement (FR-MTERM-000) since they were referring to the same characteristic of this functionality. Moreover, we grouped together several requirements related to monitoring into a new one, i.e., FR-MTERM-004. Finally, all the requirements related to the management and monitoring of NI were moved to the requirements of the Network Intelligence Plane. In detail, this is the complete list of changes to MTERM requirements.

- We merged three requirements (previous FR-MTERM-000, FR-MTERM-002, and FR-MTERM-003) into the new FR-MTERM-000.
- We refined twelve requirements and moved them to the requirements of the Network Intelligence Plane.
- We grouped three requirements related to the monitoring of resources and energy consumption (previous FR-MTERM-003.02, FR-MTERM-003.03, and FR-MTERM-004) into a new monitoring requirement (FR-MTERM-004) and provided more details.
- We deleted seven requirements (previous FR-MTERM-003.00, FR-MTERM-003.01, FR-MTERM-009, FR-MTERM-014, FR-MTERM-014.00, FR-MTERM-016, FR-MTERM-019) that were found to be obvious, redundant, or not relevant.
- We added two more requirements that allow the functionality to interact with the Network Intelligence Plane and to span several domains.

Gap analysis

To update some of the requirements of the MTERM functionality, we analyzed the gaps in current standardization bodies related to edge management such as NFV MANO, MEC, and Open-Source MANO (OSM). Those gaps are reported in Appendix B of D2.1.

One of the most frequent gaps is that the analyzed frameworks are not ready to support AI/ML functions in the sense that they do not consider the closed-loop nature of network intelligence. Such closed-loop control has as a main pillar in monitoring data ingestion. For example, in Activity A11, which is reported in D3.1(Section 5.1) [5] and D5.1 (Section 4.4.2) [4], we show how much communication and computational delays are contributing to the overall edge service response time, when collecting monitoring data from distributed edges. Since this monitoring data comes from multiple sources, AI/ML techniques could help in the data pre-processing phase and in reducing such data's dimensionality.

Moreover, historical data can be used to build predictive AI/ML modules that allow improving MANO operation of service relocation towards achieving service continuity and the required service quality.

As suggested in state-of-the-art literature, several use cases for network intelligence provide data analytics to decision-making algorithms, with the goal of improving the quality of the decisions. An example of that is shown in Activity A12 (Section 5.2 of D3.1 [5] and Section 4.4.3 of D5.1 [4]), where we trained and tested several ML-based WLAN performance prediction modules that assess the behavior of a WLAN using a given configuration. Based on this prediction, a WLAN controller can evaluate if a network configuration heavily affects the throughput of the connected users or not. Unfortunately, due to the limitations of current frameworks, such use cases are functional in very limited conditions, which demands a broader integration with AI/ML workflows.

Regarding MANO operations, some of the proposed works follow a reactive approach that may cause delayed operations. Therefore, there is a need to switch to more proactive ones, characterized by automation and intelligence in operations of orchestrating both services and resources. Some initial results of this shift were shown in Activity 16 (Section 6.2.1 of D4.1 [6] and Section 4.4.7 of D5.1 [4]), where the traditional reactive scaling approach was compared to a Deep Reinforcement Learning (DRL) agent based on Q-Learning. The results shown that the ML-based scaler performs better than the reactive one in terms of mean number of created VNFs and lower SLO violations. The main reason for this performance is that the ML-based scaler can anticipate the changes in workload, while the reactive scaler reacts once the change in workload is perceived.

Furthermore, if we consider network intelligence to be deployed in different segments, there must be a coordination between such segments to achieve system stability (e.g., resolve conflicting policies). Finally, when considering the interaction between the Network Intelligence Orchestrator and the network management framework, the latter should expose information of their intelligent functions to facilitate their management.

Table 2 summarizes the connection between the proposed requirements and the existing gaps for the first layer of the requirements tree. The full description of the complete tree is presented in Appendix A. In this table, we make use of two different terms for referring to NI instances: (i) the complex NI instances are referred to as NI services (NIS), which are composed of multiple atomic (ii) NI functions (NIF). This terminology is one of elemental concepts introduced in the project, and it is formally defined later in Section 3.2 of this document.

Table 2. *Summary of MTERM requirements and related gaps.*

Functional Requirement ID	Description	Gap
↻ FR-MTERM-000	DAEMON's Multi-timescale Edge Resource Management (MTERM) shall perform automated management and orchestration of resources and services in distributed edges and different timescales.	Current frameworks do not provide automation in the form of flexible and dynamic NFV management and orchestration; Current frameworks do not coordinate intelligence or resources across different network segments and timescales.
↻ FR-MTERM-004	DAEMON's MTERM shall continuously perform multi-timescale monitoring of resources (e.g., computing and network), data traffic and mobility pattern of users, as well as the energy consumption of network services and edge platforms. The monitoring is aided by AI/ML, providing data dimensionality reduction.	Current frameworks do not incorporate real-time data analytics.
↻ FR-MTERM-006	DAEMON's MTERM shall use NIFs and NISs to support orchestration of edge resources.	Current frameworks do not incorporate real-time data analytics
+ FR-MTERM-020	DAEMON's MTERM shall coordinate the decisions between different edges domains and timescales.	Current frameworks do not coordinate intelligence or resources across different network segments and timescales.
+ FR-MTERM-021	DAEMON's MTERM shall expose information of their NIFs (e.g., CPU/GPU consumption, accuracy, timescale, input data format) to the Network Intelligence Orchestrator to facilitate their management.	Current frameworks do not support the lifecycle management of AI/ML-based functions.
FR-MTERM-007	DAEMON's MTERM shall provide on-the-fly automated reconfiguration of VNFs.	Current frameworks do not provide automation in the form of flexible and dynamic NFV management and orchestration.

2.1.3 In-Backhaul Support for Service Intelligence (IBSSI)

Research Activity A18, which is reported in Section 4.5.1 of D5.1 [4], targets in-backhaul learning and builds up on a thorough literature study that finds that all the existing works perform only the inference phase of the machine learning process into programmable user planes. The computationally expensive training phase and the encoding of the model into a specific hardware target are both performed in the control plane. The different proposals differ in the family of machine learning models considered and in the nature of the targeted programmable hardware. Our comparative evaluation of such proposals showed that Random Forest models are a promising candidate for the deployment of machine learning into the user plane. The follow-up of our research, which focused on P4 programmable switches based on PISA architecture, has targeted the encoding and actual implementation of Random Forests into PISA architecture during the second year of the project. Based on current results of such implementation, we are adding two new Non-Functional Requirements (NFR) as follows.

First, we add NFR-IBSSI-000, stating that Network Intelligence algorithms should be adapted to PISA architecture. This requirement deals with the necessity to encode machine learning models into the switch's pipeline, which is constituted by Match-Action Units (MAUs). An ad-hoc encoding of the model is needed to map the machine learning algorithm into a sequence of actions (i.e., simple logical/mathematical operations) that are performed according to the hit (or miss) of one or more key values that can be matched against the switch's memory tables.

Second, we introduce NFR-IBSSI-001, which considers that Network Intelligence algorithms should be resource-prudent. This requirement stems from the fact that the memory of such programmable switches is a very limited resource. For this reason, the design of models to implement into the MAUs must take into account strategies to minimize the memory footprint of the resulting encoded model. Such strategies include, and may not be limited to, features approximation and compression.

We plan to develop a complete solution for In-backhaul learning that meets also the two new requirements and to report its design and results of performed evaluations in the upcoming deliverables D3.2, and D5.2.

Gap analysis

In-backhaul learning represents an innovative research area that is receiving growing interest because of emerging programmable network devices that have the potential to perform machine learning inference at line-rate. Despite the obstacle that represents the limited amount of memory and mathematical operations of such hardware, a comprehensive solution for in-switch machine learning would allow to meet the very stringent latency requirements of a set of use cases such as traffic classification and anomaly detection. In this context, DAEMON's contribution aims at closing the gap in the literature by proposing a viable in-switch machine learning system that fits the architecture of such programmable pipelines and optimizes its memory footprint. We are currently working on models that can meet the two new requirements and to report the results of design and evaluations in D3.2 and D5.2.

The following table summarizes the connection between the proposed functional requirements and the existing gaps for the first layer of the requirements tree. The full description of the complete tree is presented in Appendix A.

Table 3. *Summary of IBSSI requirements and related gaps.*

Functional Requirement ID	Description	Gap
🔄 FR-IBSSI-001	DAEMON's in-backhaul support for service intelligence (IBSSI) shall provide Intelligence-as-a-Service to vertical 3rd parties.	DAEMON will provide algorithms for the execution of network intelligence directly related to the vertical service (e.g., video analytics directly in the u-plane) and will allow the efficient and secure resource provisioning through the usage of solutions based on e.g. distributed ledger platform.
FR-IBSSI-002	DAEMON's IBSSI shall integrate Network Intelligence within programmable switches.	Programmable user planes are starting to be leveraged for network telemetry functionalities. However, these are limited to data collection and pre-processing, which are then fed to NI located in the control plane to take network management decisions. There is a gap in our understanding of what portion of the decision process can be moved to the switches directly, at line rate and avoiding the delay of interacting with the control plane. DAEMON's activities aim at closing such a gap.

2.1.4 Compute-Aware Radio Scheduling (CAWRS)

These research activities are performed within the WP3 context, as reported in sections 4.1 and 4.2 of D3.1 [5], and are dedicated to design, from an experimental perspective, novel technologies to enable an efficient usage of non-deterministic computing platforms for RAN virtualization. First, enabling reliable DU virtualization over computing platforms, to then pursue efficiency in heterogeneous environments with a huge diversity in the nature of the computational resources.

Given the importance of time constraints on virtualized DUs, there is a need of including network intelligence solutions to effectively offload heavy tasks performed by the DU, such as decoding operations. For that reason, we have included a new requirement FR-CAWRS-001 standing for the addition of predictive HARQ mechanisms for vRAN solutions. These mechanisms take advantage of the information provided by the decoder to make predictions about the decodability of transport blocks.

Furthermore, the heterogeneous nature of virtualization platforms, together with the disaggregation of current RAN approaches, demands network intelligence scheduling solutions. More precisely, we require allocation solutions for radio and computing resources where resources are heterogeneous and must operate in real-time timescale. Consequently, the requirement FR-CAWRS-002 has been included to better state the identified needs.

The inclusion of new requirements implies the addition of design constraints. NFR-CAWRS-001 design constraint states the inference time at which scheduling solutions must operate to provide real-time inference. Then, NFR-CAWRS-002 asserts the main goal of the scheduling solutions: addressing spectral efficiency given the computing capacity constraints. Finally, NFR-CAWRS-003 defines the accuracy and the false positive rate that HARQ inference mechanisms must accomplish.

Gap analysis

Although predictive HARQ mechanisms are widely studied, to the best of our knowledge, there are no experimental designs for distributed units including such prediction-based mechanisms.

Computing and RAN resources scheduling is a widely studied topic within the research community, but the novel architectural designs bringing disaggregation, together with the increasing heterogeneity of the computing platforms, bring us the opportunity to design a joint RAN and computing resources scheduling network intelligence algorithm.

The following table summarizes the connection between the proposed requirements and the existing gaps for the first layer of the requirements tree. The full description of the complete tree is presented in Appendix A.

Table 4. *Summary of CAWRS requirements and related gaps.*

Functional Requirement ID	Description	Gap
+ FR-CAWRS-001	DAEMON's compute-aware radio scheduling (CAWRS) shall integrate predictive HARQ.	Current solutions do not incorporate this solution for preserving synchronization when using virtualized RAN.
+ FR-CAWRS-002	DAEMON's CAWRS shall integrate intelligent algorithms to allocate radio and computing resources in real time.	Current frameworks do not incorporate joint radio and computing resources allocation.
NFR-CAWRS-000	DAEMON's CAWRS shall have reaction times below 10s.	Current approaches do not incorporate this stringent requirement.

2.1.5 Energy-Aware VNF Control & Orchestration (EAWVNF)

The research reported in Activity A15 of deliverable D5.1 [4] of the DAEMON project was targeted at investigating energy-aware VNF placement as part of the network orchestration. Specifically, the research focused on testing the reduction in both dynamic (due to computation) and base (idle) energy consumption in a simulated edge-based infrastructure serving workloads provided by the Shanghai Telecom dataset [7] [8]. To this end, we defined *Green fit,* an energy-aware task allocation policy that considers energy consumption due to auto-scaling, and a proactive horizontal auto-scaling solution. Those new functionalities regarding the energy footprint of VNFs scaling have been included as requirement in FR-EAWVNF-006. This requirement has been split into measuring the energy footprint of VNFs vertical scaling EAWVNF-006.00 and measuring the horizontal scaling EAWVNF-006.01.

In a normal scenario, an orchestrator manages the edge nodes by automating the deployment, management, scaling, interconnection, and availability of applications. The master nodes (as there may be more than one) are responsible for orchestrating the workloads between the associated devices (or worker nodes), while master nodes can simultaneously be worker nodes. Once a user request occurs, the orchestrator assigns an application (service) to an edge node, which executes the application packaged in a container (e.g., Docker). The scheduler decides which worker node will run that container according to a certain policy. We developed the *Green Fit* policy, which assigns tasks to the most energy-efficient nodes to minimize energy consumption, thus reducing the dynamic energy consumption. Regarding the base energy consumption, our proactive horizontal autoscaling framework for edge infrastructures is responsible for determining, for a given time window and expected workload (with certain resource requirements), which nodes should be deactivated/activated. The objective is to minimize the number of active nodes while maintaining a high request acceptance rate; the longer the nodes remain off, the lower their base energy consumption. Despite its benefits, switching on and off cloudlets frequently incurs considerable "switching" penalty such as start-up delay and activation energy consumption. These factors are also considered in our model, along with (i) latency and energy consumption due to computation and communication (dynamic energy consumption) and (ii) base energy consumption.

Currently, our auto-scaling framework only considers horizontal scaling, although we plan to extend it to support vertical scaling. In the context of this work, we plan to evaluate the impact of the auto-scaling framework itself on the energy consumption of the system.

In addition, the research reported in Section 4.2 of D3.1 [5] described that the overall throughput of the VRAN systems could be optimized considering the energy consumption due to GPU and CPU utilization, either as a penalizing term for the overall throughput or as a hard constraint of the system. This energy efficiency inclusion in vRANs has been documented in the new requirements FR-EAWVNF-005, NFR-EAWVNF-004, NFR-EAWVNF-005, NFR-EAWVNF-006.

Gap analysis

In the literature, the impact of the auto-scaling mechanism has considered both the delay to activate/deactivate the nodes (that may affect time response) and its associated energy cost. However, the cost of keeping the auto-scaling service running is usually neglected. Likewise, other factors that may affect the energy consumption of both horizontal and vertical auto-scaling should be identified.

The following table summarizes the connection between the proposed requirements and the existing gaps for the first layer of the requirements tree. The full description of the complete tree is presented in Appendix A.

Table 5. *Summary of EAWVNF requirements and related gaps.*

Functional Requirement ID	Description	Gap
FR-EAWVNF-001	DAEMON's energy-aware VNF control & orchestration (EAWVNF) shall measure the energy footprint of VNFs in terms of CPU usage and communication traffic.	Current frameworks do not provide means to reason about the energy footprint of VNFs.
FR-EAWVNF-002	DAEMON's EAWVNF shall measure the impact of hardware resources usage by VNFs in the calculation of the energy footprint.	
FR-EAWVNF-003	DAEMON s EAWVNF shall measure the energy footprint of VNFs migration.	
FR-EAWVNF-004	DAEMON's EAWVNF shall consider how the context of the location of VNFs affects the energy footprint of VNFs.	
+ FR-EAWVNF-005	DAEMON 's EAWVNF shall configure virtualized radio access networks to increase their energy efficiency.	Current solutions configure virtualized base stations without their energy consumption in mind.
+ FR-EAWVNF-006	DAEMON's EAWVNF shall measure the energy footprint of VNFs scaling.	Current frameworks do not provide means to reason about the energy footprint of VNFs while scaling.

2.1.6 Self-Learning MANO (SLMANO)

The research reported in Section 6 of D4.1 [6] and in Activity A16 of deliverable D5.1 [4] was targeted at investigating components of a self-learning MANO. In particular, service initiation (via VNF forward graph embedding) and service scaling were studied.

For the former, a table-based reinforcement learning approach was compared with a deep learning approach for the case that all VNFs need to be placed in one datacenter based on limited knowledge on the status of the datacenter. Both approaches turn out to perform about equally well. In the second year of the project, we plan to study more complex environments such as dynamic environments involving more datacenters where only limited information can be exchanged between datacenters. That is why we added a new requirement FR-SLMANO-007. The outcome of this study will be reported in deliverable D4.2.

For the latter, a reinforcement learning (RL) scaling approach was compared with a scaling method from control theory, where both only take the observed quality of experience (QoE) as input to take scaling decisions. It was shown that the latter outperforms the former. However, in the current state of the art the parameters of the classic control theoretic algorithm need to be meticulously tuned, while the RL approach automatically tunes its parameters. That is why we detailed FR-SLMANO-002 with the additional requirement that, for these classical algorithms, it is needed to study and develop methods to automatically tune such parameters. Results will be reported in deliverable D4.2.

Part of the research reported in Section 3 of D4.1 [6] was targeted at investigating the algorithms provided by DAEMON that make orchestration decisions. These algorithms consider latency/time response/timescale needs of service function chains and at the same time the energy footprint. Thus, it is desired to obtain the tradeoff between time and energy footprint, and that is why we detailed FR-SLMANO-002.00, which explicitly states that the self-learning MANO should take into account energy constraints to be aligned with FR-EAWVNF-000.

Gap analysis

The standards describe MANO frameworks and their interfaces, but not how to automate it. Standards typically do not describe algorithms that implement the MANO functionality.

There are works in the literature that have investigated algorithms for forward graph embedding and scaling with reinforcement learning and traditional approaches, but, as far as we know, always under the assumption that all relevant knowledge is available to the agent that needs to make the decision. In DAEMON we differ from the state of the art by alleviating this assumption. In particular, we feed the decision-making agent only with information that is deemed to be available in real situations.

The following table summarizes the connection between the proposed requirements and the existing gaps for the first layer of the requirements tree. The full description of the complete tree is presented in Appendix A.

Table 6. *Summary of SLMANO requirements and related gaps.*

Functional Requirement ID	Description	Gap
FR-SLMANO-000	DAEMON's Self-learning MANO (SLMANO) shall design autonomous and self-learning orchestrators and controllers that can operate with minimal human intervention.	Current frameworks do not allow a high degree of automation in the management and orchestration of network and compute resources: 1) the setting up and scaling of network services are based on a priori designed rules; 2) the interfaces to access the information upon which NI decisions can be made do not exist.
NFR-SLMANO-000	DAEMON's SLMANO shall design controllers and orchestrators steered by high-level QoE targets and business KPIs (high level intents), rather than strict QoS goals and technical KPIs.	Intent-based networking is Beginning to be discussed in standard bodies (e.g., IETF), but beyond defining what it should be, there are no standards yet.
FR-SLMANO-002	DAEMON's SLMANO shall design controllers and orchestrators that support diverse intent-based objectives combinations provided by application developers, in terms of high-level application properties (possibly unknown at design time).	Intent-based networking is starting to be discussed in standard bodies (e.g., IETF), but there are no standardizations yet beyond defining what it should be.
FR-SLMANO-003	DAEMON's SLMANO shall design controllers and orchestrators that self-converge to stable control loops.	Although stability (i.e., avoiding that positive feedback loops spin out of control) lies at the heart of classical (linear) control theory, it is poorly understood in the context of autonomic computing systems.

FR-SLMANO-004	DAEMON's SLMANO shall design controllers and orchestrators that are trustworthy and explainable, where decisions can be traced back to the key intents that have driven a specific action.	In some areas of artificial intelligence trustworthiness and explainability have been proposed and investigated, but these techniques are not widely applicated in networking yet.
FR-SLMANO-005	DAEMON's SLMANO shall design controllers and orchestrators that are able to report that the systems they control behave unexpectedly, indicating possible need for retraining to cope with unseen or changed dynamics.	To the best of our knowledge, separating normal from abnormal behavior has not been investigated in the context of a SLMANO yet.
⟳ FR-SLMANO-006	DAEMON's SLMANO shall implement mechanisms to detect when learned information becomes stale.	To the best of our knowledge, there are no tests yet to check if the statistics of recent seen data sufficiently differs from the statistics of the data with which the algorithm was trained.
+ FR-SLMANO-007	DAEMON's SLMANO shall be able to gradually adapt to changing environments.	In some areas of artificial intelligence techniques to alter the balance between exploration and exploitation during the lifecycle of a machine learning algorithm are used to adapt to changing circumstances. Yet, these are not widely used in networking problems.

2.1.7 Capacity Forecasting (CFORE)

The results of DAEMON's research activities A21 and A22, targeting capacity forecasting network functionalities and reported in Section 4.6 of D5.1 of the DAEMON project [4], showed how NI based on loss meta-learning can largely benefit anticipatory networking tasks and pave the road for practical Intent-Based Networking (IBN) systems.

In follow-up works carried out during the second year of the project, we are exploring the boundaries of loss meta-learning approaches, so as to understand up to what extent these techniques are applicable across anticipatory networking use cases, but also to investigate what their limitations are. Based on current results of such new analyses, we are updating the requirements for the CFORE functionality by including two new Functional Requirements (FR) as follows.

First, we introduce FR-CFORE-006, stating that loss meta-learning should occur with minimum training time. This FR stems from the consideration that the loss is meta-learned (along with the model parameters) at runtime in the production system: therefore, the initial lack of accuracy of the loss representation determines substantial errors in the predictions, hence significant costs for the operator. Initial explorations of the exact impact of these costs make it clear that the economic penalty for the operator can be significant. Hence minimizing the training time is an important FR when loss meta-learning models are trained from a cold start situation.

Second, we add FR-CFORE-007, which indicates that loss meta-learning shall support losses that combine multiple predictions. This FR is motivated by additional experiments to those performed in year 1 of the project, which showed how the models presented in Section 4 of D4.1 [6] and evaluated in Section 4.6 of D5.1 [4] could not scale to situations where the system performance does not depend on a single prediction but on a composition of multiple forecasting tasks. This problem emerges, for instance, in use cases of admission control over multiple predicted traffic flows, or in network slice brokering. Learning the correct loss function in those situations implies capturing the correlations among the different predictions and the performance metric, which calls for more complex meta-learning tools than those reported in D4.1, and let us extend our list of requirements for CFORE.

Finally, we added two new functional requirements FR-CFORE-004 and FR-CFORE-005. These were in fact indicated as non-functional requirements (NFR-CFORE-001, NFR-CFORE-002) in the previous version of the DAEMON requirements but have been moved as they better fit the former definition. These requirements state that DAEMON capacity forecast models shall (a) be able to learn autonomously their objective/loss function and (b) provide information about their level of accuracy.

Gap analysis

The loss meta-learning approaches developed by the DAEMON project not only represent an extremely innovative tool for NI to support IBN systems, but they also address an open problem in the machine learning community, i.e., learning appropriate losses for regression from experience. Therefore, the whole concept underpinning DAEMON's activities in this direction aims at closing gaps in both the networking and machine learning literatures, and the additional requirements detailed above open the way for novel loss meta-learning solutions for that are more comprehensive than those developed during the first year of the project. We are presently working on models that can meet these new requirements, and expect that those will be reported as part of D4.2 of the DAEMON project.

The following table summarizes the connection between the proposed requirements and the existing gaps for the first layer of the requirements tree. The full description of the complete tree is presented in Appendix A.

Table 7. *Summary of CFORE requirements and related gaps.*

Functional Requirement ID	Description	Gap
FR-CFORE-000	DAEMON's Capacity Forecasting (CFORE) shall design models capable of anticipating the amount of resources needed to accommodate future mobile service demands, so as to support Network Intelligence (NI) algorithms across the mobile network architecture.	Decision-making concerning the capacity that orchestrators and controllers shall allocate in their micro-domain of competence is a key task for the NI operating across the whole network. Yet, current forecasting models target the prediction of network traffic only, meaning that they require a subsequent block that translates the expected demand into a capacity requirement. Capacity forecasting models close such a gap by directly outputting the future capacity information required for automated network management.
FR-CFORE-001	DAEMON's CFORE shall operate at very different timescales.	Current models for mobile traffic forecasting do not consider multi-timescale operation.
FR-CFORE-002	DAEMON's CFORE shall account for monetary costs in order to produce a practical prediction.	Current models for mobile traffic forecasting only target the minimization of the error in the predicted demand and are thus agnostic to monetary costs.
FR-CFORE-003	DAEMON's CFORE shall operate over streaming data.	Current models for mobile traffic forecasting are not designed or optimized for operation on streaming data.
🔄 FR-CFORE-004	DAEMON's CFORE shall provide information about their level of accuracy.	Current models for mobile traffic forecasting do not provide side information about the expected accuracy of their predictions.
🔄 FR-CFORE-005	DAEMON's CFORE shall be able to learn autonomously their objective/loss function.	Current models for mobile traffic forecasting require that a target loss is provided a priori, e.g., designed based on expert knowledge. However, in many network management use cases, the exact relationship between the prediction and the system performance is not fully known in advance, which calls for models capable of learning such a relationship and tune the prediction to it.

2.1.8 Automated Anomaly Response (AARES)

The research activities performed within the Task 4.3 in WP4 (reported in Section 5. of D4.1 [6]) were focused on building specific NI solutions to automatically detect unexpected behaviors that emerge in the network and trigger relevant signals. While working on the features for IoT anomaly detection, we found out that the distribution of the features of some particular devices changed over time. This is especially important in the first step of the IoT anomaly detection pipeline, since, depending on an initial clustering based on traffic volume, a different model is applied to detect the possible deviances. If a device increases its traffic over time following a seasonal pattern, it could trigger a false alarm for an anomaly.

Therefore, a new functional requirement FR-AARES-004 has been defined that describes the risk of temporal distribution shift in the captured data and that enough historical data needs to be used in order to be able to analyze the possible distribution shift in the data.

The analysis of temporal distribution shift in the data will enable us to find out the typical time segment where we can assume that the data is stationary along with the selection of features that are more invariant to shift. Based on that we will be able to develop robust models with higher stability over time and also the frequency at which we should update them with new data.

Gap analysis

The concept of distribution shift in machine learning is a well-known and studied problem. However, it is still one of the main open and general problems that the community is facing, especially in the anomaly detection task, where distribution shifts can be mistaken as out-of-distribution anomalous samples. Working on the task for mitigating distribution shifts in the anomaly detection task aims at closing the gap in machine learning state-of-the-art. We are currently working on the feature analysis based on this new requirement and expect that those will be reported as part of D4.2 of the DAEMON project.

The following table summarizes the connection between the proposed requirements and the existing gaps for the first layer of the requirements tree. The full description of the complete tree is presented in Appendix A.

Table 8. *Summary of AARES requirements and related gaps.*

Functional Requirement ID	Description	Gap
FR-AARES-001	DAEMON's Automated Anomaly Response (AARES) shall operate at different timescales, depending on the input from the system DAEMON is monitoring.	Each anomaly detection task should take into consideration the requirements in terms of timescale to generate anomaly warnings. For finer granularities, the performance of the models might implicitly decrease, as the time available for the model to produce results also decreases. Such trade-off has not yet been analyzed.
FR-AARES-002	DAEMON's AARES models shall have specific data requirements, including a sizable amount of historical data to establish normal behavior and ground-truth occurrences of anomalies to develop a feasible solution.	The data quality is of paramount importance for the anomaly detection approaches we aim to integrate in DAEMON. Moreover, the lack of expert knowledge brings an extra risk when building data features to train the anomaly detection tools for DAEMON.
FR-AARES-003	DAEMON's AARES shall take into consideration the cost of system monitoring, developing and deploying the anomaly detection models in order to produce a feasible anomaly detection solution.	Usually, solutions derived for anomaly detection do not account for the monetary and energy cost of the proposed algorithms.
+ FR-AARES-004	DAEMON's AARES shall account for a possible temporal distribution shift in unseen data.	Current models for anomaly detection fail to distinguish between true anomalies and temporal shifts on data distribution. Solving this issue is an open problem.

2.2 Requirements for NI management and coordination

The DAEMON project sets forth a vision for end-to-end NI coordination, aimed at ensuring a conflict-free and synergic operation of the many NI algorithms running across schedulers, controllers, and orchestrators in the network. As a first step in the rigorous design of a complete framework for the joint operation of NI instances, we outline a clear set of requirements, both functional and non-functional, for NI coordination.

We name the framework ***Network Intelligence plane (NIP)***, as we conceive it as a novel plane in the mobile network, complementing those already existing in 5G architectures, i.e., the user or data plane, the control plane, and the management plane. The tree of requirements for the NIP proposed by the DAEMON project is organized into five major blocks in terms of functional requirements (FR).

- NI orchestration. The first subtree of NIP requirements concerns the capability of the NI plane to decompose complex NI instances and represent them as a combination of atomic NI elements. This is a paramount requisite for the NIP to have the level of control necessary to orchestrate all network-wide NI-related operations, by handling closed control-loop in different micro-domains. The ultimate goal of NI orchestration is the ability to create, when needed and in an automated manner, an end-to-end Network Intelligence Service that can achieve specific KIPs and meet business needs. We recall that, in the context of the project, we define the complex NI instances as ***NI services (NIS)*** that are composed by multiple atomic ***NI functions (NIF)***. We will provide formal definitions of the NIS and NIF concepts later in Section 3 of this document.
- NI interfaces. The second subtree of NIP requirements targets specifications for the interfaces needed to enable the necessary communication among the different building blocks of the NI plane and existing external elements. For instance, when a NIS is created/composed, training the NIFs that constitute the NIS calls for the creation and deployment of MLOps frameworks, for

which several commercial solutions exist. Similarly, NIS are deployed within network controllers or orchestrators that are managed by traditional MANO frameworks, making interactions of the NIP with those mandatory. Reinventing such MLOps or MANO frameworks is not a sensible choice, and establishing interfaces that allow the DAEMON's NIP to communicate with such existing frameworks makes the solution developed by the project more practical, since it favors its fast integration into existing industry's initiatives. As an additional requisite, these interfaces shall also be compliant with relevant current standardization efforts, e.g., by 3GPP, O-RAN or ETSI.

- NI lifecycle management. The third subtree of NIP requirements sets forth the requisite for the NI plane of handling the management of complete lifecycles of both NIS and NIF. Specifically, once a NIS is released for production, the NIP shall support its onboarding, instantiation, termination, scaling, and state retrieval. The same should happen with the different NIFs that compose each NIS. Note that in the context of NI instances, lifecycle management includes the monitoring of the health of the NIF: this includes typical diagnostic information if the NIF is being used in inference or it is an online learning solution, or other metrics such as the loss and the training loops if the NIF is currently being trained. Moreover, the NIP needs to provide feedback on the NIF performance so that higher-level decisions can be made (e.g., about the need for the model to be updated or replaced).
- NI coordination. The fourth subtree of NIP requirements defines the preconditions about the capability of the NI plane to perform conflict resolution and guarantee overall stability of NI instance operation in an end-to-end fashion, possibly taking advantage of synergies across NI. Coordination of NI can include, but is not limited to, (i) sharing measurement and input data among different NIFs, (ii) arbitration policies in case of two NIFs that share the same sink, that is, the configuration APIs, or (iii) control of system stability among conflicting policies, actions or decisions, e.g., when optimizing a certain objective function at one network domain may be counterproductive to equivalent processed in other domains, hence jeopardizing end-to-end stability of the automated network management.
- NI catalogues. The fifth and final subtree of NIP requirements defines the need for the NI plane to be able to access catalogues of both NIS and NIF, which have already been onboarded and feature varied performance and complexity for the same specific network functionality. These catalogues are paramount for the NIP to take informed choices about the most appropriate NIS or NIF to instantiate at a given time instant and in a specific controller or orchestrator, based on, e.g., available computing resources, inference latency requirements, or accuracy constraints.

As far as non-functional requirements (NFR) are concerned, the main specifications that the DAEMON project drafted for the NI plane to date are as follows:

- Support for virtualization environments. Mobile network infrastructures are characterized by a variety of virtualization environments across micro-domains. As a basic example, resource-limited edge platforms employ different virtualization techniques than large and resourceful core network datacenters, which calls for the capability of the NI plane to support heterogeneous virtualization environments for deploying services/applications in distributed domains, hence providing specific maintenance and virtualization-specific policies for orchestration operations.
- Federated multi-domain management. Management-level agreements are necessary for establishing collaboration between orchestration and management entities, and NIFs in different (e.g., edge) domains. Due to the high mobility of users in mobile ecosystems, applications are deployed in distributed way across different edge platforms. Thus, the NI-assisted management and orchestration provided by the NI plane needs to support cross-domain/cross-edge service orchestration for achieving seamless service operation.

Note that the NIP requirements above, both functional and non-functional, concern the NI plane, which in turn optimizes the operation of NI instances across all mobile network architectural (micro-)domains. Therefore, NIP requirements are orthogonal to the functioning of each NI-assisted functionality and can be seen as enablers that concern all KPIs targeted by DAEMON: for this reason, all elements of the NIP requirement tree are marked as relevant to all KPIs targeted by the project. Overall, the requirements presented above account for the feedback coming from the initial algorithmic design described in D3.1 [5] and D4.1 [6], allowing us to streamline the NI architectural framework that fulfils common tasks related to the management of the NI in the network.

For the sake of consistency, we have also created a tree-shaped representation of the set of NIP requirements, in resemblance of the representation of the functionalities' requirements created in D2.1 [1] and updated in the previous Section 0, which is fully disclosed in Appendix A. The following Figure 3 depicts the first level of the tree-shaped representation of the NIP requirements, while the full description of the complete NIP requirement tree is provided in Appendix A.

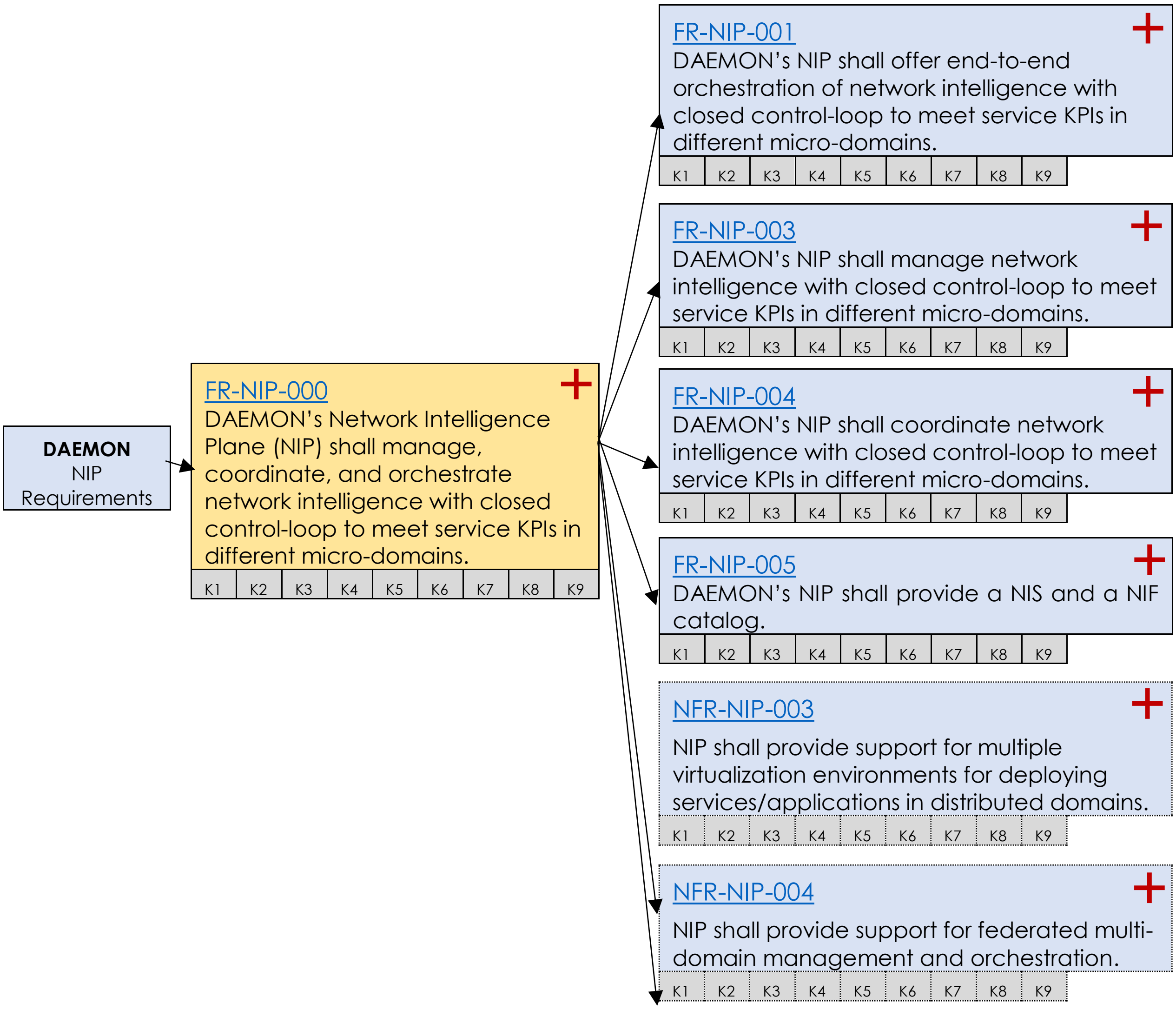

DAEMON
NIP
Requirements
FR-NIP-000
DAEMON's Network Intelligence Plane (NIP) shall manage, coordinate, and orchestrate network intelligence with closed control-loop to meet service KPIs in different micro-domains.
K1 K2 K3 K4 K5 K6 K7 K8 K9
FR-NIP-001
DAEMON's NIP shall offer end-to-end orchestration of network intelligence with closed control-loop to meet service KPIs in different micro-domains.
K1 K2 K3 K4 K5 K6 K7 K8 K9
FR-NIP-003
DAEMON's NIP shall manage network intelligence with closed control-loop to meet service KPIs in different micro-domains.
K1 K2 K3 K4 K5 K6 K7 K8 K9
FR-NIP-004
DAEMON's NIP shall coordinate network intelligence with closed control-loop to meet service KPIs in different micro-domains.
K1 K2 K3 K4 K5 K6 K7 K8 K9
FR-NIP-005
DAEMON's NIP shall provide a NIS and a NIF catalog.
K1 K2 K3 K4 K5 K6 K7 K8 K9
NFR-NIP-003
NIP shall provide support for multiple virtualization environments for deploying services/applications in distributed domains.
K1 K2 K3 K4 K5 K6 K7 K8 K9
NFR-NIP-004
NIP shall provide support for federated multi-domain management and orchestration.
K1 K2 K3 K4 K5 K6 K7 K8 K9

Figure 3. *First-level organization of requirements tree of the DAEMON project for the proposed Network Intelligence Plane (NIP).*

Gap analysis

The requirements set forth above outline a NI plane that closes several gaps in the current frameworks for mobile network management that are proposed by the main Standard-Defining Organizations (SDOs). More precisely, in Section 5 and Appendix B of D2.1 [1], we provided a thorough discussion of existing solutions, their operation, and their limitations in terms of NI support. While we refer the reader to that deliverable for full details, Table 9 summarizes the main conclusions of the analysis.

As apparent from the table, current standards and platforms proposed by ETSI, O-RAN, or 3GPP, as well as implementations of the same institutions like OSM or ONAP do not provide (i) mechanisms to coordinate intelligence across different network micro-domains, or (ii) solutions for decentralized and unified data management across NI instances. Also (iii) support for the management of NI lifecycles is very limited, and there is only an early consideration for (iv) methodologies for the definition and representation of NI models. These are key functionalities for end-to-end NI coordination.

The functional requirements we define for the DAEMON's NI plane help removing the current barriers so as to enable full support for all aspects that are not necessarily covered by current frameworks. Indeed, we create specific requirements targeting the coordination of NI instances in an end-to-end fashion, which includes developing synergies in terms of data management and handling of interactions with MLOps platforms; the management of the complete lifecycles of both complex NI instances and atomic NI functions; the maintenance of catalogues of NI models that ease de-composition and orchestration.

Table 9. *Summary of the functionalities for NI management supported by the existing frameworks for mobile network control and orchestration.*

Framework	Provide a methodology to define AI-based functionalities	Provide mechanism to manage lifecycle of AI-based functionalities	Provide mechanism to coordinate intelligence across different network segments	Decentralized and unified data management for NI instances
ETSI MEC	No	No	No	No
ETSI NFV	No	No	No	No
ETSI ENI	**Yes**	No	No	No
O-RAN	**Yes**	**Partially**	No	No
OSM	No	No	No	No
3GPP	No	No	No	No
ONAP	No	No	No	No
NI plane	**Yes**	**Yes**	**Yes**	**Yes**

3 NI plane: initial architectural design

In this section, we present the initial design for an architectural model that stems from and integrates with current standards (e.g., O-RAN, 3GPP, ETSI), and realizes the DAEMON vision of native support for end-to-end NI coordination. We recall that the DAEMON's NI-native architecture is expected to bring together a variety of NI-assisted functions that span different timescales and network domains in a coordinated manner. To achieve this result, our design builds upon and adheres to the functional and non-functional requirements laid down in Section 2.2 above. Importantly, abiding by such requirements ensures that the design is also compliant with the initial guidelines provided in Section 6 of D2.1 [1], which covered in a less formal way a subset of the current requirements, and concerned the relationship among NI degrees of freedom, input-output relationship, and real-time constraints subject to infrastructure observability and controllability. In the following, we discuss the architectural framework proposed by DAEMON in detail.

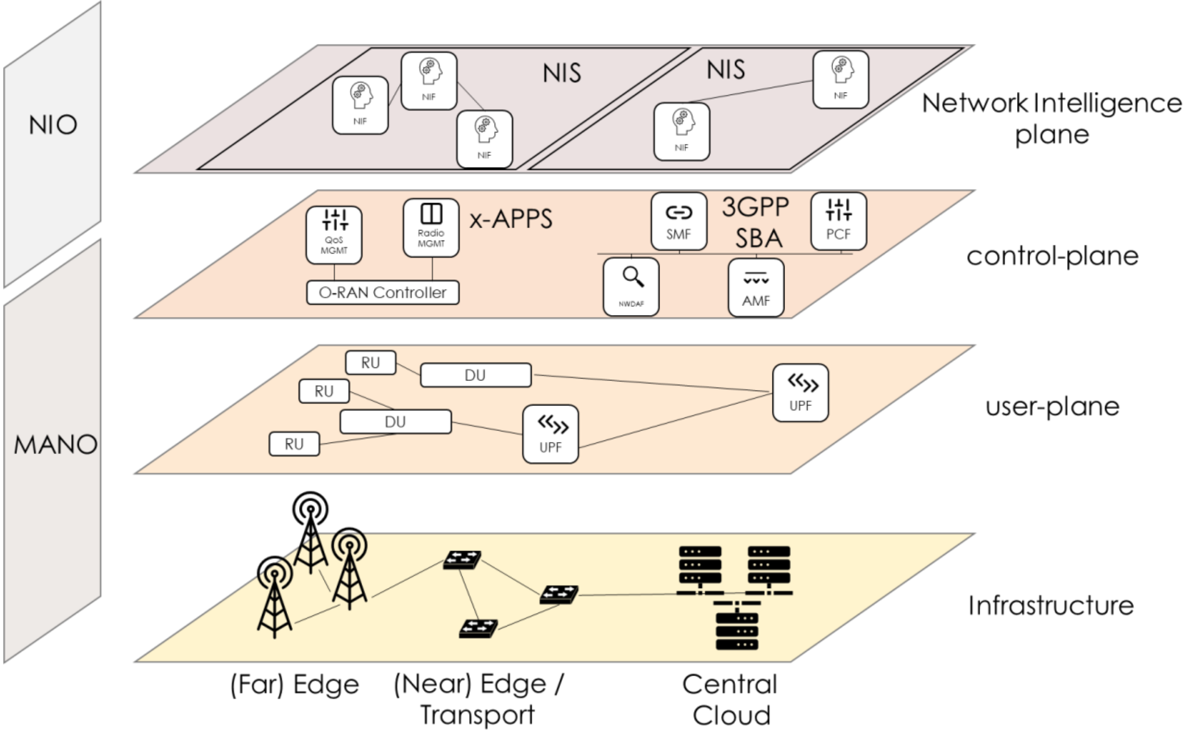


***Figure** 4. The overall DAEMON framework.*

Figure 4 describes the overall framework envisioned by DAEMON. Owing to the softwarization and the data-driven trends of current networks, we decompose the structure into four complementary layers: in addition to the legacy infrastructure layer, the control plane, and the user plane (which are the three fundamental building blocks of a software network such as the 5G one), in DAEMON we envision one additional layer, the ***Network Intelligence plane (NIP)***, that integrates the functions related to network intelligence, such as the ones detailed in D3.1 [5] and D4.1 [6], in the network architecture.

The Management and Orchestration (MANO) of this compound network is performed by two modules: the MANO, as traditionally done in 5G Networks, which handles the typical lifecycle management of the network and VNFs [9], and a new sibling element, the ***Network Intelligence Orchestrator (NIO)***, which takes care of all the operations related to the management of the intelligence of the network (represented by a variety of NI instances deployed across micro-domains). These operations include:

- The *selection* of the ***Network Intelligence Functions (NIF)*** that come together to build a ***Network Intelligence Service (NIS)*** to pursue one of the KPIs envisioned by DAEMON, such as e.g. energy efficiency (more details on this in Section 3.2**Error! Reference source not found.**).
- The *monitoring* of such functions, including the monitoring of their KPIs (e.g., their accuracy) and of the specific actions that may be taken to optimize them (e.g., meta parameter change, re-training, or model changes).
- The specific *training* procedures in case of learning models.
- The *interaction* with the MANO to handle service and resource orchestration.

For MANO, we reuse all the definitions and functional components from ETSI [9]. We omit these to avoid clutter, and we refer to [9] for a detailed explanation of such terminology. Instead, we focus on discussing in detail and defining the internals of the novel NI orchestration, specifying interfaces and procedures.

3.1 Detailed architecture

In this section, we provide an overall representation of architectural models and list relevant definitions from standards. We then introduce the new components proposed by DAEMON (e.g., NIF, NIS, NIO) and explain how they fit such existing architectural models (e.g., by O-RAN and 3GPP).

Specifically, Section 3.1.1 provides a high-level view of the NIP operation and introduces basic concepts underpinning the NIP design. Section 3.1.2 introduces the framework used by the NIP for a representation of NIFs that allows their orchestration. Section 3.1.3 details the organization of the NIO based on such a NIF representation. Finally, Section 3.1.4 discusses the interaction of the NIO with MLOps frameworks.

3.1.1 NIP operation and basic concepts

In DAEMON, we envision the management of NI in a similar manner as the management of Network Services is designed for 5G Networks. This allows us to re-use well known concepts, adapting them to the context of network intelligence. The high-level interactions are depicted in Figure 5, showing how the interactions take place.

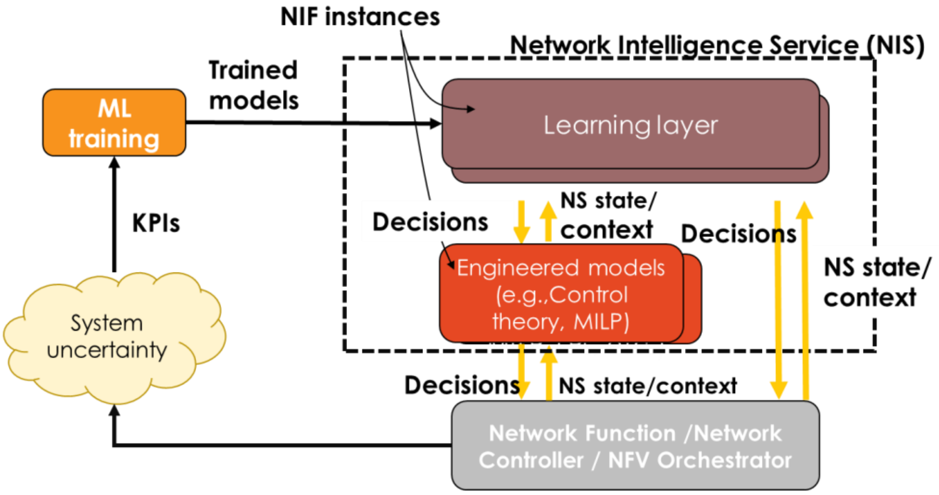


Figure 5. Taxonomy of NIP operations.

Following this strategy, and analogously to the information model specified for network management by for instance 3GPP, we define the concepts of NI Service (cf. Network Service, i.e., a 5G service class such as eMBB or URLLC) and NI Function (like any function specified by, e.g., 3GPP or O-RAN), as follows.

Network Intelligence Function (NIF). *Functional block in a network intelligence instance that implements a decision-making functionality to be deployed in a controller, NFV orchestrator, or Network Function and has well-defined interfaces and behavior. A NIF thus corresponds to an individual NI instance that assists a specific functionality, and comprehensive lists of NIFs are in deliverables of D3.1* [5] *and D4.1* [6] *of the project.*

Network Intelligence Service (NIS). *Composition of Network Intelligence Functions (NIFs) that has a specific target, usually related to a specific set of targeted KPIs. Table 10 shows examples of NISs derived from NI Functionalities developed in DAEMON.*

Table 10. *Examples of NIs derived by DAEMON NI functionalities [5] [6].*

NIS	KPIs	NIFs
Reliable Virtualized RAN	Reliability	• Reliable distributed unit (DU) for virtualized RAN • Orchestration of radio and computing resources in vRANs
Sustainable network operation	VNF Energy Savings Compute Resource Savings OPEX Savings	• Cloud Acceleration for virtualized RAN • Compute Aware scheduling analytics • AI-enhanced edge orchestration • Data-driven resource orchestration • Multi-timescale network slice reservation
Network capacity management	Wireless Capacity Increase	• Reconfigurable Intelligent Surfaces Control
Edge orchestration	OPEX Savings	• Network Service Auto-scaling • Capacity forecasting

There is a one-to-many relationship between NIS and NIFs, as the former could be provided by one or more instances of the latter. Consequently, network operators or service providers can for example request specific sustainability and reliability services targeting one or more KPIs. The NI orchestration will take care of providing such service by composing specific instances of NIFs.

NIFs themselves could be of different kinds: they could be learning models, based on, e.g., Deep Neural Networks or Engineered Models, or they could be built upon specific optimization algorithms such as the ones based on control theory or Mixed-Integer Linear Programming (MILP). ***The heterogeneous definition of NIFs is fully aligned with the vision of the DAEMON project of NI as the result of algorithms that are not limited to complex AI models but also encompass traditional and interpretable models that are not necessarily data-driven***. Also, these NIFs have two main interactions with the underlying layers (c-plane, u-plane, or infrastructure, proxied by an NFV Orchestrator): as a matter of fact, NIFs both (a) inject decisions and (b) receive information about the Network Slice State and the context of such state. A NIS is hence a coordinated effort of one or more NIF that could be arranged hierarchically. For example, a NIS could be composed of a Learning-type NIF sending decisions to an engineered model NIF, which in turn acts on the underlying infrastructure.

3.1.2 NIF representation framework

In order to manage the interaction between different NIFs, in DAEMON we define a further level of detail that de-composes each NIF into atomic elements that perform a specific operation. That is, besides the specific requirements associated with the algorithms, as discussed in Section 0, we need a mechanism to create a common framework to map the most common features of NI algorithms, subsequently integrate them into the overall architecture, and design the necessary interfaces that algorithms use to interact with their environment.

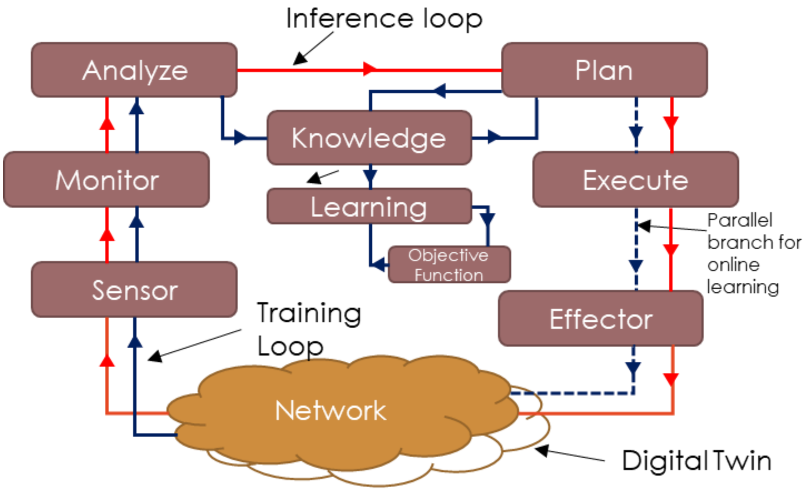


Figure 6. *Diagram summarizing the proposed N-MAPE-K framework for NIF representation. Note that the Effector and the Sensors can also be redirected to a Digital Twin element in order to disentangle the learning-loop process from the real operation of the network.*

For this purpose, we adopt within the activities of the DAEMON project a methodology that is already used by the MAPE-K (*Monitor-Analyze-Plan-Execute over a shared Knowledge*) feedback loop—one of the most influential reference control models for autonomous and self-adaptive systems [10]. We introduced this nomenclature adopted to label NI requirements within DAEMON was first introduced in [11], and it allows for classifying the algorithms that run at NI instances in a unified manner, based on how they interact with the other elements of the network.

It is worth noting that the original MAPE-K framework has limitations in the context of mobile network functionalities supported by NI, which represents our target in the project. To overcome such limitations, we propose changes to the legacy MAPE-K to consider the specificities of the network environment, as depicted in Figure 6. In this figure, we illustrate the different training and control loops that may be implemented by a NIF: (i) the inference loop, (ii) the training loop, and (iii) a different training loop with a branch for online learning. The model emerging from this adaptation is coined the ***Network Monitor-Analyze-Plan-Execute over a shared Knowledge (N-MAPE-K)***.

The extensions of N-MAPE-K over the original MAPE-K concern in particular the following two dimensions.

- The purpose of the NIF, i.e., whether the Knowledge is (a) being trained or (b) being used for inference during the operation of the network, following the MLOps paradigm.
- The nature of NIF algorithm, i.e., telling apart online learning or pre-trained/engineered models.

For the latter, the Knowledge module shall be integrated with a Training definition, which contains all the attributes of the algorithm and specifies aspects such as the input data format, training batches, and training epochs. Most importantly, consistently with the NI design guidelines set forth by DAEMON (see Section 4 of this document for details), the Training shall specify the used loss function (which could be dynamically adjusted), and the State/Action representation –depending on whether the NIF algorithm belongs to the family of supervised learning models or to the one of online learning ones. Additionally, the Effector and the Sensors can also be redirected to a Digital Twin element, if needed by the specific NI instance, in order to disentangle the learning-loop process from the real operation of the network.

Based on the resulting N-MAPE-K representation adopted by the DAEMON project, each NIF can be further split into atomic ***NIF Components (NIF-C)*** as follows.

- The ***Sensors*** block specifies all the probes that are needed to gather the input data and the kind of input data we have to gather. In principle, Sensors within the NIF correspond to the APIs that are used to interact with the software and hardware measurement probes and data repositories deployed in the network infrastructure.
- The ***Monitor*** block specifies how the NIF interacts with the Sensors, i.e., when and how it accesses the APIs mentioned above.
- The ***Analyze*** block includes any pre-processing, summary, or preparation of the data, such as those implemented by averaging, autoencoding, or clustering algorithms.
- The ***Plan*** block constitutes the specific NI algorithm that is implemented by the NIF, for instance a Neural Network (NN) performing a classification task.
- The ***Execute*** block specifies how the algorithm is going to interact with the system and how to possibly change its configuration parameters.
- Finally, the ***Effector*** block includes specific configuration parameters updated in the Network Function, again specifying the API to be used to that end.

The N-MAPE-K framework is the result of a joint effort of the DAEMON consortium and is presented in full details in a joint work involving the partners active in WP2 of the project [11]. We refer the reader to that scientific publication for more information. Also, we will present later in Section 3.2 how the N-MAPE-K framework can be leveraged in practice to model in a unified way two algorithms developed in the project and presented in [12] and [13], which in turn enables their joint NI lifecycle management.

3.1.3 NI Orchestrator

We now detail the structure of the proposed NI Orchestrator (NIO), which is designed so as to fulfil the NI management and coordination requirements outlined in Section 2.2.

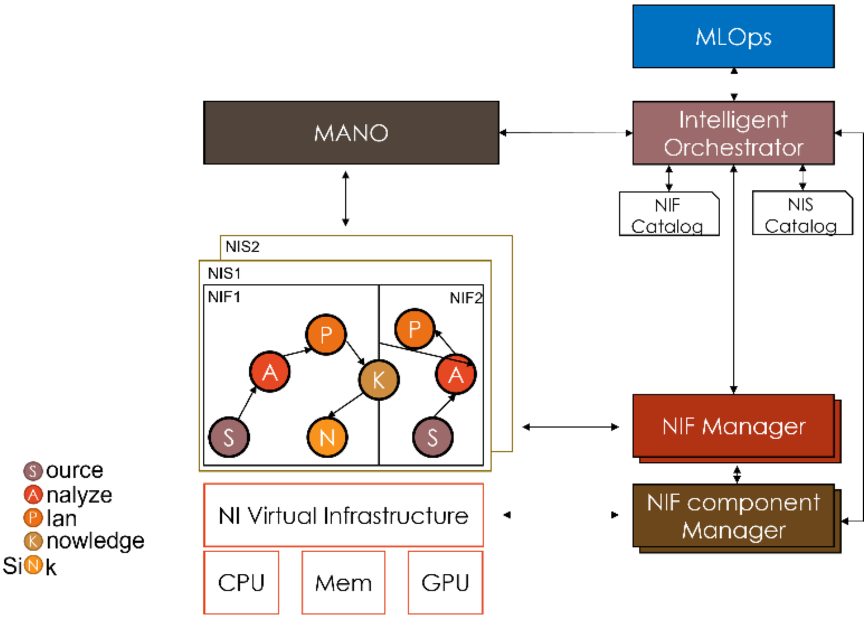


Figure 7. *The initial design of the NI Orchestration framework proposed by the DAEMON project.*

To manage and orchestrate the NIS, NIF, and NIF-C that build the network intelligence, we mutated the layered structure of the ETSI NFV MANO framework, tailoring the components to the specificities of network intelligence. The resulting Network Intelligence Orchestration framework, depicted in Figure 7, is organized into three levels, i.e., (i) the Intelligence Orchestrator, (ii) the NIF Manager, and (iii) the NIF-C Manager. We next detail the functions and operation of these three levels.

NIF Component Manager. The NIF Component Manager is in charge of handling the lifecycle of the NIF-Cs. By lifecycle management, we refer to operations that include onboarding, instantiation, termination, scaling, and state retrieval. All these are handled by the NIF-C Manager independently of their Kind (I.e., independently of whether they are Source / Analyze / Plan / Knowledge/ Sink) and their connection towards the Infrastructure. For instance, in the case of Sources, the IP addresses of the different data producers shall be provided, while for the case of Sinks the specific configuration API endpoints have to be configured. This will have specific instantiations according to where this interaction shall take place. For instance, if the NIF is executed from the core, then Sinks and Sources shall integrate with the Network Registry Function and the Network Exposure Function, properly synchronizing with the Network Data Analytics Function (NWDAF) [14], whose analytics are actually captured as a set of Analyze, Plan, and Knowledge boxes. Similar considerations apply also for other network domains, such as the RAN, where this framework can be fully integrated in the O-RAN x-Apps or r-Apps ecosystems.

NIF Manager. The NIF Manager, instead, has a global view of the set of NIF-C that compose every Network Intelligence Function: besides the lifecycle management of the NIF, this module is in charge of monitoring the health of the intelligence functions. This includes typical diagnostic information (e.g., constantly checking the KPIs yielded by the NIFs, such as its accuracy) if the NIF is being used in inference or it is an online learning solution; other metrics would be monitored, such as the loss and the training loops, in case the NIF is currently being trained (more details on this in Section 3.1.4). The NIF Manager is also responsible for setting the meta-parameters of the models (through the interaction with the NIF-C manager) and reporting the health status of the NIF to the upmost module in the hierarchy, i.e., the Intelligence Orchestrator.

Intelligence Orchestrator. The Intelligence Orchestrator is in charge of the lifecycle management of the NIS, by properly coordinating the NIFs that build each of them. This includes the possibility of sharing NIF-C among different NIFs (e.g., two NIFs that require the same input) and also the arbitration policies in case of two NIFs that share the same sink, that is, the configuration APIs. Note that this is performed at the level of the Intelligence Orchestrator, and it is not a responsibility of the NIF Manager anymore: indeed, this coordination is not within a single NIF (which is a task for the NIF Manager), rather across NIFs, hence requires a higher-level view that only the Intelligence Orchestrator has.

This module also manages the connections towards the network management and orchestration to gather important information such as the expected network KPIs for the managed slice and service, and the information of the underlying network infrastructure. The Intelligence Orchestrator has catalogs of already onboarded NIS and NIFs. In particular, NIFs may need to be re-trained to cope with changing or different conditions, or on a periodical basis. In this case, the Network Orchestration interfaces with an external platform to build ML pipelines and perform such operation: such an interaction with MLOps platforms is the specific subject of the next section.

3.1.4 MLOps and NI Orchestration

Several of the functionalities of the NI Plane framework described above can sound very familiar to MLOps engineers (cf. Section 11, Annex C of D2.1 [1]) since modern MLOps frameworks are composed of well-known workflows such as building ML models, deploying, monitoring them, and, if required, re-training them. Let us recall what is MLOps. In simple terms, MLOps [15] is a methodology that combines Machine Learning (ML) with software development operations (DevOps) and data engineering with the goal of building, training, deploying, and maintaining ML systems in productions with high reliability and efficiency guarantees. *Figure* 8 illustrates a generic, modular, and flexible MLOps workflow that is designed to bring ML solutions to any business or industry.

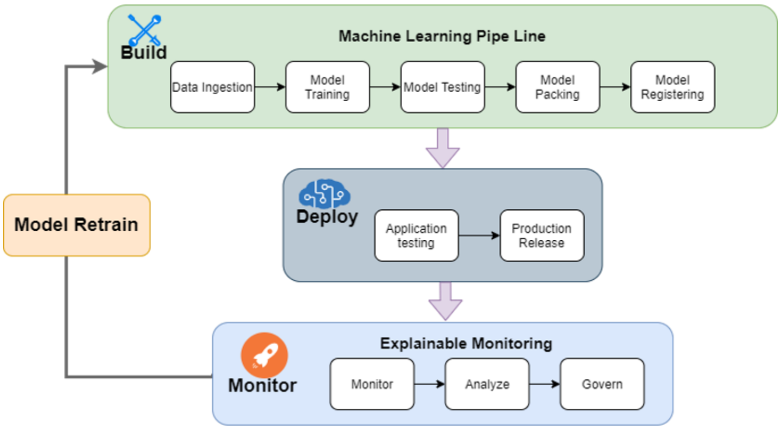


Figure 8. *A generic MLOps workflow.*

The DAEMON architecture explicitly indicates that building ML model functionalities (i.e., the ML pipelines) is delegated to an external platform, and MLOps frameworks are the de-facto platform to do this task. In general, the DAEMON architecture already embraces MLOps workflows in its own functionalities. Let us describe the three main components of a generic MLOps (Build, Deploy, and Monitor) in terms of DAEMON architecture functionalities:

- ***Build.*** When a NIS is composed of NIF empowered by ML models, training such models will be performed via the creation and deployment of ML pipelines. Once the models are trained, they will be registered in the NIF/NIS catalog and will be ready to be deployed in a test/production environment.
- ***Deploy.*** Once the models are created, trained and stored in a register in the previous module, the NI orchestrator will select and start the deployment of a given NIS to test the model performance and behavior in a production or production-like environment to ensure its robustness and scalability. This implies verifying the capability of the model for inference from data batches or data streams, and/or in making business decisions. After that, it can be released for production. Once the NI Orchestrator has determined the deployment of a NIS/NIF, the NI Orchestrator will take care of the lifecycle management of the NIS while the NIF manager will take care of the lifecycle management of the NIFs composing the NIS.
- ***Monitor.*** In MLOps, this module is used to monitor, analyze and govern the performance of the ML application (i.e., the combination of the model and its DevOps application). In DAEMON, these functionalities are split among the NIF Manager and the NI Orchestrator. In runtime, the performance of the ML model is monitored via pre-defined metrics and the application via telemetry data, exposed via well-defined NIS/NIF's APIs. All the extracted information via the monitoring functionality will be analyzed by the NI Orchestrator to determine the performance of the ML model (or set of models) in the NIS. The analysis can be enhanced with an explainability framework for better understanding of the decision-making process of the models. The outcome of the analysis step will be used by the governance module to trigger actions based on the model's performance such as conflict resolution, triggering a new retraining phase, change the model, etc.

In summary, the DAEMON architecture already embraces MLOps workflows and therefore is designed to support natively lifecycle management of NI. Table 11 shows the parallel between MLOps workflows and the DAEMON NI Plane relate to each other.

Table 11. *MLOps and DAEMON NI Orchestration Framework comparison.*

MLOps workflows	MLOps Component	DAEMON NI Plane component exposing such functionality	Description
Build	ML pipeline	ML pipelines	Creation, training and registration of training models.
Deploy	Testing and Release	NI Orchestrator	Selection and trigger NIS deployment (testing/release)
		NI Orchestrator	Runtime and lifecycle management of deployed NIS
Monitor	Monitor	NIF Manager	Capture and expose performance metric of the ML model
	Analysis	NI Orchestrator	Determine the performance of the ML model(s) in the NIS. May include explainability functionalities to keep accountability of ML models.
	Governance	NI Orchestrator	Send alerts and trigger actions based on the model's performance such as conflict resolution, trigger a new retraining phase, change the model,

Well-known frameworks such as Kubeflow [16] and MLFlow [17] are MLOps frameworks that provide the basic workflows explained above. Moreover, some recent efforts have shown that the integration of MLOps workflows in ML pipelines and MANO frameworks is possible [18]. The DAEMON NIP design we propose is orthogonal and compatible with all these emerging approaches, as it does not attempt to reinvent them, rather it enables interactions with them. Ultimately, this favors the adoption of the design set forth by the project within ongoing standardization and implementation efforts.

3.2 Preliminary case study

To conclude our presentation of the initial design of the DAEMON NI plane, we showcase who the NI information model introduced by the project can effectively support orchestration and synergy of intelligence within mobile networks. Specifically, we target two representative NI algorithms: vrAIn [19], which is a state-the-art solution for vRAN orchestration, and SBC-vRAN [20], which is an algorithm developed by the DAEMON project to address the energy-aware vRAN orchestration problem described in Section 3.2 of D4.1 [6], and which will be detailed in D4.2.

We first instantiate the two algorithms as NIFs, in alignment with the NIP operation model outlined in Section 3.1.1. We then represent these two NIFs as atomic NIF-Cs using the proposed N-MAPE-K model discussed in Section 3.1.2. This allows the NIO to orchestrate the NIF-Cs as presented in Section 3.1.3, over a NIF-C ecosystem based on the Zenoh-Flow paradigm [19]; Zenoh-flow is briefly described in Section 4.1.4.

The process above entails a split of the NIFs into NIF-Cs following the proposed N-MAPE-K framework, as detailed in Table 12 below.

Table 12. *Description of the two NIFs considered and their relationship with the N-MAPE-K framework.*

Analytics	Description	
	vrAIn	SBP-vRAN
Sensors + Monitor	Channel conditions: SNR measurements, traffic demands: as Buffer State Reports (BSRs) from the terminals.	Channel conditions: SNR measurements, Traffic demands: BSRs and Downlink buffer occupation.
Analyze	Inputs are passed through an autoencoder to reduce their dimensions, forming an encoding that is used in the execution algorithm.	Inputs are passed through an autoencoder to reduce their dimensions, forming an encoding that is used in the execution algorithm.
Plan	An actor-critic deep learning algorithm takes the encodings as input and generates two outputs: the amount of CPU required and the MCS policy.	A Bayesian Learning model takes the encodings as input and generates three outputs: MCS policy, airtime policy, TX power policy.
Execution + Effector	Two APIs exposed by the virtualization environment (for the CPU quota) and the base station (for the MCS policy, via O-RAN A1/E2 interface).	Three APIs exposed by the base station (O-RAN E2 interface).
Knowledge	A model of the CPU behavior of a base station.	A model of the power consumption behavior of a base station.
Training / Loss / State / Actions / Rewards	• States: Latent representation of the input data. • Actions: compute and radio policies. • Rewards: latency tolerance.	• States: Latent representation of the input data. • Actions: radio policies. • Rewards: maximum throughput subject to power budget.

Figure 9 shows a possible mapping of our target vRAN orchestration algorithms into Zenoh Flow. The topmost part of the graph depicts the Sources and the Sinks NIF-Cs, that implement the Monitor / Sensor and the Execute / Effector in Zenoh Flow. In a nutshell, these are the input and output variables of the NIFs, which in this case are shared by the two NIFs.

Four sources provide the input to the two algorithms: the user SNR and BSR reports (common to vrAIn and SBP-vRAN), the CPU load (only used by vrAIn), and the Power Consumption (only used by SBP-vRAN). Thanks to the Zenoh Flow features, these sinks can produce the data needed for the NI algorithms at a fixed pace (e.g., every TTI, that is, at intervals of 1 ms) or on-demand. The former is the case of SNR (in both uplink and downlink direction) and user load requirements (obtained by BSR reported by users and the inspection of the internal buffers). On-demand data is instead requested for, e.g., the CPU consumption used in vrAIn or the Power used in SBP-vRAN, which could be retrieved in larger batches. These variables are not directly involved in the NI algorithm operation but are instead used to build accurate models of the vRAN system for the computing profile and power consumption.

In the considered NI examples, the Analyze blocks can be similarly shared. Driven by the same objective (i.e., the dimensionality reduction from a very informative yet large data), both vrAIn and SBP-vRAN build on an autoencoder-based analysis of the data, yielding succinct input data for the downstream learning agent. Here, the NI implementation can take advantage of Zenoh Flow feature of recursively defining each node: that is, each Analyze block can be composed of a set of Zenoh Flow nodes that, when joined together, provide the autoencoder functionality. Note that different outlets of each block can be placed at diverse positions yielding to, e.g., an autoencoder that compresses down to 4 or 8 dimensions according to the given use. These items can directly feed the Plan part, in charge of the actual decision in the system, or be stored in the knowledge blocks (e.g., for NI training).

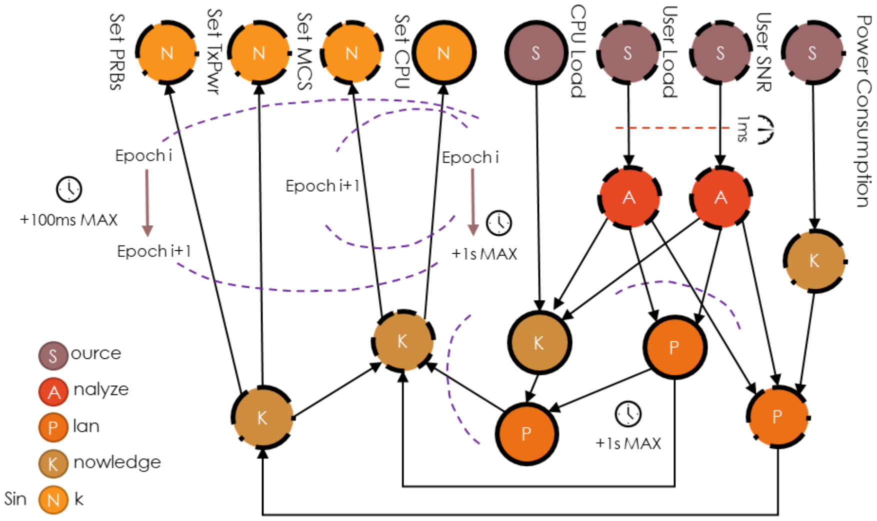


Figure 9. *The Zenoh-flow arrangement of the N-MAPE-K blocks for vrAIn and SBP-vRAN. Dashed nodes are common to both algorithms, while solid and dot-dashed nodes are related to vrAIn and SBP-vRAN, respectively.*

The Plan blocks implement the trained learning agents for the two algorithms. They could have cascading relations (such as the CPU and Radio schedulers in vrAIn) or be completely independent, like the one used by SBP-vRAN. In principle, other relations may exist, such as peer-to-peer exchanges of information (e.g., for encodings) or achieved rewards computed by the Plan blocks.

Besides the data used for training, the knowledge nodes are fed with the policies and decisions computed by the Plan nodes, resulting in a database for the sink nodes in the network. Again, sinks can be shared across NIFs instances, as in the case of the set MCS API, which is common to both vrAIn and SBP-vRAN. The end-to-end computation of decision (e.g., the Plan to Sink timing constraint, from epoch to epoch) can be implemented using the timestamping feature of Zenoh. For instance, the timing requirements of vrAIn and SBP-vRAN (i.e., 1s and 100ms, respectively) can be indicated to the Zenoh orchestrator, allowing us to identify late inputs. Ultimately, the discussion above shows how, building on the proposed NMAPE-K representation, the Zenoh Flow framework can be used to implement heterogeneous NI algorithms.

4 Guidelines on machine learning for network management

In the previous section, we have seen the overall picture of the proposed architectural design, which follows from the analysis on the requirements for NI management and orchestration from Section 2.2. Next, we delve into each of the eight different functionalities described in Section 0, and we provide a specific set of guidelines for the incorporation of machine-learning-based functions in the design and implementation of each of these functionalities. We provide two sets of guidelines: The first focuses on the modifications required to adapt AI/ML solutions into specific networking applications, whereas the second set comprises insights on whether AI/ML solutions are the best choice for different network use cases.

4.1 Tailored AI design for NI

One of the objectives of DEAMON is establishing methodologies to adapt legacy AI models based on recent deep learning approaches to the specificities of real-world NI problems. Indeed, networking operation, optimization, and management is a complex and particular framework with many singularities that distinguish it from other fields. This idiosyncrasy is important when it comes to incorporating new approaches into the network, since top-level solutions with unmatched performance in other less constrained fields may fail to provide the envisioned operability in networking.

In the light of these considerations, the DAEMON project challenges the current practice of addressing NI problems by directly adopting general-purpose AI models or models that have been successfully employed in other domains, without significant modifications. Instead, a sensible integration of AI models into NI calls for substantial customization and contextualization. In this section, ***we provide a comprehensive list of the current outcomes of the research conducted within the DAEMON project and aimed at adapting and tailoring AI/ML solutions so as to better fit network functionalities***. We link these adaptations to the requirements of the target functionalities described in Section 0. The connection between requirements and the content presented here is twofold: on the one hand, the functional requirements set the constraints and goals that the derived solutions must fulfil; on the other hand, the developed solutions allow us to unveil limitations of the requirements from the outcomes of the research work, hence triggering updates of the requirements. This second connection has been instrumental to develop the second project iteration.

Table 13 summarizes the six guidelines produced by the project to date, indicating the requirements they relate to and providing a brief description of their key message. Full details on each guideline are then presented in the remainder of this section. Note that the focus there is on the extrapolation of the design guidelines of AI for NI, guidelines that arise from the activities carried out during the first iteration of the DAEMON project. Therefore, when applicable, we also link guidelines to their implementation for some specific NI-assisted functionality that are presented in other deliverables of the project.

Table 13. *Summary of the DAEMON project's guidelines on tailoring AI for NI.*

Guideline	Requirements	Description
Incorporating prior knowledge in decision making schemes	FR-SLMANO-002 FR-SLMANO-005	AI models for NI shall incorporate prior knowledge about the network system by design, e.g., as restrictions on the coefficients of the neural network, or as simplifications to the training data. This reduces the amount of data needed for training without impairing AI performance.
Avoiding the loss-metric mismatch	FR-CFORE-002 FR-CFORE-005	AI models for NI shall be trained using customized loss functions that are carefully developed based on expert system knowledge. Unlike legacy loss functions that are designed to be generic enough to work well in a wide range of scenarios, task-tailored losses can capture the specific performance targets and dramatically improve results.
Loss function meta-learning	FR-CFORE-002 FR-CFORE-005FR-CFORE-006 FR-CFORE-007	AI models for NI may adopt when relevant a design that meta-learns the loss function that best suits the network management objective at hand. This is the case, e.g., when the performance metric to be optimized by anticipatory MANO actions is not known a priori by the network operator.
Self-learning models based on dataflow programming	FR-MTERM-004	AI models for NI shall be informed by tailored data feeds. The input to AI models for NI requires decentralized and distributed data management, unification of data patterns, support for heterogeneous devices, support for eventual consistency models, or support for different timescales and real-time communications. In turn, these call for both decentralized data pipelines as well as the ability for declaring deadlines for real time operations and the reusability of components.

Adapting a known reward function to networking	FR-SLMANO-003 FR-MTERM-007.00	AI models for NI that are based on RL may adapt known rewards instead of defining new ones. Contrary to most of the RL applications in networking, where the states, actions, and reward function are defined using a networking rationale, the DAEMON project commends that the many different reward expressions used in well-known applications of RL can be leveraged and adapted to suitable rewards that drive NI decisions in specific network functionalities.
Low inference time and energy consumption	NFR-RIS-001 NFR-RIS-002 NFR-EAWVNF-003 NFR-EAWVNF-004 NFR-CAWRS-000 NFR-CAWRS-001 NFR-CAWRS-003	AI models of NI may be designed for extremely low inference latency and energy consumption. This requirement applies to a number of mobile network applications such as traffic classifiers or load balancers in multi-gigabit-per-second backhaul segments, or in baseband processing operations in the radio interfaces, where the processing latency budget for inference is well below 100 microseconds. Techniques for AI design that meet such specifications include (i) distribution of complexity across simple and fast models, e.g., via multi-actor-critic RL, (ii) in-subsystem inference that avoids time-consuming communication with a GPU, e.g., by running AI directly in the network interface card (NIC), or (iii) use of low-complexity AI models, e.g., Binarized Neural Networks (BNN).

We remark that, collectively, these guidelines address the different items related to the tailored design of AI for NI, as presented in the DoA of the DAEMON project. We make these links explicit as follows.

- The guidelines on (i) incorporating prior knowledge in decision making schemes, (ii) avoiding the loss-metric mismatch, and (iii) adapting a known reward function to networking address the issue of "*closing the loss-metric mismatch, by deriving general guidelines for the design of dedicated loss functions that are perfectly aligned with the actual performance metrics of interest*".
- The guidelines on (i) loss function meta-learning and (ii) self-learning models based on dataflow programming address the problem of "*designing a methodology for self-learning AI models that dynamically and automatically balance costs and efficiency, by learning the loss function indirectly from the feedback of the end-customers, without requiring them to explicitly identify their objectives*".
- The guideline on (i) low inference time and energy consumption addresses the problem of "*developing elastic NI models capable of adapting their own complexity to the context, trading off (computational) complexity for accuracy, responsiveness or energy efficiency as needed*".

4.1.1 Incorporating prior knowledge in decision making schemes

The NI design guidelines set forth next are relevant to the following requirements of DAEMON, which were presented in Section 4 of D2.1 [1], updated in Section 2 of this document, and which are reported in full in Appendix A of this same document:

- FR-SLMANO-002, FR-SLMANO-005

For ML-based approaches in many networking problems, the probability of taking an action a when a certain observation ω is made, i.e., the policy $\pi(\omega, a)$, is modelled as a neural network. In such problems, ***there often exists some prior knowledge inherent to the problem that constrains the action space***; for instance, it may be known a priori that, for two observations ω_1, ω_2 in different regions of the observation space, a certain action a should be more likely for observation ω_1 than for observation ω_2. The simplest example of this feature is a monotonicity constraint: $\pi(\omega_1, a) > \pi(\omega_2, a)$ if $\omega_1 > \omega_2$. Plain vanilla, not-tailored neural networks do not possess such a property; in contrast, they can only learn this property after being trained on a large amount of data. Incorporating this prior knowledge in the decision making scheme can be achieved in various ways: (i) by putting adequate restrictions on the coefficients of the neural network, either by imposing hard restrictions (e.g., all of the coefficients being positive in the case of the monotonicity constraint) or by penalizing unfit coefficients in an additional loss term; (ii) by preprocessing the training data such that pairs of data that do not expose the desired behavior are suitably altered or removed. In both cases, by incorporating prior knowledge, less data is needed to train the neural network to achieve a reasonable performance.

Figure 10 shows a typical example in which the acceptance probability of a network service (with a given service level agreement or SLA) was learned based on data containing how previous SLAs were accepted. The figure shows how the performance increases as the number of samples in the training data set increases for various neural network models. The plain vanilla neural network (labelled "Vanilla"), which does not take prior knowledge into account, needs much more data than the models in which prior knowledge is incorporated. The models labeled as "Reg.", "AVWT" (absolute value weight transformation) and "MOL" (mini-batch order loss) are variants of the first type, where prior knowledge is incorporated by putting a hard or soft constraint on the coefficients of the neural network, while "CSE" (conflicting sample elimination) and "PO" (probability optimization) are variants of the second type, in which the data is preprocessed. More results will be reported in deliverable D4.2 of the DAEMON Project.

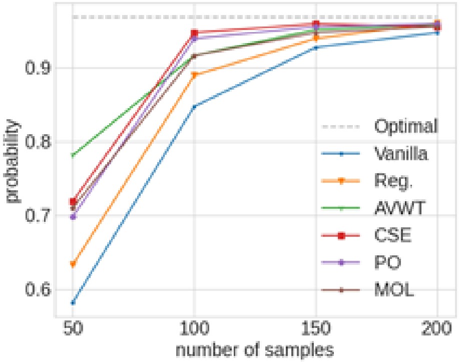


Figure 10. *Performance of neural network with embedded prior knowledge.*

4.1.2 Avoiding the loss-metric mismatch in network intelligence

The NI design guidelines set forth next are relevant to the following requirements of DAEMON, which were presented in Section 4 of D2.1 [1], updated in Section 2 of this document, and which are reported in full in Appendix A of this same document:

- FR-CFORE-002, FR-CFORE-005 for Capacity Forecasting NI.

Loss functions drive the training process of supervised machine learning models. In the vast majority of cases, loss functions are designed to be generic enough to work well in a wide range of scenarios. In regression problems, including forecasting tasks, Mean Absolute Error (MAE), Mean Square Error (MSE), or Mean squared logarithmic error (MSLE) are common choices for expressing the loss.

However, in many practical cases in network management, such traditional losses do not characterize well the target performance metric of forecasting tasks. For instance, in anticipatory resource allocation problems encountered across mobile network infrastructure domains, the goal is anticipating a capacity that is *sufficient* to accommodate the future traffic demand. Indeed, underprovisioning of capacity leads to the disruption of the offered service and violations of the Service-Level Agreements (SLAs) with the service providers, while overprovisioning causes a more affordable squandering of resources. There, it is critical that the predictor learns to forecast a minimum quantity that is always above the demand.

Using a traditional loss function to perform forecasts in cases such as those outlined above results in a so-called loss-metric mismatch, where the regression objective, represented by the loss to be minimized, does not correspond to the optimization of the actual performance metric. As a result, the AI model that implements the regression model does not learn predictions that are aligned with the expected network management objective.

As part of its guidelines for the tailored design of AI for networking, ***the DAEMON project supports the use of customized loss functions that are carefully developed based on expert system knowledge***, i.e., a deep understanding of the network engineering or management task at hand, as well as of the variables that affect it and how they do so. *Figure 11* illustrates how the tailored design of loss functions for NI shall occur. In the left plot (a), a pure traffic predictor is trained using a legacy loss, e.g., MAE or MSE for regression. The resulting forecast cannot be used as is, but serves as an input to the actual decision block, which is manually designed by human experts to output the anticipatory MANO actions so that the target network performance objective is met. Yet, the decision block is agnostic of the inherent accuracy of the predictor and just trust the forecast it receives. In the right plot (b), the novel approach proposed by the DAEMON project is outlined: expert knowledge is used to directly design a dedicated loss that encodes the relationship between the prediction and the performance objective. As a result, the predictor is trained to produce forecasts that optimize the performance and can be directly used to drive the MANO actions. Importantly, the action decision is now aware of the unavoidable prediction error (e.g., lower accuracy in predicting small traffic volumes), and automatedly compensates for it (e.g., by taking more conservative actions to accommodate small-traffic future demands).

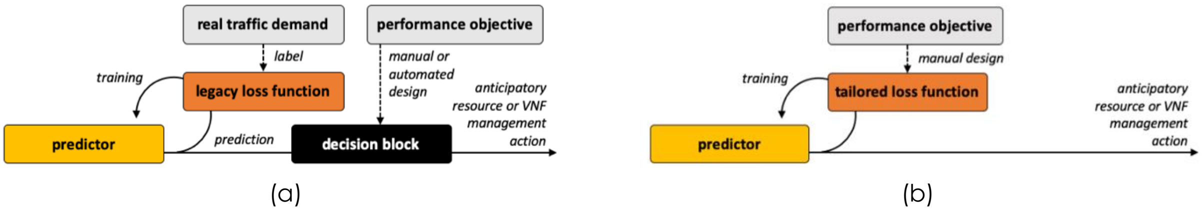


Figure 11. *Different approaches for solving the loss-metric mismatch.*

4.1.3 Loss meta-learning for network intelligence

The NI design guidelines set forth next are relevant to the following requirements of DAEMON, which were presented in Section 4 of D2.1 [1], updated in Section 2 of this document, and which are reported in full in Appendix A of this same document:

- FR-CFORE-002, FR-CFORE-005, FR-CFORE-006, FR-CFORE-007 for Capacity Forecasting NI.

The performance metric to be optimized by anticipatory MANO actions is not always known a priori by the network operator. This is the case, for instance. when the performance must be measured at the application layer (i.e., in the service provider domain), or when it concerns end user satisfaction (e.g., if it relates to mean opinion scores or quality of experience). In these situations, designing tailored loss functions as presented in Section 4.1.2 above is not possible, since the human expert (e.g., network manager or system engineer) does not know the exact relationship between the forecast and target performance.

DAEMON sets forth innovative guidelines to deal with NI design in the complex situations described above. Specifically, instead of imposing a predefined expression of the loss function used to train the predictor, ***the DAEMON project advocates a design of forecasting models that is free to meta-learn the loss function that best suits the network management objective at hand***. In practice, this is realized by combining a loss-learning block with the actual predictor, as shown in Figure 12. This block is responsible for learning the loss function, or, equivalently, capturing the relationship between the forecast produced by the predictor and the target management objective. Once ready, the loss-learning block can operate as a tailored loss function: it receives the output of the predictor and determines its quality for the precise management task. Therefore, it can be employed to train the predictor so as to steer the optimization of its parameters towards minimizing the actual MANO objective.

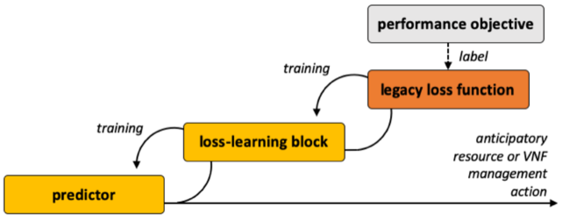


Figure 12. *Loss meta-learning for NI.*
The network management objective is learnt and encoded into a loss-learning block. This block then serves as the loss function to train the predictor, so that it directly outputs the anticipatory action.

Our (not-limiting) choice for the implementation of the loss-learning block is a deep neural network (DNN), in cascade to a first DNN that implements the predictor. This is illustrated in Figure 13. The loss-learning DNN can be trained as a regular neural network, by minimizing the MSE between its output and the objective. It is worth noting that such loss training can use performance measurements collected in the target system as a direct representation of the objective, without any need to formalize it as a mathematical function. This model, which we name *LossLeaP,* for *Loss-Learning Predictor*, solves a regression problem and outputs a continuous-valued action, but does so by learning from experience, similarly to Reinforcement Learning (RL) approaches.

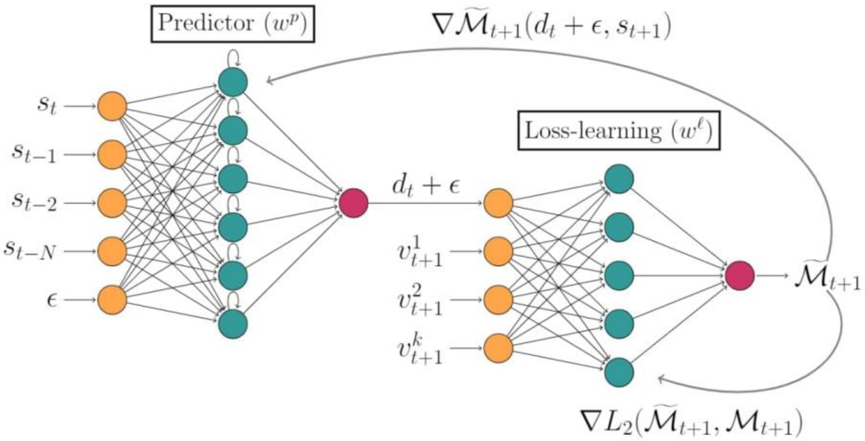

Predictor (w^p)
$\nabla\widetilde{\mathcal{M}}_{t+1}(d_t+\epsilon, s_{t+1})$
Loss-learning (w^ℓ)
s_t
s_{t-1}
s_{t-2}
s_{t-N}
ϵ
$d_t+\epsilon$
v^1_{t+1}
v^2_{t+1}
v^k_{t+1}
$\widetilde{\mathcal{M}}_{t+1}$
$\nabla L_2(\widetilde{\mathcal{M}}_{t+1}, \mathcal{M}_{t+1})$

Figure 13. *Proposed architecture of the loss meta-learning model set forth by DAEMON.*

This design has several key advantages:

- The loss-learning DNN can learn the relationships between the prediction and the objective from measurement data, without need for human intervention.
- Without any need for prior knowledge of the system, the loss-learning DNN can model tangled non-linear and multivariate objectives that may characterize practical MANO decisions.
- The only prerequisite on the objective is that its values should be minimum to attain the best performance, which is very supple: if not implicit in the performance target, the requirement can be met with naive transformations of the performance measurement data during training.

Full details on the LossLeaP design and operation are available in [20], and a preliminary performance evaluation showing the advantages of loss meta-learning over legacy DNNs is presented in Section 4.6.2 and Section 4.6.3 of D5.1 of the DAEMON project [4]. Overall, the DAEMON guideline above paves the road to the design of more adapted and automated NI models for MANO operations.

4.1.4 Self-learning models based on dataflow programming

The NI design guidelines set forth next are relevant to the following requirements of DAEMON, which were presented in Section 4 of D2.1 [1], updated in Section 2 of this document, and which are reported in full in Appendix A of this same document:

- FR-MTERM-004 on Multi-timescale edge resource management.

Designing an NI-native architecture for B5G systems requires clear requirements and specifications, related to data-driven features such as: decentralized and distributed data management, unification of data patterns, support for heterogeneous devices, support for eventual consistency models, or support for different timescales and real-time communications. Based on these considerations, ***the DAEMON project advises that we need both decentralized data pipelines as well as the ability for declaring deadlines for real time operations and the reusability of components*** [11].

Eclipse Zenoh-Flow *[19]* provides the mechanism for simplifying and structuring (i) the declaration, (ii) the deployment, and (iii) the writing of complex applications that can span from the Cloud to the Edge or beyond edge. As illustrated in

Figure 14. , Zenoh-Flow offers flexibility and extensibility for data flow programming structures computations. The main benefit of this approach is that it enables us to decorrelate applications from the underlying infrastructure: data are published and subscribed to without the need to know where they are actually located, e.g., cloud, edge, or beyond edge.

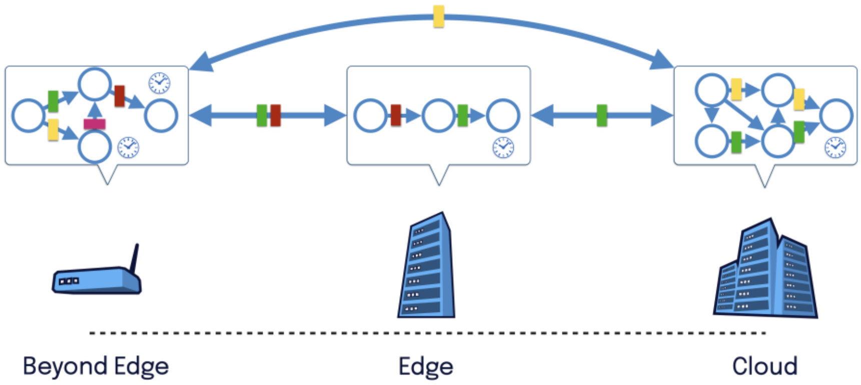


***Figure** 14. Decentralized data pipeline - using Eclipse Zenoh-Flow.*

During DAEMON's second period, we have tackled the challenge of integrating NI algorithms into the overall DAEMON's architecture presented in Figure 4. Thus, it is fundamental to understand which are the needed interfaces. For this purpose, we adopted a methodology named the MAPE-K (Monitor-Analyze-Plan-Execute over a shared Knowledge) feedback loop, one of the most influential reference control models for autonomic and self-adaptive systems [21]. Having MAPE-K as a reference, the algorithms that run at NI instances can be classified in a unified manner, according to how they interact with the other elements of the network.

Figure 9, presented in the case study of Section 3.2, illustrates the implementation of a MAPE-K extended definition for the vrAIn [13] algorithm as an example of NI. vrAIn is a reinforcement learning algorithm that tailors the available computing capacity on a vRAN platform to the expected load introduced by the different physical layer tasks such as frame decoding. Thus, by gathering information from the network operation (e.g., the link conditions and the load introduced by the different terminals), it can compute a suitable policy for computing assigned resources and a cap on the maximum modulation and coding scheme, which limits the stress on the computing platform.

4.1.5 Autonomous service scaling: Adapting a known reward function to networking

The NI design guidelines set forth next are relevant to the following requirements of DAEMON, which were presented in Section 4 of D2.1 [1], updated in Section 2 of this document, and which are reported in full in Appendix A of this same document:

- FR-SLMANO-003 for Self-learning MANO, FR-MTERM-007.00 for Multi-timescale resource allocation.

In NFV-based networks, it is of vital importance to fulfill the Service Level Objectives (SLOs) of different services. By assigning more resources, a network operator can cope with the requirements imposed by the network services. Scaling is a challenging problem, mainly because it decides the exact amount of resources that a running service requires to meet an SLO. Recently, ML strategies are being proposed for flexible resource scaling in NFV-based networks, given their ability to learn from data and past experiences. Reinforcement Learning (RL) is also explored as a solution for scaling network resources. An agent's objective in RL is to learn a policy that maximizes an expected reward function by interacting with an environment through actions. Following the learned policy, the agent proactively adapts the network resources, similar to predictive auto-scalers, but without any a priori knowledge of the system.

However, designing a reward function requires expert knowledge, since the agent will try to follow the least effort path, i.e., the sequence of actions that maximize the reward. If wrongly designed, an RL agent can be led to wrong behaviors that do not match the expected behavior [22], which can be costly in networking applications.

Contrary to most of the RL applications in networking, where the states, actions, and reward function are defined using a networking rationale, ***the DAEMON project commends that the many different reward expressions used in known applications of Reinforcement Learning can be leveraged to identify suitable rewards that drive NI decisions in specific network functionalities***. Specifically, in the context of the activities of the project, we map the auto-scaling problem to known applications of RL [23], via the Gym Open-AI project that provides a set of classical problems for RL algorithm benchmarking. We noticed that our problem closely resembles the *Cart-Pole*[1] environment. In our problem, the agent tries to guarantee a given SLO by taking discrete actions (i.e., increase, decrease or maintain). Similarly, in the *Cart-Pole*, the cart tries to keep the pole upright by taking discrete actions (i.e., go left or right). These results were reported in the document D5.1 of the DAEMON project [4].

Following the same rationale as in the *Cart-Pole* environment, we define the information retrieved by the monitor as the network state. At time step t, the state $s^{(t)}$ is defined as:

- Mean CPU usage among the active VNFs
- Mean number of jobs waiting in the queue
- Peak (maximum) latency from the active VNFs
- The number of active VNFs.

Based on this information, a Deep Q Network (DQN) agent decides if the number of VNF instances must be increased, decreased, or kept the same. The reward function is also defined in a similar way as in the *Cart-Pole* problem. Our agent takes discrete actions to maintain a given continuous variable (e.g., latency) at a certain level. Consequently, the agent is rewarded if the actions are leading towards that goal. More specifically, the reward function at time step t is defined as

$$r^{(t)} = \begin{cases} 1 & \left|d^{(t)} - d_{tgt}\right| < \epsilon \cdot d_{tgt} \ \vee \ \left|cpu^{(t)} - cpu_{tgt}\right| < \epsilon \cdot cpu_{tgt} \\ 0 & \left|d^{(t)} - d_{tgt}\right| \geq \epsilon \cdot d_{tgt} \ \vee \ \left|cpu^{(t)} - cpu_{tgt}\right| \geq \epsilon \cdot cpu_{tgt} \\ -100 & in\ episode\ termination\ cases \end{cases}$$

In this reward function, $d^{(t)}$ is the peak latency from the active VNFs at time step t (taken from the network state), d_{tgt} is the target latency as defined by the SLO, and ϵ is a range of tolerance (e.g., 20%). Note that, if the reward function is only defined based on the perceived latency, the agent will take the most obvious action: to keep increasing the number of VNF instances, disregarding the economic impact of such a decision. To keep the number of VNF instances at an adequate level, we also let the agent be rewarded if the current CPU usage is within a predefined range. If the CPU usage is low, probably the workload can be served using fewer VNFs and vice versa. Moreover, the agent is hardly penalized if it incurs in episode termination situations. We defined two situations where the episode is terminated: when the agent creates more than 20 VNFs or the jobs that are waiting in the queue (overload) are above 200. These two situations represent wrong agent behavior and must be penalized. In such situations, the episode ends, and a simulation is restarted.

[1] https://gym.openai.com/envs/CartPole-v1

4.1.6 Low inference time and low energy-consuming NI

The NI design guidelines set forth next are relevant to the following requirements of DAEMON, which were presented in Section 4 of D2.1 [1], updated in Section 2 of this document, and which are reported in full in Appendix A of this same document:

- NFR-RIS-001 and NFR-RIS-002 for Reconfigurable Intelligent Surfaces Control NI;
- NFR-EAWVNF-003 and NFR-EAWVNF-004 for Energy-aware VNF Orchestration NI;
- NFR-CAWRS-000, NFR-CAWRS-001, and NFR-CAWRS-003 for Compute-aware Radio Scheduling NI.

Extremely low inference latency and energy consumption is a requirement for NI models in a number of mobile network applications such as traffic classifiers or load balancers in multi-gigabit-per-second backhaul segments, or in baseband processing operations in the radio interfaces. In such network intelligence functions, the processing latency budget for inference is well below 100 microseconds. Most of the existing AI/ML solutions are resource-demanding and do not consider so stringent constraints for the inference task***. In DAEMON, we advocate for the development of tailored highly-efficient ML solutions that focus on the said limiting processing latency while minimizing the loss of performance.***

Hereafter, we analyze three techniques to meet such tight requirements:

1. Low complexity. Evidently, the computational complexity of an NI model has a strong correlation with its latency performance and energy consumption. Single-agent models require consolidating all the state/context information across all the users in the system and all the actions involving each of these users in a single model (e.g., a neural network). As a result, this approach usually renders scalability problems and overly complex models with poor inference performance (in terms of latency and energy consumption) and overly long training periods. A more natural approach to reduce the complexity. This has been well studied in the literature and techniques such as distributed learning and multi-agent models have been proposed to alleviate this problem. This type of approaches let us distribute the complexity of the problem across individual agents. To avoid stability or fairness issues, multi-actor-single-critic methods with attention mechanisms can be used to integrate some notion of coordination across agents, as discussed in [24].
2. In-subsystem inference. Having the feature extraction and inference steps of a NI solution in different subsystems (e.g., CPU and GPU), as is common in machine learning applications, requires moving data through PCIe buses, which incurs a latency toll. Moreover, CPUs and GPUs require batching input data to improve the per-core processing efficiency. This improves data locality, avoiding stalls in the CPU pipeline due to data read delay, and it allows to fill vector processing registers; but this comes at a cost in latency as well. A good solution to this problem is to perform in-subsystem inference, i.e., directly within the subsystem (CPU or network interface card (NIC)) that collects the input data. This has been demonstrated, e.g., in [24], where it is shown an 18x increase in latency performance when using a common pipeline of NIC+CPU for data collection and ML inference, compared to performing both steps directly on the NIC (see Figure 15).

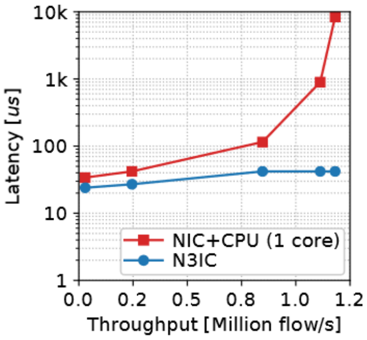


***Figure** 15. Inference latency comparison as function of the input load.*
This figure is reproduced from [24]. for a simple model performing feature extraction and inference on two different subsystems (NIC + CPU) or both steps directly on a NIC (N3IC).

3. Binarized Neural networks. The most popular type of neural networks used in ML solutions use 32 bits to encode each weight and activation, which often incurs in prohibitive latency and energy consumption. A solution to this problem is to reduce the precision of the model in a process called quantization. Binarized Neural Networks (BNNs), first proposed by [25], represent the extreme case of this process, where both weights and activations are restricted to two possible values: -1 and +1. As shown in [25], such extreme quantization requires special tools for training. First, compared to common training techniques based on backpropagation and stochastic gradient descent, the gradient of the binarization function often vanishes; second, the weight space {-1, 1} cannot absorb

small update steps. Using the appropriate training tools, BNNs show a great potential to provide extremely low-latency inference and low energy consumption. Compared to an equivalent 8-bit quantized network, BNNs require 8 times smaller memory size and 8 times fewer memory accesses, with drastic gains on optimized hardware, e.g., exploiting SIMD extensions on intel or AMD CPUs.

4.1.7 Anomaly detection

The NI design guidelines set forth next are relevant to the following requirements of DAEMON, which were presented in Section 4 of D2.1 [1], updated in Section 2 of this document, and which are reported in full in Appendix A of this same document:

- FR-AARES-000, FR-AARES-001 and FR-AARES-003 for anomaly detection.

B5G/6G infrastructures are deployed to serve diverse verticals. These verticals will be requesting a set of services, which can generate diverse traffic profiles. For instance, a utility can generate traffic from sensors, from video streams, or even audio; these streams can vary in requirements, for instance in terms delay and reliability.

In such a heterogeneous environment, things do not always operate as they should, or as agreed. Therefore, systems should be prepared to handle situations that exhibit a behavior beyond what is "ordinary", agreed, etc. This is important for preserving the services of all verticals served by a network segment. The reason for the unordinary behavior can be due to malevolent reasons or to something else, e.g., some device malfunctioning, extraordinary requirements from the vertical, or underestimation (or overestimation) of resources. In other words, anomalies can be security incidents, may indicate faulty sensors, or may be related to aspects of interest to the vertical domain (i.e., the concern is on the data and on the control plane).

In light of the above, a key problem that needs to be solved can be generally stated as follows: "Given (a) an area / network segment, (b) the verticals / services supported in the area and their anticipated / agreed behavior in time and space, and (c) the network configuration set up for supporting the services, find the sources that exhibit and unordinary behavior".

The problem above can fall in the class of problems that is generally called "anomaly detection". The problem has attracted attention in various domains and for various solution approaches, such as unsupervised learning and pattern recognition, which are representative solutions that enable the experimentation with diverse levels of data availability. Yet, the direct application of these approaches in a use case with highly particular characteristics such as network management does not provide robust and powerful outcomes; and the preliminary experiments indicate that the direct application of standard AI/ML approaches is not enough to obtain the required network KPIs.

Therefore, in the DAEMON project, we are focusing on finding possible adapted solutions to the particularities of network functioning, and the results will be reported in the next deliverables of the other Work Packages of the project. As a result, guidelines for tailored AI design for anomaly detection are not yet included in Table 13 reported at the beginning of the section.

4.2 Limits of AI for NI

One of the cornerstones of the DAEMON project is its critical approach towards AI, intended as complex data-hungry black-box models based on deep learning and as a sliver bullet to solve any task in network management. Following this stance, the project is exploring the limits of AI in the case of the eight NI-assisted network functionalities targeted in the DoA, so as to identify potential limitations of AI in such practical tasks. At the same time, we are investigating alternative methods that allow to broaden the spectrum of learning and optimization tools that are best suited to concrete networking problems, including classical statistical models, simple ML techniques, optimization tools, or heuristics. Such tools can be employed in stand-alone approaches or jointly in hybrid approaches if the latter are found to work better for the functionality at hand.

In this section, we summarize the results of the activities in the project that aim at responding to the question: *"When should be AI preferred to (or combined with) other approaches in order to maximize the efficiency and performance of NI?"* We therefore provide the insights and conclusions that outcome from the research derived in the context of DAEMON about whether/when AI is the most appropriate solution to network management problems. Next, ***we provide a list of the current outcomes of the research conducted within the DAEMON project and aimed at understanding limitations of AI/ML solutions by demonstrating that other classes of models are better suited to empower the NI-assisted network functionalities we target***. We link these adaptations to the requirements of the target functionalities described in Section 0. As in the case of the guidelines for a tailored design of AI for NI, the connection between requirements and these new guidelines is bidirectional, as (i) the functional requirements set the constraints that the guidelines fulfil, and (ii) the evaluation of solutions built on the guidelines allows for revealing limitations of the requirements, which shall be updated accordingly.

Table 14 summarizes the seven guidelines produced by the project to date, indicating the requirements they relate to and providing a brief description of their key message. Full details on each guideline are then presented in the remainder of this section. Note that the focus there is on the extrapolation of the design guidelines of AI for NI, which arise from the activities carried out during the first iteration of the DAEMON project. Therefore, when applicable, we also link guidelines to their implementation for some specific NI-assisted functionality that are presented in other deliverables of the project.

Table 14. *Summary of the DAEMON project's guidelines on the limits of AI for NI.*

Guideline for	Requirements	Description
Traffic classification	FR-MTERM-001.01 FR-MTERM-006	For unencrypted traffic, in DAEMON we propose the use of simple statistical algorithms for traffic classification of unencrypted data, as we have shown that they perform as well as complex AI/ML approaches. In such situation, the statistical approaches are preferred due to the huge difference in complexity.
Wireless Network performance inference	FR-MTERM-001.01 FR-MTERM-006	While ML approaches outperform classic mathematical approaches that rely on simplified assumptions, the former suffer from the limitations on the fixed size of input and scalability. It has been shown that hybrid approaches based on machine learning algorithms that make use of graph theory clearly improve the performance over both standard ML and mathematical solutions.
Self-learning MANO	FR-SLMANO-000	In auto-scaling of virtual resources, it has been proven that classical control theory approaches outperform RL-based controllers in terms of the trade-off between resource requirements and QoE. It turns out that the flexibility that the RL approach brings incur the cost of having a lower performance.
Forecasting in mobile networks	FR-CFORE-000	While there has been an extensive research on ML approaches for forecasting, showing that such approaches usually outperform more classical statistical solutions, in DAEMON we have proposed a truly hybrid approach that takes the best from both paradigms, and which improves the results of state-of-the-art predictors. The concept is simple: instead of applying a global normalization of the traffic time series before it is input to the DNN predictor, a dynamic normalization is performed at each time step; the level used for such a dynamic normalization is decided by a statistical model. Both the DNN and the statistical model's parameters are trained through the same gradient descent mechanism. From this, as well as the second point of this table, DAEMON advocates for the use of hybrid solutions that provide synergistic gains.
In-backhaul inference	FR-IBSSI-002 NFR-IBSSI-000 NFR-IBSSI-001	The feasibility of realizing inference in programmable user planes at line rate is a challenging network environment for NI, because of the strong limitations of the programmable switch's hardware. In such application, highly-elaborated, complex non-interpretable deep learning models for the user-plane tasks analyzed provide a similar performance as much simpler and interpretable tree-based approaches. The DAEMON project advocates the use of Random Forest models instead of other approaches, including those based on deep learning, for in-backhaul inference. Indeed, apart from not achieving a better performance, neuron-based approaches are challenging to implement in resource-constrained programmable switches.
Federated learning powered NI functionalities	FR-SLMANO-000 FR-SLMANO-003 FR-AARES-000 FR-AARES-001	While the main question is whether ML should be preferred to non-ML-based approaches for some NI applications, another related question is which ML framework should be considered, which also falls within the questions about the best practices and limits of each of the AI frameworks. For example, in DAEMON we recommend the use of Federated Learning (FL) over centralized or distributed learning for applications that require several intelligent agents acting cooperatively. In cases where the decisions taken at distant parts of the network are intertangled and impact each other. FL allows for a low response time due to the existence of the local module, and a high scalability due to the exchange of limited traffic between FL clients and the FL controller.
Predictive HARQ	NFR-CAWRS-000 NFR-CAWRS-001 NFR-CAWRS-003	Predictive HARQ is a network application that required extremely low latency, while maintaining both ultra-high accuracy and low false positive rate. Complex ML-based algorithms fail to provide performance guarantees, and they consume excessive time in the inference task. In DAEMON, we have found clearly identifiable patterns that distinguish decodable and non-decodable code blocks, which can be detected through simple algorithms with minimum computation delay. Hence, in DAEMON we suggest the use of simple statistical or control-theory approaches to implement predictive HARQ and other ultra-low-latency-inference applications.

4.2.1 Traffic classification

The NI design guidelines set forth next are relevant to the following requirements of DAEMON, which were presented in Section 4 of D2.1 [1], updated in Section 2 of this document, and which are reported in full in Appendix A of this same document:

- FR-MTERM-001.01, FR-MTERM-006 for Multi-timescale resource allocation.

Managing and optimizing the network capacity to provide Quality of Service (QoS) becomes even more challenging due to the omnipresence of wireless networks with increased network capacity to support the ever-increasing number of devices and applications [26] [27]. Traditionally, the Network Monitoring Service (NMS) performs a set of tasks to analyze the behavior of the networks and services throughout their traffic. The information provided by this system can be used to determine which applications affect network the most in terms of total bandwidth usage or to identify the most critical links, such that decision-making engines for network management can ensure fast troubleshooting and secure high QoS to the users. Focusing on Traffic Classification (TC), this network management task allows us to infer the application that is generating the traffic [27] [28]. Knowing the traffic class provides a mechanism to enforce specific security and QoS policies on the analyzed traffic.

In the recent years, TC based on Deep Learning (DL) have outperformed the ones that are empowered by traditional methods such as port-based, Deep Packet Inspection (DPI), and flow-based traffic analysis using statistical Machine Learning (ML), and this result is more evident on traffic that is encrypted [29]One of the reasons for such success is that DL models can automatically extract the features required to solve the given classification task with high accuracy from a large amount of raw data [30] . Many proposed DL-based TCs achieve high accuracy on raw packets but lack interpretability. Therefore, the lower the interpretability of a ML model is, the harder is to humans to understand why some decisions or predictions have been made. We believe that this lack of interpretability of a ML-based TC may lead to several risks for the systems that consume such information, e.g., billing, when deployed in real environments, since it would be hard to identify the root cause of poor performance in such systems. Moreover, as DL models are computationally expensive, using them as "one size fits all" may leave them useless due to poor response time.

Motivated by the lack of interpretability of DL models for packet-based TC on raw input data, [31] presented a set of experiments to measure the performance (e.g., prediction accuracy) of DL-based TCs when the input representation of the packet uses different combinations of the IP and transport protocol headers and payloads. Based on the evaluation results, it was concluded that the DL models are reduced to a statistical IP/port-based architecture whose features are automatically extracted from raw (byte) representation of the packet headers. Moreover, when analyzing the transport payload, it was found that this feature does not improve the performance of the DL models. On the contrary, the transport payload may negatively impact the performance. In general, we can see how a combination of data analysis, feature engineering and domain knowledge allow building ML models with high interpretability, high accuracy, and low computational complexity for non-encrypted packets. Features like packet lengths, transport protocol type, and source-destination port pairs are enough to solve the traffic classification tasks.

Focusing on wireless encrypted packets, [32] performed several experiments on three traffic classification tasks. In the first task, the TC algorithm uses L1 packets to determine if the transmitted packet is a Management, Control, or Data L2 frame in 802.11. In the second task, the TC algorithm uses L1 packets to determine the type of application inside the transmitted packet (e.g., audio or video). In the third task, the TC algorithm discriminates between the actual applications generating the L7 traffic. The experiments result showed that a DL model based on Convolutional Neural Network (CNN) could achieve the best performance on the three proposed tasks, achieving above 99.9% in task accuracy discriminating among classes in task 1, 97.8% in task 2, and 92% in task 3. Moreover, the CNN outperformed a state-of-the-art Recurrent Neural Network (RNN) when using shorter IQ sequences. The performance of the RNN might improve with longer sequences at expenses of increasing the training time.

In summary, when dealing with encrypted traffic, the automatic feature extraction procedure of DL models helps in the Traffic Classification task, given that the features used in simple ML models based on statistical ML are not enough to properly identify among traffic classes. On the contrary, ***in unencrypted traffic, DL models are behaving as simple statistical IP/port-based architecture and can be replaced by more simple ML models***. TC can be seen as a time-series analysis. Traditionally, RNNs are well suited for this kind of problem. However, when dealing with encrypted wireless traffic, a CNN outperforms an RNN for shorter IQ sequences. This facilitates the training process and produces shorter prediction times, which may be compelling for integrating into spectrum-based real-time traffic analyzers.

4.2.2 Inferring wireless networks performance using Graph Neural Networks

The NI design guidelines set forth next are relevant to the following requirements of DAEMON, which were presented in Section 4 of D2.1 [1], updated in Section 2 of this document, and which are reported in full in Appendix A of this same document:

- FR-MTERM-001.01, FR-MTERM-006 for Multi-timescale resource allocation.

The 5G and beyond 5G networks require a complex performance analysis and monitoring to support the stringent demands of QoS Quality of Experience (QoE) for new services. The modeling of concrete QoS parameters such as delay, jitter, loss, throughput, and other network's KPIs for each service is crucial to develop prediction models that decision-making algorithms can use to manage and control the dynamic adaptation of the network in a more efficient way and achieve the performance requirements.

As network-oriented applications evolve toward increasingly stringent QoS/QoE requirements, Wi-Fi amendments have defined new strategies that improve the offered bandwidth to users. Notwithstanding the novel mechanisms for next-generation Wi-Fi deployments, addressing the demands of highly dense scenarios adds more stress to the already scarce spectrum, as more devices will fight for medium access. To predict its performance, controllers or Access Points (APs) in a wireless network can use network models to assess how positively or negatively a given decision will impact the performance of the network. Typically, Markov models [33] and mathematical models [34] have been used to characterize wireless networks. However, given the increasing heterogeneity in network services and functionalities that are developed to accommodate the extreme imposed requirements, traditional mathematical models do not hold. On the one hand, the Markov chain states increase exponentially with the number of considered devices and their configurations. On the other hand, mathematical approaches rely on simplifying assumptions in order to keep tractability. Moreover, new services and technologies alter the traffic patterns, forcing prediction models to be continuously updated. Therefore, new models are needed that adapt to varying network conditions and environments.

ML models are powerful tools for different aspects of network optimization, as they avoid making *a priori* assumptions, typically found on analytical models. However, ***traditional ML models faces challenges when learning from structured data represented as graphs, as the relationships among nodes are not captured or have to be represented differently***. For example, typical data pre-processing steps include the generation of a fixed-size matrix (e.g., images of a given size, audio samples or sentences of a fixed length), which represents the sample's information and serves as training data. On the contrary, approaches such as Graph Neural Networks (GNNs) [35] have been proposed as neural networks that operate on graphs.

The work in [36] evaluated a GNN model that adapts well in graph-based problems that exhibit combinatorial behavior. Additionally, the GNN model was compared to more traditional ML models and analyzed the impact of different features on the model's performance. Depending on the available data, a controller might use a trained model with a given set of features or another. According to the evaluation, the GNN approach can obtain a 64% increase in the performance regarding a naive approach and around 55% for other ML approaches when using all training features.

4.2.3 Self-learning MANO – reinforcement learning

The NI design guidelines set forth next are relevant to the following requirements of DAEMON, which were presented in Section 4 of D2.1 [1], updated in Section 2 of this document, and which are reported in full in Appendix A of this same document:

- FR-SLMANO-000 for Self-learning MANO.

For autonomous service scaling (a key component of self-learning MANO), the number of VNFs needs to be scaled according to the work that is offered. The decision to add or remove a VNF often needs to be made based on the observed QoE, e.g., the latency incurred in processing the work. Scaling algorithms can rely on reinforcement learning (RL), see Section 4.1.5, or on control theory (CT). In the latter, the observed latency is compared to a threshold and the latency trend is observed: if both are sufficiently positive, the number of VNFs is increased; if both are sufficiently negative, the number is decreased. This is often referred as a proportional integral (PI) controller, which is a simplified form of the general PID (i.e., proportional, integral and derivative) controller that reduces the derivative control part because it is very sensitive to feedback noise. In contrast to the RL approach, which has a neural network at its core with as many parameters as there are synapses, the CT approach has only a few parameters. A conceptual figure of a PI controller is in Figure 16, where K_p and K_I are the proportional gain an integral gain, respectively. These two parameters are key of designing the CT approach for the considered process.

It turns out that ***the CT approach outperforms an RL based controller in terms of the trade-off resource requirements versus QoE***, at the expense of carefully training the parameters of the CT algorithm. In other words, the flexibility that the RL approach brings comes at the cost of having a lower performance.

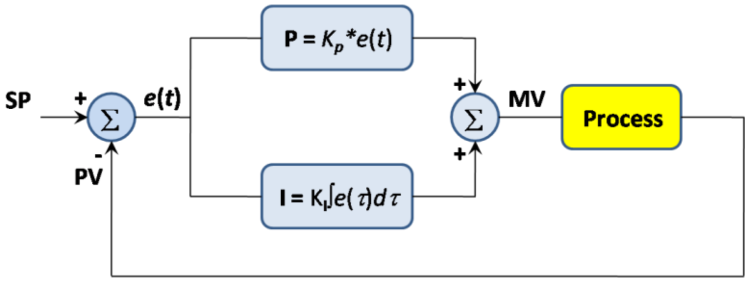


***Figure 16**. General PI Controller.*

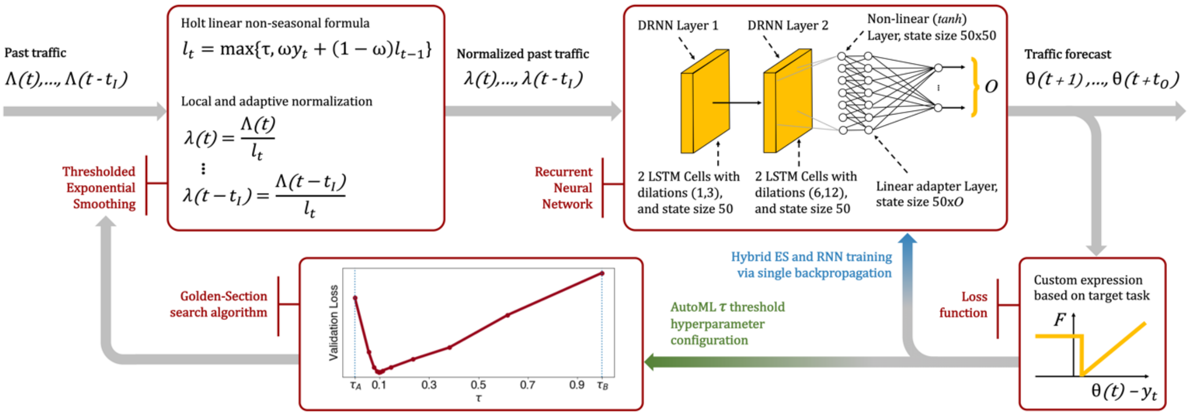
4.2.4 Forecasting in mobile networks

The NI design guidelines set forth next are relevant to the following requirements of DAEMON, which were presented in Section 4 of D2.1 [1], updated in Section 2 of this document, and which are reported in full in Appendix A of this same document:

- FR-CFORE-000 for Capacity Forecasting NI.

Current trends in forecasting for anticipatory networking lean towards the systematic adoption of models that are purely based on deep learning approaches, with a fairly large body of works that have explored varied Deep Neural Network (DNN) architectures for diverse forecasting objectives. Thanks to the proven gains of DNNs in terms of accuracy over legacy statistical models, many architectures have been proposed, which rely, among others, on Long Short-Term Memory (LSTM) [37] [38] [39], Stacked AutoEncoder (SAE) [37], and Multi-Layer Perceptron (MLP) [40] structures. Convolutional layers have also been extensively tested, in their baseline [38] [39], three-dimensional [40], or graph [41] versions.

However, very recent results from the machine learning community suggest that hybrid engines integrating statistical modelling and DNN can in fact substantially outperform pure DNN approaches in time series forecasting tasks [42]. ***In DAEMON, we adapt a hybrid statistical-learning paradigm to the problem of forecasting for Network Intelligence (NI), and demonstrate the superior performance of such a model over pure DNN solutions***, including state-of-the-art dedicated DNN-based predictors from the literature. Specifically, we propose a model named Thresholded Exponential Smoothing with Recurrent Neural Network (TES-RNN), which is a general-purpose network traffic forecasting technique that can be tailored to perform predictions for different NI functions. The concept behind TES-RNN is simple: instead of applying a global normalization of the traffic time series before it is input to the DNN predictor, a dynamic normalization is performed at each time step; the level used for such a dynamic normalization is decided by a statistical model, whose parameters are optimized jointly with those of the DNN during training.



Past traffic
$\Lambda(t),\dots,\Lambda(t-t_I)$
Holt linear non-seasonal formula
$l_t = \max\{\tau, \omega y_t + (1-\omega) l_{t-1}\}$
Local and adaptive normalization
$\lambda(t) = \frac{\Lambda(t)}{l_t}$
$\vdots$
$\lambda(t-t_I) = \frac{\Lambda(t-t_I)}{l_t}$
Thresholded Exponential Smoothing
Normalized past traffic
$\lambda(t),\dots,\lambda(t-t_I)$
Recurrent Neural Network
DRNN Layer 1
DRNN Layer 2
Non-linear (tanh) Layer, state size 50x50
O
2 LSTM Cells with dilations (1,3), and state size 50
2 LSTM Cells with dilations (6,12), and state size 50
Linear adapter Layer, state size 50xO
Traffic forecast
$\theta(t+1),\dots,\theta(t+t_O)$
Hybrid ES and RNN training via single backpropagation
AutoML τ threshold hyperparameter configuration
Loss function
Custom expression based on target task
F
$\theta(t) - y_t$
Golden-Section search algorithm
Validation Loss
τ_A
0.1
0.3
0.5
0.7
0.9
τ_B
τ

***Figure 17**. Architecture of the hybrid TES-RNN model proposed by DAEMON.*
The traffic observed over the past time steps is input to the TES component for a local and adaptive normalization of non-negligible traffic above a threshold τ. The resulting traffic λ is fed to the RNN component, which outputs a forecast θ of traffic within a horizon of O time steps. During training, the loss computed from θ is used to learn the threshold hyperparameter τ of the TES block in an AutoML style via a Golden-Section search.

The structure of the proposed TES-RNN model is depicted in Figure 17. It combines a classical Exponential Smoothing (ES) statistical model with a Recurrent Neural Network (RNN) architecture, following current recommendations for machine learning design towards time series forecasting [43] . For the ES part, our model employs a Holt linear non-seasonal ES formula [44] , which is recommended for time series with daily periodicity as those often encountered in mobile network traffic [43]. For the predictor part, the TES-RNN model uses a Dilated Recurrent Network (DRNN) [45], which, unlike vanilla Long Short-Term Memory (LSTM), realizes an RNN attention mechanism. We remark how TES-RNN is a true hybrid model, since the parameters of the ES part are optimized concurrently with the RNN weights using a unified gradient descent. Thanks to this joint training, the ES-RNN model represents a leap forward with respect to previous attempts at mixing different statistical and/or machine learning methods: unlike simple combination [46] or ensemble [47] strategies used to date, this technique takes full advantage of the strengths of statistical and machine learning methods, while mitigating their respective limitations.

An important element is the threshold τ used to bound the minimum value of the ES level. This is a key parameter that avoids low-traffic situations (often encountered in antenna-level demands) to cause ES normalization by extremely low values, which we prove to disrupt the hybrid predictor's operation. The configuration of this threshold is in fact challenging, as it introduces an interesting trade-off: in general, a threshold closer to the traffic peak ensures higher robustness to low-traffic situations; however, it also forces a less dynamic normalization that limits the benefits of the ES. In order to identify the best value of τ, we employ an AutoML approach, implemented as a Golden-Section search algorithm: at each step, the validation loss F(τ) used to train the ES and RNN models is also computed for two threshold values inside the search interval: the threshold yielding the higher validation loss is used to update either the left or the right extreme of the interval. The algorithm iterates until the length of the search interval falls below a target tolerance: then, the final τ is the mean of the last interval.

Full details on the TES-RNN design and operation are available in [48], and a preliminary performance evaluation showing the advantages of a hybrid design over legacy DNNs is presented in Section 4.6.1 of D5.1 of the DAEMON project [4]. Overall, by proposing this very first hybrid approach to forecasting for NI, DAEMON paves the way for a different strategy to the design of predictors for mobile network environments.

4.2.5 In-backhaul inference

The NI design guidelines set forth next are relevant to the following requirements of DAEMON, which were presented in Section 4 of D2.1 [1], updated in Section 2 of this document, and which are reported in full in Appendix A of this same document:

- FR-IBSSI-002, NFR-IBSSI-000, NFR-IBSSI-001 for In-backhaul Support for Service Intelligence NI.

As part of the project activities, we investigate the feasibility of realizing inference in programmable user planes at line rate. This is a challenging network environment for NI, given the strong limitations of the programmable switch hardware, as detailed in Section 7.1 of D3.1 of DAEMON [5].

Based on the results of extensive tests with multiple real-world use cases for network traffic classification and anomaly detection, ***the DAEMON project advocates the use of Random Forest models instead of other approaches, including those based on deep learning, for in-backhaul inference***. Indeed, we did not identify any significant advantage in relying on complex non-interpretable deep learning models for the user-plane tasks analyzed: simpler approaches based on multiple decision trees achieve an accuracy that is similar or even superior in such tasks. Instead, we found that deep learning models are much more challenging to implement in resource-constrained programmable switches, which dramatically limits their internal complexity (e.g., in terms of layer depth or number of neurons per layer) and thus an inference potential that is classically largely dependent on their architectural complexity.

Results supporting the guideline above are available in Section 4.5.1 of D5.1 of the DAEMON project [4].

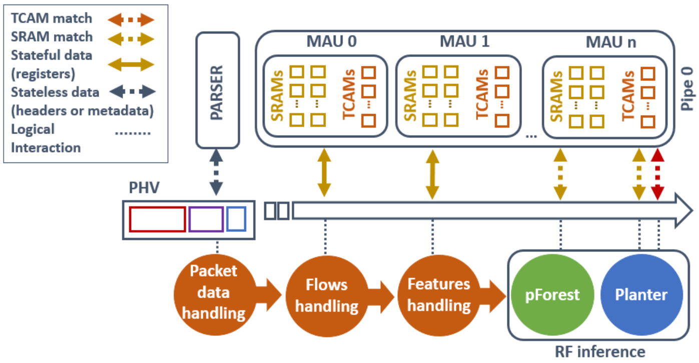


Figure 18. *Proposed framework for in-backhaul inference.*

In order to validate the approach based on Random Forest models, we thoroughly analyzed the two most interesting proposals that we found in literature, namely pForest [49] and Planter [50]. Since no public implementation of either solutions was available, we reproduced the two works. Furthermore, as the original methods target flow-level versus target-level classification, we developed a unified framework for Random Forest based inference (that can be fully implemented on programmable switches) to provide a fair comparison. Figure 18. shows the proposed framework, whose workflow is largely aligned with pForest paper [50]. It consists of four phases: (i) Packet data handling refers to the extraction by the Parser of relevant packet-level features from the incoming packet, which are stored into the Packet Header Vector (PHV) to be carried through the whole forwarding pipe. (ii) Flow handling is the process that assigns a flow identifier via a CRC32 hash checksum of the 5-tuple composed of IP source and destination addresses, source and destination ports, and transport protocol identifier. Different CRC16 and CRC32 hash codes of the same tuple are used to store such a flow identifier into a flow table of the switch MAUs. Allowing to store information about whether the flow has been already observed or not. (iii) Feature handling is responsible of the compression of the flow-level features with a technique proposed in pForest. (iv) RF inference is the final phase of the workflow, where the packet-level and the flow-level features stored into the PHV are run across a Random Forest model encoded into the MAU pipeline.

The framework ultimately realizes a generic RF-based early flow inference in the switch, and it can be configured to reproduce any binary RF model.

4.2.6 Federated learning powered NI functionalities

The NI design guidelines set forth next are relevant to the following requirements of DAEMON, which were presented in Section 4 of D2.1 [1], updated in Section 2 of this document, and which are reported in full in Appendix A of this same document:

- FR-SLMANO-000 and FR-SLMANO-003 for Federated Learning powered Controller.
- FR-AARES-000 and FR-AARES-001 for anomaly detection.

Most of the advances in ML approaches are based on the idea of a single intelligent agent that computes and executes the learning process. When we consider multi-agent environments, the most direct approach and the most considered in initial advances is the *distributed* version of the single-agent approach, where each agent acts independently and attempts to learn a selfish (or common) goal at the same time as all the other agents. In addition to applying these legacy approaches (centralized or distributed) to power NI functionalities, another approach is to base the solution on the use of Federated Learning (FL) [51]. ***The FL approach can be used for reasons of scalability and data protection***. Scalability is crucial for many DAEMON functionalities and applications. For example, if we consider anomaly detection at some section of the network, there may be an enormous amount of traffic sources to analyze. Likewise, the aspect of keeping data locally is important, either due to privacy and security reasons, or due to delay constraints. In addition, the exchange of a limited set of data keeps the throughput between local and central segments assuming a high number of data sources.

In the FL approach, the distributed knowledge bases contain local information regarding the performance of the model. Certain information is communicated to a centralized controller of the FL model, and such controller will be tuning the algorithm in accordance with the FL paradigm.

For example, if we focus on anomaly detection, the FL approach provides several advantages with respect to other ML approaches. While the discussion on Section 4.1.7 revolve about whether the application of ML requires modifications to achieve good performance in network's anomaly detection task, here we comment on *the type* of ML approach that is suitable for the same application, even if such approach requires further refinements and tailoring to unveil its potential. In this case, FL is the recommended solution over RL other centralized and distributed choices, since it provides the following advantages:

- Low fault tolerance in anomaly detection process due to FL enhancement.
- Low response time due to existent of the local anomaly detection module.
- High scalability due to the exchange of limited traffic between FL clients and FL controller.
- Access to local database for the execution of anomaly detection process, while keeping small central database in the FL controller.
- Low network traffic exchanged between FL clients and FL controller.

4.2.7 Predictive HARQ

The NI design guidelines set forth next are relevant to the following requirements of DAEMON, which were presented in Section 4 of D2.1 [1], updated in Section 2 of this document, and which are reported in full in Appendix A of this same document:

- NFR-CAWRS-000, NFR-CAWRS-001, and NFR-CAWRS-003 for Compute-aware Radio Scheduling NI.

Hybrid ARQ (HARQ) is an essential operation at the physical layer of a 5G Distributed Unit. Predictive HARQ enables the inference of the decodability of Uplink data essentially using feedback from the decoder, minimizing situations where uplink subframes are discarded because they cannot be processed in time. The produced prediction allows the subsequent tasks to be performed without waiting for the decoding process to be finished.

However, the latency required to execute the inference process itself consumes part of the latency budget for the whole HARQ process. This fact limits ourselves to extremely fast-inference approaches and prevents the use of GPUs or other computing resources which incur in additional latency access. Moreover, ***complex data-driven models based on neural networks are not necessarily the best tool for HARQ operations***.

To illustrate this, we propose exploiting *extrinsic information*, which spawns organically by belief propagation algorithms used by both turbo and LDPC codes. *Belief information* is encoded into log-likelihood ratios (LLRs) and the key to iterative decoding is the sequence of a posteriori LLRs of the information symbols, exchanged every iteration between the different nodes of the decoder (each node takes advantage of the information computed by the others). Due to the nature of the decoding process, extrinsic LLRs can be good estimators of the decodability of code blocks (CBs) constituting a transport block. **Figure** 19 shows the mean of extrinsic LLRs for both undecodable (in red) and decodable (in other colors) CBs. The color of the decodable CBs indicates the maximum number of iterations required to validate their successful decoding. We can observe that the mean extrinsic magnitude of undecodable CBs is small and steady across iterations. In contrast, the decodable CBs have a growing mean extrinsic magnitude over iterations.

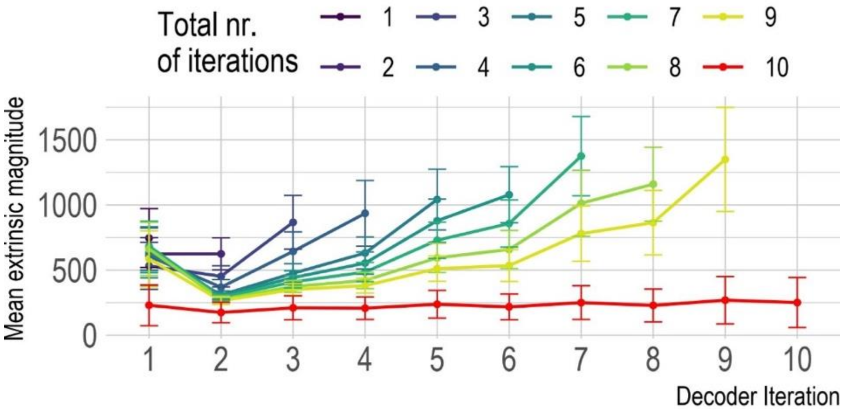


***Figure** 19. Mean extrinsic magnitude for each iteration of a turbodecoder. Dot/line indicate the average value across multiple PUSCH TBs with different MCS, TBS and SNR. Error bars indicate the standard deviation.*

These results evidence clearly identifiable patterns between decodable and non-decodable CBs, which could be detected via simple, non-ML approaches. Indeed, to effectively take advantage of the decodability forecasting, the inference time of the proposed solution must be extremely low as stated in the design constraint NFR-CAWRS-001. With the information observed in the aforementioned figure and the time requirements imposed by design, rule- or threshold-based algorithms will better fit for the design of predictive HARQ mechanisms rather than ML-based techniques.

Moreover, it must accomplish both ultra-high accuracy and low false positive rate imposed by the design requirement NFR-CAWRS-003. The main reason lies on the high cost associated to the recovery process after such situations. It usually requires recovery mechanisms from other protocol stack layers.

5 State of the art on intelligent network management

One of the duties of the DAEMON project is to be aware of the current State of the Art, as well as of the coetaneous advances that materialize during the development of the project. As a consortium, we are carrying out a literature review on the research topics in which we are currently interested. The review process was guided by a straightforward protocol that included objectives, research questions, and methodology. Regarding *the objectives*, we wanted to systematically review and analyze the current research on applied Machine Learning (ML) for solving networking problems in Mobile Network Operators (MNO). Additionally, we wanted to identify the advantages and limits of ML when solving networking problems. Regarding *the research questions*, we evaluated the ML solutions from three different perspectives: the network itself, the ML approach, and the data produced to train and test different algorithms. Finally, *the methodology* included inclusion and exclusion criteria, the labeling method, and search strings. We would like to highlight that this literature analysis is a live and on-going work, since we are constantly keeping track of new research. The results reported here consist of the current report, which follows the proposed methodology.

5.1 Research Questions (RQs)

As aforementioned, we evaluate the ML solutions from three different perspectives:

1. Networking perspective**:** The RQs were designed to identify the networking problem and where it could fit within DAEMON's architecture. Therefore, the RQs are grouped into five categories: (1) the networking problem, (2) the application area, (3) the micro-domain, (4) the location of the algorithm, and (5) its operation timescale.
2. ML approach: We classify the obtained papers regarding the ML approach. Since one of DAEMON's objectives is to set clear guidelines on how and where to deploy Network Intelligence (NI), we first need to identify how the proposed NI compares with traditional methods and the NI limitations.
3. Data characteristics: Since NI needs to be trained and tested, we need to know what kind of data is used for these two purposes and how it was obtained.

5.1.1 Network-related questions

Next, we detail each of the formulated research questions. We start with network-related questions.

1. *Which networking problems in MNOs are solved by ML?*

   This question tries to identify the networking problem. In section 5 of D2.1 [1], we suggested some representative networking problems which were used as the answer to this question. These problems are traffic prediction, traffic classification, link evaluation, fault management, intrusion, and anomaly detection. This RQ was open, meaning that every partner was free to add a new networking problem.

2. *What is the application area of the paper?*

   According to the latest 5GPPP white paper [52], three major application areas were identified, namely, (1) Network Planning, (2) Network Diagnostics, (3) Network Optimization and Control. We include Network Security inside the Network Diagnostics category, as they are correlated.

3. *Can the networking problem be placed inside any of the micro-domains identified in DAEMON's proposal? If yes, which one?*

   In DAEMON proposal, we identified several network segments. These network segments are subscriber, beyond edge, far edge, edge, transport, and core.

4. *Is it clear where the algorithm will be deployed?*

   This RQ tries to identify where the algorithm is placed. For instance, traffic classification can be made at routers (Data Plane) or at controllers (Control Plane). The categories for this RQ are data, control, and orchestration plane.

5. *What is the operation timescale of the algorithm?*

   This RQ identifies the rate at which outcomes are obtained. An outcome can be a prediction or a decision, and it is typically measured at inference. Section 2 of D2.1 [1] introduced different timescales. For example, network reconfiguration is usually performed at longer timescales (minutes to hours, hours to days) than other low-latency decisions. Consequently, their associated ML algorithms must take decisions within those time windows.

5.1.2 Machine Learning related questions

Below, we list machine learning-related questions.

1. *What is the ML method associated with the networking problem?*

ML is typically classified as Supervised, Unsupervised, and Reinforcement Learning. However, it has been identified that several combinations can be used to tackle networking problems. These combinations include Semi-Supervised Learning and hybrid solutions (e.g., a mix of mathematical programming and supervised learning).

2. *To which ML problem is the networking problem reduced?*

   The previous question can be subdivided into several categories. Typical ML problems include classification, regression, and forecasting for supervised learning, or clustering and dimension reduction for unsupervised learning [53].

3. *Which algorithm/architecture was used to solve the problem?*

   This RQ identifies which specific algorithm is behind the NI. For example, the NI consists of a Deep Neural Network (DNN), or an autoencoder, among others.

4. *Is the algorithm designed to be resource aware?*

   The network is mainly composed of resource-constrained environments (e.g., network edge). With this RQ, we wanted to identify which techniques are applied to allow the deployment of such algorithms in resource-constrained environments. For instance, some DNNs prune some of their parameters to reduce their complexity while maintaining a good prediction accuracy [54].

5. *How is the ML algorithm defined in terms of its input/output relationship?*

   This RQ describes what features are used as input and what value is generated as output. If the NI algorithm is Reinforcement Learning based, then this RQ describes the Markov Decision Process (MDP) i.e., states, actions and rewards.

6. *Which and how the loss function/reward function is defined?*

   This RQ identifies what is being optimized and how it is associated to the KPIs of the networking problem. In ML approaches, we usually consider loss functions to measure the performance of the solution; for example, some traditional loss functions for classification are probabilistic (e.g., Cross-entropy), and in regression we usually measure some distance metric from the actual value (e.g., Mean Square Error, Mean Absolute Error).

7. *Which benchmark algorithms are used to compare the performance of the NI?*

   This RQ specifies against which algorithm was compared the ML solution. Typically, the ML solutions are compared against heuristics, mathematical optimization solutions, and analytic models.

8. *Is there any discussion about the advantages of the proposed ML-based algorithm? If yes, what are the benefits of the ML approaches in comparison with the benchmark?*

   This RQ identifies the advantages of the proposed approach against the benchmark. For example, a GNN exploits better topological/relational data properties than other approaches (see section 4.2 for more details).

9. *Is there any discussion about the disadvantages of the proposed ML-based algorithm? If yes, what are the limitations of the ML approaches in comparison with the benchmark?*

   This RQ stablishes the limitations of the proposed approach against the benchmark. For example, the performance of an RNN has been demonstrated to be lower than a CNN when processing long sequences in terms of execution time and accuracy (see section 4.2).

10. *How far is the solution found by the ML approach with respect to the optimal? And/or regarding the benchmark?*

    This RQ identifies the optimality gap with respect to the benchmark algorithms. However, in some cases, the optimality cannot be determined. This fact is encountered especially in highly complex problems where the ML approach is considered precisely because the optimal solution is unknown and not possible to derive.

5.1.3 Data for ML training and testing

Finally, we list below questions concerning machine learning models training and testing.

1. *How is the dataset used for training and evaluation of the NI obtained/generated?*

   Synthetic datasets are normally generated using simulators/emulators while real datasets are generated by deploying a live testbed. The process of collection and aggregation occurs at the Mobile Network Operator (MNO) premises. The measurement data is aggregated over geographical areas, temporal intervals, or flows generated by a significant number of users, depending on which aggregation preserves data utility for the subsequent analysis carried out by the project partners.

2. *What was the network setup to generate the dataset?*

The description of the generation setup is required for a complete characterization of the work. A network setup consists of number of nodes, node capacity, network topology, among others. Additionally, existing anonymized datasets from MNO's production network will be also used.

3. *Is the dataset available? Where?*

   This RQ tries to measure how many networking datasets are available and accessible, which is a highly important feature in research due to the required reproducibility and repeatability.

4. *What are the 4V's (velocity, variety, veracity, volume) of the generated data?*

   This RQ provides the standard four-properties description that is often used to represent and characterize datasets. These four properties sufficiently describe the generated datasets, such that their quality can be verified. *Data velocity* refers to the input data rate, how fast the data is incoming to the system. *Data variety* refers to the form of data that is used for training and testing: usually, data can be represented in a structured or an unstructured form; examples of structured data are tables, time-series, whereas among the examples of unstructured data we can highlight text, images, and video [55]. Packets are normally unstructured data. *Data veracity* refers to the accuracy of the data. Finally, *data volume* denotes the amount of data available.

5.2 Methodology

The previously defined questions circumscribe the aspects of the works in which we are interested. Besides this, we also need to establish to which set of works we are going to ask those questions. Next, we proceed to provide both the inclusion and the exclusion factors that determine whether a paper or work can be considered for this review.

5.2.1 Inclusion factors

For a work to be included in the DAEMON's literature review, it should fulfil the following points.

1. The publication proposes new ML-based or hybrid algorithms to solve networking problems in one or multiple micro-domains as defined in DAEMON.
2. The publication is an academic and peer-reviewed study or a pre-print that is under review on a conference/journal that is indexed on one of the databases presented in the next point.
3. The publication is indexed in Scopus, Web of Science, IEEE Xplore, ACM, ScienceDirect, Springer, Directory of Open Access Journals (DOAJ), or JSTOR.

5.2.2 Exclusion factors

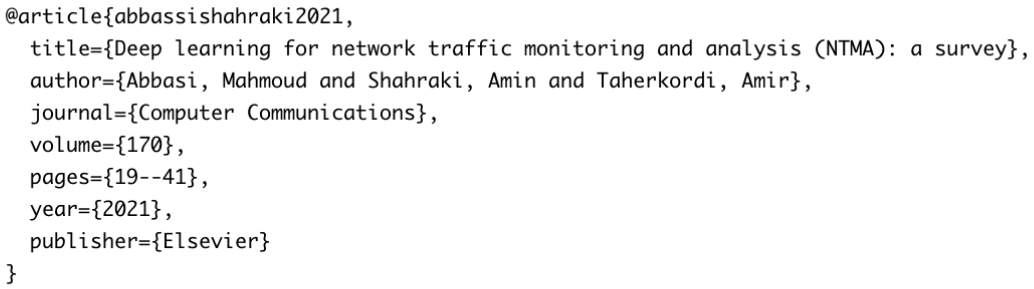
Additionally, we exclude from the search the following publications.

1. Surveys and literature reviews.
4. Papers that are published before 2018 (older than four years). The more recent survey regarding the application of ML into networking in general (not in a specific domain) is from this year [56].
2. Papers with less than six pages. Short papers and magazine papers are excluded since they typically present initial ideas without further development.
3. Papers that cannot be accessible via traditional affiliations (e.g., University). Papers that require very specific kind of access should not be part of the collection.

5.2.3 Labelling and search strings

The collected papers were included in a shared bib file using as key a combination of the last name of the first and second author and the year of publication. For example, the publication in [57] has the bib entry specified in **Figure 20**.

Each partner has focused on certain topics within DAEMON's scope for the search of works, mainly based on the corresponding area of expertise of each partner. For the sake of cooperation and coordination, each partner provides a set of search strings that determine the topics in which they center their search. These search strings collected from every partner were saved in a shared file, to ensure that two different partners were not searching the same topics.



```
@article{abbassishahraki2021,
  title={Deep learning for network traffic monitoring and analysis (NTMA): a survey},
  author={Abbasi, Mahmoud and Shahraki, Amin and Taherkordi, Amir},
  journal={Computer Communications},
  volume={170},
  pages={19--41},
  year={2021},
  publisher={Elsevier}
}
```

Figure 20. *Bibtex reference.*

5.3 Literature analysis

Following the proposed methodology in previous section, we made a first collective attempt to gather state-of-the-art research covering the topics that are of interest within DAEMON. In this first attempt, we were able to gather 39 papers in total. The number of papers is limited due to the strict exclusion factors. The detailed results from this literature review can be seen in Appendix B. In this section, we analyze the main outcomes.

As it can be seen from **Table** 15, most ML applications in networking have been focused on optimization and control at the network edge or in cross-domains (e.g., edge-cloud). Consequently, control and orchestration plane are the preferred locations to place the NI solutions (cf. **Figure** 21). One of the main findings is that most of the reviewed literature does not include the operation timescale or it operates at short timescale.

Table *15. Cross-Domain analysis of publications.*

	Network Application Areas			
Network Micro-Domain	**Network Diagnostics and Security**	**Network Optimization and Control**	**Network Planning**	**Grand Total**
Access	2	1	1	4
Core	1	1	0	2
Cross-domain	0	9	0	9
Edge	1	11	0	12
Edge and Client	0	1	0	1
Edge/core	0	4	0	4
Subscriber	1	0	0	1
Transport	3	2	0	5
UE	0	0	1	1
Grand Total	**8**	**29**	**2**	**39**

The application of NI solutions at the edge bring along a parameter optimization required to optimize different network KPIs. Consequently, every configuration needs to evaluate its impact on a network KPI (e.g., throughput or delay), such that we select the configuration providing the best network KPI.

We can see from **Figure** 22 that the overall preferred ML method is Reinforcement Learning. However, there is no clear dominant option regarding the ML algorithms, despite being RL the most used ML method, and this decision considerably varies depending on the specific application. The use of the RL setup to solve networking problems comes from the fact that, typically, these problems can be easily defined as an MDP. Given a state, the agent applies an action and obtains a reward (e.g., the network KPI).

However, the optimization method may vary, depending on the specificities of the problem. For example, in [58], the authors formulate two problems in cellular networks as a RL problem. The first problem aims to ensure that the serving base station radio link power is constantly tuned so that the target downlink Signal to Interference plus Noise Ratio (SINR) is met. To do this, the authors used tabular Q-learning as an optimization method. Tabular Q-Learning can be used in this problem since the number of states (e.g., network state, SINR) and actions (e.g., power commands) can be determined. The second problem tracks the faults and their impact to the serving base station SINR. Each fault is registered in a registry, which is set whenever a fault happens in the network and unset when the fault is cleared. Since each base station can have its own set of faults and actions to correct them, in a network with thousands of base stations, tabular Q-Learning does not scale. Therefore, the authors use Deep Q-Networks (DQNs) as an optimization method.

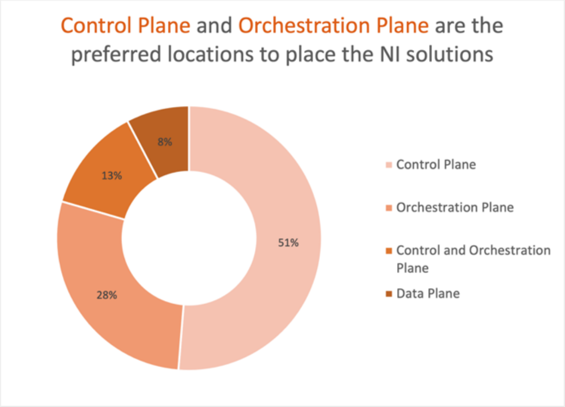


***Figure** 21. Algorithm Location.*

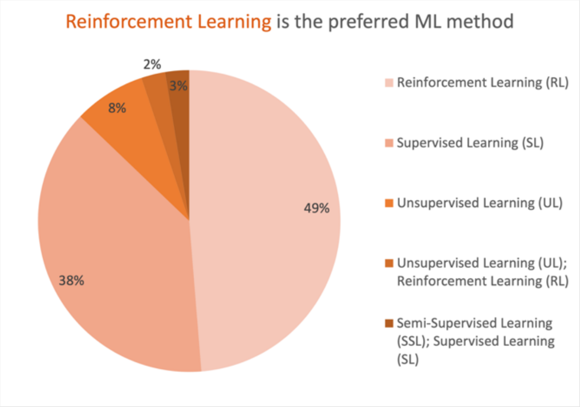


***Figure** 22. ML Method.*

Another interesting finding regarding the ML approach is that, despite being located on the network edge, where most of the deployed devices are resource-constrained (i.e., not enough computing power), most published papers are not resource-aware. Compression, pruning, and quantization are recent techniques that reduce the learning model complexity. The complexity is reduced by removing non-essential parameters to the model performance, which facilitates the deployment of deep learning models in devices with limited resources and improves its applicability for real-time applications [54]. Nonetheless, current research does not consider such techniques when designing NI to be deployed at the network edge.

Moreover, few works consider the optimality gap, which is understood as the difference on performance between the proposed ML method and the optimal solution. This lack of comparisons comes from the fact that, for many elaborate networking problems, such optimal solution is not computable and unknown. Because of this limitation, this gap is only given for small deployments or works with simplified models. We circumvent this shortcoming by extending the comparison analysis also to other possible benchmarks. These comparisons against benchmarks are much more common in the recent works, where ML algorithms usually outperform benchmark methods. However, the fact that the optimality gap cannot be obtained in all cases casts doubt on the applicability of ML in networking.

We also wanted to analyze the literature from the dataset point of view. In general, there is a good balance between real and synthetic datasets, as it can be seen in **Table** 16. Synthetic datasets are normally generated using simulators/emulators, while real datasets are generated by deploying a live testbed. We noticed that the data to train and test the NI in bigger deployments is typically obtained through simulations. There are different challenges for obtaining accurate and real data: On one hand, deploying a large number of devices is expensive; on the other hand, the testbed should be isolated from external interference to measure the performance of a learning model, which in some cases is difficult to achieve (e.g., in wireless environments). Additionally, as shown in **Table** 17, private datasets are more common than open datasets. This situation prevents the community from verifying both reproducibility and repeatability, and it can also lead to NI solutions tailored to a specific case, making difficult the generalization of ML algorithms.

***Table** 16. Dataset Generation Method*

Dataset Generation	Amount of works
Synthetic	18
Real	16
N/A	5
Grand Total	39

***Table** 17. Dataset Availability*

Dataset Availability	Amount of works
N/A	16
Private	15
Open	8
Grand Total	39

As intuition suggests, the importance of the data characteristics and properties in ML applications, which are in essence data-driven applications, is crucial. When fed with quality data, ML algorithms produce optimal results. However, it seems that the data curation is often overlooked in the networking area. Most of the time, it is unclear how to obtain the data, or the datasets cannot be sufficiently described from a data science point of view. As a consortium, we had great difficulties filling the 4V's of data, i.e., the four main features that should characterize large measurement datasets such as those collected for network performance evaluation: Volume, Velocity, Variety and Veracity. Most of the works do not specify the incoming data rate or the volume of the data. This consideration is vital for designing real-life NI systems since it is related to the capacity of the system to process such data, and it makes clear that the networking community should incorporate data science mechanisms and mindset in order to correctly exploit the potential of ML approaches.

6 Conclusions

The deliverable presented the very first contributions of the second iteration of the DAEMON project.

Firstly, it detailed the updated requirements, both functional and non-functional, for the eight network functionalities supported by Network Intelligence (NI) as targeted by DAEMON. Also, it introduced the original functional and non-functional requirements for the NI plane intended to orchestrate NI instances in the end-to-end mobile network architecture. This body of requirements lays the foundations to the work in WP3 and WP4 during the second iteration of the project, and we expect that the NI algorithms and models developed in those WPs will adhere to the specifications in this document.

Secondly, the deliverable presented the initial proposal of the project for a NI plane that meets the requirements laid out above, detailing its internal organization and its relationships with existing functional blocks like network management and orchestration (MANO) or machine learning operations (MLOps). It also presented a model for the representation of NI algorithms and operation in the NI plane, which was showcased via a practical example of usage to coordinate different NI instances. The blueprint for a NI plane presented in the deliverable is a first step towards the full design of a coordinated end-to-end NI orchestration that represents one of the targets of the project. The work in WP3 and WP4 will focus during the second iterations also on ensuring that the NI models developed in the project align with the NI plane specifications, so that they can be easily integrated into the blueprint.

Thirdly, the deliverable outlined two different sets of guidelines for the design of NI. The first set concerned recommendations on how to tailor AI models for operation in mobile network environments and how to best match the expectations of operators in terms of automated infrastructure management. The second set concerned specific contexts, among those tackled by the project, where complex data-hungry models based on deep learning approaches are not necessarily the best choice for network automation, and the DAEMON project results actually show that more traditional or hybrid solutions provide clear advantages. Both sets of guidelines will be considered during the improvement of the NI algorithms in WP3 and WP4 of the project during the second iteration of activities.

Finally, the document presented the current status of a live survey of research in NI-assisted network functionalities, introducing the methodology adopted by DAEMON to carry out a comprehensive literature review along the lifetime of the project. This is an ongoing activity that will ultimately lead to a comprehensive review of the state of the art in NI design and evaluation – a consortium-wide effort that will serve as a key reference for researcher and practitioners in the field, and further foster the vision, approach and methods developed by the DAEMON project.

7 References

[1] ICT-52, "D2.1: Initial report on requirements analysis and state-of-the-art frameworks and toolsets.," DAEMON, Available at: https://zenodo.org/record/5060979#, June, 2021.

[2] 5GPPP, "AI and ML – Enablers for Beyond 5G Networks. White Paper.," 5GPPP, https://doi.org/10.5281/zenodo.4299895, 2020.

[3] I. Standards, "IEEE/ISO/IEC 29148-2018," 2018. [Online]. Available: https://standards.ieee.org/standard/29148-2018.html .

[4] ICT-52, "D5.1: Preliminary evaluation results and plan for proof-of-concept demonstrations," DAEMON, March 2022.

[5] ICT-52, "D3.1: Initial design of real-time control and VNF intelligence mechanisms," DAEMON, Available at: https://zenodo.org/record/5745433#, November, 2021.

[6] ICT-52, "D4.1: Initial design of intelligent orchestration and management mechanisms," DAEMON, Available at: https://zenodo.org/record/5745456#, November, 2021.

[7] S. Telecom, "The Telecom Dataset," Shanghai Telecom, June 2018. [Online]. Available: http://sguangwang.com/TelecomDataset.html.

[8] S. Wang, Y. Guo, N. Zhang, P. Yang, A. Zhou and X. Shen, "Delay-aware microservice coordination in mobile edge computing: A reinforcement learning approach," *IEEE Transactions on Mobile Computing,* vol. 20, no. 3, pp. 939-951, 2019.

[9] ETSI GS NFV 003, "Network Functions Virtualisation (NFV); Terminology for Main Concepts in NFV - v1.2.1," ETSI, Dec. 2014.

[10] O. Gheibi, D. Weyns and F. Quin, "Applying machine learning in self-adaptive systems: A systematic literature review," *ACM Transactions on Autonomous and Adaptive Systems (TAAS),* vol. 15, no. 3, pp. 1-37, 2021.

[11] M. Camelo, L. Cominardi, M. Gramaglia, M. Fiore, A. Garcia-Saavedra, F. L., D. De Vleeschauwer, P. Soto, N. Slamnik-Kriještorac, J. Ballesteros, C. Chang, G. Baldoni, J. Marquez-Barja, P. Hellinckx and S. Latré, "Requirements and Specifications for the Orchestration of Network Intelligence in 6G. Zenodo," in *IEEE Consumer Communications & Networking Conference (CCNC)*, 2022.

[12] J. A. Ayala-Romero, A. Garcia-Saavedra, M. Gramaglia, X. Costa-Perez, A. Banchs and J. Alcaraz, "vrAIn: A deep learning approach tailoring computing and radio resources in virtualized RANs," in *The 25th Annual International Conference on Mobile Computing and Networking*, Los Cabos Mexico, 2019.

[13] J. A. Ayala-Romero, A. Garcia-Saavedra, X. Costa-Perez and G. Iosifidis, "Bayesian Online Learning for Energy-Aware Resource Orchestration in Virtualized RANs," in *IEEE Conference on Computer Communications (INFOCOM)*, 2021.

[14] 3. G. P. P. (3GPP), "5G System - Network Data Analytics Services - Stage 3, Technical Specification (TS) 29.520, version 17.5.0.," 3GPP, [Online]. Available: https://www.3gpp.org/DynaReport/29520.html, Dec. 2021.

[15] E. Raj, Engineering MLOps, Packt Publishing, 2021.

[16] Kubeflow, "Kubeflow: The Machine Learning Toolkit for Kubernetes," [Online]. Available: https://www.kubeflow.org/. [Accessed 06 06 2022].

[17] M. Project, "MLFLow: An open source platform for the machine learning lifecycle," LLC, [Online]. Available: https://mlflow.org/. [Accessed 06 06 2022].

[18] S. Sultana, P. Dooze and V. Kumar, "Realization of an Intrusion Detection use-case in ONAP with Acumos," in *International Conference on Computer Communications and Networks (ICCCN)*, 2021.

[19] ZettaScale Technology, "Zenoh-Flow," Eclipse Foundation - Zenoh -Flow project, 2022. [Online]. Available: https://github.com/eclipse-zenoh/zenoh-flow.

[20] A. Collet and A. F. M. Banchs, "LossLeaP: Learning to Predict for Intent-Based Networking," in *IEEE INFOCOM*, 2022.

[21] J. O. Kephart and D. M. Chess, "The vision of autonomic computing," *Computer,* vol. 36, no. 1, pp. 41-50, 2003.

[22] OpenAI, "Faulty Reward Functions in the Wild," OpenAI, 2022. [Online]. Available: https://openai.com/blog/faulty-reward-functions/ . [Accessed 1 July 2022].

[23] P. Soto, D. De Vleeschauwer, M. Camelo, Y. De Bock, K. De Schepper, C. Chang, P. Hellinckx, J. F. Botero and S. Latré, "Towards autonomous VNF auto-scaling using deep reinforcement learning," in *IEEE International Conference on Software Defined Systems (SDS)*, 2021.

[24] L. Buşoniu, R. Babuška and B. De Schutter, "Multi-agent reinforcement learning: An overview," in *Innovations in Multi-Agent Systems and Applications - 1*, D. Srinivasan and L. Jain, Eds., Springer, 2010.

[25] I. Hubara, M. Courbariaux, D. Soudry, R. El-Yaniv and Y. Bengio, "Binarized neural networks," *Advances in neural information processing systems,* vol. 29, 2016.

[26] M. Roughan, S. Sen, O. Spatscheck and N. Duffield, "Class-of-service mapping for QoS: a statistical signature-based approach to IP traffic classification," in *ACM SIGCOMM conference on Internet measurement*, Oct. 2004.

[27] S. Valenti, D. Rossi, A. Dainotti, A. Pescapè, A. Finamore and M. Mellia, "Reviewing traffic classification," in *Data Traffic Monitoring and Analysis*, Springer, 2013, pp. 123-147.

[28] G. Aceto, D. Ciuonzo, A. Montieri and A. Pescapé, "Toward effective mobile encrypted traffic classification through deep learning," *Neurocomputing,* vol. 409, pp. 306-315, 2020.

[29] M. Lotfollahi, M. Jafari Siavoshani, R. Shirali Hossein Zade and M. Saberian, "Deep packet: A novel approach for encrypted traffic classification using deep learning," *Soft Computing,* vol. 24, no. 3, pp. 1999-2012, 2020.

[30] F. Shaheen, B. Verma and M. Asafuddoula, "Impact of automatic feature extraction in deep learning architecture," in *International conference on digital image computing: techniques and applications (DICTA)* , 2016.

[31] K. Ismailaj, M. Camelo and S. Latré, "When deep learning may not be the right tool for traffic classification," in *IFIP/IEEE International Symposium on Integrated Network Management (IM)*, 2021.

[32] M. Camelo, P. Soto and S. Latré, "A General Approach for Traffic Classification in Wireless Networks using Deep Learning," *IEEE Transactions on Network and Service Management.*

[33] L. Qiu, Y. Zhang, F. Wang, M. K. Han and R. Mahajan, "A general model of wireless interference," in *ACM international conference on Mobile computing and networking* , Sept. 2007.

[34] G. Bianchi, "Performance analysis of the IEEE 802.11 distributed coordination function," *IEEE Journal on selected areas in communications,* vol. 18, no. 3, pp. 535-547, 2000.

[35] F. Scarselli, M. Gori, A. C. Tsoi, M. Hagenbuchner and G. Monfardini, "The graph neural network model," *IEEE transactions on neural networks,* vol. 20, no. 1, pp. 61-80, 2008.

[36] P. Soto, M. Camelo, K. Mets, F. Wilhelmi, D. Góez, L. A. Fletscher, N. Gaviria, P. Hellinckx, J. Botero and S. Latré, " ATARI: A graph convolutional neural network approach for performance prediction in next-generation WLANs," *Sensors,* vol. 21, no. 13, 2021.

[37] J. Wang, J. Tang, Z. Xu, Y. Wang, G. Xue, X. Zhang and D. Yang, "Spatiotemporal modeling and prediction in cellular networks: A big data enabled deep learning approach," in *IEEE International Conference on Computer Communications (INFOCOM)*, 2017.

[38] C. Zhang, M. Fiore and P. Patras, "Multi-Service Mobile Traffic Forecasting via Convolutional Long Short-Term Memories," in *IEEE International Symposium on Measurements and Networking (IEEE M&N)* , Jun. 2019.

[39] C. Huang, C. Chiang and Q. Li, "A study of deep learning networks on mobile traffic forecasting," in *IEEE International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC)*, 2017.

[40] C. Zhang and P. Patras, "Long-term mobile traffic forecasting using deep spatio-temporal neural networks," in *ACM International Symposium on Mobile Ad Hoc Networking and Computing, (Mobihoc)*, 2018.

[41] X. Wang, Z. Zhou, F. Xiao, K. Xing, Z. Yang, Y. Liu and C. Peng, "Spatio-temporal analysis and prediction of cellular traffic in metropolis," *IEEE Transactions on Mobile Computing,* vol. 18, no. 9, 2019.

[42] S. Makridakis, E. Spiliotis and V. Assimakopoulos, "The M4 competition: 100,000 time series and 61 forecasting methods," *International Journal of Forecasting,* vol. 36, no. 1, pp. 54-74, 2020.

[43] S. Smyl, "A hybrid method of exponential smoothing and recurrent neural networks for time series forecasting," *International Journal of Forecasting,* vol. 36, no. 1, pp. 75-85, 2020.

[44] R. Hyndman, A. Koehler, J. Ord and R. Snyder, Forecasting with Exponential Smoothing: The State Space Approach, Springer, 2008.

[45] S. Chang, Y. Zhang, W. Han, M. Yu, X. Guo, W. Tan, X. Cui, M. J. Witbrock, M. Hasegawa-Johnson and T. S. Huang, *Dilated recurrent neural networks,* available at arxiv: abs/1710.02224.

[46] R. T. Clemen, "Combining forecasts: A review and annotated bibliography," *International Journal of Forecasting,* vol. 5, no. 4, 1989.

[47] O. Sagi and L. Rokach, "Ensemble learning: A survey," *WIREs Data Mining and Knowledge Discovery,* vol. 8, no. 4, 2018.

[48] L. Lo Schiavo, M. Fiore, M. Gramaglia, A. Banchs and X. Costa-Perez, "Forecasting for Network Management with Joint Statistical Modelling and Machine Learning," in *IEEE WoWMoM*, Belfast, Ireland, 2022.

[49] C. Busse-Grawitz, R. Meier, A. Dietmüller, T. Bühler and L. Vanbever, *pForest: In-Network Inference with Random Forests,* arXiv:1909.05680 http://arxiv.org/abs/1909.05680, 2019.

[50] C. Zheng and N. Zilberman, "Planter: Seeding Trees within Switches," in *SIGCOMM Poster and Demo Sessions*, 2021.

[51] Q. Yang, Y. Liu, T. Chen and Y. Tong, "Federated machine learning: Concept and applications," *ACM Transactions on Intelligent Systems and Technology (TIST),* vol. 10, no. 2, pp. 1-19, 2019.

[52] 5G PPP Technology Board, "TO REMOVEEAI and ML – Enablers for Beyond 5G Networks," 2021, May.

[53] P. R. Nicolas, Scala for machine learning, Packt Publishing Ltd., 2015.

[54] Y. Cheng, D. Wang, P. Zhou and T. Zhang, "Model compression and acceleration for deep neural networks: The principles, progress, and challenges," *IEEE Signal Processing Magazine,* vol. 35, no. 1, pp. 126-136, 2018.

[55] "Structured vs unstructured data," MongoDB, [Online]. Available: https://www.mongodb.com/unstructured-data/structured-vs-unstructured. [Accessed 1 July 2022].

[56] R. Boutaba, M. A. Salahuddin, N. Limam, S. Ayoubi, N. Shahriar, F. Estrada-Solano and O. M. Caicedo, "A comprehensive survey on machine learning for networking: evolution, applications and research opportunities," *Journal of Internet Services and Applications,* vol. 9, no. 1, pp. 1-99, 2018.

[57] M. Abbasi, A. Shahraki and A. Taherkordi, "Deep Learning for Network Traffic Monitoring and Analysis (NTMA): A Survey," *Computer Communications,* vol. 170, pp. 19-41, 2021.

[58] F. B. Mismar, J. Choi and B. L. Evans, "A framework for automated cellular netowrk tuning with reinforcement learning," *IEEE Transactions on Communications,* vol. 67, no. 10, pp. 7152-7167, 2019.

[59] 5G-ppp, "AI-ML for Networks," 5G-ppp, 2021. [Online]. Available: https://5g-ppp.eu/wp-content/uploads/2021/05/AI-MLforNetworks-v1-0.pdf.

[60] T. Engel, "SS7: Locate, Track, Manipulate.," [Online]. Available: https://berlin.ccc.de/~tobias/31c3-ss7-locate-track-manipulate.pdf.

[61] K. Nohl and L. Melette, "Advanced interconnect attacks," 15 08 2015. [Online]. Available: https://media.ccc.de/v/camp2015-6785-advanced_interconnect_attacks. [Accessed 15 08 2021].

[62] N. Slamnik-Kriještorac, E. de Britto e Silva, E. Municio, H. Carvalho de Resende, S. Hadiwardoyo and J. Marquez-Barja, "Network Service and Resource Orchestration: A Feature and Performance Analysis within the MEC-Enhanced Vehicular Network Context.," no. 20, January 2020.

[63] A. Sapio, I. Abdelaziz, A. Aldilaijan, M. Canini and P. Kalnis, "In-network computation is a dumb idea whose time has come," in *Proceedings of the 16th ACM Workshop on Hot Topics in Networks*, 2017.

[64] A. Sapio, M. Canini, C.-Y. Ho, J. Nelson, P. Kalnis, C. Kim, A. Krishnamurthy, M. Moshref, D. R. Ports and P. Richtárik, *Scaling distributed machine learning with in-network aggregation,* arXiv preprint, 2019.

[65] S. Fu, F. Yang and Y. Xiao, "AI Inspired Intelligent Resource Management in Future Wireless Network," *IEEE Access,* vol. 8, pp. 22425-22433, 2020.

[66] F. Yang, Z. Wang, X. Ma, G. Yuan and X. An, *SwitchAgg: a further step towards in-network computation,* arXiv preprint arXiv:1904.04024, 2019.

[67] C. Papagianni, J. Mangues-Bafalluy, P. Bermudez, S. Barmpounakis, D. De Vleeschauwer, J. Brenes, E. Zeydan, C. Casetti, C. Guimarães, P. Murillo, A. Garcia-Saavedra, D. Corujo and T. Pepe, "5Growth: AI-driven 5G for Automation in Vertical Industries," in *2020 European Conference on Networks and Communications (EuCNC)*, Dubrovnik (Croatia), 2020.

[68] G. Siracusano, S. Galea, D. Sanvito, M. Malekzadeh, G. Antichi, P. Costa, H. Haddadi and R. Bifulco, "Re-architecting Traffic Analysis with Neural Network Interface Cards," in *USENIX Symposium on Networked Systems Design and Implementation (NSDI 22)*, 2022.

[69] "“5G System; Network Data Analytics Services; Stage 3,””.

[70] A. Manousis, R. A. Sharma, V. Sekar and J. Sherry, "Contention-Aware Performance Prediction For Virtualized Network Functions.," in *Conf. of the ACM Special Interest Group on Data Communication on the application, technologies, architectures, and protocols for computer communication*, 2020.

[71] N. Xiong and Z. Zilberman, "Do switches dream of machine learning? Toward in-network classification," in *Proceedings of the 18th ACM workshop on hot topics in networks*, 2019.

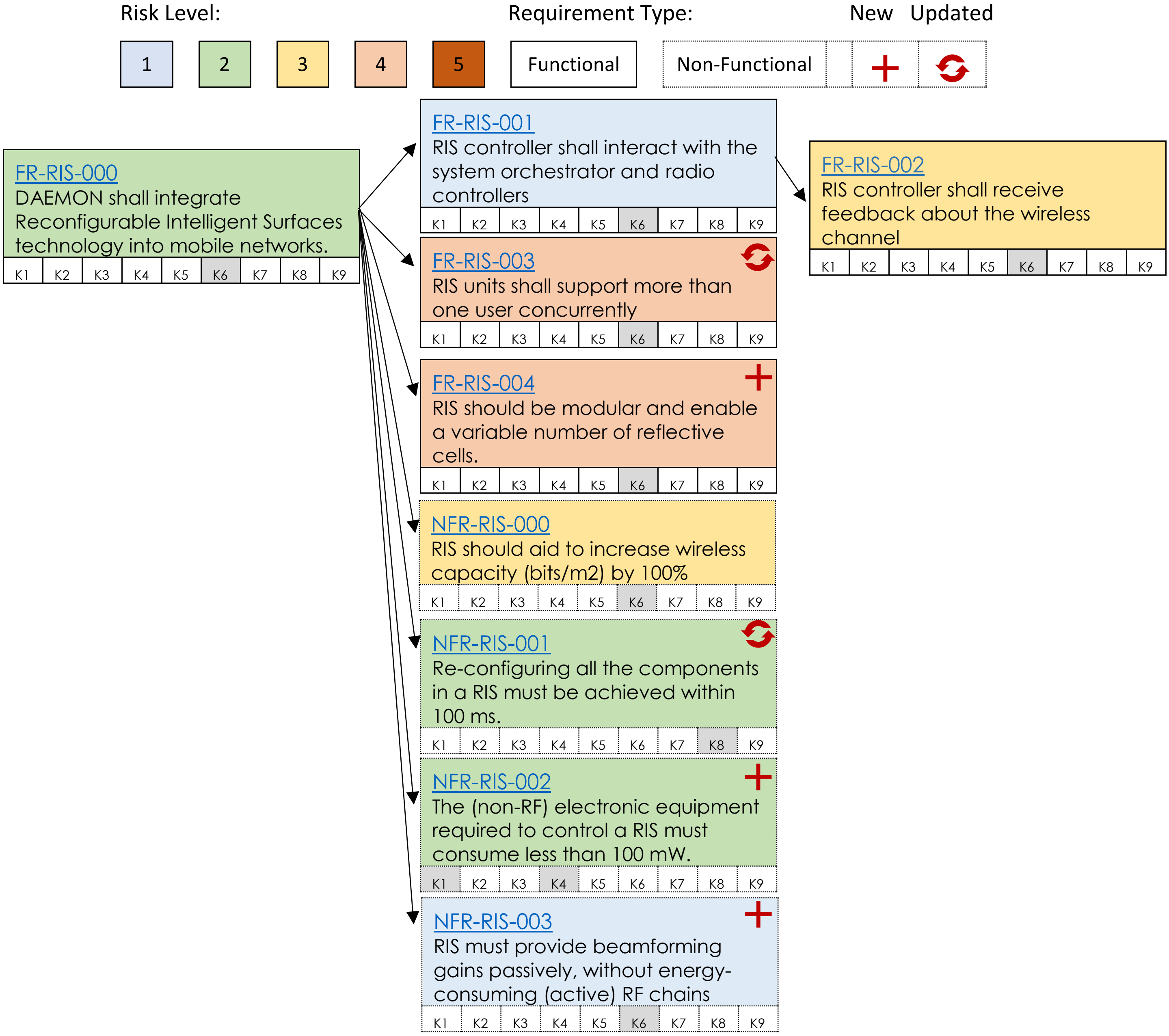
A Appendix: NI Use Cases Functional Requirements

In this appendix, we present the complete requirements tree designed by DAEMON. For the sake of readability, we divide the presentation of the tree in different blocks. First, by separating the tree of requirements of the NI plane from the requirements' tree for the functionalities. Then, we show separately the section of the tree dedicated to each one of the eight functionalities. Besides this, we also divide the description of the requirements depending on whether they are functional or non-functional requirements.

We start by describing the functional requirements' tree for each one of the eight functionalities. Then, we describe the functional requirements of the NI plane. We continue by describing the non-functional requirements for the functionalities and the NI plane, and we will conclude this appendix by presenting the design constraints for each one of the cases.

A.1 Functional requirements: Network Functionalities

A.1.1 RIS control


Risk Level: 1 2 3 4 5

Requirement Type: Functional Non-Functional

New + Updated ⟳

FR-RIS-000
DAEMON shall integrate Reconfigurable Intelligent Surfaces technology into mobile networks.
K1 K2 K3 K4 K5 K6 K7 K8 K9

FR-RIS-001
RIS controller shall interact with the system orchestrator and radio controllers
K1 K2 K3 K4 K5 K6 K7 K8 K9

FR-RIS-002
RIS controller shall receive feedback about the wireless channel
K1 K2 K3 K4 K5 K6 K7 K8 K9

FR-RIS-003 ⟳
RIS units shall support more than one user concurrently
K1 K2 K3 K4 K5 K6 K7 K8 K9

FR-RIS-004 +
RIS should be modular and enable a variable number of reflective cells.
K1 K2 K3 K4 K5 K6 K7 K8 K9

NFR-RIS-000
RIS should aid to increase wireless capacity (bits/m2) by 100%
K1 K2 K3 K4 K5 K6 K7 K8 K9

NFR-RIS-001 ⟳
Re-configuring all the components in a RIS must be achieved within 100 ms.
K1 K2 K3 K4 K5 K6 K7 K8 K9

NFR-RIS-002 +
The (non-RF) electronic equipment required to control a RIS must consume less than 100 mW.
K1 K2 K3 K4 K5 K6 K7 K8 K9

NFR-RIS-003 +
RIS must provide beamforming gains passively, without energy-consuming (active) RF chains
K1 K2 K3 K4 K5 K6 K7 K8 K9

<table>
<tr><th colspan="18">FR-RIS-000</th></tr>
<tr><td colspan="3"><b>Description</b></td><td colspan="15">DAEMON shall integrate Reconfigurable Intelligent Surfaces (RIS) technology into mobile networks.</td></tr>
<tr><td colspan="3"><b>Version</b></td><td colspan="15">001M1</td></tr>
<tr><td colspan="3"><b>Owner</b></td><td colspan="15">NEC</td></tr>
<tr><td colspan="3"><b>Priority</b></td><td colspan="15">High</td></tr>
<tr><td colspan="3"><b>Risk</b></td><td colspan="15">2</td></tr>
<tr><td colspan="3"><b>Risk Description</b></td><td colspan="15">There is a mild risk that the project will not be able to build a RIS prototype. Should this happen, the project will rely on simulations and mathematical models.</td></tr>
<tr><td colspan="3"><b>Rationale</b></td><td colspan="15">RIS technologies will play a key role in increasing the wireless network capacity of next-generation networks, reduce energy consumption, and create new privacy and security applications. However, optimal RIS operation can only be achieved in coordination with the radio access network controller. To this end, native support by DAEMON platform and open interfaces that integrate RIS controllers into the rest of the mobile network control ecosystem is required.</td></tr>
<tr><td><b>K1</b></td><td></td><td><b>K2</b></td><td></td><td><b>K3</b></td><td></td><td><b>K4</b></td><td></td><td><b>K5</b></td><td></td><td><b>K6</b></td><td>X</td><td><b>K7</b></td><td></td><td><b>K8</b></td><td></td><td><b>K9</b></td><td></td></tr>
<tr><td colspan="3"><b>Parents</b></td><td colspan="15">None</td></tr>
<tr><td colspan="3"></td><td colspan="15"></td></tr>
</table>

<table>
<tr><th colspan="18">FR-RIS-001</th></tr>
<tr><td colspan="3"><b>Description</b></td><td colspan="15">RIS controller shall interact with the system orchestrator and radio controllers</td></tr>
<tr><td colspan="3"><b>Version</b></td><td colspan="15">002M5</td></tr>
<tr><td colspan="3"><b>Owner</b></td><td colspan="15">NEC</td></tr>
<tr><td colspan="3"><b>Priority</b></td><td colspan="15">High</td></tr>
<tr><td colspan="3"><b>Risk</b></td><td colspan="15">1</td></tr>
<tr><td colspan="3"><b>Risk Description</b></td><td colspan="15">No risk</td></tr>
<tr><td colspan="3"><b>Rationale</b></td><td colspan="15">An interface between the mobile network orchestrator, the gNB controllers, and the RIS controller shall enable joint optimization of gNBs, UEs and surfaces.</td></tr>
<tr><td><b>K1</b></td><td></td><td><b>K2</b></td><td></td><td><b>K3</b></td><td></td><td><b>K4</b></td><td></td><td><b>K5</b></td><td></td><td><b>K6</b></td><td>X</td><td><b>K7</b></td><td></td><td><b>K8</b></td><td></td><td><b>K9</b></td><td></td></tr>
<tr><td colspan="3"><b>Parents</b></td><td colspan="15">FR-RIS-000</td></tr>
</table>

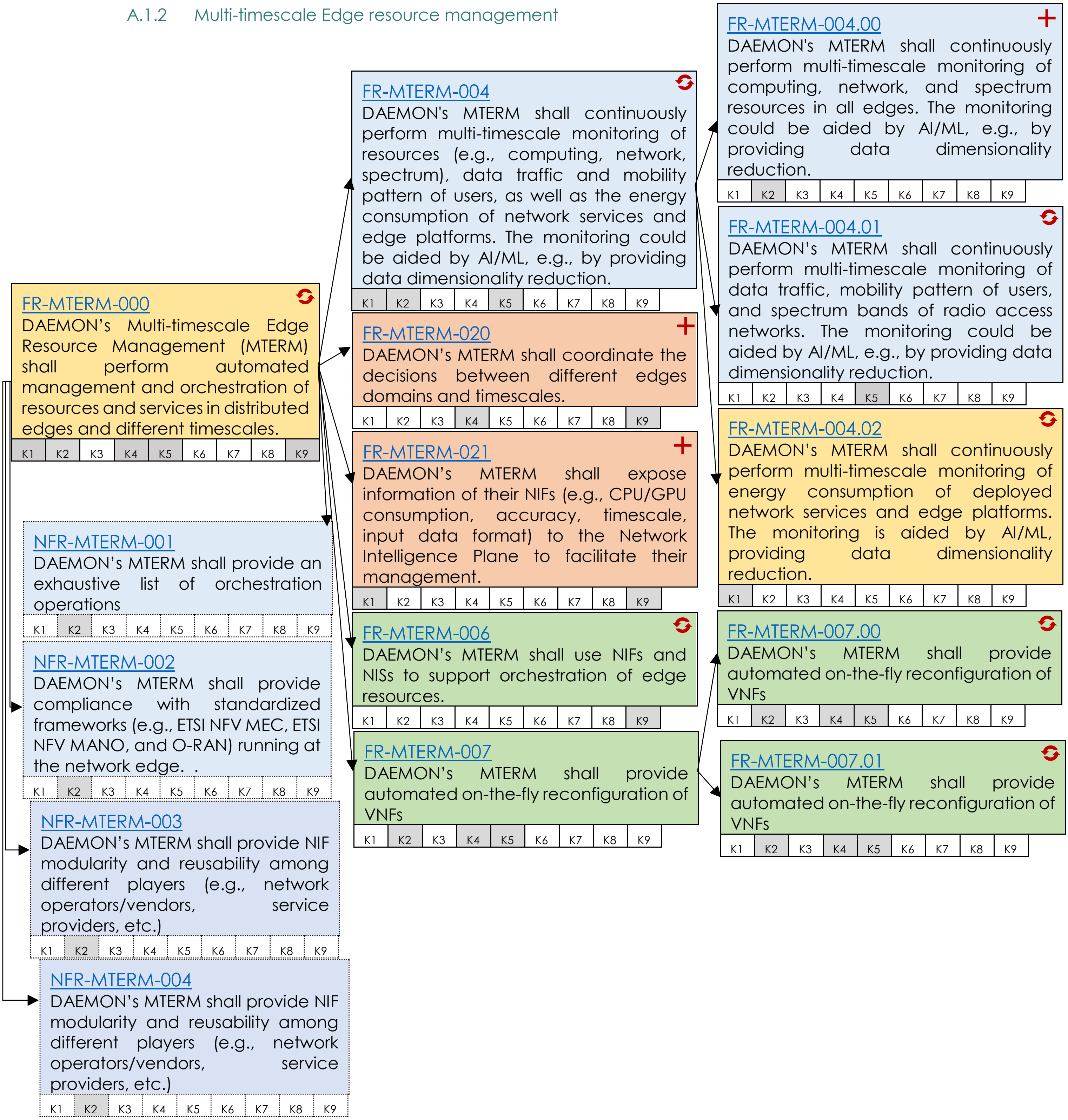
<table>
<tr><th colspan="18">FR-RIS-002</th></tr>
<tr><td colspan="3"><b>Description</b></td><td colspan="15">RIS controller shall receive feedback about the wireless channel</td></tr>
<tr><td colspan="3"><b>Version</b></td><td colspan="15">002M4</td></tr>
<tr><td colspan="3"><b>Owner</b></td><td colspan="15">NEC</td></tr>
<tr><td colspan="3"><b>Priority</b></td><td colspan="15">Medium</td></tr>
<tr><td colspan="3"><b>Risk</b></td><td colspan="15">3</td></tr>
<tr><td colspan="3"><b>Risk Description</b></td><td colspan="15">Channel feedback may not be received in a timely manner or with the required accuracy so as to be useful information.</td></tr>
<tr><td colspan="3"><b>Rationale</b></td><td colspan="15">Reconfigurable Intelligent Surfaces modify the propagation properties of impinging wireless signals in a controllable manner. To this end, good estimations about the wireless environment based upon feedback from users and gNBs, i.e., channel information, are required to perform optimal RIS operation.</td></tr>
<tr><td><b>K1</b></td><td></td><td><b>K2</b></td><td></td><td><b>K3</b></td><td></td><td><b>K4</b></td><td></td><td><b>K5</b></td><td></td><td><b>K6</b></td><td>X</td><td><b>K7</b></td><td></td><td><b>K8</b></td><td></td><td><b>K9</b></td><td></td></tr>
<tr><td colspan="3"><b>Parents</b></td><td colspan="15">FR-RIS-001</td></tr>
</table>

<table>
<tr><th colspan="18">FR-RIS-003</th></tr>
<tr><td colspan="3"><b>Description</b></td><td colspan="15">RIS units shall support more than one user concurrently</td></tr>
<tr><td colspan="3"><b>Version</b></td><td colspan="15">003M17</td></tr>
<tr><td colspan="3"><b>Owner</b></td><td colspan="15">NEC</td></tr>
<tr><td colspan="3"><b>Priority</b></td><td colspan="15">Medium</td></tr>
<tr><td colspan="3"><b>Risk</b></td><td colspan="15">4</td></tr>
<tr><td colspan="3"><b>Risk Description</b></td><td colspan="15">Tight and timely coordination between gNB MAC schedulers may be required</td></tr>
<tr><td colspan="3"><b>Rationale</b></td><td colspan="15">This enables increasing the system capacity for multiple users.</td></tr>
<tr><td><b>K1</b></td><td></td><td><b>K2</b></td><td></td><td><b>K3</b></td><td></td><td><b>K4</b></td><td></td><td><b>K5</b></td><td></td><td><b>K6</b></td><td>X</td><td><b>K7</b></td><td></td><td><b>K8</b></td><td></td><td><b>K9</b></td><td></td></tr>
</table>

Parents	FR-RIS-000

<table>
<tr><th colspan="18">FR-RIS-004</th></tr>
<tr><td colspan="3"><b>Description</b></td><td colspan="15">RIS should be modular and enable a variable number of reflective cells.</td></tr>
<tr><td colspan="3"><b>Version</b></td><td colspan="15">002M17</td></tr>
<tr><td colspan="3"><b>Owner</b></td><td colspan="15">NEC</td></tr>
<tr><td colspan="3"><b>Priority</b></td><td colspan="15">Low</td></tr>
<tr><td colspan="3"><b>Risk</b></td><td colspan="15">4</td></tr>
<tr><td colspan="3"><b>Risk Description</b></td><td colspan="15">Modularity may be overly hard to achieve when designing a RIS.</td></tr>
<tr><td colspan="3"><b>Rationale</b></td><td colspan="15">The ability to change the amount of reflective surface would enable a RIS to adapt itself to the surface, which may be highly irregular.</td></tr>
<tr><td><b>K1</b></td><td></td><td><b>K2</b></td><td></td><td><b>K3</b></td><td></td><td><b>K4</b></td><td></td><td><b>K5</b></td><td></td><td><b>K6</b></td><td><b>X</b></td><td><b>K7</b></td><td></td><td><b>K8</b></td><td></td><td><b>K9</b></td><td></td></tr>
<tr><td colspan="3"><b>Parents</b></td><td colspan="15">FR-RIS-000</td></tr>
</table>

A.1.2 Multi-timescale Edge resource management



FR-MTERM-000 ↻
DAEMON’s Multi-timescale Edge Resource Management (MTERM) shall perform automated management and orchestration of resources and services in distributed edges and different timescales.
K1 K2 K3 K4 K5 K6 K7 K8 K9

FR-MTERM-004 ↻
DAEMON's MTERM shall continuously perform multi-timescale monitoring of resources (e.g., computing, network, spectrum), data traffic and mobility pattern of users, as well as the energy consumption of network services and edge platforms. The monitoring could be aided by AI/ML, e.g., by providing data dimensionality reduction.
K1 K2 K3 K4 K5 K6 K7 K8 K9

FR-MTERM-004.00 +
DAEMON's MTERM shall continuously perform multi-timescale monitoring of computing, network, and spectrum resources in all edges. The monitoring could be aided by AI/ML, e.g., by providing data dimensionality reduction.
K1 K2 K3 K4 K5 K6 K7 K8 K9

FR-MTERM-004.01 ↻
DAEMON’s MTERM shall continuously perform multi-timescale monitoring of data traffic, mobility pattern of users, and spectrum bands of radio access networks. The monitoring could be aided by AI/ML, e.g., by providing data dimensionality reduction.
K1 K2 K3 K4 K5 K6 K7 K8 K9

FR-MTERM-004.02 ↻
DAEMON’s MTERM shall continuously perform multi-timescale monitoring of energy consumption of deployed network services and edge platforms. The monitoring is aided by AI/ML, providing data dimensionality reduction.
K1 K2 K3 K4 K5 K6 K7 K8 K9

FR-MTERM-020 +
DAEMON’s MTERM shall coordinate the decisions between different edges domains and timescales.
K1 K2 K3 K4 K5 K6 K7 K8 K9

FR-MTERM-021 +
DAEMON’s MTERM shall expose information of their NIFs (e.g., CPU/GPU consumption, accuracy, timescale, input data format) to the Network Intelligence Plane to facilitate their management.
K1 K2 K3 K4 K5 K6 K7 K8 K9

FR-MTERM-006 ↻
DAEMON’s MTERM shall use NIFs and NISs to support orchestration of edge resources.
K1 K2 K3 K4 K5 K6 K7 K8 K9

FR-MTERM-007
DAEMON’s MTERM shall provide automated on-the-fly reconfiguration of VNFs
K1 K2 K3 K4 K5 K6 K7 K8 K9

FR-MTERM-007.00 ↻
DAEMON’s MTERM shall provide automated on-the-fly reconfiguration of VNFs
K1 K2 K3 K4 K5 K6 K7 K8 K9

FR-MTERM-007.01 ↻
DAEMON’s MTERM shall provide automated on-the-fly reconfiguration of VNFs
K1 K2 K3 K4 K5 K6 K7 K8 K9

NFR-MTERM-001
DAEMON’s MTERM shall provide an exhaustive list of orchestration operations
K1 K2 K3 K4 K5 K6 K7 K8 K9

NFR-MTERM-002
DAEMON’s MTERM shall provide compliance with standardized frameworks (e.g., ETSI NFV MEC, ETSI NFV MANO, and O-RAN) running at the network edge. .
K1 K2 K3 K4 K5 K6 K7 K8 K9

NFR-MTERM-003
DAEMON’s MTERM shall provide NIF modularity and reusability among different players (e.g., network operators/vendors, service providers, etc.)
K1 K2 K3 K4 K5 K6 K7 K8 K9

NFR-MTERM-004
DAEMON’s MTERM shall provide NIF modularity and reusability among different players (e.g., network operators/vendors, service providers, etc.)
K1 K2 K3 K4 K5 K6 K7 K8 K9

Risk Level: 1 2 3 4 5

Requirement Type: Functional Non-Functional

New + Updated ↻

<table>
<tr><th colspan="18">FR-MTERM-000</th></tr>
<tr><td colspan="2"><b>Description</b></td><td colspan="16">DAEMON's Multi-timescale Edge Resource Management (MTERM) shall perform automated management and orchestration of resources and services in distributed edges and different timescales.</td></tr>
<tr><td colspan="2"><b>Version</b></td><td colspan="16">003M18</td></tr>
<tr><td colspan="2"><b>Owner</b></td><td colspan="16">IMEC</td></tr>
<tr><td colspan="2"><b>Priority</b></td><td colspan="16">High</td></tr>
<tr><td colspan="2"><b>Risk</b></td><td colspan="16">3</td></tr>
<tr><td colspan="2"><b>Risk Description</b></td><td colspan="16">The decisions made by Network Intelligent Functions (NIFs) and network Intelligent Services (NIS) distributed across the edge networks might be out of sync, since they can make decisions in different timescales. We might need to assign the level of priority to decision-making entities in different tiers and have a control loop that will track the effect of these decisions on the service KPIs.</td></tr>
<tr><td colspan="2"><b>Rationale</b></td><td colspan="16">Services are deployed in a distributed fashion, due to the high mobility of users, and an uneven distribution of resources across the edge networks. Thus, a proper management and orchestration of these distributed services needs to be achieved. The network intelligence in the form of AI-based NIFs and NISs needs to be distributed to different edges in the management and orchestration architecture in order to treat different service dynamics in coarse/fine granular timescale. This should be done in an automated way. Unfortunately, current management frameworks do not provide automation in the form of flexible and dynamic NFV management and orchestration and therefore, this gap should be addressed. Moreover, management frameworks should be able to coordinate intelligence or resources across different network segments and timescales.
<b>Services</b>: MEC application services (specific to use cases, i.e., vertical services), Value-added services (e.g., location services, Radio Network Information Service), NISs and NIFs (e.g., traffic classifiers), energy consumption analyzers, etc.
<b>Resources</b>: CPU, memory, spectrum, storage, and network</td></tr>
<tr><td><b>K1</b></td><td>X</td><td><b>K2</b></td><td>X</td><td><b>K3</b></td><td></td><td><b>K4</b></td><td>X</td><td><b>K5</b></td><td>X</td><td><b>K6</b></td><td></td><td><b>K7</b></td><td></td><td><b>K8</b></td><td></td><td><b>K9</b></td><td>X</td></tr>
<tr><td colspan="2"><b>Parents</b></td><td colspan="16">None</td></tr>
</table>

<table>
<tr><th colspan="18">FR-MTERM-004</th></tr>
<tr><td colspan="2"><b>Description</b></td><td colspan="16">DAEMON's MTERM shall continuously perform multi-timescale monitoring of resources (e.g., computing, network, spectrum), data traffic and mobility pattern of users, as well as the energy consumption of network services and edge platforms. The monitoring could be aided by AI/ML, e.g., by providing data dimensionality reduction.</td></tr>
<tr><td colspan="2"><b>Version</b></td><td colspan="16">002M18</td></tr>
<tr><td colspan="2"><b>Owner</b></td><td colspan="16">IMEC</td></tr>
<tr><td colspan="2"><b>Priority</b></td><td colspan="16">High</td></tr>
<tr><td colspan="2"><b>Risk</b></td><td colspan="16">1</td></tr>
<tr><td colspan="2"><b>Risk Description</b></td><td colspan="16">The amount of data that is being collected might burden the resource-constrained edge nodes. Thus, we need to assess the resource requirements of monitoring services that will be running along with other services on the edge platforms, and to perform a corresponding management of these services in order to produce meaningful and credible results.</td></tr>
<tr><td colspan="2"><b>Rationale</b></td><td colspan="16">Monitoring is one of the main pillars of any automated and adaptative system. By monitoring, any system can verify that its decisions were correctly applied, achieving closed-loop control. However, given the diversity of network operators/vendors/infrastructure/providers/service providers, monitored data stems from multiple sources. In that sense, AI/ML techniques could help to pre-process and reduce such data's dimensionality. However, current frameworks do not incorporate real-time data analytics, making difficult the monitoring of data.</td></tr>
<tr><td><b>K1</b></td><td>X</td><td><b>K2</b></td><td>X</td><td><b>K3</b></td><td></td><td><b>K4</b></td><td></td><td><b>K5</b></td><td>X</td><td><b>K6</b></td><td></td><td><b>K7</b></td><td></td><td><b>K8</b></td><td></td><td><b>K9</b></td><td></td></tr>
<tr><td colspan="2"><b>Parents</b></td><td colspan="16">FR-MTERM-000</td></tr>
</table>

<table>
<tr><th colspan="18">FR-MTERM-004.00</th></tr>
<tr><td colspan="4"><b>Description</b></td><td colspan="14">DAEMON's MTERM shall continuously perform multi-timescale monitoring of computing, network, and spectrum resources in all edges. The monitoring could be aided by AI/ML, e.g., by providing data dimensionality reduction.</td></tr>
<tr><td colspan="4"><b>Version</b></td><td colspan="14">001M18</td></tr>
<tr><td colspan="4"><b>Owner</b></td><td colspan="14">IMEC</td></tr>
<tr><td colspan="4"><b>Priority</b></td><td colspan="14">High</td></tr>
<tr><td colspan="4"><b>Risk</b></td><td colspan="14">1</td></tr>
<tr><td colspan="4"><b>Risk Description</b></td><td colspan="14">Risk FR-MTERM-004</td></tr>
<tr><td colspan="4"><b>Rationale</b></td><td colspan="14">The constant monitoring input of computing, network and spectrum resources will feed the orchestration entities that perform orchestration operations, to control and provide an on-the-fly reconfiguration of deployed virtualized network functions, to migrate them, and to identify anomalies in service and/or framework operation. These metrics shall be monitored at different timescales, depending on the granularity required by the service consuming the data and the available resources at the edge.</td></tr>
<tr><td><b>K1</b></td><td></td><td><b>K2</b></td><td><b>X</b></td><td><b>K3</b></td><td></td><td><b>K4</b></td><td></td><td><b>K5</b></td><td></td><td><b>K6</b></td><td></td><td><b>K7</b></td><td></td><td><b>K8</b></td><td></td><td><b>K9</b></td><td></td></tr>
<tr><td colspan="4"><b>Parents</b></td><td colspan="14">FR-MTERM-004</td></tr>
</table>

<table>
<tr><th colspan="18">FR-MTERM-004.01</th></tr>
<tr><td colspan="4"><b>Description</b></td><td colspan="14">DAEMON's MTERM shall continuously perform multi-timescale monitoring of data traffic, mobility pattern of users, and spectrum bands of radio access networks. The monitoring could be aided by AI/ML, e.g., by providing data dimensionality reduction.</td></tr>
<tr><td colspan="4"><b>Version</b></td><td colspan="14">002M18</td></tr>
<tr><td colspan="4"><b>Owner</b></td><td colspan="14">IMEC</td></tr>
<tr><td colspan="4"><b>Priority</b></td><td colspan="14">Low</td></tr>
<tr><td colspan="4"><b>Risk</b></td><td colspan="14">2</td></tr>
<tr><td colspan="4"><b>Risk Description</b></td><td colspan="14">The value-added services that collect and parse data from the network traffic, the UE mobility, and the spectrum bands impose additional burden to the resource-constrained edge nodes. Thus, we need to assess the resource requirements of those services that will be running along with other services on the edge platforms, and to perform a corresponding management of these service in order to produce meaningful and credible results.</td></tr>
<tr><td colspan="4"><b>Rationale</b></td><td colspan="14">The constant monitoring input of data traffic, mobility patterns and spectrum bands will provide input about the UEs to the orchestration entities that perform orchestration operations, to proactively deploy additional VNFs when and where needed, to migrate them, and to reconfigure existing VNFs to meet demands of all UEs in the system. These metrics shall be monitored at different timescales, depending on the granularity required by the service consuming the data and the available resources at the edge.</td></tr>
<tr><td><b>K1</b></td><td></td><td><b>K2</b></td><td></td><td><b>K3</b></td><td></td><td><b>K4</b></td><td></td><td><b>K5</b></td><td><b>X</b></td><td><b>K6</b></td><td></td><td><b>K7</b></td><td></td><td><b>K8</b></td><td></td><td><b>K9</b></td><td></td></tr>
<tr><td colspan="4"><b>Parents</b></td><td colspan="14">FR-MTERM-004</td></tr>
</table>

<table>
<tr><th colspan="2">FR-MTERM-004.02</th></tr>
<tr><td><b>Description</b></td><td>DAEMON's MTERM shall continuously perform multi-timescale monitoring of energy consumption of deployed network services and edge platforms. The monitoring is aided by AI/ML, providing data dimensionality reduction.</td></tr>
<tr><td><b>Version</b></td><td>002M18</td></tr>
<tr><td><b>Owner</b></td><td>IMEC</td></tr>
<tr><td><b>Priority</b></td><td>Low</td></tr>
</table>

<table>
<tr><td colspan="2"><b>Risk</b></td><td colspan="16">3</td></tr>
<tr><td colspan="2"><b>Risk Description</b></td><td colspan="16">The energy consumption calculation of isolated services might be a complex task, while at the same time, an aggregated energy consumption per edge platform might severely affect accuracy of energy-aware NIFs. Furthermore, although those NIFs that manage energy consumption in the whole system run in cloud, they still need to have probes installed on the edges, and proper assessment of their energy consumption and resource requirements needs to be obtained.</td></tr>
<tr><td colspan="2"><b>Rationale</b></td><td colspan="16">The constant monitoring of energy consumption per service/per edge platform is needed to make an optimal decision on VNF placement and VNF migration from one edge to another. With such an energy consumption footprint in the whole system, cloud orchestrator can perform load balancing between edge platforms, and accordingly turn off certain NIFs and deployed services if energy consumption needs to be decreased.</td></tr>
<tr><td><b>K1</b></td><td>X</td><td><b>K2</b></td><td></td><td><b>K3</b></td><td></td><td><b>K4</b></td><td></td><td><b>K5</b></td><td></td><td><b>K6</b></td><td></td><td><b>K7</b></td><td></td><td><b>K8</b></td><td></td><td><b>K9</b></td><td></td></tr>
<tr><td colspan="2"><b>Parents</b></td><td colspan="16">FR-MTERM-004</td></tr>
</table>

<table>
<tr><th colspan="18">FR-MTERM-020</th></tr>
<tr><td colspan="2"><b>Description</b></td><td colspan="16">DAEMON's MTERM shall coordinate the decisions between different edges domains and timescales.</td></tr>
<tr><td colspan="2"><b>Version</b></td><td colspan="16">001M18</td></tr>
<tr><td colspan="2"><b>Owner</b></td><td colspan="16">IMEC</td></tr>
<tr><td colspan="2"><b>Priority</b></td><td colspan="16">Low</td></tr>
<tr><td colspan="2"><b>Risk</b></td><td colspan="16">4</td></tr>
<tr><td colspan="2"><b>Risk Description</b></td><td colspan="16">Depending on the amount of decision-making engines, the coordination can be cumbersome.</td></tr>
<tr><td colspan="2"><b>Rationale</b></td><td colspan="16">Several management and orchestration operations are based on the decisions of different decision-making engines. Such engines can be based on AI/ML. To guarantee service continuity, coordination between distributed orchestrators in different edge domains is almost mandatory. However, current frameworks do not coordinate intelligence or resources across different network segments and timescales.</td></tr>
<tr><td><b>K1</b></td><td></td><td><b>K2</b></td><td></td><td><b>K3</b></td><td></td><td><b>K4</b></td><td>X</td><td><b>K5</b></td><td></td><td><b>K6</b></td><td></td><td><b>K7</b></td><td></td><td><b>K8</b></td><td></td><td><b>K9</b></td><td>X</td></tr>
<tr><td colspan="2"><b>Parents</b></td><td colspan="16">FR-MTERM-000</td></tr>
</table>

<table>
<tr><th colspan="2">FR-MTERM-021</th></tr>
<tr><td><b>Description</b></td><td>DAEMON's MTERM shall expose information of their NIFs (e.g., CPU/GPU consumption, accuracy, timescale, input data format) to the Network Intelligence Plane to facilitate their management.</td></tr>
<tr><td><b>Version</b></td><td>001M18</td></tr>
<tr><td><b>Owner</b></td><td>IMEC</td></tr>
<tr><td><b>Priority</b></td><td>Low</td></tr>
<tr><td><b>Risk</b></td><td>4</td></tr>
<tr><td><b>Risk Description</b></td><td>The huge amounts of collected data from surrounding infrastructure might represent a risk, since that data might be incomplete or inconsistent. This lack of sufficient and consistent input data leads to inefficiencies in decision-making, e.g., when to replace a NIF .</td></tr>
<tr><td><b>Rationale</b></td><td>Information about NIFs like CPU/GPU consumption, accuracy, timescale, and input data format should be exposed to the Network Intelligence Plane. Based on this information, the intelligent orchestrator(s) should take a decision (e.g., change NIFs because of its poor performance). This would facilitate the lifecycle management of AI/ML-based functions, which current frameworks do not support.</td></tr>
</table>

<table>
<tr><td><b>K1</b></td><td>X</td><td><b>K2</b></td><td></td><td><b>K3</b></td><td></td><td><b>K4</b></td><td></td><td><b>K5</b></td><td></td><td><b>K6</b></td><td></td><td><b>K7</b></td><td></td><td><b>K8</b></td><td></td><td><b>K9</b></td><td>X</td></tr>
<tr><td colspan="4"><b>Parents</b></td><td colspan="14">FR-MTERM-000</td></tr>
</table>

<table>
<tr><th colspan="18">FR-MTERM-006</th></tr>
<tr><td colspan="4"><b>Description</b></td><td colspan="14">DAEMON’s MTERM shall use NIFs and NISs to support orchestration of edge resources.</td></tr>
<tr><td colspan="4"><b>Version</b></td><td colspan="14">002M28</td></tr>
<tr><td colspan="4"><b>Owner</b></td><td colspan="14">IMEC</td></tr>
<tr><td colspan="4"><b>Priority</b></td><td colspan="14">Low</td></tr>
<tr><td colspan="4"><b>Risk</b></td><td colspan="14">2</td></tr>
<tr><td colspan="4"><b>Risk Description</b></td><td colspan="14">The interfaces created to support the instantiation of NIFs and NISs could be tightly coupled which difficult their integration with existing management and orchestration frameworks.</td></tr>
<tr><td colspan="4"><b>Rationale</b></td><td colspan="14">Current research has shown that management and orchestration operations can be improved by using Network Intelligence Functions (ML-based solutions). However, existing management and orchestration frameworks that operate at the network edge (e.g., NFV MANO, OSM, ETSI MEC) do not fully integrate and support the instantiation of such intelligent functions. These frameworks do not provide the necessary interfaces to enable services and application to be data-driven.</td></tr>
<tr><td><b>K1</b></td><td></td><td><b>K2</b></td><td></td><td><b>K3</b></td><td></td><td><b>K4</b></td><td></td><td><b>K5</b></td><td></td><td><b>K6</b></td><td></td><td><b>K7</b></td><td></td><td><b>K8</b></td><td></td><td><b>K9</b></td><td>X</td></tr>
<tr><td colspan="4"><b>Parents</b></td><td colspan="14">FR-MTERM-000</td></tr>
</table>

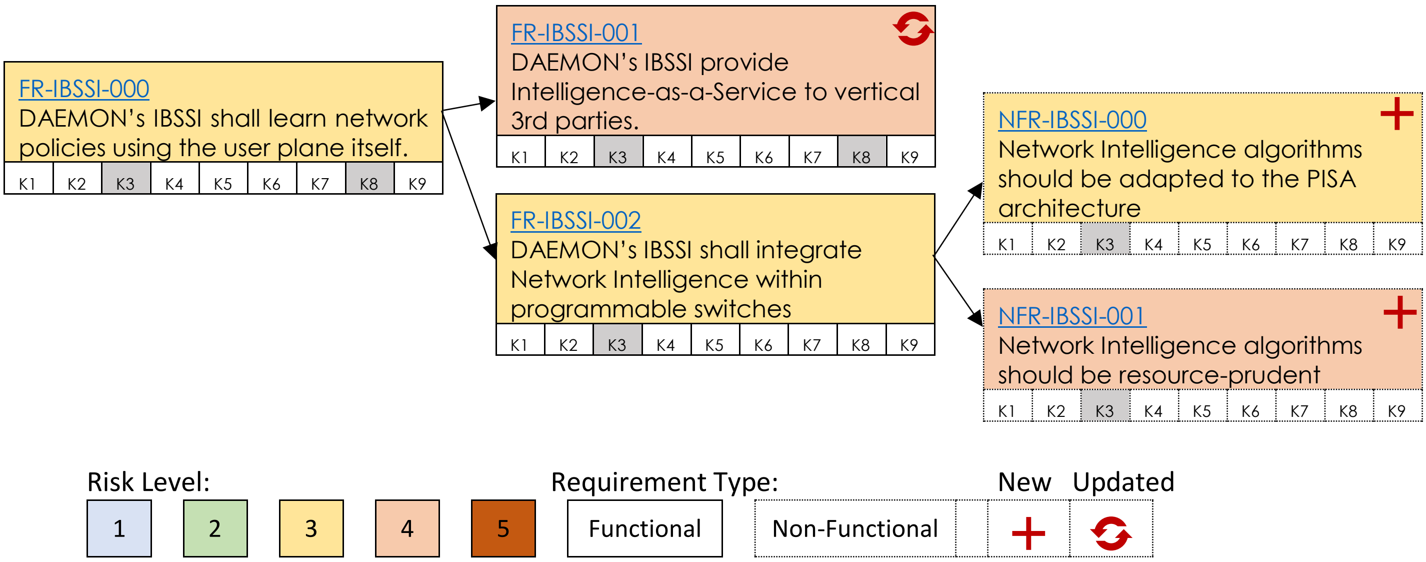
<table>
<tr><th colspan="18">FR-MTERM-007</th></tr>
<tr><td colspan="4"><b>Description</b></td><td colspan="14">DAEMON’s MTERM shall provide automated on-the-fly reconfiguration of VNFs</td></tr>
<tr><td colspan="4"><b>Version</b></td><td colspan="14">003M18</td></tr>
<tr><td colspan="4"><b>Owner</b></td><td colspan="14">IMEC</td></tr>
<tr><td colspan="4"><b>Priority</b></td><td colspan="14">High</td></tr>
<tr><td colspan="4"><b>Risk</b></td><td colspan="14">2</td></tr>
<tr><td colspan="4"><b>Risk Description</b></td><td colspan="14">The reconfiguration of VNFs in the service function chain might impose a risk of service unavailability during the reconfiguration.</td></tr>
<tr><td colspan="4"><b>Rationale</b></td><td colspan="14">Following the cloud-native service design, the service function chains consist of loosely-coupled VNFs that can be replaced and separately configured. Orchestration entities can make decisions to scale up/down/out/in any of these VNFs, and to replace the faulty ones, while maintaining the service continuity.</td></tr>
<tr><td><b>K1</b></td><td></td><td><b>K2</b></td><td><b>X</b></td><td><b>K3</b></td><td></td><td><b>K4</b></td><td>X</td><td><b>K5</b></td><td><b>X</b></td><td><b>K6</b></td><td></td><td><b>K7</b></td><td></td><td><b>K8</b></td><td></td><td><b>K9</b></td><td></td></tr>
<tr><td colspan="4"><b>Parents</b></td><td colspan="14">FR-MTERM-000</td></tr>
</table>

<table>
<tr><th colspan="18">FR-MTERM-007.00</th></tr>
<tr><td colspan="4"><b>Description</b></td><td colspan="14">DAEMON’s MTERM shall provide automated on-the-fly VNF scaling.</td></tr>
<tr><td colspan="4"><b>Version</b></td><td colspan="14">002M18</td></tr>
<tr><td colspan="4"><b>Owner</b></td><td colspan="14">IMEC</td></tr>
<tr><td colspan="4"><b>Priority</b></td><td colspan="14">High</td></tr>
<tr><td colspan="4"><b>Risk</b></td><td colspan="14">2</td></tr>
<tr><td colspan="4"><b>Risk Description</b></td><td colspan="14">The VNF scaling in the service function chain might impose a risk of service unavailability during the reconfiguration.</td></tr>
<tr><td colspan="4"><b>Rationale</b></td><td colspan="14">Rationale FR-MTERM-007</td></tr>
<tr><td><b>K1</b></td><td></td><td><b>K2</b></td><td><b>X</b></td><td><b>K3</b></td><td></td><td><b>K4</b></td><td>X</td><td><b>K5</b></td><td><b>X</b></td><td><b>K6</b></td><td></td><td><b>K7</b></td><td></td><td><b>K8</b></td><td></td><td><b>K9</b></td><td></td></tr>
</table>

Parents	FR-MTERM-007

<table>
<tr><th colspan="18">FR-MTERM-007.01</th></tr>
<tr><td colspan="4"><b>Description</b></td><td colspan="14">DAEMON's MTERM shall provide automated on-the-fly update of VNFs</td></tr>
<tr><td colspan="4"><b>Version</b></td><td colspan="14">002M18</td></tr>
<tr><td colspan="4"><b>Owner</b></td><td colspan="14">IMEC</td></tr>
<tr><td colspan="4"><b>Priority</b></td><td colspan="14">Low</td></tr>
<tr><td colspan="4"><b>Risk</b></td><td colspan="14">2</td></tr>
<tr><td colspan="4"><b>Risk Description</b></td><td colspan="14">The update of VNFs (e.g., change of VNF image, VNF descriptor, IP address, etc.) in the service function chain might impose a risk of service unavailability during the reconfiguration.</td></tr>
<tr><td colspan="4"><b>Rationale</b></td><td colspan="14">Following the cloud-native service design, the service function chains are consisted of loosely-coupled VNFs that can be replaced and separately configured. Orchestration entities can make decisions to update VNFs e.g., if an updated image or descriptor are needed.</td></tr>
<tr><td><b>K1</b></td><td></td><td><b>K2</b></td><td><b>X</b></td><td><b>K3</b></td><td></td><td><b>K4</b></td><td>X</td><td><b>K5</b></td><td><b>X</b></td><td><b>K6</b></td><td></td><td><b>K7</b></td><td></td><td><b>K8</b></td><td></td><td><b>K9</b></td><td></td></tr>
<tr><td colspan="4"><b>Parents</b></td><td colspan="14">FR-MTERM-007</td></tr>
</table>

A.1.3 In-backhaul support for service intelligence

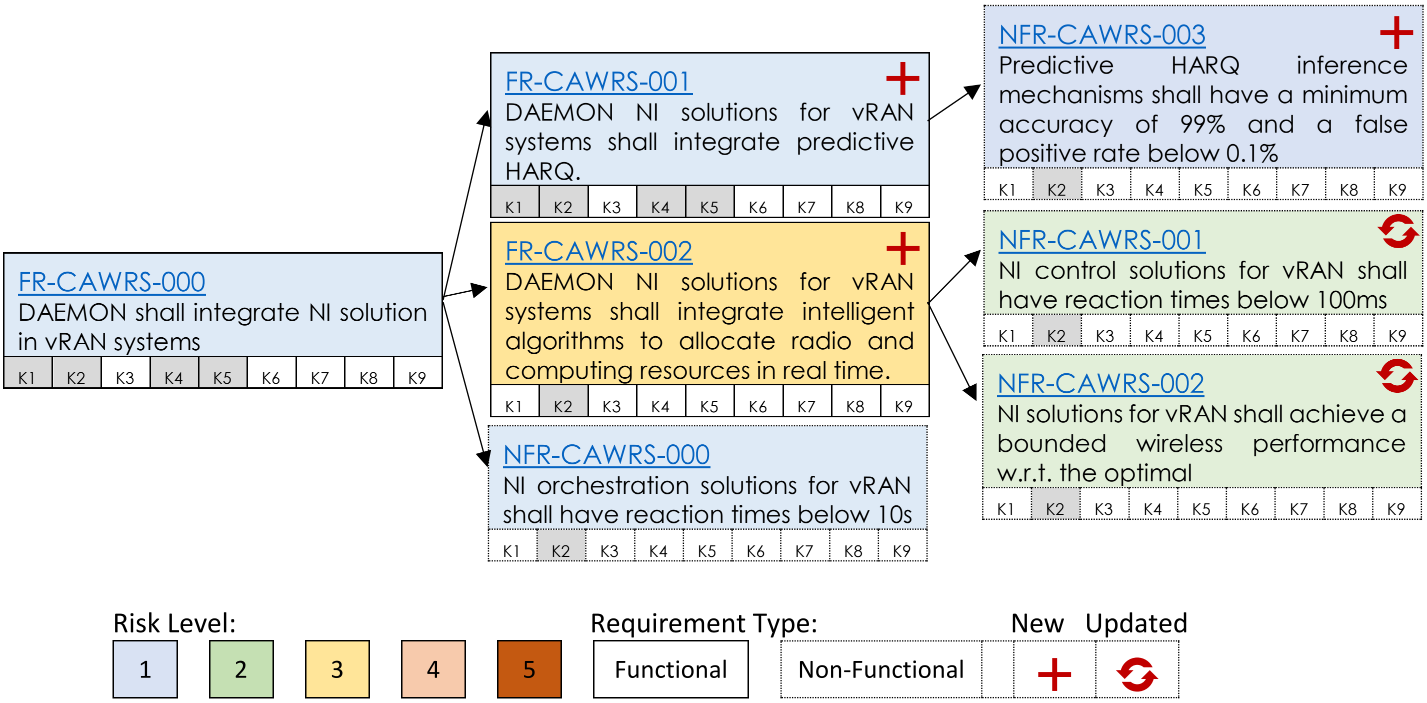

FR-IBSSI-000
DAEMON's IBSSI shall learn network policies using the user plane itself.
K1 K2 K3 K4 K5 K6 K7 K8 K9
FR-IBSSI-001
DAEMON's IBSSI provide Intelligence-as-a-Service to vertical 3rd parties.
K1 K2 K3 K4 K5 K6 K7 K8 K9
FR-IBSSI-002
DAEMON's IBSSI shall integrate Network Intelligence within programmable switches
K1 K2 K3 K4 K5 K6 K7 K8 K9
NFR-IBSSI-000
Network Intelligence algorithms should be adapted to the PISA architecture
K1 K2 K3 K4 K5 K6 K7 K8 K9
NFR-IBSSI-001
Network Intelligence algorithms should be resource-prudent
K1 K2 K3 K4 K5 K6 K7 K8 K9
Risk Level:
1
2
3
4
5
Requirement Type:
Functional
Non-Functional
New
Updated

<table>
<tr><th colspan="18">FR-IBSSI-000</th></tr>
<tr><td colspan="4">Description</td><td colspan="14">DAEMON's IBSSI shall learn network policies using the user plane itself.</td></tr>
<tr><td colspan="4">Version</td><td colspan="14">003M18</td></tr>
<tr><td colspan="4">Owner</td><td colspan="14">UC3M</td></tr>
<tr><td colspan="4">Priority</td><td colspan="14">Low</td></tr>
<tr><td colspan="4">Risk</td><td colspan="14">3</td></tr>
<tr><td colspan="4">Risk Description</td><td colspan="14">To ensure fast reaction times for orchestration mechanisms upon network changes, the network shall learn directly from data-plane network functions, providing triggers for the required re-orchestrations, or re-configurations of the network functions.</td></tr>
<tr><td colspan="4">Rationale</td><td colspan="14">Besides monitoring of KPIs, the network shall already understand and detect malfunctioning already from the analysis of specific traffic patterns or control-plane interactions. This is especially important for operations such as anomaly detection.</td></tr>
<tr><td>K1</td><td></td><td>K2</td><td></td><td>K3</td><td>X</td><td>K4</td><td></td><td>K5</td><td></td><td>K6</td><td></td><td>K7</td><td></td><td>K8</td><td>X</td><td>K9</td><td></td></tr>
<tr><td colspan="4">Parents</td><td colspan="14">None</td></tr>
</table>

<table>
<tr><th colspan="18">FR-IBSSI-001</th></tr>
<tr><td colspan="4">Description</td><td colspan="14">DAEMON's IBSSI provide Intelligence-as-a-Service to vertical 3rd parties</td></tr>
<tr><td colspan="4">Version</td><td colspan="14">003M17</td></tr>
<tr><td colspan="4">Owner</td><td colspan="14">UC3M</td></tr>
<tr><td colspan="4">Priority</td><td colspan="14">High</td></tr>
<tr><td colspan="4">Risk</td><td colspan="14">4</td></tr>
<tr><td colspan="4">Risk Description</td><td colspan="14">Third parties will be allowed to be included in the network operation through specific APIs that are used to i) manage the kind of provided intelligence and ii) ensure that the resources are provided to them. Also, these interfaces shall accommodate different intelligence instances running in the third-party premises and in the network domain.</td></tr>
<tr><td colspan="4">Rationale</td><td colspan="14">DAEMON will provide algorithms for the execution of network intelligence directly related to the vertical service (e.g., video analytics directly in the u-plane) and allow the efficient and secure resource provisioning through the usage of solutions based on e.g., distributed ledger platform.</td></tr>
<tr><td>K1</td><td></td><td>K2</td><td></td><td>K3</td><td>X</td><td>K4</td><td></td><td>K5</td><td></td><td>K6</td><td></td><td>K7</td><td></td><td>K8</td><td>X</td><td>K9</td><td></td></tr>
<tr><td colspan="4">Parents</td><td colspan="14">FR-IBSSI-000</td></tr>
</table>

<table>
<tr><th colspan="18">FR-IBSSI-002</th></tr>
<tr><td colspan="3"><b>Description</b></td><td colspan="15">DAEMON's IBSSI shall integrate Network Intelligence within programmable switches.</td></tr>
<tr><td colspan="3"><b>Version</b></td><td colspan="15">002M17</td></tr>
<tr><td colspan="3"><b>Owner</b></td><td colspan="15">IMDEA</td></tr>
<tr><td colspan="3"><b>Priority</b></td><td colspan="15">Medium</td></tr>
<tr><td colspan="3"><b>Risk</b></td><td colspan="15">3</td></tr>
<tr><td colspan="3"><b>Risk Description</b></td><td colspan="15">Programmable switches have extremely limited computational capabilities and memory, which constrains substantially what they can do in terms of learning.</td></tr>
<tr><td colspan="3"><b>Rationale</b></td><td colspan="15">Programmable user planes are starting to be leveraged for network telemetry functionalities. However, these are limited to data collection and pre-processing, which are then fed to NI located in the control plane to take network management decisions. DAEMON will investigate what portion of the decision process can be moved to the switches directly, at line rate and avoiding the delay of interacting with the control plane.</td></tr>
<tr><td><b>K1</b></td><td></td><td><b>K2</b></td><td></td><td><b>K3</b></td><td><b>X</b></td><td><b>K4</b></td><td></td><td><b>K5</b></td><td></td><td><b>K6</b></td><td></td><td><b>K7</b></td><td></td><td><b>K8</b></td><td></td><td><b>K9</b></td><td></td></tr>
<tr><td colspan="3"><b>Parents</b></td><td colspan="15">FR-IBSSI-000</td></tr>
</table>

A.1.4 Compute-aware radio scheduling



FR-CAWRS-000
DAEMON shall integrate NI solution in vRAN systems

FR-CAWRS-001 +
DAEMON NI solutions for vRAN systems shall integrate predictive HARQ.

FR-CAWRS-002 +
DAEMON NI solutions for vRAN systems shall integrate intelligent algorithms to allocate radio and computing resources in real time.

NFR-CAWRS-000
NI orchestration solutions for vRAN shall have reaction times below 10s

NFR-CAWRS-003 +
Predictive HARQ inference mechanisms shall have a minimum accuracy of 99% and a false positive rate below 0.1%

NFR-CAWRS-001
NI control solutions for vRAN shall have reaction times below 100ms

NFR-CAWRS-002
NI solutions for vRAN shall achieve a bounded wireless performance w.r.t. the optimal

Risk Level: 1 2 3 4 5

Requirement Type: Functional Non-Functional

New + Updated

FR-CAWRS-000	
Description	DAEMON shall integrate NI solution in vRAN systems
Version	001M3
Owner	UC3M
Priority	High
Risk	1
Risk Description	There is a low risk that DAEMON will not integrate NI solutions into vRAN systems, as DAEMON partners were already capable of integrating such kind of solutions in Open Source vRAN environments.
Rationale	The mobile network industry is moving towards virtual network function solutions, and RAN Functions are not an exception. Being among the most resource consuming functions (in terms of computation), thus allowing the re-design of such function by taking into account the computing resource optimization as further objective will improve the overall spending (both CAPEX and OPEX, for the resource provisioning) for the network operation.

K1	X	**K2**	X	**K3**		**K4**	X	**K5**	X	**K6**		**K7**		**K8**		**K9**	

Parents	None

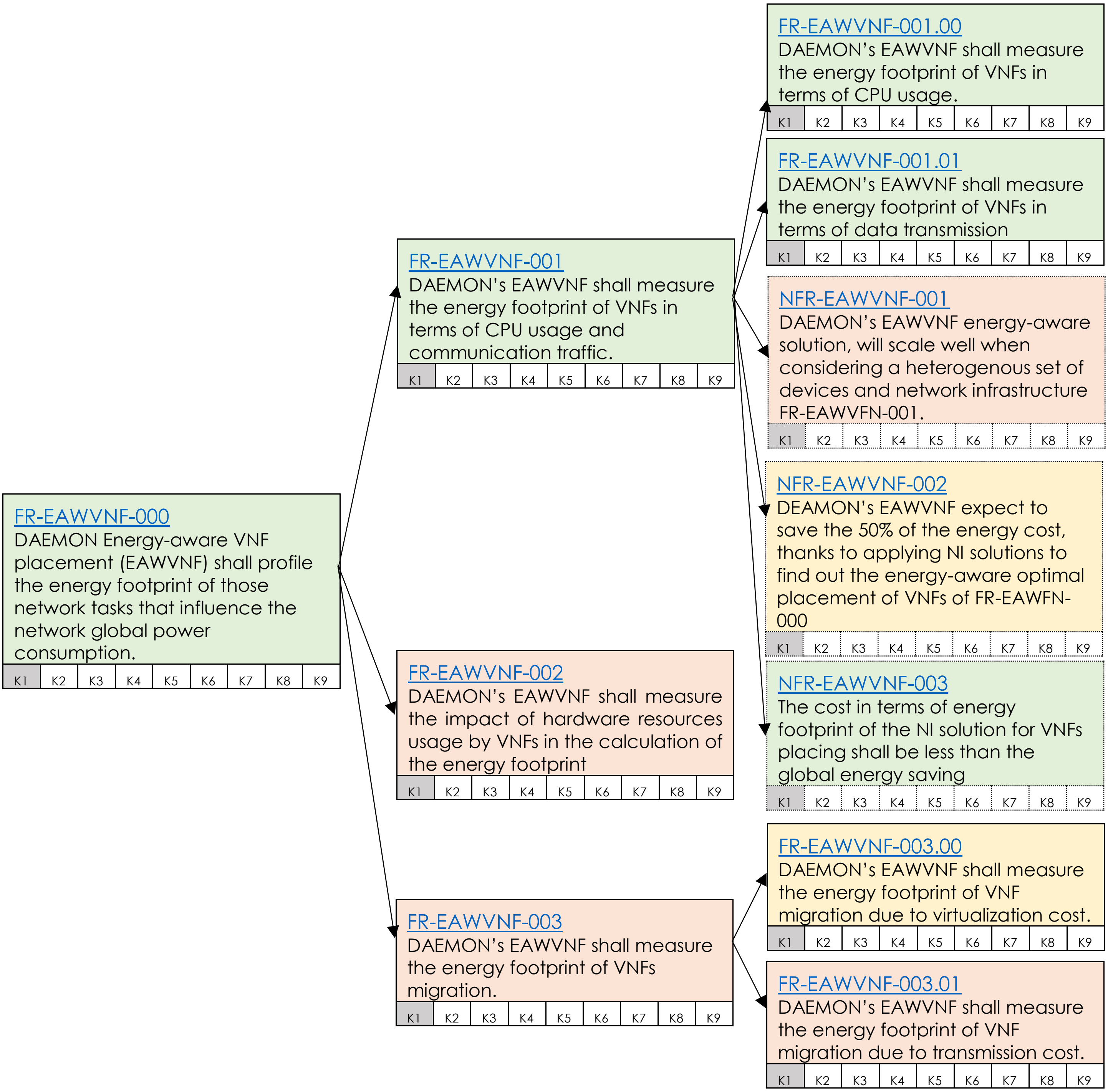
FR-CAWRS-001	
Description	DAEMON NI solutions for vRAN systems shall integrate predictive HARQ.
Version	001M17
Owner	i2CAT
Priority	High
Risk	1
Risk Description	There is a low risk that DAEMON will not integrate predictive HARQ solutions, because they have been widely studied in other contexts before.
Rationale	Predictive HARQ mechanisms collect data from the subframe decoding process and makes a prediction about the decodability of the corresponding transport blocks. This enables the usage of transport blocks that otherwise would have been dropped because they were not decoded on time.

K1	X	**K2**	X	**K3**		**K4**	X	**K5**	X	**K6**		**K7**		**K8**		**K9**	

Parents	FR-CAWRS-000

<table>
<tr><th colspan="18">FR-CAWRS-002</th></tr>
<tr><td colspan="3"><b>Description</b></td><td colspan="15">DAEMON NI solutions for vRAN systems shall integrate intelligent algorithms to allocate radio and computing resources in real time.</td></tr>
<tr><td colspan="3"><b>Version</b></td><td colspan="15">001M17</td></tr>
<tr><td colspan="3"><b>Owner</b></td><td colspan="15">i2CAT</td></tr>
<tr><td colspan="3"><b>Priority</b></td><td colspan="15">High</td></tr>
<tr><td colspan="3"><b>Risk</b></td><td colspan="15">3</td></tr>
<tr><td colspan="3"><b>Risk Description</b></td><td colspan="15">There is a medium risk integrating intelligent algorithms for radio and computing resources allocation in real time because of the reduced operation timescale.</td></tr>
<tr><td colspan="3"><b>Rationale</b></td><td colspan="15">Intelligent radio and computing allocation algorithms provide mechanisms to efficiently distribute the available radio and computing resources, being at the same time crucial for providing latency guarantees and for maximizing the performance of the overall system.</td></tr>
<tr><td><b>K1</b></td><td></td><td><b>K2</b></td><td><b>X</b></td><td><b>K3</b></td><td></td><td><b>K4</b></td><td></td><td><b>K5</b></td><td></td><td><b>K6</b></td><td></td><td><b>K7</b></td><td></td><td><b>K8</b></td><td></td><td><b>K9</b></td><td></td></tr>
<tr><td colspan="3"><b>Parents</b></td><td colspan="15">FR-CAWRS-000</td></tr>
</table>

A.1.5 Energy-aware VNF placement



FR-EAWVNF-000
DAEMON Energy-aware VNF placement (EAWVNF) shall profile the energy footprint of those network tasks that influence the network global power consumption.
K1 K2 K3 K4 K5 K6 K7 K8 K9

FR-EAWVNF-001
DAEMON’s EAWVNF shall measure the energy footprint of VNFs in terms of CPU usage and communication traffic.
K1 K2 K3 K4 K5 K6 K7 K8 K9

FR-EAWVNF-002
DAEMON’s EAWVNF shall measure the impact of hardware resources usage by VNFs in the calculation of the energy footprint
K1 K2 K3 K4 K5 K6 K7 K8 K9

FR-EAWVNF-003
DAEMON’s EAWVNF shall measure the energy footprint of VNFs migration.
K1 K2 K3 K4 K5 K6 K7 K8 K9

FR-EAWVNF-001.00
DAEMON’s EAWVNF shall measure the energy footprint of VNFs in terms of CPU usage.
K1 K2 K3 K4 K5 K6 K7 K8 K9

FR-EAWVNF-001.01
DAEMON’s EAWVNF shall measure the energy footprint of VNFs in terms of data transmission
K1 K2 K3 K4 K5 K6 K7 K8 K9

NFR-EAWVNF-001
DAEMON’s EAWVNF energy-aware solution, will scale well when considering a heterogenous set of devices and network infrastructure FR-EAWVFN-001.
K1 K2 K3 K4 K5 K6 K7 K8 K9

NFR-EAWVNF-002
DEAMON’s EAWVNF expect to save the 50% of the energy cost, thanks to applying NI solutions to find out the energy-aware optimal placement of VNFs of FR-EAWFN-000
K1 K2 K3 K4 K5 K6 K7 K8 K9

NFR-EAWVNF-003
The cost in terms of energy footprint of the NI solution for VNFs placing shall be less than the global energy saving
K1 K2 K3 K4 K5 K6 K7 K8 K9

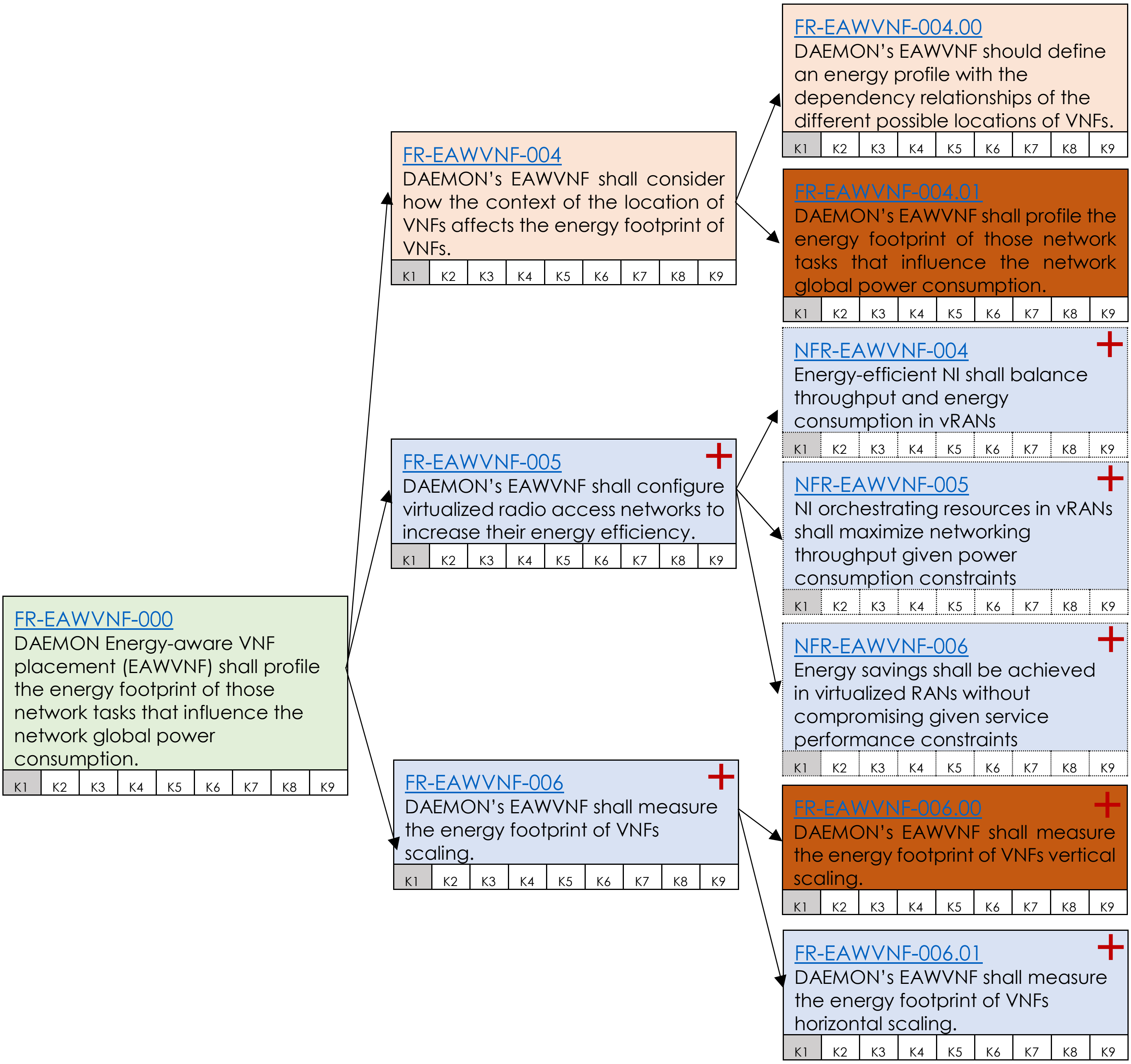
FR-EAWVNF-003.00
DAEMON’s EAWVNF shall measure the energy footprint of VNF migration due to virtualization cost.
K1 K2 K3 K4 K5 K6 K7 K8 K9

FR-EAWVNF-003.01
DAEMON’s EAWVNF shall measure the energy footprint of VNF migration due to transmission cost.
K1 K2 K3 K4 K5 K6 K7 K8 K9

Risk Level: 1 2 3 4 5

Requirement Type: Functional Non-Functional

New ➕ Updated 🔄



FR-EAWVNF-000
DAEMON Energy-aware VNF placement (EAWVNF) shall profile the energy footprint of those network tasks that influence the network global power consumption.

FR-EAWVNF-004
DAEMON's EAWVNF shall consider how the context of the location of VNFs affects the energy footprint of VNFs.

FR-EAWVNF-004.00
DAEMON's EAWVNF should define an energy profile with the dependency relationships of the different possible locations of VNFs.

FR-EAWVNF-004.01
DAEMON's EAWVNF shall profile the energy footprint of those network tasks that influence the network global power consumption.

FR-EAWVNF-005 +
DAEMON's EAWVNF shall configure virtualized radio access networks to increase their energy efficiency.

NFR-EAWVNF-004 +
Energy-efficient NI shall balance throughput and energy consumption in vRANs

NFR-EAWVNF-005 +
NI orchestrating resources in vRANs shall maximize networking throughput given power consumption constraints

NFR-EAWVNF-006 +
Energy savings shall be achieved in virtualized RANs without compromising given service performance constraints

FR-EAWVNF-006 +
DAEMON's EAWVNF shall measure the energy footprint of VNFs scaling.

FR-EAWVNF-006.00 +
DAEMON's EAWVNF shall measure the energy footprint of VNFs vertical scaling.

FR-EAWVNF-006.01 +
DAEMON's EAWVNF shall measure the energy footprint of VNFs horizontal scaling.

Risk Level: 1 2 3 4 5

Requirement Type: Functional Non-Functional

New + Updated ⟳

<table>
<tr><th colspan="18">FR-EAWVNF-000</th></tr>
<tr><td colspan="4"><b>Description</b></td><td colspan="14">DAEMON Energy-aware VNF placement (EAWVNF) shall profile the energy footprint of those network tasks that influence the network global power consumption.</td></tr>
<tr><td colspan="4"><b>Version</b></td><td colspan="14">001M1</td></tr>
<tr><td colspan="4"><b>Owner</b></td><td colspan="14">UMA</td></tr>
<tr><td colspan="4"><b>Priority</b></td><td colspan="14">High</td></tr>
<tr><td colspan="4"><b>Risk</b></td><td colspan="14">2</td></tr>
<tr><td colspan="4"><b>Risk Description</b></td><td colspan="14">The reliability of the measurement depends on a complete identification of the external factors that affect the energy footprint (e.g., temperature, processor, or noisy neighbor problem), the accuracy of the energy measurement methods used, and the dependency on specific hardware. We should be able to estimate the energy consumption of VNFs both in simulated and real environments, obtaining possibly similar results.</td></tr>
<tr><td colspan="4"><b>Rationale</b></td><td colspan="14"></td></tr>
<tr><td><b>K1</b></td><td><b>X</b></td><td><b>K2</b></td><td></td><td><b>K3</b></td><td></td><td><b>K4</b></td><td></td><td><b>K5</b></td><td></td><td><b>K6</b></td><td></td><td><b>K7</b></td><td></td><td><b>K8</b></td><td></td><td><b>K9</b></td><td></td></tr>
<tr><td colspan="4"><b>Parents</b></td><td colspan="14">None</td></tr>
</table>

<table>
<tr><th colspan="18">FR-EAWVNF-001</th></tr>
<tr><td colspan="4"><b>Description</b></td><td colspan="14">DAEMON's EAWVNF shall measure the energy footprint of VNFs in terms of CPU usage and communication traffic.</td></tr>
<tr><td colspan="4"><b>Version</b></td><td colspan="14">001M2</td></tr>
<tr><td colspan="4"><b>Owner</b></td><td colspan="14">UMA</td></tr>
<tr><td colspan="4"><b>Priority</b></td><td colspan="14">High</td></tr>
<tr><td colspan="4"><b>Risk</b></td><td colspan="14">2</td></tr>
<tr><td colspan="4"><b>Risk Description</b></td><td colspan="14">The reliability of the measurement depends on the ability to identify and quantify the influence of external factors in the energy consumption calculation (e.g., noisy neighbor problem or distance to base station). Calculating the cost of executing code and transmitting and receiving information on specific hardware accurately is a complex task. It is possible to mitigate this risk by going through calculating an upper bound of its energy footprint.</td></tr>
<tr><td colspan="4"><b>Rationale</b></td><td colspan="14">The main factors that influence the energy consumption are the CPU usage and the data sent and received by a given VNF. We need to identify what are the factors that should be considered in the formula that calculates the total energy footprint of the network, in terms of computation and communication. We should consider not only the internal factors, as we said, computation and communication, but also the external ones, such as the neighboring traffic. Since our main goal is not to report absolute energy footprint values, but relative ones, we need to find a sound method to quantify the revenue of placing a VNF in one or another location in terms of power saving.</td></tr>
<tr><td><b>K1</b></td><td><b>X</b></td><td><b>K2</b></td><td></td><td><b>K3</b></td><td></td><td><b>K4</b></td><td></td><td><b>K5</b></td><td></td><td><b>K6</b></td><td></td><td><b>K7</b></td><td></td><td><b>K8</b></td><td></td><td><b>K9</b></td><td></td></tr>
<tr><td colspan="4"><b>Parents</b></td><td colspan="14">FR-EAWVNF-000</td></tr>
</table>

FR-EAWVNF-001.00	
Description	DAEMON's EAWVNF shall measure the energy footprint of VNFs in terms of CPU usage.
Version	001M2
Owner	UMA
Priority	High
Risk	2
Risk Description	Calculating the cost of executing any kind of code, on specific hardware accurately is a complex task, since there are several factors that we need to quantify in order to calculate the energy footprint. The theoretical values given by CPU providers usually do not coincide with the real ones.
Rationale	We need to identify what are the factors that should be considered in the formula that calculates the global energy footprint of the VNFs instantiated for each application, in terms of computation. We know that the processor type of the device where a VNFs is running influences the energy footprint, but there are also other parameters that make the software provoke the hardware to consume more energy, like the size of VNF input.

<table>
<tr><td><b>K1</b></td><td><b>X</b></td><td><b>K2</b></td><td></td><td><b>K3</b></td><td></td><td><b>K4</b></td><td></td><td><b>K5</b></td><td></td><td><b>K6</b></td><td></td><td><b>K7</b></td><td></td><td><b>K8</b></td><td></td><td><b>K9</b></td><td></td></tr>
<tr><td colspan="3"><b>Parents</b></td><td colspan="15">FR-EAWVNF-001</td></tr>
</table>

<table>
<tr><th colspan="18">FR-EAWVNF-001.01</th></tr>
<tr><td colspan="3"><b>Description</b></td><td colspan="15">DAEMON's EAWVNF shall measure the energy footprint of VNFs in terms of data transmission.</td></tr>
<tr><td colspan="3"><b>Version</b></td><td colspan="15">001M2</td></tr>
<tr><td colspan="3"><b>Owner</b></td><td colspan="15">UMA</td></tr>
<tr><td colspan="3"><b>Priority</b></td><td colspan="15">High</td></tr>
<tr><td colspan="3"><b>Risk</b></td><td colspan="15">2</td></tr>
<tr><td colspan="3"><b>Risk Description</b></td><td colspan="15">Calculating the cost of data transmission over different types of network links, accurately is a complex task, since there are several factors that we need to quantify in order to calculate the energy footprint. The network throughput is something that varies a lot and depends on some external factors like the current traffic or transmitting neighboring devices.</td></tr>
<tr><td colspan="3"><b>Rationale</b></td><td colspan="15">In the energy footprint calculation, we need to consider that some VNFs will produce some data that might need to be transmitted to other devices. We know that the transmission power, the payload and the transmission rate should be considered, along with other terms.</td></tr>
<tr><td><b>K1</b></td><td><b>X</b></td><td><b>K2</b></td><td></td><td><b>K3</b></td><td></td><td><b>K4</b></td><td></td><td><b>K5</b></td><td></td><td><b>K6</b></td><td></td><td><b>K7</b></td><td></td><td><b>K8</b></td><td></td><td><b>K9</b></td><td></td></tr>
<tr><td colspan="3"><b>Parents</b></td><td colspan="15">FR-EAWVNF-001</td></tr>
</table>

<table>
<tr><th colspan="18">FR-EAWVNF-002</th></tr>
<tr><td colspan="3"><b>Description</b></td><td colspan="15">DAEMON's EAWVNF shall measure the impact of hardware resources usage by VNFs in the calculation of the energy footprint.</td></tr>
<tr><td colspan="3"><b>Version</b></td><td colspan="15">001M2</td></tr>
<tr><td colspan="3"><b>Owner</b></td><td colspan="15">UMA</td></tr>
<tr><td colspan="3"><b>Priority</b></td><td colspan="15">Low</td></tr>
<tr><td colspan="3"><b>Risk</b></td><td colspan="15">4</td></tr>
<tr><td colspan="3"><b>Risk Description</b></td><td colspan="15">The accuracy of the energy consumption measurement depends on specific hardware, including not only the computing device processor. Other hardware, such as memory use or access to HDD, could also influence the total energy footprint, but it is difficult to assess in which percentage. So, it is not easy to estimate it accurately.</td></tr>
<tr><td colspan="3"><b>Rationale</b></td><td colspan="15">Measuring the hardware resources usage of VNFs and their energy footprint provides extra information to accurately estimate the overall energy footprint of a VNF. We are seeking to find additional factors to the energy consumption formula, to calculate more precisely the network energy footprint. Although the DAEMON approach does not need absolute values of energy consumption, we need to find out if there are certain situations where the excessive use of additional resources by a certain VNF, strongly impact the decision of, for example, migrating it to another location.</td></tr>
<tr><td><b>K1</b></td><td><b>X</b></td><td><b>K2</b></td><td></td><td><b>K3</b></td><td></td><td><b>K4</b></td><td></td><td><b>K5</b></td><td></td><td><b>K6</b></td><td></td><td><b>K7</b></td><td></td><td><b>K8</b></td><td></td><td><b>K9</b></td><td></td></tr>
<tr><td colspan="3"><b>Parents</b></td><td colspan="15">FR-EAWVNF-000</td></tr>
</table>

<table>
<tr><th colspan="2">FR-EAWVNF-003</th></tr>
<tr><td><b>Description</b></td><td>DAEMON's EAWVNF shall measure the energy footprint of VNFs migration.</td></tr>
<tr><td><b>Version</b></td><td>001M2</td></tr>
<tr><td><b>Owner</b></td><td>UMA</td></tr>
<tr><td><b>Priority</b></td><td>High</td></tr>
<tr><td><b>Risk</b></td><td>4</td></tr>
<tr><td><b>Risk Description</b></td><td>The cost in terms of energy consumption of code migration in general, and in particular considering VNFs, depends on several factors that we need to identify. Also, there are different mechanisms to perform code migration and each of them requires a different formula for energy footprint calculation, affecting the accuracy of the final result.</td></tr>
</table>

Rationale	The migration of a certain VNF has an energy cost that should be analyzed. It is essential to understand this energy cost to prioritize migrations to other systems if needed.																
K1	**X**	**K2**		**K3**		**K4**		**K5**		**K6**		**K7**		**K8**		**K9**	
Parents	FR-EAWVNF-000																

FR-EAWVNF-003.00																	
Description	DAEMON's EAWVNF shall measure the energy footprint of VNF migration due to virtualization cost.																
Version	001M2																
Owner	UMA																
Priority	High																
Risk	3																
Risk Description	The main risk is that we do not consider all the factors relative to virtualization that affect the energy consumption of migrating a certain VNF. Another risk is that, even when we find a formula to calculate this energy footprint for a certain virtualization technology (or a few of them), later new technologies may appear.																
Rationale	The virtualization has an energy cost and should be analyzed. We should find out if this cost depends on the device (mainly Edge devices and Cloud), and how we can calculate it for both simulated and real environments. It is essential to understand this energy cost to prioritize migrations to other systems if needed. Also, we need to choose the list of virtual machines we are going to consider in this requirement.																
K1	**X**	**K2**		**K3**		**K4**		**K5**		**K6**		**K7**		**K8**		**K9**	
Parents	FR-EAWVNF-003																

FR-EAWVNF-003.01																	
Description	DAEMON's EAWVNF shall measure the energy footprint of VNF migration due to transmission cost.																
Version	001M2																
Owner	UMA																
Priority	High																
Risk	4																
Risk Description	The main risk is that we do not consider all the factors relative to virtualization that affect the energy consumption of migrating a certain VNF. Another risk is that, even when we find out a formula to calculate this energy footprint for a certain virtualization technology (or a few of them), later new technologies appear.																
Rationale	The main factors that affect the energy footprint of VNF migration are the code and data transmission. There are different mechanisms to move a VNF to a different location and each one implies to transfer more or less data. So, the code migration mechanism strongly influences the energy footprint since it varies the amount of information to be transmitted. We should find out how we can calculate it for both simulated and real environments. It is essential to understand this energy cost to prioritize migrations to other systems if needed. Also, we need to decide a single migration mechanism if possible, to be able to calculate its energy footprint.																
K1	**X**	**K2**		**K3**		**K4**		**K5**		**K6**		**K7**		**K8**		**K9**	
Parents	FR-EAWVNF-003																

FR-EAWVNF-004	
Description	DAEMON's EAWVNF shall consider how the context of the location of VNFs affects the energy footprint of VNFs.
Version	001M3
Owner	UMA
Priority	High
Risk	4

<table>
<tr><td colspan="2"><b>Risk Description</b></td><td colspan="16">The main risk is to do not model the context properly due to external and non-measurable artifacts. Moreover, DAEMON could not capture all the possible scenarios related within the location context to model the data's location to feed</td></tr>
<tr><td colspan="2"><b>Rationale</b></td><td colspan="16">The VNF placement cost in terms of energy footprint should consider the execution context where a VNF will be running, and the location of the data that will feed this function. The goal is to adapt the energy footprint of the needed VNFs to the context of the location where they are running.</td></tr>
<tr><td><b>K1</b></td><td><b>X</b></td><td><b>K2</b></td><td></td><td><b>K3</b></td><td></td><td><b>K4</b></td><td></td><td><b>K5</b></td><td></td><td><b>K6</b></td><td></td><td><b>K7</b></td><td></td><td><b>K8</b></td><td></td><td><b>K9</b></td><td></td></tr>
<tr><td colspan="2"><b>Parents</b></td><td colspan="16">FR-EAWVNF-000</td></tr>
</table>

<table>
<tr><th colspan="18">FR-EAWVNF-004.00</th></tr>
<tr><td colspan="2"><b>Description</b></td><td colspan="16">DAEMON's EAWVNF should define an energy profile with the dependency relationships of the different possible locations of VNFs.</td></tr>
<tr><td colspan="2"><b>Version</b></td><td colspan="16">001M3</td></tr>
<tr><td colspan="2"><b>Owner</b></td><td colspan="16">UMA</td></tr>
<tr><td colspan="2"><b>Priority</b></td><td colspan="16">High</td></tr>
<tr><td colspan="2"><b>Risk</b></td><td colspan="16">4</td></tr>
<tr><td colspan="2"><b>Risk Description</b></td><td colspan="16">One possible risk is that we cannot capture all the possible scenarios related with the location context. The variability of execution location contexts and their relationship with the energy footprint could be so high that it is not possible to consider all the cases in the AI algorithms that compute the best solution to deploy a set of VNFs.</td></tr>
<tr><td colspan="2"><b>Rationale</b></td><td colspan="16">To compute the energy footprint of a VNF we need to consider the energy cost depending on the location of the input data, and also the context of the execution location. One possible context could be the quality of the energy consumed, if it is green and renewable energy or polluting energy.</td></tr>
<tr><td><b>K1</b></td><td><b>X</b></td><td><b>K2</b></td><td></td><td><b>K3</b></td><td></td><td><b>K4</b></td><td></td><td><b>K5</b></td><td></td><td><b>K6</b></td><td></td><td><b>K7</b></td><td></td><td><b>K8</b></td><td></td><td><b>K9</b></td><td></td></tr>
<tr><td colspan="2"><b>Parents</b></td><td colspan="16">FR-EAWVNF-004</td></tr>
</table>

<table>
<tr><th colspan="18">FR-EAWVNF-004.01</th></tr>
<tr><td colspan="2"><b>Description</b></td><td colspan="16">DAEMON's EAWVNF should characterize the different variants of VNFs regarding to the context of the location where the VNF will be running.</td></tr>
<tr><td colspan="2"><b>Version</b></td><td colspan="16">001M3</td></tr>
<tr><td colspan="2"><b>Owner</b></td><td colspan="16">UMA</td></tr>
<tr><td colspan="2"><b>Priority</b></td><td colspan="16">High</td></tr>
<tr><td colspan="2"><b>Risk</b></td><td colspan="16">5</td></tr>
<tr><td colspan="2"><b>Risk Description</b></td><td colspan="16">Sometimes the proposed solutions for energy saving cost about the same or sometimes even more than applying a non-energy aware policy. So, we need to assess the cost of computing NI solutions in terms of energy, by adding this cost to the global energy footprint of the solution proposed by DAEMON.</td></tr>
<tr><td colspan="2"><b>Rationale</b></td><td colspan="16">The energy footprint of a VNF could depend on the energy cost of getting the input information depending on the location of the input data. Sometimes, the best solution could be to migrate the VNFs, but other times DAEMON could propose to adapt VNFs so that we can instantiate the most energy efficient version.</td></tr>
<tr><td><b>K1</b></td><td><b>X</b></td><td><b>K2</b></td><td></td><td><b>K3</b></td><td></td><td><b>K4</b></td><td></td><td><b>K5</b></td><td></td><td><b>K6</b></td><td></td><td><b>K7</b></td><td></td><td><b>K8</b></td><td></td><td><b>K9</b></td><td></td></tr>
<tr><td colspan="2"><b>Parents</b></td><td colspan="16">FR-EAWVNF-004</td></tr>
</table>

<table>
<tr><th colspan="2">FR-EAWVNF-005</th></tr>
<tr><td><b>Description</b></td><td>DAEMON's EAWVNF shall configure virtualized radio access networks to increase their energy efficiency</td></tr>
<tr><td><b>Version</b></td><td>001M17</td></tr>
<tr><td><b>Owner</b></td><td>NEC</td></tr>
<tr><td><b>Priority</b></td><td>High</td></tr>
<tr><td><b>Risk</b></td><td>1</td></tr>
<tr><td><b>Risk Description</b></td><td>There is a risk that DAEMON is unable to configure virtualized radio access networks. This risk is low because O-RAN specification shall permit this.</td></tr>
<tr><td><b>Rationale</b></td><td>RAN virtualization promises high flexibility and lower costs but current virtualization techniques render higher energy consumption in the RAN. Hence, it is of</td></tr>
</table>

	paramount important to configure virtualized base stations with their energy consumption in mind																
K1	**X**	**K2**		**K3**		**K4**		**K5**		**K6**		**K7**		**K8**		**K9**	
Parents	FR-EAWVNF-000																

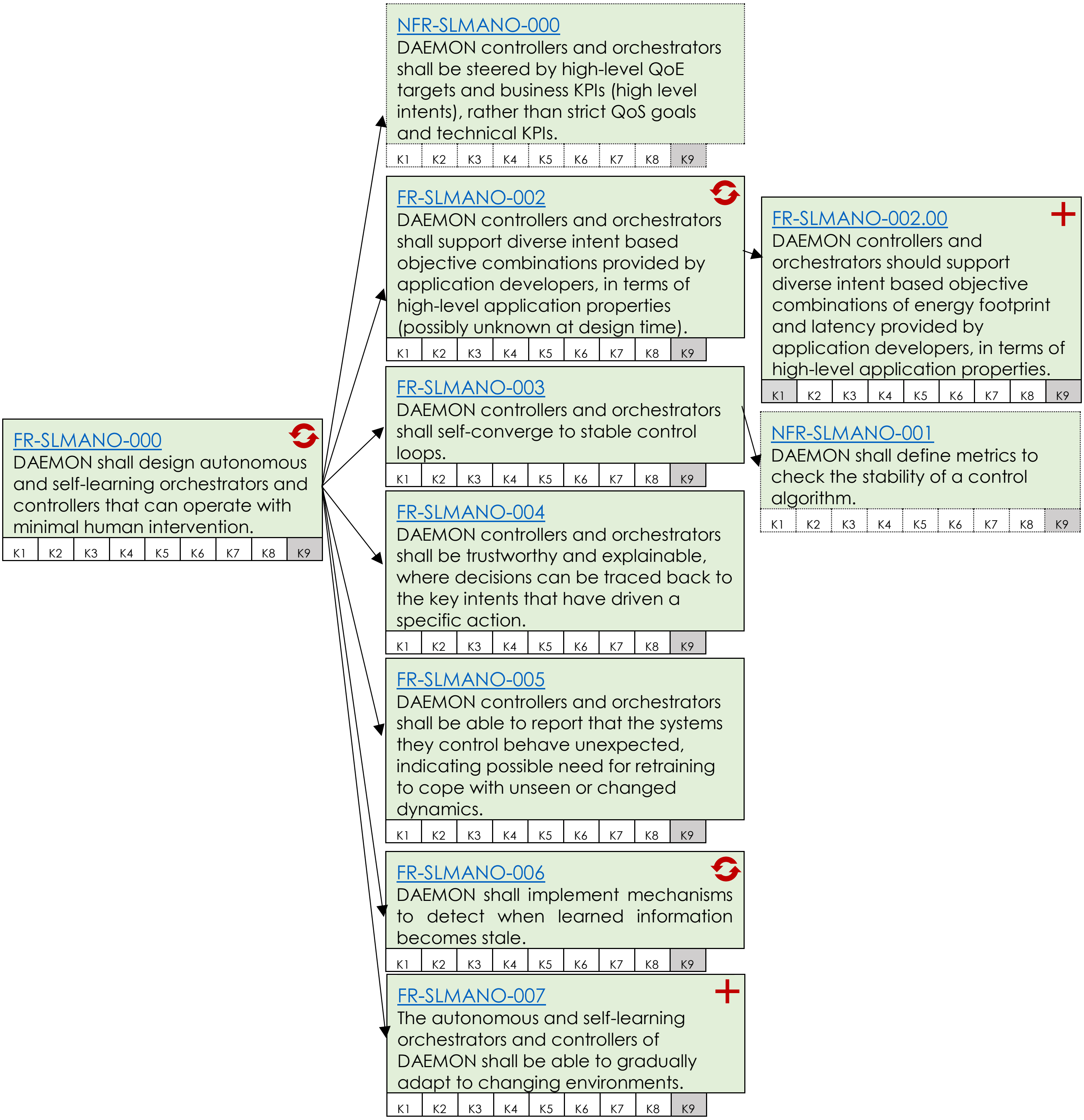
FR-EAWVNF-006																	
Description	DAEMON's EAWVNF shall measure the energy footprint of VNFs scaling.																
Version	001M18																
Owner	UMA																
Priority	Medium																
Risk	2																
Risk Description	The cost in terms of energy consumption of VNFs scaling, depends on several factors that we need to identify. Depending on the approach used to calculate or estimate energy footprint the accuracy of the final result will be more or less adjusted to adjusted to reality.																
Rationale	there are different proposals to perform VNF scaling and each of them need to incorporate an energy profile to calculate or estimate the energy footprint. Scaling up or down a certain VNF frequently according a dynamic demand has an energy cost that should be analyzed. It is essential to understand this energy cost to prioritize if scaling up and down is needed.																
K1	**X**	**K2**		**K3**		**K4**		**K5**		**K6**		**K7**		**K8**		**K9**	
Parents	FR-EAWVNF-000																

FR-EAWVNF-006.00																	
Description	DAEMON's EAWVNF shall measure the energy footprint of VNFs vertical scaling.																
Version	001M18																
Owner	UMA																
Priority	Low																
Risk	5																
Risk Description	The cost of VNF vertical (up/down) scaling to augment (i.e., scale up) the provision of VNF resources depends on several factors that we need to identify. Also, there are different approaches to perform VNF resource allocation and each of them requires a different formula for energy footprint calculation, affecting the accuracy of the final result.																
Rationale	The cost of VNF vertical scaling to augment (i.e., scale up) the provision of VNF resources has an energy cost that should be analyzed. It is essential to understand the energy cost of resource provision to decide when VNF vertical scaling is needed, while taking into account the cost of resource provision actions.																
K1	**X**	**K2**		**K3**		**K4**		**K5**		**K6**		**K7**		**K8**		**K9**	
Parents	FR-EAWVNF-006																

FR-EAWVNF-006.01																	
Description	DAEMON's EAWVNF shall measure the energy footprint of VNFs horizontal scaling.																
Version	001M18																
Owner	UMA																
Priority	Low																
Risk	2																
Risk Description	The main risk is that we do not consider all the factors relative to VNFs horizontal scaling due to virtualization that affect the energy consumption of scaling a certain VNF. Another risk is that, even when we find a formula to calculate this energy footprint for a certain virtualization technology (or a few of them), later new technologies may appear																
Rationale	The horizontal scaling (I.e., in/out) of a VNF has an energy cost due to the virtualization process, which should be analyzed. DAEMON should propose mechanisms to find out the elements that influence this energy cost, such as the HW of target devices (mainly Edge devices and Cloud). Also, DAEMON will propose mechanisms to estimate energy footprint for both simulated and real																

	environments. It is essential to understand this energy cost to prioritize horizontal scaling if needed.																
K1	**X**	**K2**		**K3**		**K4**		**K5**		**K6**		**K7**		**K8**		**K9**	
Parents	FR-EAWVNF-006																

A.1.6 Self-learning MANO



FR-SLMANO-000 🔄
DAEMON shall design autonomous and self-learning orchestrators and controllers that can operate with minimal human intervention.
K1 K2 K3 K4 K5 K6 K7 K8 K9

NFR-SLMANO-000
DAEMON controllers and orchestrators shall be steered by high-level QoE targets and business KPIs (high level intents), rather than strict QoS goals and technical KPIs.
K1 K2 K3 K4 K5 K6 K7 K8 K9

FR-SLMANO-002 🔄
DAEMON controllers and orchestrators shall support diverse intent based objective combinations provided by application developers, in terms of high-level application properties (possibly unknown at design time).
K1 K2 K3 K4 K5 K6 K7 K8 K9

FR-SLMANO-002.00 +
DAEMON controllers and orchestrators should support diverse intent based objective combinations of energy footprint and latency provided by application developers, in terms of high-level application properties.
K1 K2 K3 K4 K5 K6 K7 K8 K9

FR-SLMANO-003
DAEMON controllers and orchestrators shall self-converge to stable control loops.
K1 K2 K3 K4 K5 K6 K7 K8 K9

NFR-SLMANO-001
DAEMON shall define metrics to check the stability of a control algorithm.
K1 K2 K3 K4 K5 K6 K7 K8 K9

FR-SLMANO-004
DAEMON controllers and orchestrators shall be trustworthy and explainable, where decisions can be traced back to the key intents that have driven a specific action.
K1 K2 K3 K4 K5 K6 K7 K8 K9

FR-SLMANO-005
DAEMON controllers and orchestrators shall be able to report that the systems they control behave unexpected, indicating possible need for retraining to cope with unseen or changed dynamics.
K1 K2 K3 K4 K5 K6 K7 K8 K9

FR-SLMANO-006 🔄
DAEMON shall implement mechanisms to detect when learned information becomes stale.
K1 K2 K3 K4 K5 K6 K7 K8 K9

FR-SLMANO-007 +
The autonomous and self-learning orchestrators and controllers of DAEMON shall be able to gradually adapt to changing environments.
K1 K2 K3 K4 K5 K6 K7 K8 K9

Risk Level: 1 2 3 4 5

Requirement Type: Functional Non-Functional

New + Updated 🔄

<table>
<tr><th colspan="18">FR-SLMANO-000</th></tr>
<tr><td colspan="4"><b>Description</b></td><td colspan="14">DAEMON shall design autonomous and self-learning orchestrators and controllers that can operate with minimal human intervention.</td></tr>
<tr><td colspan="4"><b>Version</b></td><td colspan="14">002M17</td></tr>
<tr><td colspan="4"><b>Owner</b></td><td colspan="14">NBL</td></tr>
<tr><td colspan="4"><b>Priority</b></td><td colspan="14">High</td></tr>
<tr><td colspan="4"><b>Risk</b></td><td colspan="14">2</td></tr>
<tr><td colspan="4"><b>Risk Description</b></td><td colspan="14">Only regularly repeating patterns can be learned. Stochastic fluctuation on top of these regular patterns hamper learning and need to be filtered out. The learning rate, number of epoch (the number of times that is run through the data) and exploitation versus exploration balance need to be carefully chosen. Moreover, the behavior of the system can change either slowly (as the system evolves) or suddenly (when, e.g., new software is installed on some of the components. Both need to be handled.</td></tr>
<tr><td colspan="4"><b>Rationale</b></td><td colspan="14">Any decision that the orchestration and control functions can be envisioned to be automatized in the following way. First, the software agent taking the decisions needs to be provided (in a timely way) with the data necessary to take its decisions. The agent relies on the policy currently in force to take the appropriate action. With each action taken (given the provided data) the agent is provided with feedback that expresses how good that action was given the current data. Based on this feedback the agent can change its policy to steer the system in the desired direction.</td></tr>
<tr><td><b>K1</b></td><td></td><td><b>K2</b></td><td></td><td><b>K3</b></td><td></td><td><b>K4</b></td><td></td><td><b>K5</b></td><td></td><td><b>K6</b></td><td></td><td><b>K7</b></td><td></td><td><b>K8</b></td><td></td><td><b>K9</b></td><td>X</td></tr>
<tr><td colspan="4"><b>Parents</b></td><td colspan="14">None.</td></tr>
</table>

<table>
<tr><th colspan="18">FR-SLMANO-002</th></tr>
<tr><td colspan="4"><b>Description</b></td><td colspan="14">DAEMON controllers and orchestrators should support diverse intent based objective combinations provided by application developers, in terms of high-level application properties (possibly unknown at design time).</td></tr>
<tr><td colspan="4"><b>Version</b></td><td colspan="14">002M17</td></tr>
<tr><td colspan="4"><b>Owner</b></td><td colspan="14">NBL</td></tr>
<tr><td colspan="4"><b>Priority</b></td><td colspan="14">High</td></tr>
<tr><td colspan="4"><b>Risk</b></td><td colspan="14">2</td></tr>
<tr><td colspan="4"><b>Risk Description</b></td><td colspan="14">Can a single algorithm fulfil this requirement for all use cases (e.g., URLLC (ultra-low latency reliable communication), EMBB (enhanced mobile broadband), and MMTC (massive machine type communication) defined in 5G)? Will it be too complex? Is it better to train multiple competing algorithms for each specific use case and select the best performing?</td></tr>
<tr><td colspan="4"><b>Rationale</b></td><td colspan="14">Business KPIs will change frequently due to highly varying markets. The algorithms provided by DAEMON need to be flexible enough to self-learn and converge to sufficiently optimized behavior to avoid human intervention for retuning or redesigning the algorithms and mechanisms. If a classical algorithm has still parameters to tune a procedure to tune (i.e., learn) these parameters needs to be designed and investigated.</td></tr>
<tr><td><b>K1</b></td><td></td><td><b>K2</b></td><td></td><td><b>K3</b></td><td></td><td><b>K4</b></td><td></td><td><b>K5</b></td><td></td><td><b>K6</b></td><td></td><td><b>K7</b></td><td></td><td><b>K8</b></td><td></td><td><b>K9</b></td><td>X</td></tr>
<tr><td colspan="4"><b>Parents</b></td><td colspan="14">FR-SLMANO-000</td></tr>
</table>

<table>
<tr><th colspan="2">FR-SLMANO-002.00</th></tr>
<tr><td><b>Description</b></td><td>DAEMON controllers and orchestrators should support diverse intent based objective combinations of energy footprint and latency provided by application developers, in terms of high-level application properties.</td></tr>
<tr><td><b>Version</b></td><td>002M17</td></tr>
<tr><td><b>Owner</b></td><td>UMA</td></tr>
<tr><td><b>Priority</b></td><td>High</td></tr>
<tr><td><b>Risk</b></td><td>2</td></tr>
<tr><td><b>Risk Description</b></td><td>The energy consumption calculation of placement decision should take into consideration also the requirements in terms of the latency/time response/timescale needs of service function chains. The algorithm should find</td></tr>
</table>

<table>
<tr><td colspan="3"></td><td colspan="15">the best tradeoff between the most fitting timescale and the energy saving requirements for the VNF placement.</td></tr>
<tr><td colspan="3"><b>Rationale</b></td><td colspan="15">Following the cloud-native service design, the service function chains consist of loosely-coupled VNFs that can be separately placed. Orchestration managers can make the VNFs placement decisions, while maintaining the service provision. The algorithms provided by DAEMON that make orchestration decisions should consider latency/time response/timescale needs of service function chains and at the same time the energy footprint. So, to make a tradeoff between time and energy footprint is desired.</td></tr>
<tr><td><b>K1</b></td><td>X</td><td><b>K2</b></td><td></td><td><b>K3</b></td><td></td><td><b>K4</b></td><td></td><td><b>K5</b></td><td></td><td><b>K6</b></td><td></td><td><b>K7</b></td><td></td><td><b>K8</b></td><td></td><td><b>K9</b></td><td>X</td></tr>
<tr><td colspan="3"><b>Parents</b></td><td colspan="15">FR-SLMANO-002</td></tr>
</table>

<table>
<tr><th colspan="18">FR-SLMANO-003</th></tr>
<tr><td colspan="3"><b>Description</b></td><td colspan="15">DAEMON controllers and orchestrators should self-converge to stable control loops.</td></tr>
<tr><td colspan="3"><b>Version</b></td><td colspan="15">002M17</td></tr>
<tr><td colspan="3"><b>Owner</b></td><td colspan="15">NBL</td></tr>
<tr><td colspan="3"><b>Priority</b></td><td colspan="15">High</td></tr>
<tr><td colspan="3"><b>Risk</b></td><td colspan="15">3</td></tr>
<tr><td colspan="3"><b>Risk Description</b></td><td colspan="15">Can we capture realistic dynamic behavior in the systems, emulators or simulators that we will use? How can stability be verified?</td></tr>
<tr><td colspan="3"><b>Rationale</b></td><td colspan="15">Different parts of the system will have different dynamics, potentially changing over time due to SW and HW upgrades. The algorithms provided by DAEMON need to be intelligent enough to self-learn and converge to a stable though responsive behavior without human intervention for retuning or redesigning the algorithms and mechanisms.
In general, a system operating in steady state is stable if after an infinitesimally short, small enough perturbation applied to it dies out exponentially fast so that it returns to that steady state working point. The perturbation needs to be short compared to the reaction time inherent to the system and small so that it does not jump to another working point.</td></tr>
<tr><td><b>K1</b></td><td></td><td><b>K2</b></td><td></td><td><b>K3</b></td><td></td><td><b>K4</b></td><td></td><td><b>K5</b></td><td></td><td><b>K6</b></td><td></td><td><b>K7</b></td><td></td><td><b>K8</b></td><td></td><td><b>K9</b></td><td>X</td></tr>
<tr><td colspan="3"><b>Parents</b></td><td colspan="15">FR-SLMANO-000</td></tr>
</table>

<table>
<tr><th colspan="18">FR-SLMANO-004</th></tr>
<tr><td colspan="3"><b>Description</b></td><td colspan="15">DAEMON controllers and orchestrators should be trustworthy and explainable, where decisions can be traced back to the key intents that have driven a specific action.</td></tr>
<tr><td colspan="3"><b>Version</b></td><td colspan="15">003M17</td></tr>
<tr><td colspan="3"><b>Owner</b></td><td colspan="15">NBL</td></tr>
<tr><td colspan="3"><b>Priority</b></td><td colspan="15">Low</td></tr>
<tr><td colspan="3"><b>Risk</b></td><td colspan="15">5</td></tr>
<tr><td colspan="3"><b>Risk Description</b></td><td colspan="15">Often ML tools are black boxes that after training work well, but do not give any indication of why they work. Human network operators might distrust such tools and hence, be reluctant to use them. Moreover, when decisions are taken at multiple layers at different timescales, conflicts may arise amongst agents operating at different timescales (possibly due to a human error when setting the goals).</td></tr>
<tr><td colspan="3"><b>Rationale</b></td><td colspan="15">In a complex composition of multi-layer controllers, conflicts between different levels of intents need to be visualized, such that unexpected unwanted behavior can be analyzed and revised in terms of the active and potentially erroneously specified intents (due to human errors).</td></tr>
<tr><td><b>K1</b></td><td></td><td><b>K2</b></td><td></td><td><b>K3</b></td><td></td><td><b>K4</b></td><td></td><td><b>K5</b></td><td></td><td><b>K6</b></td><td></td><td><b>K7</b></td><td></td><td><b>K8</b></td><td></td><td><b>K9</b></td><td>X</td></tr>
<tr><td colspan="3"><b>Parents</b></td><td colspan="15">FR-SLMANO-000</td></tr>
</table>

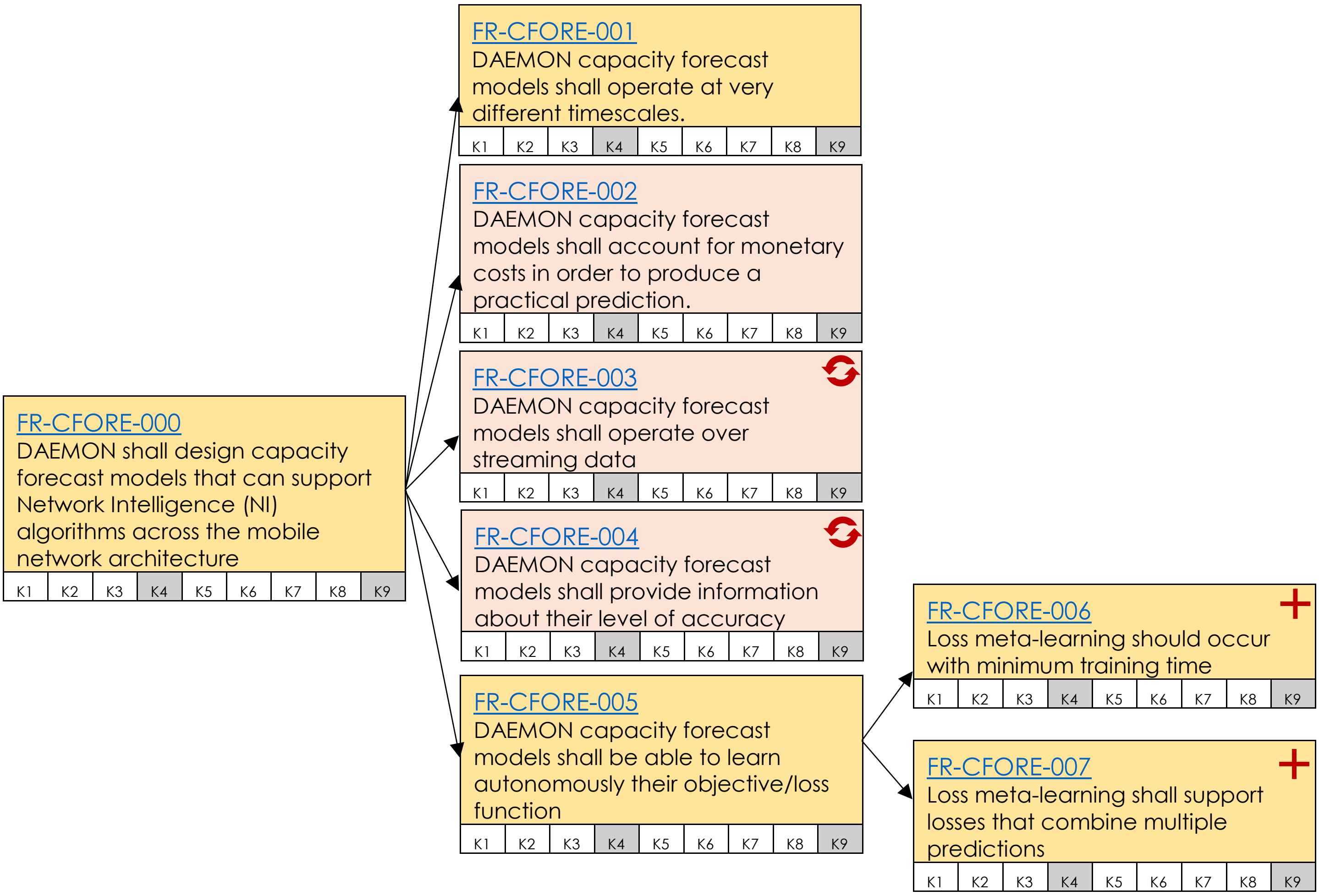
<table>
<tr><th colspan="2">FR-SLMANO-005</th></tr>
<tr><td><b>Description</b></td><td>DAEMON controllers and orchestrators should be able to report that the systems they control behave unexpected, indicating possible need for retraining to cope with unseen or changed dynamics.</td></tr>
</table>

<table>
<tr><td><b>Version</b></td><td colspan="17">001M2</td></tr>
<tr><td><b>Owner</b></td><td colspan="17">NBL</td></tr>
<tr><td><b>Priority</b></td><td colspan="17">High</td></tr>
<tr><td><b>Risk</b></td><td colspan="17">3</td></tr>
<tr><td><b>Risk Description</b></td><td colspan="17">There is a risk that spurious changes are seen by the system as changes in the environment, causing the system to retrain (doing a lot of exploration and making the associated wrong decisions) where it is not needed.</td></tr>
<tr><td><b>Rationale</b></td><td colspan="17">If online retraining is prohibited, or if the algorithms are incapable of self-converging to a sufficient solution, the need for human intervention needs to be reported.</td></tr>
<tr><td><b>K1</b></td><td></td><td><b>K2</b></td><td></td><td><b>K3</b></td><td></td><td><b>K4</b></td><td></td><td><b>K5</b></td><td></td><td><b>K6</b></td><td></td><td><b>K7</b></td><td></td><td><b>K8</b></td><td></td><td><b>K9</b></td><td>X</td></tr>
<tr><td><b>Parents</b></td><td colspan="17">FR-SLMANO-000</td></tr>
</table>

<table>
<tr><th colspan="18">FR-SLMANO-006</th></tr>
<tr><td><b>Description</b></td><td colspan="17">DAEMON shall implement mechanisms to detect when learned information becomes stale.</td></tr>
<tr><td><b>Version</b></td><td colspan="17">002M17</td></tr>
<tr><td><b>Owner</b></td><td colspan="17">NBL</td></tr>
<tr><td><b>Priority</b></td><td colspan="17">Medium</td></tr>
<tr><td><b>Risk</b></td><td colspan="17">3</td></tr>
<tr><td><b>Risk Description</b></td><td colspan="17">The behavior of the system can change suddenly (when, e.g., a flash crowd arrives generating a lot of traffic, when new software is installed on some of the components, when there is an outage of part of the infrastructure), which makes that the policy learned on past system behavior is no longer applicable. Therefore, a system is needed to detect when learned information becomes stale indicating when retraining is required.</td></tr>
<tr><td><b>Rationale</b></td><td colspan="17">In the framework defined under FR-SLMANO-000-001M2, where the software agent taking the decisions is provided (in a timely way) with the data necessary to take its decisions and where with each action taken (given the provided data), the agent is provided with quantitative feedback that expresses how good that action was, a change in behavior can be detected by observing the evolution of the feedback. If there is a drastic change, the balance between exploration and exploitation needs to be tilted in favor of exploration so that the system can be retrained to work properly in the new environment.</td></tr>
<tr><td><b>K1</b></td><td></td><td><b>K2</b></td><td></td><td><b>K3</b></td><td></td><td><b>K4</b></td><td></td><td><b>K5</b></td><td></td><td><b>K6</b></td><td></td><td><b>K7</b></td><td></td><td><b>K8</b></td><td></td><td><b>K9</b></td><td>X</td></tr>
<tr><td><b>Parents</b></td><td colspan="17">FR-SLMANO-000</td></tr>
</table>

<table>
<tr><th colspan="18">FR-SLMANO-007</th></tr>
<tr><td><b>Description</b></td><td colspan="17">The autonomous and self-learning orchestrators and controllers of DAEMON shall be able to gradually adapt to changing environments.</td></tr>
<tr><td><b>Version</b></td><td colspan="17">001M17</td></tr>
<tr><td><b>Owner</b></td><td colspan="17">NBL</td></tr>
<tr><td><b>Priority</b></td><td colspan="17">Medium</td></tr>
<tr><td><b>Risk</b></td><td colspan="17">2</td></tr>
<tr><td><b>Risk Description</b></td><td colspan="17">The behavior of the system can change slowly together with the usage patterns. DAEMON self-learning MANO need to follow these changes otherwise the decisions it takes will gradually become worse. This can be achieved by setting a good balance between exploration and exploitation.</td></tr>
<tr><td><b>Rationale</b></td><td colspan="17">A learning system that relies on feedback to improve its policy, can gradually learn by taking from time to time exploratory actions (i.e., random actions which are deemed not to be optimal by the current policy). Usually, the fraction of exploration actions is large (close to 100%) at the start of the learning process, and gradually reduces to 0 as the system learns. In order to be able to adapt to a changing environment, the exploration fraction is kept at, say 10%.</td></tr>
<tr><td><b>K1</b></td><td></td><td><b>K2</b></td><td></td><td><b>K3</b></td><td></td><td><b>K4</b></td><td></td><td><b>K5</b></td><td></td><td><b>K6</b></td><td></td><td><b>K7</b></td><td></td><td><b>K8</b></td><td></td><td><b>K9</b></td><td>X</td></tr>
<tr><td><b>Parents</b></td><td colspan="17">FR-SLMANO-000</td></tr>
</table>

A.1.7 Capacity forecasting



FR-CFORE-000
DAEMON shall design capacity forecast models that can support Network Intelligence (NI) algorithms across the mobile network architecture

FR-CFORE-001
DAEMON capacity forecast models shall operate at very different timescales.

FR-CFORE-002
DAEMON capacity forecast models shall account for monetary costs in order to produce a practical prediction.

FR-CFORE-003
DAEMON capacity forecast models shall operate over streaming data

FR-CFORE-004
DAEMON capacity forecast models shall provide information about their level of accuracy

FR-CFORE-005
DAEMON capacity forecast models shall be able to learn autonomously their objective/loss function

FR-CFORE-006
Loss meta-learning should occur with minimum training time

FR-CFORE-007
Loss meta-learning shall support losses that combine multiple predictions

Risk Level: 1 2 3 4 5

Requirement Type: Functional Non-Functional

New + Updated 🔄

<table>
<tr><th colspan="18">FR-CFORE-000</th></tr>
<tr><td colspan="3"><b>Description</b></td><td colspan="15">DAEMON Capacity Forecasting (CFORE) shall design models capable of anticipating the amount of resources needed to accommodate future mobile service demands, so as to support Network Intelligence (NI) algorithms across the mobile network architecture.</td></tr>
<tr><td colspan="3"><b>Version</b></td><td colspan="15">001M2</td></tr>
<tr><td colspan="3"><b>Owner</b></td><td colspan="15">IMDEA</td></tr>
<tr><td colspan="3"><b>Priority</b></td><td colspan="15">High</td></tr>
<tr><td colspan="3"><b>Risk</b></td><td colspan="15">3</td></tr>
<tr><td colspan="3"><b>Risk Description</b></td><td colspan="15">The main risk is that the forecasting models do not achieve the accuracy needed to support efficient decision-making, hence limiting the effectiveness of NI.</td></tr>
<tr><td colspan="3"><b>Rationale</b></td><td colspan="15">Many decisions to be taken by orchestrators and controllers deployed across different micro-domains of the mobile network must be taken in anticipatory manner, i.e., proactively with respect to the actual demand or requirements. Such decisions concern the capacity that orchestrators and controllers shall allocate in their micro-domain of competence. Predicting such capacity is thus a key enabler for the NI operating across the whole network.</td></tr>
<tr><td><b>K1</b></td><td></td><td><b>K2</b></td><td></td><td><b>K3</b></td><td></td><td><b>K4</b></td><td><b>X</b></td><td><b>K5</b></td><td></td><td><b>K6</b></td><td></td><td><b>K7</b></td><td></td><td><b>K8</b></td><td></td><td><b>K9</b></td><td><b>X</b></td></tr>
<tr><td colspan="3"><b>Parents</b></td><td colspan="15">None</td></tr>
</table>

<table>
<tr><th colspan="18">FR-CFORE-001</th></tr>
<tr><td colspan="3"><b>Description</b></td><td colspan="15">DAEMON capacity forecast models shall operate at very different timescales</td></tr>
<tr><td colspan="3"><b>Version</b></td><td colspan="15">001M2</td></tr>
<tr><td colspan="3"><b>Owner</b></td><td colspan="15">IMDEA</td></tr>
<tr><td colspan="3"><b>Priority</b></td><td colspan="15">Medium</td></tr>
<tr><td colspan="3"><b>Risk</b></td><td colspan="15">3</td></tr>
<tr><td colspan="3"><b>Risk Description</b></td><td colspan="15">The risk of insufficient accuracy in the prediction is exacerbated in as timescales become faster, as traffic demands are increasingly bursty, and the changes in requirements more and more rapid.</td></tr>
<tr><td colspan="3"><b>Rationale</b></td><td colspan="15">Orchestrators and controllers operate at very different timescales across the diverse network domains, and take decisions over intervals that range from hours to seconds or less depending on the nature of the concerned resources (e.g., computing resources, transport capacity, spectrum, etc.). Capacity forecasting models must be adapted to such diverse settings.</td></tr>
<tr><td><b>K1</b></td><td></td><td><b>K2</b></td><td></td><td><b>K3</b></td><td></td><td><b>K4</b></td><td><b>X</b></td><td><b>K5</b></td><td></td><td><b>K6</b></td><td></td><td><b>K7</b></td><td></td><td><b>K8</b></td><td></td><td><b>K9</b></td><td><b>X</b></td></tr>
<tr><td colspan="3"><b>Parents</b></td><td colspan="15">FR-CFORE-000</td></tr>
</table>

<table>
<tr><th colspan="18">FR-CFORE-002</th></tr>
<tr><td colspan="3"><b>Description</b></td><td colspan="15">DAEMON capacity forecast models shall account for monetary costs in order to produce a practical prediction</td></tr>
<tr><td colspan="3"><b>Version</b></td><td colspan="15">001M2</td></tr>
<tr><td colspan="3"><b>Owner</b></td><td colspan="15">IMDEA</td></tr>
<tr><td colspan="3"><b>Priority</b></td><td colspan="15">High</td></tr>
<tr><td colspan="3"><b>Risk</b></td><td colspan="15">4</td></tr>
<tr><td colspan="3"><b>Risk Description</b></td><td colspan="15">Considering a high number of cost sources makes the forecasting problem more involved and identifying the correct capacity prediction becomes harder in general.</td></tr>
<tr><td colspan="3"><b>Rationale</b></td><td colspan="15">Predicting the sheer capacity needed to accommodate the traffic demand is not sufficient in many practical applications of capacity forecasting to network orchestration and control. Often, decisions on the allocation resources and Virtual Network Functions (VNFs) must consider the costs incurred by the network operator (e.g., unnecessarily assigned resources that go unused, Service Level Agreement violations, VNF reconfiguration delays that determine subscriber churn, energy consumption generated by running VNFs at different network elements, etc.). Designing models that can capture such costs, and output a capacity that jointly reduces them, is critical to the economic sustainability of the network management process.</td></tr>
<tr><td><b>K1</b></td><td></td><td><b>K2</b></td><td></td><td><b>K3</b></td><td></td><td><b>K4</b></td><td><b>X</b></td><td><b>K5</b></td><td></td><td><b>K6</b></td><td></td><td><b>K7</b></td><td></td><td><b>K8</b></td><td></td><td><b>K9</b></td><td><b>X</b></td></tr>
<tr><td colspan="3"><b>Parents</b></td><td colspan="15">FR-CFORE-000</td></tr>
</table>

<table>
<tr><th colspan="18">FR-CFORE-003</th></tr>
<tr><td colspan="3"><b>Description</b></td><td colspan="15">DAEMON capacity forecast models shall operate over streaming data</td></tr>
<tr><td colspan="3"><b>Version</b></td><td colspan="15">001M5</td></tr>
<tr><td colspan="3"><b>Owner</b></td><td colspan="15">IMDEA</td></tr>
<tr><td colspan="3"><b>Priority</b></td><td colspan="15">High</td></tr>
<tr><td colspan="3"><b>Risk</b></td><td colspan="15">4</td></tr>
<tr><td colspan="3"><b>Risk Description</b></td><td colspan="15">Adapting capacity forecasting to support a streaming model adds complexity and challenges to the design of the solution, which may reduce its efficiency.</td></tr>
<tr><td colspan="3"><b>Rationale</b></td><td colspan="15">While many traffic forecasting models are trained off-line and tested on historical data, the operation of such models in production calls for training and operation on traffic data as it is measured in the network. This implicitly means that capacity forecasting models must be adapted to work on streaming data.</td></tr>
<tr><td><b>K1</b></td><td></td><td><b>K2</b></td><td></td><td><b>K3</b></td><td></td><td><b>K4</b></td><td><b>X</b></td><td><b>K5</b></td><td></td><td><b>K6</b></td><td></td><td><b>K7</b></td><td></td><td><b>K8</b></td><td></td><td><b>K9</b></td><td><b>X</b></td></tr>
<tr><td colspan="3"><b>Parents</b></td><td colspan="15">FR-CFORE-000</td></tr>
</table>

<table>
<tr><th colspan="18">FR-CFORE-004</th></tr>
<tr><td colspan="3"><b>Description</b></td><td colspan="15">DAEMON capacity forecast models shall provide information about their level of accuracy</td></tr>
<tr><td colspan="3"><b>Version</b></td><td colspan="15">001M17</td></tr>
<tr><td colspan="3"><b>Owner</b></td><td colspan="15">IMDEA</td></tr>
<tr><td colspan="3"><b>Priority</b></td><td colspan="15">Low</td></tr>
<tr><td colspan="3"><b>Risk</b></td><td colspan="15">4</td></tr>
<tr><td colspan="3"><b>Risk Description</b></td><td colspan="15">Anticipating not only the target variable but also the uncertainty of its estimate makes the prediction task sensibly more complex.</td></tr>
<tr><td colspan="3"><b>Rationale</b></td><td colspan="15">Having information on the uncertainty of the prediction can help fine-tuning resource allocation, e.g., by including safety margins dimensioned on the level of expected accuracy of the forecasting model.</td></tr>
<tr><td><b>K1</b></td><td></td><td><b>K2</b></td><td></td><td><b>K3</b></td><td></td><td><b>K4</b></td><td><b>X</b></td><td><b>K5</b></td><td></td><td><b>K6</b></td><td></td><td><b>K7</b></td><td></td><td><b>K8</b></td><td></td><td><b>K9</b></td><td><b>X</b></td></tr>
<tr><td colspan="3"><b>Parents</b></td><td colspan="15">FR-CFORE-000</td></tr>
</table>

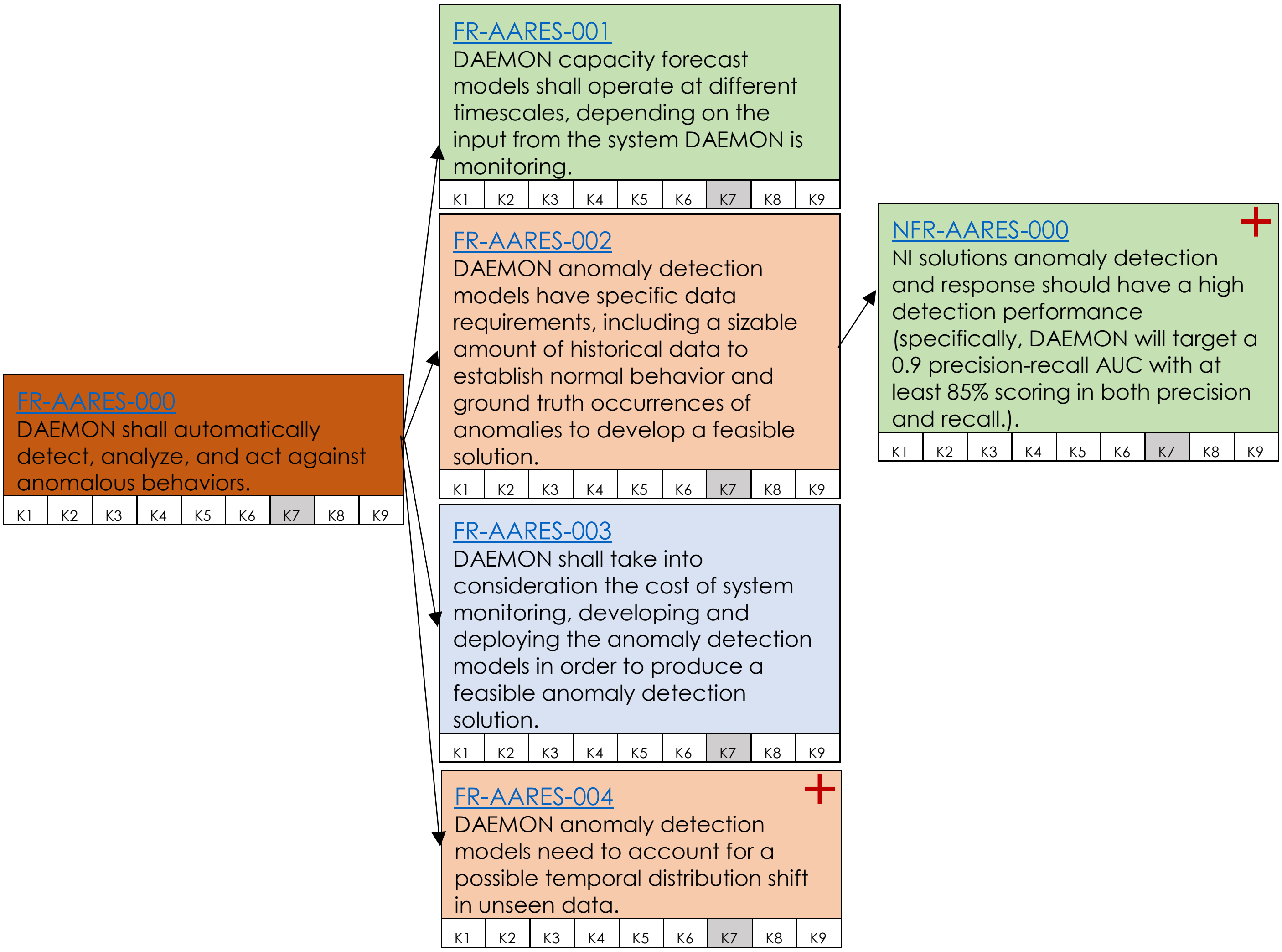
<table>
<tr><th colspan="18">FR-CFORE-005</th></tr>
<tr><td colspan="3"><b>Description</b></td><td colspan="15">DAEMON capacity forecast models shall be able to learn autonomously their objective/loss function</td></tr>
<tr><td colspan="3"><b>Version</b></td><td colspan="15">001M17</td></tr>
<tr><td colspan="3"><b>Owner</b></td><td colspan="15">IMDEA</td></tr>
<tr><td colspan="3"><b>Priority</b></td><td colspan="15">High</td></tr>
<tr><td colspan="3"><b>Risk</b></td><td colspan="15">3</td></tr>
<tr><td colspan="3"><b>Risk Description</b></td><td colspan="15">Meta-learning the correct loss from scratches is a challenging task, for which no solution exists in the machine learning community.</td></tr>
<tr><td colspan="3"><b>Rationale</b></td><td colspan="15">Many network management tasks involve situations where the relationship between the prediction (e.g., of resources to be allocated) and the performance (e.g., quality of experience of users) is unknown a-priori. In these settings, designing a correct loss function for machine learning is not possible, and meta-learning the loss is the only viable option.</td></tr>
<tr><td><b>K1</b></td><td></td><td><b>K2</b></td><td></td><td><b>K3</b></td><td></td><td><b>K4</b></td><td><b>X</b></td><td><b>K5</b></td><td></td><td><b>K6</b></td><td></td><td><b>K7</b></td><td></td><td><b>K8</b></td><td></td><td><b>K9</b></td><td><b>X</b></td></tr>
<tr><td colspan="3"><b>Parents</b></td><td colspan="15">FR-CFORE-000</td></tr>
</table>

<table>
<tr><th colspan="2">FR-CFORE-006</th></tr>
<tr><td><b>Description</b></td><td>Loss meta-learning should occur with minimum training time</td></tr>
<tr><td><b>Version</b></td><td>001M17</td></tr>
<tr><td><b>Owner</b></td><td>IMDEA</td></tr>
<tr><td><b>Priority</b></td><td>Medium</td></tr>
<tr><td><b>Risk</b></td><td>3</td></tr>
<tr><td><b>Risk Description</b></td><td>Meta-learning the loss inherently increases the time to convergence of a machine learning model, and reducing that time is challenging.</td></tr>
<tr><td><b>Rationale</b></td><td>In meta-learning models, the loss is learned (along with the model parameters) at runtime in the production system. Therefore, the initial lack of accuracy of the loss representation determines substantial errors in the predictions, hence significant</td></tr>
</table>

<table>
<tr><td colspan="3"></td><td colspan="15">costs for the operator. It is thus key to minimize the training time and the economic penalty for the operator of training the whole model from a cold start situation.</td></tr>
<tr><td>K1</td><td></td><td>K2</td><td></td><td>K3</td><td></td><td>K4</td><td>X</td><td>K5</td><td></td><td>K6</td><td></td><td>K7</td><td></td><td>K8</td><td></td><td>K9</td><td>X</td></tr>
<tr><td colspan="3">Parents</td><td colspan="15">FR-CFORE-005</td></tr>
</table>

<table>
<tr><th colspan="18">FR-CFORE-007</th></tr>
<tr><td colspan="3">Description</td><td colspan="15">Loss meta-learning shall support losses that combine multiple predictions</td></tr>
<tr><td colspan="3">Version</td><td colspan="15">001M17</td></tr>
<tr><td colspan="3">Owner</td><td colspan="15">IMDEA</td></tr>
<tr><td colspan="3">Priority</td><td colspan="15">Medium</td></tr>
<tr><td colspan="3">Risk</td><td colspan="15">3</td></tr>
<tr><td colspan="3">Risk Description</td><td colspan="15">Having multiple forecasting models depend on the same loss implies correlations in the predictions, which are typically very complex to learn, making the problem more involved than single-input loss meta-learning.</td></tr>
<tr><td colspan="3">Rationale</td><td colspan="15">In many network management tasks, the performance does not depend on a single prediction but on a composition of multiple forecasting tasks. This is the case, for instance, in admission control problems over many predicted traffic flows, or in network slice brokering. Learning the correct loss function in those situations implies capturing the correlations among the different predictions and the performance metric, which calls for even more complex meta-learning tools.</td></tr>
<tr><td>K1</td><td></td><td>K2</td><td></td><td>K3</td><td></td><td>K4</td><td>X</td><td>K5</td><td></td><td>K6</td><td></td><td>K7</td><td></td><td>K8</td><td></td><td>K9</td><td>X</td></tr>
<tr><td colspan="3">Parents</td><td colspan="15">FR-CFORE-005</td></tr>
</table>

A.1.8 Automated anomaly response



FR-AARES-000
DAEMON shall automatically detect, analyze, and act against anomalous behaviors.

FR-AARES-001
DAEMON capacity forecast models shall operate at different timescales, depending on the input from the system DAEMON is monitoring.

FR-AARES-002
DAEMON anomaly detection models have specific data requirements, including a sizable amount of historical data to establish normal behavior and ground truth occurrences of anomalies to develop a feasible solution.

NFR-AARES-000
NI solutions anomaly detection and response should have a high detection performance (specifically, DAEMON will target a 0.9 precision-recall AUC with at least 85% scoring in both precision and recall.).

FR-AARES-003
DAEMON shall take into consideration the cost of system monitoring, developing and deploying the anomaly detection models in order to produce a feasible anomaly detection solution.

FR-AARES-004
DAEMON anomaly detection models need to account for a possible temporal distribution shift in unseen data.

Risk Level: 1 2 3 4 5

Requirement Type: Functional Non-Functional

New + Updated ⟳

<table>
<tr><th colspan="18">FR-AARES-000</th></tr>
<tr><td colspan="4"><b>Description</b></td><td colspan="14">DAEMON shall automatically detect, analyze, and act against anomalous behaviors.</td></tr>
<tr><td colspan="4"><b>Version</b></td><td colspan="14">002M5</td></tr>
<tr><td colspan="4"><b>Owner</b></td><td colspan="14">TID</td></tr>
<tr><td colspan="4"><b>Priority</b></td><td colspan="14">High</td></tr>
<tr><td colspan="4"><b>Risk</b></td><td colspan="14">5</td></tr>
<tr><td colspan="4"><b>Risk Description</b></td><td colspan="14">Anomaly detection tasks might not correctly capture new previously unseen anomalies.</td></tr>
<tr><td colspan="4"><b>Rationale</b></td><td colspan="14">Most communication platforms use a reactive approach to deal with communication issues (i.e., operation teams react when incidents are severe only, and the service is often compromised already). DAEMON requires a proactive approach to anomaly detection that can detect both malicious and benign anomalies in the different systems it integrates.</td></tr>
<tr><td><b>K1</b></td><td></td><td><b>K2</b></td><td></td><td><b>K3</b></td><td></td><td><b>K4</b></td><td></td><td><b>K5</b></td><td></td><td><b>K6</b></td><td></td><td><b>K7</b></td><td><b>X</b></td><td><b>K8</b></td><td></td><td><b>K9</b></td><td></td></tr>
<tr><td colspan="4"><b>Parents</b></td><td colspan="14">None</td></tr>
</table>

<table>
<tr><th colspan="18">FR-AARES-001</th></tr>
<tr><td colspan="4"><b>Description</b></td><td colspan="14">DAEMON anomaly detection shall operate at different timescales, depending on the input from the system DAEMON is monitoring.</td></tr>
<tr><td colspan="4"><b>Version</b></td><td colspan="14">002M5</td></tr>
<tr><td colspan="4"><b>Owner</b></td><td colspan="14">TID</td></tr>
<tr><td colspan="4"><b>Priority</b></td><td colspan="14">Medium</td></tr>
<tr><td colspan="4"><b>Risk</b></td><td colspan="14">2</td></tr>
<tr><td colspan="4"><b>Risk Description</b></td><td colspan="14">Each anomaly detection task should take into consideration the requirements in terms of the timescale it needs to generate anomaly warnings. For finer granularities, the performance of the models might implicitly decrease, as the time available for the model to produce results also decreases. We will work to find the best tradeoff between the most fitting timescale and the performance requirements for the DAEMON anomaly detection tasks.</td></tr>
<tr><td colspan="4"><b>Rationale</b></td><td colspan="14">Anomalies can become easier to spot depending on the timescale that fits to the particular system with which DAEMON interacts.</td></tr>
<tr><td><b>K1</b></td><td></td><td><b>K2</b></td><td></td><td><b>K3</b></td><td></td><td><b>K4</b></td><td></td><td><b>K5</b></td><td></td><td><b>K6</b></td><td></td><td><b>K7</b></td><td><b>X</b></td><td><b>K8</b></td><td></td><td><b>K9</b></td><td></td></tr>
<tr><td colspan="4"><b>Parents</b></td><td colspan="14">FR-AARES-000</td></tr>
</table>

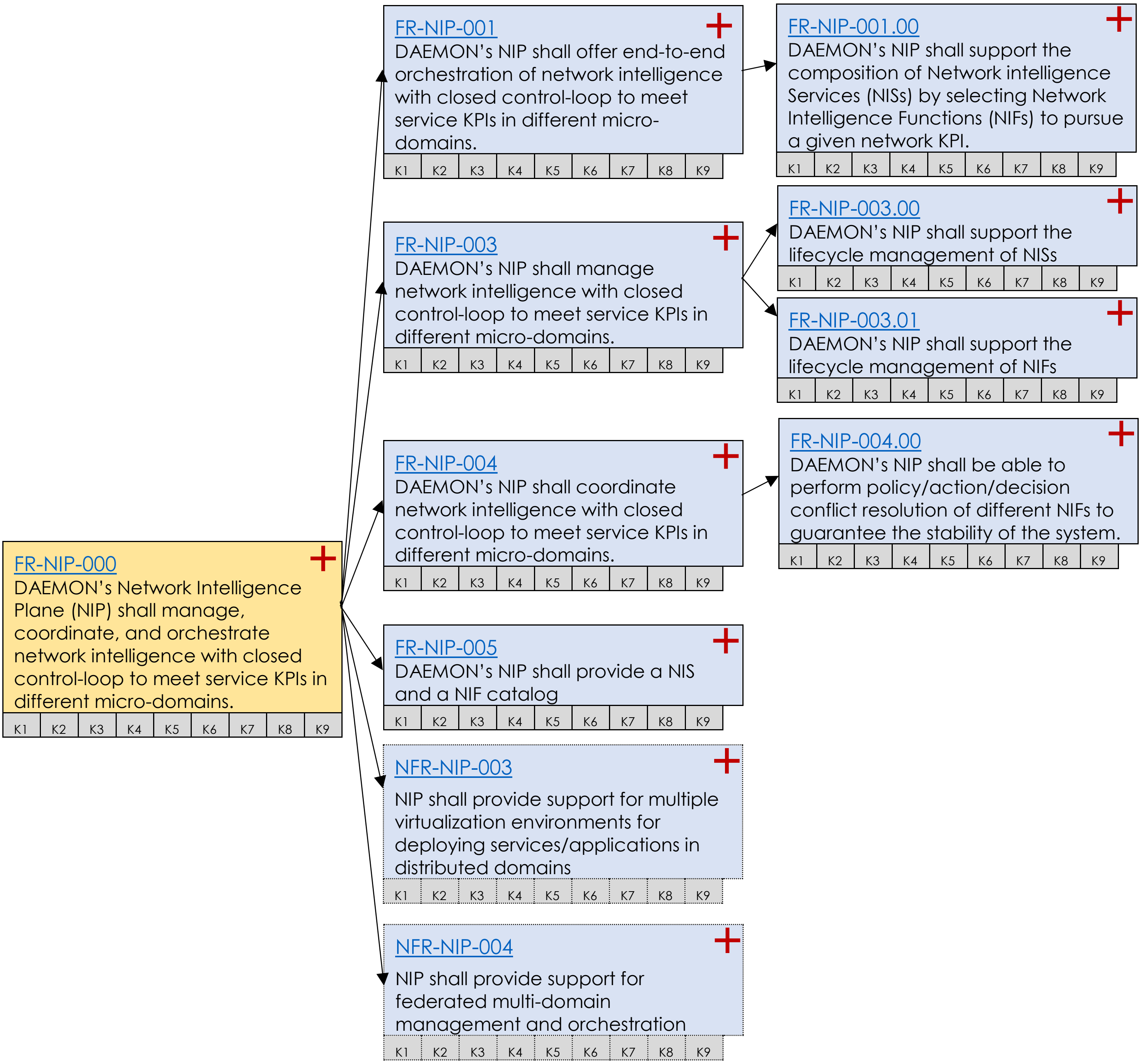
<table>
<tr><th colspan="18">FR-AARES-002</th></tr>
<tr><td colspan="4"><b>Description</b></td><td colspan="14">DAEMON anomaly detection models have specific data requirements, including a sizable amount of historical data to establish normal behavior and ground truth occurrences of anomalies to develop a feasible solution.</td></tr>
<tr><td colspan="4"><b>Version</b></td><td colspan="14">002M5</td></tr>
<tr><td colspan="4"><b>Owner</b></td><td colspan="14">TID</td></tr>
<tr><td colspan="4"><b>Priority</b></td><td colspan="14">High</td></tr>
<tr><td colspan="4"><b>Risk</b></td><td colspan="14">4</td></tr>
<tr><td colspan="4"><b>Risk Description</b></td><td colspan="14">The lack of high-quality historical data to establish the baseline behavior of the system (i.e., anomaly-free state for training) poses a high risk to developing anomaly detection approaches for DAEMON. Similarly, the lack of ground truth anomalies that have been detected in the system will make the validation of any anomaly detection approach challenging. Finally, the lack of expert knowledge brings an extra risk when building data features to train the anomaly detection tools for DAEMON.</td></tr>
<tr><td colspan="4"><b>Rationale</b></td><td colspan="14">The data quality is of paramount importance for the anomaly detection approaches we aim to integrate in DAEMON. Specifically, we aim to build on high quality ground truth for establishing the normal baseline for the system DAEMON monitors. Similarly, in order to validate the performance of our DAEMON anomaly detection solutions, we require a diverse set of ground-truth anomalies that operators previously captured in the systems DAEMON integrates. Furthermore, in order to craft data features that respond to the purpose of each system, we require expert knowledge and the operators' support in this process.</td></tr>
<tr><td><b>K1</b></td><td></td><td><b>K2</b></td><td></td><td><b>K3</b></td><td></td><td><b>K4</b></td><td></td><td><b>K5</b></td><td></td><td><b>K6</b></td><td></td><td><b>K7</b></td><td><b>X</b></td><td><b>K8</b></td><td></td><td><b>K9</b></td><td></td></tr>
</table>

Parents	FR-AARES-000

<table>
<tr><th colspan="18">FR-AARES-003</th></tr>
<tr><td colspan="4"><b>Description</b></td><td colspan="14">DAEMON shall take into consideration the cost of system monitoring, developing and deploying the anomaly detection models in order to produce a feasible anomaly detection solution.</td></tr>
<tr><td colspan="4"><b>Version</b></td><td colspan="14">003M5</td></tr>
<tr><td colspan="4"><b>Owner</b></td><td colspan="14">TID</td></tr>
<tr><td colspan="4"><b>Priority</b></td><td colspan="14">Low</td></tr>
<tr><td colspan="4"><b>Risk</b></td><td colspan="14">1</td></tr>
<tr><td colspan="4"><b>Risk Description</b></td><td colspan="14">The high cost of training and running DAEMON anomaly detection tools might suppose a high expenditure for the operators of the system in question.</td></tr>
<tr><td colspan="4"><b>Rationale</b></td><td colspan="14">DAEMON anomaly detection tools must run in real-world production systems, where we must also consider the actual monetary cost of running a state-of-the-art system for ML/DL tasks. We will work to produce solutions that adapt to different tiers of existing resources.</td></tr>
<tr><td><b>K1</b></td><td></td><td><b>K2</b></td><td></td><td><b>K3</b></td><td></td><td><b>K4</b></td><td></td><td><b>K5</b></td><td></td><td><b>K6</b></td><td></td><td><b>K7</b></td><td><b>X</b></td><td><b>K8</b></td><td></td><td><b>K9</b></td><td></td></tr>
<tr><td colspan="4"><b>Parents</b></td><td colspan="14">FR-AARES-000</td></tr>
</table>

<table>
<tr><th colspan="18">FR-AARES-004</th></tr>
<tr><td colspan="4"><b>Description</b></td><td colspan="14">DAEMON anomaly detection models need to account for a possible temporal distribution shift in unseen data.</td></tr>
<tr><td colspan="4"><b>Version</b></td><td colspan="14">004M17</td></tr>
<tr><td colspan="4"><b>Owner</b></td><td colspan="14">TID</td></tr>
<tr><td colspan="4"><b>Priority</b></td><td colspan="14">High</td></tr>
<tr><td colspan="4"><b>Risk</b></td><td colspan="14">4</td></tr>
<tr><td colspan="4"><b>Risk Description</b></td><td colspan="14">The data captured in a network environment is indeed a temporal series that can have seasonal patterns or data can even be non-stationary. This issue poses a high risk for any anomaly detection approach that learns some normal behavior or statistics from the data. A possible distribution shift where features extracted from the captured data diverge too much over time will make the detection of anomalies in unseen data challenging.</td></tr>
<tr><td colspan="4"><b>Rationale</b></td><td colspan="14">The data used for anomaly detection should cover historical data for an analysis of seasonal shifts and temporal distribution shifts. The features extracted from the features should be tested against stationarity and temporal covariance shift. Feature selection should select features that show high stability over time to avoid this issue. Nevertheless, anomaly detection models can age over time and new data should be captured regularly to update such models.</td></tr>
<tr><td><b>K1</b></td><td></td><td><b>K2</b></td><td></td><td><b>K3</b></td><td></td><td><b>K4</b></td><td></td><td><b>K5</b></td><td></td><td><b>K6</b></td><td></td><td><b>K7</b></td><td><b>X</b></td><td><b>K8</b></td><td></td><td><b>K9</b></td><td></td></tr>
<tr><td colspan="4"><b>Parents</b></td><td colspan="14">FR-AARES-000</td></tr>
</table>

A.2 Functional requirements: Network Intelligence Plane



FR-NIP-000
DAEMON's Network Intelligence Plane (NIP) shall manage, coordinate, and orchestrate network intelligence with closed control-loop to meet service KPIs in different micro-domains.

FR-NIP-001
DAEMON's NIP shall offer end-to-end orchestration of network intelligence with closed control-loop to meet service KPIs in different micro-domains.

FR-NIP-001.00
DAEMON's NIP shall support the composition of Network intelligence Services (NISs) by selecting Network Intelligence Functions (NIFs) to pursue a given network KPI.

FR-NIP-003
DAEMON's NIP shall manage network intelligence with closed control-loop to meet service KPIs in different micro-domains.

FR-NIP-003.00
DAEMON's NIP shall support the lifecycle management of NISs

FR-NIP-003.01
DAEMON's NIP shall support the lifecycle management of NIFs

FR-NIP-004
DAEMON's NIP shall coordinate network intelligence with closed control-loop to meet service KPIs in different micro-domains.

FR-NIP-004.00
DAEMON's NIP shall be able to perform policy/action/decision conflict resolution of different NIFs to guarantee the stability of the system.

FR-NIP-005
DAEMON's NIP shall provide a NIS and a NIF catalog

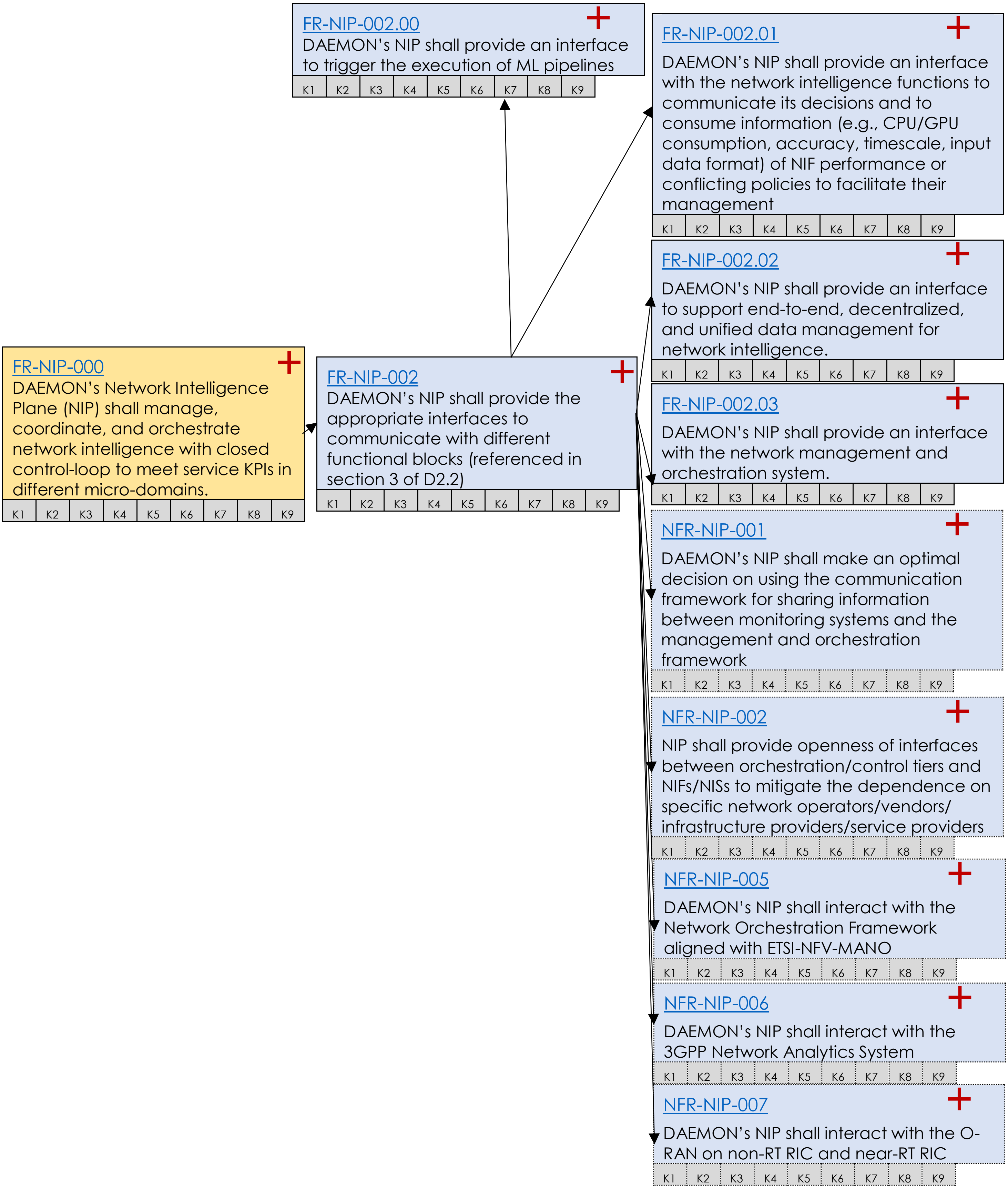
NFR-NIP-003
NIP shall provide support for multiple virtualization environments for deploying services/applications in distributed domains

NFR-NIP-004
NIP shall provide support for federated multi-domain management and orchestration

Risk Level: 1 2 3 4 5

Requirement Type: Functional Non-Functional

New + Updated 🔄



FR-NIP-000 +
DAEMON's Network Intelligence Plane (NIP) shall manage, coordinate, and orchestrate network intelligence with closed control-loop to meet service KPIs in different micro-domains.
K1 K2 K3 K4 K5 K6 K7 K8 K9

FR-NIP-002 +
DAEMON's NIP shall provide the appropriate interfaces to communicate with different functional blocks (referenced in section 3 of D2.2)
K1 K2 K3 K4 K5 K6 K7 K8 K9

FR-NIP-002.00 +
DAEMON's NIP shall provide an interface to trigger the execution of ML pipelines
K1 K2 K3 K4 K5 K6 K7 K8 K9

FR-NIP-002.01 +
DAEMON's NIP shall provide an interface with the network intelligence functions to communicate its decisions and to consume information (e.g., CPU/GPU consumption, accuracy, timescale, input data format) of NIF performance or conflicting policies to facilitate their management
K1 K2 K3 K4 K5 K6 K7 K8 K9

FR-NIP-002.02 +
DAEMON's NIP shall provide an interface to support end-to-end, decentralized, and unified data management for network intelligence.
K1 K2 K3 K4 K5 K6 K7 K8 K9

FR-NIP-002.03 +
DAEMON's NIP shall provide an interface with the network management and orchestration system.
K1 K2 K3 K4 K5 K6 K7 K8 K9

NFR-NIP-001 +
DAEMON's NIP shall make an optimal decision on using the communication framework for sharing information between monitoring systems and the management and orchestration framework
K1 K2 K3 K4 K5 K6 K7 K8 K9

NFR-NIP-002 +
NIP shall provide openness of interfaces between orchestration/control tiers and NIFs/NISs to mitigate the dependence on specific network operators/vendors/ infrastructure providers/service providers
K1 K2 K3 K4 K5 K6 K7 K8 K9

NFR-NIP-005 +
DAEMON's NIP shall interact with the Network Orchestration Framework aligned with ETSI-NFV-MANO
K1 K2 K3 K4 K5 K6 K7 K8 K9

NFR-NIP-006 +
DAEMON's NIP shall interact with the 3GPP Network Analytics System
K1 K2 K3 K4 K5 K6 K7 K8 K9

NFR-NIP-007 +
DAEMON's NIP shall interact with the O-RAN on non-RT RIC and near-RT RIC
K1 K2 K3 K4 K5 K6 K7 K8 K9

Risk Level: 1 2 3 4 5

Requirement Type: Functional Non-Functional

New + Updated ⟳

<table>
<tr><th colspan="18">FR-NIP-000</th></tr>
<tr><td colspan="3"><b>Description</b></td><td colspan="15">DAEMON's Network Intelligence Plane (NIP) shall manage, coordinate, and orchestrate network intelligence with closed control-loop to meet service KPIs in different micro-domains.</td></tr>
<tr><td colspan="3"><b>Version</b></td><td colspan="15">002M18</td></tr>
<tr><td colspan="3"><b>Owner</b></td><td colspan="15">IMEC</td></tr>
<tr><td colspan="3"><b>Priority</b></td><td colspan="15">Low</td></tr>
<tr><td colspan="3"><b>Risk</b></td><td colspan="15">1</td></tr>
<tr><td colspan="3"><b>Risk Description</b></td><td colspan="15">Given the wide range of NI solutions we need a common framework to map the most common features of NI algorithms, integrate them into a defined architecture, and design the necessary interfaces that algorithms use to interact with their environment.</td></tr>
<tr><td colspan="3"><b>Rationale</b></td><td colspan="15">Network Intelligence (NI) is proposed to replace or assist network operators in their diverse set of network management tasks. However, current management frameworks (e.g., O-RAN, MANO) are not flexible enough or do not support the integration of NI instances. The DAEMON architectural framework enables the penetration of intelligence into both the user and control planes, thereby creating a hierarchical NI architecture that consists of distributed NI instances for network management, which altogether collaborate to improve their individual learning and decision-making processes.</td></tr>
<tr><td><b>K1</b></td><td><b>X</b></td><td><b>K2</b></td><td><b>X</b></td><td><b>K3</b></td><td><b>X</b></td><td><b>K4</b></td><td><b>X</b></td><td><b>K5</b></td><td><b>X</b></td><td><b>K6</b></td><td><b>X</b></td><td><b>K7</b></td><td><b>X</b></td><td><b>K8</b></td><td><b>X</b></td><td><b>K9</b></td><td>X</td></tr>
<tr><td colspan="3"><b>Parents</b></td><td colspan="15">None</td></tr>
</table>

<table>
<tr><th colspan="18">FR-NIP-001</th></tr>
<tr><td colspan="3"><b>Description</b></td><td colspan="15">DAEMON's NIP shall offer end-to-end orchestration of network intelligence with closed control-loop to meet service KPIs in different micro-domains.</td></tr>
<tr><td colspan="3"><b>Version</b></td><td colspan="15">001M18</td></tr>
<tr><td colspan="3"><b>Owner</b></td><td colspan="15">IMEC</td></tr>
<tr><td colspan="3"><b>Priority</b></td><td colspan="15">Low</td></tr>
<tr><td colspan="3"><b>Risk</b></td><td colspan="15">1</td></tr>
<tr><td colspan="3"><b>Risk Description</b></td><td colspan="15">One of the main challenges in the orchestration of NI is to translate network requirements or KPIs to meet business needs.</td></tr>
<tr><td colspan="3"><b>Rationale</b></td><td colspan="15">The NI Plane integrates the functions related to network intelligence. In several cases, these functions can be orchestrated to create an end-to-end Network Intelligence Services. The creation of such services can be done in an automatic way, similarly as in network orchestrators.</td></tr>
<tr><td><b>K1</b></td><td><b>X</b></td><td><b>K2</b></td><td><b>X</b></td><td><b>K3</b></td><td><b>X</b></td><td><b>K4</b></td><td><b>X</b></td><td><b>K5</b></td><td><b>X</b></td><td><b>K6</b></td><td><b>X</b></td><td><b>K7</b></td><td><b>X</b></td><td><b>K8</b></td><td><b>X</b></td><td><b>K9</b></td><td><b>X</b></td></tr>
<tr><td colspan="3"><b>Parents</b></td><td colspan="15">FR-NIP-000</td></tr>
</table>

<table>
<tr><th colspan="18">FR-NIP-001.00</th></tr>
<tr><td colspan="3"><b>Description</b></td><td colspan="15">DAEMON's NIP shall support the composition of Network intelligence Services (NISs) by selecting Network Intelligence Functions (NIFs) to pursue a given network KPI.</td></tr>
<tr><td colspan="3"><b>Version</b></td><td colspan="15">002M18</td></tr>
<tr><td colspan="3"><b>Owner</b></td><td colspan="15">IMEC</td></tr>
<tr><td colspan="3"><b>Priority</b></td><td colspan="15">Low</td></tr>
<tr><td colspan="3"><b>Risk</b></td><td colspan="15">1</td></tr>
<tr><td colspan="3"><b>Risk Description</b></td><td colspan="15">It is possible that the available NIFs do not address the system constraints. In this case, NIFs that try to fulfil system constraints as close as possible will be selected.</td></tr>
<tr><td colspan="3"><b>Rationale</b></td><td colspan="15">Network Intelligence Functions (NIFs) are functional blocks that implement a decision-making functionality to be deployed in a controller. Similar to the information model specified for network management by e.g., 3GPP, NIFs can be arranged to compose a Network Intelligence Service (NIS).
Depending on the available resources and the business goals or SLAs, NIP will select the best NIF model that suits the system constraints. For example, in some cases it might be feasible to sacrifice accuracy at the expenses of a lower computational complexity.</td></tr>
<tr><td><b>K1</b></td><td><b>X</b></td><td><b>K2</b></td><td><b>X</b></td><td><b>K3</b></td><td><b>X</b></td><td><b>K4</b></td><td><b>X</b></td><td><b>K5</b></td><td><b>X</b></td><td><b>K6</b></td><td><b>X</b></td><td><b>K7</b></td><td><b>X</b></td><td><b>K8</b></td><td><b>X</b></td><td><b>K9</b></td><td>X</td></tr>
</table>

Parents	FR-NIP-001

<table>
<tr><th colspan="18">FR-NIP-002</th></tr>
<tr><td colspan="3"><b>Description</b></td><td colspan="15">DAEMON's NIP shall provide the appropriate interfaces to communicate with different functional blocks (referenced in section 3 of D2.2)</td></tr>
<tr><td colspan="3"><b>Version</b></td><td colspan="15">001M18</td></tr>
<tr><td colspan="3"><b>Owner</b></td><td colspan="15">IMEC</td></tr>
<tr><td colspan="3"><b>Priority</b></td><td colspan="15">Low</td></tr>
<tr><td colspan="3"><b>Risk</b></td><td colspan="15">1</td></tr>
<tr><td colspan="3"><b>Risk Description</b></td><td colspan="15">There are common communication patterns (e.g., pub/sub) that could be replicated here. However, we must select the most suitable communication system, considering that the decisions taken by the NIP might impact network behavior.</td></tr>
<tr><td colspan="3"><b>Rationale</b></td><td colspan="15">Once a NIS is created/composed, training such models (NIFs) will be performed via the creation and deployment of MLOps frameworks. Once the models are trained, they will be registered in the NIF/NIS catalogue and will be ready to be deployed in a test/production environment. Currently there are several commercial frameworks that already do that for ML applications. The idea is not to reinvent the wheel, but to adapt such frameworks to the network domain.
Once the NISs are deployed, the appropriate interfaces to manage the lifecycle management of their NIFs shall be used. Moreover, NIFs should be able to infer the network state/context as input. For that reason, the NIP should enable an interface with the corresponding management framework.</td></tr>
<tr><td><b>K1</b></td><td>X</td><td><b>K2</b></td><td>X</td><td><b>K3</b></td><td>X</td><td><b>K4</b></td><td>X</td><td><b>K5</b></td><td>X</td><td><b>K6</b></td><td>X</td><td><b>K7</b></td><td>X</td><td><b>K8</b></td><td>X</td><td><b>K9</b></td><td>X</td></tr>
<tr><td colspan="3"><b>Parents</b></td><td colspan="15">FR-NIP-000</td></tr>
</table>

<table>
<tr><th colspan="18">FR-NIP-002.00</th></tr>
<tr><td colspan="3"><b>Description</b></td><td colspan="15">DAEMON's NIP shall provide an interface to trigger the execution of ML pipelines</td></tr>
<tr><td colspan="3"><b>Version</b></td><td colspan="15">002M18</td></tr>
<tr><td colspan="3"><b>Owner</b></td><td colspan="15">IMEC</td></tr>
<tr><td colspan="3"><b>Priority</b></td><td colspan="15">Low</td></tr>
<tr><td colspan="3"><b>Risk</b></td><td colspan="15">1</td></tr>
<tr><td colspan="3"><b>Risk Description</b></td><td colspan="15">From the architectural point of view, we need to identify or to create the interaction points between the NIP and the MLOps framework.</td></tr>
<tr><td colspan="3"><b>Rationale</b></td><td colspan="15">MLOps is a methodology that combines Machine Learning (ML) with software development operations (DevOps) and data engineering with the goal of building, training, deploying, and maintaining ML systems in productions with high reliability and efficiency guarantees. DAEMON architecture explicitly indicates that building ML models' functionality (i.e., the ML pipelines) is delegated to an external platform, and MLOps frameworks are the de-facto platform to do this task. Currently there are several commercial frameworks that already do that for ML applications. The idea is not to reinvent the wheel, but to adapt such frameworks to the network domain.</td></tr>
<tr><td><b>K1</b></td><td>X</td><td><b>K2</b></td><td>X</td><td><b>K3</b></td><td>X</td><td><b>K4</b></td><td>X</td><td><b>K5</b></td><td>X</td><td><b>K6</b></td><td>X</td><td><b>K7</b></td><td>X</td><td><b>K8</b></td><td>X</td><td><b>K9</b></td><td>X</td></tr>
<tr><td colspan="3"><b>Parents</b></td><td colspan="15">FR-NIP-002</td></tr>
</table>

<table>
<tr><th colspan="2">FR-NIP-002.01</th></tr>
<tr><td><b>Description</b></td><td>DAEMON's NIP shall provide an interface with the network intelligence functions to communicate its decisions and to consume information (e.g., CPU/GPU consumption, accuracy, timescale, input data format) of NIF performance or conflicting policies to facilitate their management</td></tr>
<tr><td><b>Version</b></td><td>001M18</td></tr>
</table>

<table>
<tr><td><b>Owner</b></td><td colspan="17">IMEC</td></tr>
<tr><td><b>Priority</b></td><td colspan="17">Low</td></tr>
<tr><td><b>Risk</b></td><td colspan="17">1</td></tr>
<tr><td><b>Risk Description</b></td><td colspan="17">Functionalities and NIFs can be very diverse. To ease the implementation of a NIP, information should be standardized (e.g., format) which can be cumbersome given the wide application domains of DAEMON functionalities.</td></tr>
<tr><td><b>Rationale</b></td><td colspan="17">NIP decisions (replacement, retraining, execution, and termination) should be made based on the information coming from the Network Intelligence Functions (NIFs). This information should be enough to take a good decision. Furthermore, this decision must be communicated using the same channel, guaranteeing the stability of the system.</td></tr>
<tr><td><b>K1</b></td><td>X</td><td><b>K2</b></td><td>X</td><td><b>K3</b></td><td>X</td><td><b>K4</b></td><td>X</td><td><b>K5</b></td><td>X</td><td><b>K6</b></td><td>X</td><td><b>K7</b></td><td>X</td><td><b>K8</b></td><td>X</td><td><b>K9</b></td><td>X</td></tr>
<tr><td><b>Parents</b></td><td colspan="17">FR-NIP-002</td></tr>
</table>

<table>
<tr><th colspan="18">FR-NIP-002.02</th></tr>
<tr><td><b>Description</b></td><td colspan="17">DAEMON’s NIP shall provide an interface to support end-to-end, decentralized, and unified data management for network intelligence.</td></tr>
<tr><td><b>Version</b></td><td colspan="17">001M8</td></tr>
<tr><td><b>Owner</b></td><td colspan="17">ZSC</td></tr>
<tr><td><b>Priority</b></td><td colspan="17">High</td></tr>
<tr><td><b>Risk</b></td><td colspan="17">1</td></tr>
<tr><td><b>Risk Description</b></td><td colspan="17">Besides managing the lifecycle of different NISs, the burden of managing data can be too big as it involves a multiplicity of data sources and data types.</td></tr>
<tr><td><b>Rationale</b></td><td colspan="17">The NI Multi-timescale Closed-loop AI Framework should provide end-to-end decentralized and unified data management to ease the development, operation, and management of any NI model. Such data is gathered with the purpose of training NI algorithms. The main characteristics are previously defined in FR-MTERM-001 in D2.1 [1].</td></tr>
<tr><td><b>K1</b></td><td><b>X</b></td><td><b>K2</b></td><td><b>X</b></td><td><b>K3</b></td><td><b>X</b></td><td><b>K4</b></td><td><b>X</b></td><td><b>K5</b></td><td><b>X</b></td><td><b>K6</b></td><td><b>X</b></td><td><b>K7</b></td><td><b>X</b></td><td><b>K8</b></td><td><b>X</b></td><td><b>K9</b></td><td>X</td></tr>
<tr><td><b>Parents</b></td><td colspan="17">FR-NIP-002</td></tr>
</table>

<table>
<tr><th colspan="18">FR-NIP-002.03</th></tr>
<tr><td><b>Description</b></td><td colspan="17">DAEMON’s NIP shall provide an interface with the network management and orchestration system.</td></tr>
<tr><td><b>Version</b></td><td colspan="17">001M18</td></tr>
<tr><td><b>Owner</b></td><td colspan="17">IMEC</td></tr>
<tr><td><b>Priority</b></td><td colspan="17">Low</td></tr>
<tr><td><b>Risk</b></td><td colspan="17">1</td></tr>
<tr><td><b>Risk Description</b></td><td colspan="17">In some cases, the interaction point with network management and orchestration systems is evident (e.g., O-RAN architecture) but in other domains it can be hard to define (e.g., NFV MANO) since they are not developed to natively support NI.</td></tr>
<tr><td><b>Rationale</b></td><td colspan="17">The NIP manages the connection towards the network management and orchestration to gather important information such as the expected network KPIs for the managed slice and service, as well as the information of the underlying network infrastructure.</td></tr>
<tr><td><b>K1</b></td><td>X</td><td><b>K2</b></td><td>X</td><td><b>K3</b></td><td>X</td><td><b>K4</b></td><td>X</td><td><b>K5</b></td><td>X</td><td><b>K6</b></td><td>X</td><td><b>K7</b></td><td>X</td><td><b>K8</b></td><td>X</td><td><b>K9</b></td><td>X</td></tr>
<tr><td><b>Parents</b></td><td colspan="17">FR-NIP-002</td></tr>
</table>

<table>
<tr><th colspan="2">FR-NIP-003</th></tr>
<tr><td><b>Description</b></td><td>DAEMON’s NIP shall manage network intelligence with closed control-loop to meet service KPIs in different micro-domains.</td></tr>
</table>

<table>
<tr><td colspan="3"><b>Version</b></td><td colspan="15">001M18</td></tr>
<tr><td colspan="3"><b>Owner</b></td><td colspan="15">IMEC</td></tr>
<tr><td colspan="3"><b>Priority</b></td><td colspan="15">Low</td></tr>
<tr><td colspan="3"><b>Risk</b></td><td colspan="15">1</td></tr>
<tr><td colspan="3"><b>Risk Description</b></td><td colspan="15">Overall system stability should be achieved. However, it can be that some NISs span several domains which require extra coordination.</td></tr>
<tr><td colspan="3"><b>Rationale</b></td><td colspan="15">Once a NIS is released for production, the NIP shall support its lifecycle management. By lifecycle management we refer to onboarding, instantiation, termination, scaling, and state retrieval. The same should happen with different NIFs that compose the NIS</td></tr>
<tr><td><b>K1</b></td><td>X</td><td><b>K2</b></td><td>X</td><td><b>K3</b></td><td>X</td><td><b>K4</b></td><td>X</td><td><b>K5</b></td><td>X</td><td><b>K6</b></td><td>X</td><td><b>K7</b></td><td>X</td><td><b>K8</b></td><td>X</td><td><b>K9</b></td><td>X</td></tr>
<tr><td colspan="3"><b>Parents</b></td><td colspan="15">FR-NIP-000</td></tr>
</table>

<table>
<tr><th colspan="18">FR-NIP-003.00</th></tr>
<tr><td colspan="3"><b>Description</b></td><td colspan="15">DAEMON's NIP shall support the lifecycle management of NISs</td></tr>
<tr><td colspan="3"><b>Version</b></td><td colspan="15">002M18</td></tr>
<tr><td colspan="3"><b>Owner</b></td><td colspan="15">IMEC</td></tr>
<tr><td colspan="3"><b>Priority</b></td><td colspan="15">Low</td></tr>
<tr><td colspan="3"><b>Risk</b></td><td colspan="15">1</td></tr>
<tr><td colspan="3"><b>Risk Description</b></td><td colspan="15">The metrics that measure the impact of a NIF in the overall performance of a NIS could be difficult to define.</td></tr>
<tr><td colspan="3"><b>Rationale</b></td><td colspan="15">NISs are composed of one or more NIFs with a specific target, usually related with a specific set of targeted KPIs. They possibly span several network domains. Therefore, it is required to not only monitor the performance of a given NIF from the NIS, but also the impact of this NIF in the performance of the NIS.</td></tr>
<tr><td><b>K1</b></td><td>X</td><td><b>K2</b></td><td>X</td><td><b>K3</b></td><td>X</td><td><b>K4</b></td><td>X</td><td><b>K5</b></td><td>X</td><td><b>K6</b></td><td>X</td><td><b>K7</b></td><td>X</td><td><b>K8</b></td><td>X</td><td><b>K9</b></td><td>X</td></tr>
<tr><td colspan="3"><b>Parents</b></td><td colspan="15">FR-NIP-003</td></tr>
</table>

<table>
<tr><th colspan="18">FR-NIP-003.01</th></tr>
<tr><td colspan="3"><b>Description</b></td><td colspan="15">DAEMON's NIP shall support the lifecycle management of NIFs</td></tr>
<tr><td colspan="3"><b>Version</b></td><td colspan="15">001M18</td></tr>
<tr><td colspan="3"><b>Owner</b></td><td colspan="15">IMEC</td></tr>
<tr><td colspan="3"><b>Priority</b></td><td colspan="15">Low</td></tr>
<tr><td colspan="3"><b>Risk</b></td><td colspan="15">1</td></tr>
<tr><td colspan="3"><b>Risk Description</b></td><td colspan="15">NIFs themselves could be of different kinds: They could be learning models, based on, e.g., Deep Neural Networks or Engineered Models, or they could be built upon specific optimization algorithms such as the ones based on control theory or Mixed-Integer Linear Programming (MILP). Thus, it's necessary to define common strategies to proper manage both types of NIFs.</td></tr>
<tr><td colspan="3"><b>Rationale</b></td><td colspan="15">According to the DAEMON architecture, the NIF manager is responsible for the lifecycle management of NIFs and monitoring the health of the intelligence functions. This includes typical diagnostic information, if the NIF is being used in inference or it is an online learning solution, or other metrics such as the loss and the training loops if the NIF is currently being trained. Moreover, the NIP needs to provide feedback on the NIFs performance so higher-level decisions can be made (e.g., that the model can be updated or replaced).</td></tr>
<tr><td><b>K1</b></td><td>X</td><td><b>K2</b></td><td>X</td><td><b>K3</b></td><td>X</td><td><b>K4</b></td><td>X</td><td><b>K5</b></td><td>X</td><td><b>K6</b></td><td>X</td><td><b>K7</b></td><td>X</td><td><b>K8</b></td><td>X</td><td><b>K9</b></td><td>X</td></tr>
<tr><td colspan="3"><b>Parents</b></td><td colspan="15">FR-NIP-003</td></tr>
</table>

<table>
<tr><th colspan="18">FR-NIP-004</th></tr>
<tr><td colspan="4">Description</td><td colspan="14">DAEMON's NIP shall coordinate network intelligence with closed control-loop to meet service KPIs in different micro-domains.</td></tr>
<tr><td colspan="4">Version</td><td colspan="14">001M18</td></tr>
<tr><td colspan="4">Owner</td><td colspan="14">IMEC</td></tr>
<tr><td colspan="4">Priority</td><td colspan="14">Low</td></tr>
<tr><td colspan="4">Risk</td><td colspan="14">1</td></tr>
<tr><td colspan="4">Risk Description</td><td colspan="14">To determine which action/policy/decision has priority on optimizing a given function is not trivial and it depends on multiple factors that need to be evaluated. Therefore, the initial selection of policies/actions/decisions is highly coupled with the use case.</td></tr>
<tr><td colspan="4">Rationale</td><td colspan="14">Coordination of NI can include, but is not limited to:
• Sharing NIF-C among different NIFs (e.g., two NIFs that require the same input)
• Arbitration policies in case of two NIFs that share the same sink, that is, the configuration APIs.
• Guarantee system stability among conflicting policies/actions/decisions.</td></tr>
<tr><td>K1</td><td>X</td><td>K2</td><td>X</td><td>K3</td><td>X</td><td>K4</td><td>X</td><td>K5</td><td>X</td><td>K6</td><td>X</td><td>K7</td><td>X</td><td>K8</td><td>X</td><td>K9</td><td>X</td></tr>
<tr><td colspan="4">Parents</td><td colspan="14">FR-NIP-000</td></tr>
</table>

<table>
<tr><th colspan="18">FR-NIP-004.00</th></tr>
<tr><td colspan="4">Description</td><td colspan="14">DAEMON's NIP shall be able to perform policy/action/decision conflict resolution of different NIFs to guarantee the stability of the system.</td></tr>
<tr><td colspan="4">Version</td><td colspan="14">003M18</td></tr>
<tr><td colspan="4">Owner</td><td colspan="14">IMEC</td></tr>
<tr><td colspan="4">Priority</td><td colspan="14">Low</td></tr>
<tr><td colspan="4">Risk</td><td colspan="14">1</td></tr>
<tr><td colspan="4">Risk Description</td><td colspan="14">When a NIF/NIS performs an action that conflicts with the action of other NIF/NIS, it is required to solve the conflict in a coordinate manner. However, designing the conflict resolution mechanism may be a very hard problem as it will depend on the multiple factors tailored to specific use cases (e.g., centralized vs. decentralized vs. federate network domains, flat vs. hierarchical decision making, etc.).</td></tr>
<tr><td colspan="4">Rationale</td><td colspan="14">Optimizations will take place in different domains of the assisted system. Therefore, the decisions/policies/actions that are taken to optimize a certain objective function (e.g., business goal, SLAs) can be counterproductive to other policies/decisions/actions. Thus, a conflict resolution system is needed that can guarantee that the system is evolving towards a stable state.</td></tr>
<tr><td>K1</td><td>X</td><td>K2</td><td>X</td><td>K3</td><td>X</td><td>K4</td><td>X</td><td>K5</td><td>X</td><td>K6</td><td>X</td><td>K7</td><td>X</td><td>K8</td><td>X</td><td>K9</td><td>X</td></tr>
<tr><td colspan="4">Parents</td><td colspan="14">FR-NIP-004</td></tr>
</table>

FR-NIP-005	
Description	DAEMON's NIP shall provide a NIS and a NIF catalog
Version	001M18
Owner	IMEC
Priority	Low
Risk	1
Risk Description	There must be some commonalities between NIFs and NISs, so they could be advertised in a general framework.

<table>
<tr><td colspan="4"><b>Rationale</b></td><td colspan="14">The NIP has catalogs of already onboarded NIS and NIFs. In particular, NIFs may need to be (re)-trained to cope with changing or different conditions, or on a periodical basis.
When a NIS is composed of NIF empowered by ML models, training such models will be performed via the creation and deployment of ML pipelines. Once the models are trained, they will be registered in the NIF/NIS catalog and will be ready to be deployed in a test/production environment.</td></tr>
<tr><td><b>K1</b></td><td>X</td><td><b>K2</b></td><td>X</td><td><b>K3</b></td><td>X</td><td><b>K4</b></td><td>X</td><td><b>K5</b></td><td>X</td><td><b>K6</b></td><td>X</td><td><b>K7</b></td><td>X</td><td><b>K8</b></td><td>X</td><td><b>K9</b></td><td>X</td></tr>
<tr><td colspan="4"><b>Parents</b></td><td colspan="14">FR-NIP-000</td></tr>
</table>

A.3 Performance requirements

Next, we provide the list of performance requirements (i.e., non-functional requirements, or NFR). We specify both the static and the dynamic numerical requirements placed on the software or on human interaction with the software.

<table>
<tr><th colspan="18">NFR-RIS-000</th></tr>
<tr><td colspan="4"><b>Description</b></td><td colspan="14">RIS should aid to increase wireless capacity (bits/m^2) by 100%</td></tr>
<tr><td colspan="4"><b>Version</b></td><td colspan="14">001M1</td></tr>
<tr><td colspan="4"><b>Owner</b></td><td colspan="14">NEC</td></tr>
<tr><td colspan="4"><b>Priority</b></td><td colspan="14">High</td></tr>
<tr><td colspan="4"><b>Risk</b></td><td colspan="14">3</td></tr>
<tr><td colspan="4"><b>Risk Description</b></td><td colspan="14">There is a risk that the performance attained in realistic environments fall below 100%</td></tr>
<tr><td colspan="4"><b>Rationale</b></td><td colspan="14">This will allow surfaces to adapt in a timely manner, following the channel dynamics.</td></tr>
<tr><td><b>K1</b></td><td></td><td><b>K2</b></td><td></td><td><b>K3</b></td><td></td><td><b>K4</b></td><td></td><td><b>K5</b></td><td></td><td><b>K6</b></td><td>X</td><td><b>K7</b></td><td></td><td><b>K8</b></td><td></td><td><b>K9</b></td><td></td></tr>
<tr><td colspan="4"><b>Parents</b></td><td colspan="14">FR-RIS-000</td></tr>
</table>

<table>
<tr><th colspan="18">NFR-RIS-001</th></tr>
<tr><td colspan="4"><b>Description</b></td><td colspan="14">Re-configuring all the components in a RIS must be achieved within 100 ms.</td></tr>
<tr><td colspan="4"><b>Version</b></td><td colspan="14">003M17</td></tr>
<tr><td colspan="4"><b>Owner</b></td><td colspan="14">NEC</td></tr>
<tr><td colspan="4"><b>Priority</b></td><td colspan="14">High</td></tr>
<tr><td colspan="4"><b>Risk</b></td><td colspan="14">3</td></tr>
<tr><td colspan="4"><b>Risk Description</b></td><td colspan="14">There is a risk that the electronic equipment required can only be re-configured in more than 100ms. For instance, nowadays shortages in electronic components may force us to resort to less performing designs.</td></tr>
<tr><td colspan="4"><b>Rationale</b></td><td colspan="14">100 ms is the timescale of O-RAN near-real-time RAN Intelligent controller and a good trade-off between tracking fast wireless channel dynamics and high overhead.</td></tr>
<tr><td><b>K1</b></td><td></td><td><b>K2</b></td><td></td><td><b>K3</b></td><td></td><td><b>K4</b></td><td></td><td><b>K5</b></td><td></td><td><b>K6</b></td><td></td><td><b>K7</b></td><td></td><td><b>K8</b></td><td>X</td><td><b>K9</b></td><td></td></tr>
<tr><td colspan="4"><b>Parents</b></td><td colspan="14">FR-RIS-000</td></tr>
</table>

<table>
<tr><th colspan="18">NFR-RIS-002</th></tr>
<tr><td colspan="4"><b>Description</b></td><td colspan="14">The (non-RF) electronic equipment required to control a RIS must consume less than 100 mW.</td></tr>
<tr><td colspan="4"><b>Version</b></td><td colspan="14">001M17</td></tr>
<tr><td colspan="4"><b>Owner</b></td><td colspan="14">NEC</td></tr>
<tr><td colspan="4"><b>Priority</b></td><td colspan="14">High</td></tr>
<tr><td colspan="4"><b>Risk</b></td><td colspan="14">3</td></tr>
<tr><td colspan="4"><b>Risk Description</b></td><td colspan="14">There is a risk that the electronic equipment required to control a RIS consumes more than 100 mW. For instance, nowadays shortages in electronic components may force us to resort to more energy-consuming solutions.</td></tr>
<tr><td colspan="4"><b>Rationale</b></td><td colspan="14">Provide smart RF reflectors that are very efficient in terms of energy consumption hence reducing OPEX.</td></tr>
<tr><td><b>K1</b></td><td>X</td><td><b>K2</b></td><td></td><td><b>K3</b></td><td></td><td><b>K4</b></td><td>X</td><td><b>K5</b></td><td></td><td><b>K6</b></td><td></td><td><b>K7</b></td><td></td><td><b>K8</b></td><td></td><td><b>K9</b></td><td></td></tr>
<tr><td colspan="4"><b>Parents</b></td><td colspan="14">FR-RIS-000</td></tr>
</table>

<table>
<tr><th colspan="18">NFR-CAWRS-000</th></tr>
<tr><td colspan="2"><b>Description</b></td><td colspan="16">NI orchestration solutions for vRAN shall have reaction times below 10s</td></tr>
<tr><td colspan="2"><b>Version</b></td><td colspan="16">002M17</td></tr>
<tr><td colspan="2"><b>Owner</b></td><td colspan="16">UC3M</td></tr>
<tr><td colspan="2"><b>Priority</b></td><td colspan="16">High</td></tr>
<tr><td colspan="2"><b>Risk</b></td><td colspan="16">1</td></tr>
<tr><td colspan="2"><b>Risk Description</b></td><td colspan="16">There is a low risk that DAEMON will not integrate succeed to provide such timings for the NI-based network orchestration. In preliminary works, DAEMON partners were able to achieve computing resources orchestration within a 10s constraint. This constraint may be lowered with the usage of more complex orchestration solutions.</td></tr>
<tr><td colspan="2"><b>Rationale</b></td><td colspan="16">In [12], DAEMON authors were able to orchestrate computing resources for vRAN by using the Docker API, with 10-second granularity. This value is already enough to bring down the computing resource usage by more than 30% in some scenario. By using the cgroups API of the Linux system, we may achieve even lower values.</td></tr>
<tr><td><b>K1</b></td><td></td><td><b>K2</b></td><td><b>X</b></td><td><b>K3</b></td><td></td><td><b>K4</b></td><td></td><td><b>K5</b></td><td></td><td><b>K6</b></td><td></td><td><b>K7</b></td><td></td><td><b>K8</b></td><td></td><td><b>K9</b></td><td></td></tr>
<tr><td colspan="2"><b>Parents</b></td><td colspan="16">FR-CAWRS-000</td></tr>
</table>

<table>
<tr><th colspan="18">NFR-CAWRS-001</th></tr>
<tr><td colspan="2"><b>Description</b></td><td colspan="16">NI control solutions to schedule computing and radio resources in real time for vRAN shall have an inference time below 500us</td></tr>
<tr><td colspan="2"><b>Version</b></td><td colspan="16">002M17</td></tr>
<tr><td colspan="2"><b>Owner</b></td><td colspan="16">UC3M</td></tr>
<tr><td colspan="2"><b>Priority</b></td><td colspan="16">High</td></tr>
<tr><td colspan="2"><b>Risk</b></td><td colspan="16">2</td></tr>
<tr><td colspan="2"><b>Risk Description</b></td><td colspan="16">There is a mild risk that NI cannot achieve sub-second timings for the vRAN control algorithms such as the radio scheduling (which needs 1ms timings). If these timings cannot be achieved, DAEMON partners will use solutions such as slower scheduling patterns, enforced every 50-100 TTIs</td></tr>
<tr><td colspan="2"><b>Rationale</b></td><td colspan="16">Ideally, scheduling decisions are taken every TTIs, thus in the ms range scale. This requirement is quite stringent and may require specialized hardware such as GPUs deployed at the edge if deep learning solutions shall be put in place. Alternatives could be the usage of mixed models between machine learning and traditional optimization</td></tr>
<tr><td><b>K1</b></td><td></td><td><b>K2</b></td><td><b>X</b></td><td><b>K3</b></td><td></td><td><b>K4</b></td><td></td><td><b>K5</b></td><td></td><td><b>K6</b></td><td></td><td><b>K7</b></td><td></td><td><b>K8</b></td><td></td><td><b>K9</b></td><td></td></tr>
<tr><td colspan="2"><b>Parents</b></td><td colspan="16">FR-CAWRS-002</td></tr>
</table>

<table>
<tr><th colspan="18">NFR-CAWRS-002</th></tr>
<tr><td colspan="2"><b>Description</b></td><td colspan="16">NI solutions for vRAN shall maximize spectral efficiency given computing capacity constraints.</td></tr>
<tr><td colspan="2"><b>Version</b></td><td colspan="16">002M17</td></tr>
<tr><td colspan="2"><b>Owner</b></td><td colspan="16">UC3M</td></tr>
<tr><td colspan="2"><b>Priority</b></td><td colspan="16">High</td></tr>
<tr><td colspan="2"><b>Risk</b></td><td colspan="16">2</td></tr>
<tr><td colspan="2"><b>Risk Description</b></td><td colspan="16">There is a mild risk that NI cannot achieve bounded performance for the wireless performance (i.e., spectrum efficiency, leading to bandwidth and latency figures). In this case, specific boundaries to the achievable computing resource saving will be defined.</td></tr>
<tr><td colspan="2"><b>Rationale</b></td><td colspan="16">With unbounded computing resource savings, the spectral efficiency may be unacceptably low. In [12] [12], DAEMON partners were capable of achieving very good tradeoffs between the achievable savings and the pure performance, by correctly understanding the traffic patterns. Other solutions may have to be designed to guarantee that this tradeoff (the ratio between the best possible performance without computing resource optimization and the one obtained by DAEMON solutions never falls below certain thresholds) maximizing hence spectral efficiency given computing capacity constraints.</td></tr>
<tr><td><b>K1</b></td><td></td><td><b>K2</b></td><td><b>X</b></td><td><b>K3</b></td><td></td><td><b>K4</b></td><td></td><td><b>K5</b></td><td></td><td><b>K6</b></td><td></td><td><b>K7</b></td><td></td><td><b>K8</b></td><td></td><td><b>K9</b></td><td></td></tr>
<tr><td colspan="2"><b>Parents</b></td><td colspan="16">FR-CAWRS-002</td></tr>
</table>

<table>
<tr><th colspan="18">NFR-CAWRS-003</th></tr>
<tr><td colspan="3"><b>Description</b></td><td colspan="15">Predictive HARQ inference mechanisms shall have a minimum accuracy of 99% and a false positive rate below 0.1%</td></tr>
<tr><td colspan="3"><b>Version</b></td><td colspan="15">001M17</td></tr>
<tr><td colspan="3"><b>Owner</b></td><td colspan="15">UC3M</td></tr>
<tr><td colspan="3"><b>Priority</b></td><td colspan="15">High</td></tr>
<tr><td colspan="3"><b>Risk</b></td><td colspan="15">1</td></tr>
<tr><td colspan="3"><b>Risk Description</b></td><td colspan="15">There is a low risk that NI cannot integrate inference mechanisms whose accuracy is at least 99% and the false positive rate below 0.1%. There are multiple previous works applying this technique in other fields.</td></tr>
<tr><td colspan="3"><b>Rationale</b></td><td colspan="15">It is critical that the inference mechanisms have a very high accuracy and a very low rate of false positives, because a wrong prediction (due to a prediction fail or a false positive result of the prediction) incurs substantially higher cost because the transport block has to be recovered by others [12].</td></tr>
<tr><td><b>K1</b></td><td></td><td><b>K2</b></td><td><b>X</b></td><td><b>K3</b></td><td></td><td><b>K4</b></td><td></td><td><b>K5</b></td><td></td><td><b>K6</b></td><td></td><td><b>K7</b></td><td></td><td><b>K8</b></td><td></td><td><b>K9</b></td><td></td></tr>
<tr><td colspan="3"><b>Parents</b></td><td colspan="15">FR-CAWRS-001</td></tr>
</table>

<table>
<tr><th colspan="18">NFR-EAWVNF-002</th></tr>
<tr><td colspan="3"><b>Description</b></td><td colspan="15">DEAMON expect to save the 50% of the energy cost, thanks to applying NI solutions to find out the energy-aware optimal placement of VNFs of <b>FR-EAWFN-000</b></td></tr>
<tr><td colspan="3"><b>Version</b></td><td colspan="15">001M2</td></tr>
<tr><td colspan="3"><b>Owner</b></td><td colspan="15">UMA</td></tr>
<tr><td colspan="3"><b>Priority</b></td><td colspan="15">High</td></tr>
<tr><td colspan="3"><b>Risk</b></td><td colspan="15">3</td></tr>
<tr><td colspan="3"><b>Risk Description</b></td><td colspan="15">We cannot achieve the 50% of energy saving in all cases, only in some of them, or simply DAEMON solutions save an inferior percentage of energy.</td></tr>
<tr><td colspan="3"><b>Rationale</b></td><td colspan="15">The performance in terms of energy consumption of the DAEMON solution should improve the current solutions in a 50%.</td></tr>
<tr><td><b>K1</b></td><td><b>X</b></td><td><b>K2</b></td><td></td><td><b>K3</b></td><td></td><td><b>K4</b></td><td></td><td><b>K5</b></td><td></td><td><b>K6</b></td><td></td><td><b>K7</b></td><td></td><td><b>K8</b></td><td></td><td><b>K9</b></td><td></td></tr>
<tr><td colspan="3"><b>Parents</b></td><td colspan="15">FR-EAWVNF-001</td></tr>
</table>

<table>
<tr><th colspan="18">NFR-EAWVNF-003</th></tr>
<tr><td colspan="3"><b>Description</b></td><td colspan="15">The cost in terms of energy footprint of the NI solution for VNFs placing shall be less than the global energy saving</td></tr>
<tr><td colspan="3"><b>Version</b></td><td colspan="15">001M2</td></tr>
<tr><td colspan="3"><b>Owner</b></td><td colspan="15">UMA</td></tr>
<tr><td colspan="3"><b>Priority</b></td><td colspan="15">High</td></tr>
<tr><td colspan="3"><b>Risk</b></td><td colspan="15">2</td></tr>
<tr><td colspan="3"><b>Risk Description</b></td><td colspan="15">The cost of the NI-assisted VNF placement could be not much less or even higher than the energy consumption savings of the proposed solutions.</td></tr>
<tr><td colspan="3"><b>Rationale</b></td><td colspan="15">The energy saving obtained by applying energy profiling to the NI algorithms for the VNFs placement should be less than the global energy saving, to be worthy. So, the cost of the energy-awareness mechanism should be a lot less than the 50% of the energy saving proposed in NRF-EAWVNF-002.</td></tr>
<tr><td><b>K1</b></td><td><b>X</b></td><td><b>K2</b></td><td></td><td><b>K3</b></td><td></td><td><b>K4</b></td><td></td><td><b>K5</b></td><td></td><td><b>K6</b></td><td></td><td><b>K7</b></td><td></td><td><b>K8</b></td><td></td><td><b>K9</b></td><td></td></tr>
<tr><td colspan="3"><b>Parents</b></td><td colspan="15">FR-EAWVNF-001</td></tr>
</table>

<table>
<tr><th colspan="2">NFR-EAWVNF-004</th></tr>
<tr><td><b>Description</b></td><td>Energy-efficient NI shall balance throughput and energy consumption in vRANs</td></tr>
<tr><td><b>Version</b></td><td>001M17</td></tr>
<tr><td><b>Owner</b></td><td>NEC</td></tr>
<tr><td><b>Priority</b></td><td>High</td></tr>
<tr><td><b>Risk</b></td><td>1</td></tr>
<tr><td><b>Risk Description</b></td><td>There is no risk</td></tr>
</table>

<table>
<tr><td colspan="3"><b>Rationale</b></td><td colspan="15">Tethered virtualized base stations may be interested to trade-off radio spectrum capacity for energy savings</td></tr>
<tr><td><b>K1</b></td><td><b>X</b></td><td><b>K2</b></td><td></td><td><b>K3</b></td><td></td><td><b>K4</b></td><td></td><td><b>K5</b></td><td></td><td><b>K6</b></td><td></td><td><b>K7</b></td><td></td><td><b>K8</b></td><td></td><td><b>K9</b></td><td></td></tr>
<tr><td colspan="3"><b>Parents</b></td><td colspan="15">FR-EAWVNF-005</td></tr>
</table>

<table>
<tr><th colspan="18">NFR-EAWVNF-005</th></tr>
<tr><td colspan="3"><b>Description</b></td><td colspan="15">NI orchestrating resources in vRANs shall maximize networking throughput given power consumption constraints</td></tr>
<tr><td colspan="3"><b>Version</b></td><td colspan="15">001M17</td></tr>
<tr><td colspan="3"><b>Owner</b></td><td colspan="15">NEC</td></tr>
<tr><td colspan="3"><b>Priority</b></td><td colspan="15">High</td></tr>
<tr><td colspan="3"><b>Risk</b></td><td colspan="15">3</td></tr>
<tr><td colspan="3"><b>Risk Description</b></td><td colspan="15">There may be cases where power constraints cannot be satisfied.</td></tr>
<tr><td colspan="3"><b>Rationale</b></td><td colspan="15">Respecting power consumption constraints, even while learning, it is of paramount importance for battery-powered small cells, solar-powered small-cells or other types of power-constraint small-cells.</td></tr>
<tr><td><b>K1</b></td><td><b>X</b></td><td><b>K2</b></td><td></td><td><b>K3</b></td><td></td><td><b>K4</b></td><td></td><td><b>K5</b></td><td></td><td><b>K6</b></td><td></td><td><b>K7</b></td><td></td><td><b>K8</b></td><td></td><td><b>K9</b></td><td></td></tr>
<tr><td colspan="3"><b>Parents</b></td><td colspan="15">FR-EAWVNF-005</td></tr>
</table>

<table>
<tr><th colspan="18">NFR-EAWVNF-006</th></tr>
<tr><td colspan="3"><b>Description</b></td><td colspan="15">Energy savings shall be achieved in virtualized RANs without compromising given service performance constraints</td></tr>
<tr><td colspan="3"><b>Version</b></td><td colspan="15">001M17</td></tr>
<tr><td colspan="3"><b>Owner</b></td><td colspan="15">NEC</td></tr>
<tr><td colspan="3"><b>Priority</b></td><td colspan="15">High</td></tr>
<tr><td colspan="3"><b>Risk</b></td><td colspan="15">3</td></tr>
<tr><td colspan="3"><b>Risk Description</b></td><td colspan="15">It may be possible that energy savings can only be achieved when service performance constraints are not satisfied.</td></tr>
<tr><td colspan="3"><b>Rationale</b></td><td colspan="15">Satisfying service-level agreements is the top priority of a mobile network. Hence, NI solutions should strive to meet service performance constraints with a minimum energy consumption toll.</td></tr>
<tr><td><b>K1</b></td><td><b>X</b></td><td><b>K2</b></td><td></td><td><b>K3</b></td><td></td><td><b>K4</b></td><td></td><td><b>K5</b></td><td></td><td><b>K6</b></td><td></td><td><b>K7</b></td><td></td><td><b>K8</b></td><td></td><td><b>K9</b></td><td></td></tr>
<tr><td colspan="3"><b>Parents</b></td><td colspan="15">FR-EAWVNF-005</td></tr>
</table>

<table>
<tr><th colspan="18">NFR-AARES-000</th></tr>
<tr><td colspan="3"><b>Description</b></td><td colspan="15">NI solutions anomaly detection and response should have a high detection performance (specifically, DAEMON will target a 0.9 precision-recall AUC with at least 85% scoring in both precision and recall.).</td></tr>
<tr><td colspan="3"><b>Version</b></td><td colspan="15">001M5</td></tr>
<tr><td colspan="3"><b>Owner</b></td><td colspan="15">TID</td></tr>
<tr><td colspan="3"><b>Priority</b></td><td colspan="15">High</td></tr>
<tr><td colspan="3"><b>Risk</b></td><td colspan="15">2</td></tr>
<tr><td colspan="3"><b>Risk Description</b></td><td colspan="15">There is a mild risk that NI for anomaly detection cannot achieve its target performance. This highly depends on the quality and availability of ground-truth datasets from the systems DAEMON will monitor.</td></tr>
<tr><td colspan="3"><b>Rationale</b></td><td colspan="15">It is important that NI solutions for anomaly detection in the different systems that DAEMON will monitor detect real (and important) anomalies, and do not flood the operators with false alarms for their systems.</td></tr>
<tr><td><b>K1</b></td><td></td><td><b>K2</b></td><td></td><td><b>K3</b></td><td></td><td><b>K4</b></td><td></td><td><b>K5</b></td><td></td><td><b>K6</b></td><td></td><td><b>K7</b></td><td><b>X</b></td><td><b>K8</b></td><td></td><td><b>K9</b></td><td></td></tr>
<tr><td colspan="3"><b>Parents</b></td><td colspan="15">FR-AARES-002</td></tr>
</table>

A.4 Design constraints

In this section, we specify constraints on the system design imposed by external standards, regulatory requirements, or project limitations.

<table>
<tr><th colspan="18">NFR-RIS-003</th></tr>
<tr><td colspan="4"><b>Description</b></td><td colspan="14">RIS must provide beamforming gains passively, without energy-consuming (active) RF chains</td></tr>
<tr><td colspan="4"><b>Version</b></td><td colspan="14">001M17</td></tr>
<tr><td colspan="4"><b>Owner</b></td><td colspan="14">NEC</td></tr>
<tr><td colspan="4"><b>Priority</b></td><td colspan="14">High</td></tr>
<tr><td colspan="4"><b>Risk</b></td><td colspan="14">1</td></tr>
<tr><td colspan="4"><b>Risk Description</b></td><td colspan="14">There is a small risk that beamforming gains can only be achieved with active RF chains that integrate RF amplifiers.</td></tr>
<tr><td colspan="4"><b>Rationale</b></td><td colspan="14">Smart RF reflectors with active RF chains already exist and are called “relays”. The main motivation for RIS is the possibility of attaining beamforming gains with minimal energy consumption and costly electronic equipment. Hence, a RIS must necessarily be passive.</td></tr>
<tr><td><b>K1</b></td><td></td><td><b>K2</b></td><td></td><td><b>K3</b></td><td></td><td><b>K4</b></td><td></td><td><b>K5</b></td><td></td><td><b>K6</b></td><td>x</td><td><b>K7</b></td><td></td><td><b>K8</b></td><td></td><td><b>K9</b></td><td></td></tr>
<tr><td colspan="4"><b>Parents</b></td><td colspan="14">FR-RIS-000</td></tr>
</table>

<table>
<tr><th colspan="18">NFR-EAWVNF-001</th></tr>
<tr><td colspan="4"><b>Description</b></td><td colspan="14">DAEMON energy-aware solution will scale well when considering a heterogenous set of devices and network infrastructure <b>FR-EAWVFN-001.</b></td></tr>
<tr><td colspan="4"><b>Version</b></td><td colspan="14">001M2</td></tr>
<tr><td colspan="4"><b>Owner</b></td><td colspan="14">UMA</td></tr>
<tr><td colspan="4"><b>Priority</b></td><td colspan="14">High</td></tr>
<tr><td colspan="4"><b>Risk</b></td><td colspan="14">4</td></tr>
<tr><td colspan="4"><b>Risk Description</b></td><td colspan="14">The variety of devices</td></tr>
<tr><td colspan="4"><b>Rationale</b></td><td colspan="14">DAEMON should be able to consider the global footprint of VNFs placement solution for a large number of different IoT and Edge devices with variable resources and networking infrastructure. The upper values of the devices’ resources considered in DAEMON will be taken from the software and hardware network or devices specifications of the underlying infrastructure.</td></tr>
<tr><td><b>K1</b></td><td><b>X</b></td><td><b>K2</b></td><td></td><td><b>K3</b></td><td></td><td><b>K4</b></td><td></td><td><b>K5</b></td><td></td><td><b>K6</b></td><td></td><td><b>K7</b></td><td></td><td><b>K8</b></td><td></td><td><b>K9</b></td><td></td></tr>
<tr><td colspan="4"><b>Parents</b></td><td colspan="14">FR-EAWVNF-001</td></tr>
</table>

NFR-MTERM-001	
Description	DAEMON’s MTERM shall provide an exhaustive list of orchestration operations
Version	002M18
Owner	IMEC
Priority	Low
Risk	1
Risk Description	The list of orchestration operations might involve subgroups of operations, depending on the policies defined for specific types of applications (e.g., value-added services require different orchestration operations than services that are directly consumed by users).
Rationale	The NI-assisted management and orchestration framework needs to provide support for at least a basic set of orchestration operations, such as on-boarding (I.e., preparation of application descriptors and images on all required edge platforms), instantiation (on all required edge platforms), scaling up/down/out/in depending on the resource (computing and network) requirements and current resource consumption, termination (I.e., releasing the allocated resources so they can be consumed by other applications, or to save energy), and state/context

	migration (I.e., migrating the state/context of the application from one edge to another due to the UE mobility, resource availability, or energy saving purposes).

K1		**K2**	X	**K3**		**K4**		**K5**		**K6**		**K7**		**K8**		**K9**	
Parents	FR-MTERM-000																

NFR-MTERM-002	
Description	DAEMON's MTERM shall provide compliance with standardized frameworks (e.g., ETSI NFV MEC, ETSI NFV MANO, and O-RAN) running at the network edge.
Version	002M18
Owner	IMEC
Priority	Low
Risk	1
Risk Description	The insufficient level of compatibility between different standardization tracks (ETSI MEC/ETSI NFV MANO & O-RAN) can potentially lead to complex and application-specific orchestration platforms, limiting their exploitability among research tracks.
Rationale	As the standardization plays a key role in ensuring that a software tool meets certain requirements that guarantee proper work in various conditions, and expanding the exploitability of such solution, NI-assisted management and orchestration framework needs to be designed and developed in accordance with the existing standardization efforts.

K1		**K2**	X	**K3**		**K4**		**K5**		**K6**		**K7**		**K8**		**K9**	
Parents	FR-MTERM-000																

NFR-MTERM-003	
Description	DAEMON's MTERM shall provide NIF modularity and reusability among different players (e.g., network operators/vendors, service providers, etc.)
Version	002M18
Owner	IMEC
Priority	Low
Risk	1
Risk Description	The lack of NIF complexity and an increased level of openness of I/O interfaces might decrease the accuracy in decision-making processes performed by those NIFs.
Rationale	Due to the heterogeneity in resource and service deployments across edge networks, the NIFs running in both framework tiers need to be application/service-agnostic, thus, no application-specific data should be considered apart from the resource requirements and KPIs stated in SLAs. With such configuration, NIFs can be maintained and used by different stakeholders.

K1		**K2**	X	**K3**		**K4**		**K5**		**K6**		**K7**		**K8**		**K9**	
Parents	FR-MTERM-000																

NFR-SLMANO-000	
Description	DAEMON controllers and orchestrators should be steered by high-level QoE targets and business KPIs (high level intents), rather than strict QoS goals and technical KPIs.
Version	002M4
Owner	NBL
Priority	Medium
Risk	2
Risk Description	The problem may be that it may be difficult to describe expected behavior in a concise way.

<table>
<tr><td><b>Rationale</b></td><td colspan="17">Application developers should have an easy way of specifying intended behavior based on their application level knowledge and requirements to guarantee QoE for their users.</td></tr>
<tr><td><b>K1</b></td><td></td><td><b>K2</b></td><td></td><td><b>K3</b></td><td></td><td><b>K4</b></td><td></td><td><b>K5</b></td><td></td><td><b>K6</b></td><td></td><td><b>K7</b></td><td></td><td><b>K8</b></td><td></td><td><b>K9</b></td><td>X</td></tr>
<tr><td><b>Parents</b></td><td colspan="17">FR-SLMANO-000</td></tr>
</table>

<table>
<tr><th colspan="18">NFR-SLMANO-001</th></tr>
<tr><td><b>Description</b></td><td colspan="17">DAEMON shall define metrics to check the stability of a control algorithm.</td></tr>
<tr><td><b>Version</b></td><td colspan="17">001M5</td></tr>
<tr><td><b>Owner</b></td><td colspan="17">NBL</td></tr>
<tr><td><b>Priority</b></td><td colspan="17">High</td></tr>
<tr><td><b>Risk</b></td><td colspan="17">2</td></tr>
<tr><td><b>Risk Description</b></td><td colspan="17">Although a rough definition of a stable control system is easy to undderstand, i.e., if after exciting the system with a short, small perturbation, it returns fast enough to the original equilibrium, it is hard to make that definition precise for nonlinear systems.</td></tr>
<tr><td><b>Rationale</b></td><td colspan="17">It is well-known that closing the control loop may lead to instable systems. In linear systems instability stems from the fact that the closed loop transfer function has poles in the positive half plane, leading to a impulse response that exponentially increases. Although some ideas some chaos engineering may be applied, in non-linear systems there is no rigorous equivalent.</td></tr>
<tr><td><b>K1</b></td><td></td><td><b>K2</b></td><td></td><td><b>K3</b></td><td></td><td><b>K4</b></td><td></td><td><b>K5</b></td><td></td><td><b>K6</b></td><td></td><td><b>K7</b></td><td></td><td><b>K8</b></td><td></td><td><b>K9</b></td><td>X</td></tr>
<tr><td><b>Parents</b></td><td colspan="17">FR-SLMANO-003</td></tr>
</table>

<table>
<tr><th colspan="18">NFR-IBSSI-000</th></tr>
<tr><td><b>Description</b></td><td colspan="17">Network Intelligence algorithms should be adapted to the PISA architecture</td></tr>
<tr><td><b>Version</b></td><td colspan="17">001M17</td></tr>
<tr><td><b>Owner</b></td><td colspan="17">IMDEA</td></tr>
<tr><td><b>Priority</b></td><td colspan="17">Medium</td></tr>
<tr><td><b>Risk</b></td><td colspan="17">3</td></tr>
<tr><td><b>Risk Description</b></td><td colspan="17">Programmable switches have specific internal architectural models that make some machine learning models more suitable than others for deployment.</td></tr>
<tr><td><b>Rationale</b></td><td colspan="17">Modern programmable switches are compliant with the Protocol Independent Switch. Architecture (PISA) model. The solutions for machine-learning-based inference implemented in such devices must thus be aligned with the internal organization into Match-Action Units (MAUs) of such a model.</td></tr>
<tr><td><b>K1</b></td><td></td><td><b>K2</b></td><td></td><td><b>K3</b></td><td>X</td><td><b>K4</b></td><td></td><td><b>K5</b></td><td></td><td><b>K6</b></td><td></td><td><b>K7</b></td><td></td><td><b>K8</b></td><td></td><td><b>K9</b></td><td></td></tr>
<tr><td><b>Parents</b></td><td colspan="17">FR-IBSSI-002</td></tr>
</table>

<table>
<tr><th colspan="18">NFR-IBSSI-001</th></tr>
<tr><td><b>Description</b></td><td colspan="17">Network Intelligence algorithms should be resource-prudent</td></tr>
<tr><td><b>Version</b></td><td colspan="17">001M17</td></tr>
<tr><td><b>Owner</b></td><td colspan="17">IMDEA</td></tr>
<tr><td><b>Priority</b></td><td colspan="17">Low</td></tr>
<tr><td><b>Risk</b></td><td colspan="17">4</td></tr>
<tr><td><b>Risk Description</b></td><td colspan="17">Programmable switches have extremely limited computational capabilities that are primarily intended to support forwarding-related policies.</td></tr>
<tr><td><b>Rationale</b></td><td colspan="17">Decision-making is not a legacy or priority task in programmable user planes. Therefore, NI solutions deployed in programmable switches must consume as little resources as possible, in a way not to hinder the regular operation of the devices and whole network. Ideally, NI models for programmable switches should not consume more than 1% of the different memory types available in these devices.</td></tr>
<tr><td><b>K1</b></td><td></td><td><b>K2</b></td><td></td><td><b>K3</b></td><td>X</td><td><b>K4</b></td><td></td><td><b>K5</b></td><td></td><td><b>K6</b></td><td></td><td><b>K7</b></td><td></td><td><b>K8</b></td><td></td><td><b>K9</b></td><td></td></tr>
<tr><td><b>Parents</b></td><td colspan="17">FR-IBSSI-002</td></tr>
</table>

<table>
<tr><th colspan="18">NFR-NIP-001</th></tr>
<tr><td colspan="2"><b>Description</b></td><td colspan="16">NIP shall make an optimal decision on using the communication framework for sharing information between monitoring systems and the management and orchestration framework</td></tr>
<tr><td colspan="2"><b>Version</b></td><td colspan="16">002M18</td></tr>
<tr><td colspan="2"><b>Owner</b></td><td colspan="16">IMEC</td></tr>
<tr><td colspan="2"><b>Priority</b></td><td colspan="16">Low</td></tr>
<tr><td colspan="2"><b>Risk</b></td><td colspan="16">1</td></tr>
<tr><td colspan="2"><b>Risk Description</b></td><td colspan="16">Additional benchmarking of communication systems (e.g., message broker) that will be used for sharing information between framework entities is needed, and different systems might be suitable for different types of applications.</td></tr>
<tr><td colspan="2"><b>Rationale</b></td><td colspan="16">As communication systems/platforms enable either synchronous or asynchronous communication between different orchestration components and NIFs, it is important to consider the complexity of using and managing the communication system (e.g., RabbitMQ is a simple and often used in most of the existing MANO solutions) for pub/sub purposes, but also the additional latency this entity involves in the communication (e.g., RabbitMQ inevitably generates additional latency because of message queuing on a central node, comparing to ZeroMQ).</td></tr>
<tr><td><b>K1</b></td><td>X</td><td><b>K2</b></td><td><b>X</b></td><td><b>K3</b></td><td>X</td><td><b>K4</b></td><td>X</td><td><b>K5</b></td><td>X</td><td><b>K6</b></td><td>X</td><td><b>K7</b></td><td>X</td><td><b>K8</b></td><td><b>X</b></td><td><b>K9</b></td><td>X</td></tr>
<tr><td colspan="2"><b>Parents</b></td><td colspan="16">FR-NIP-002</td></tr>
</table>

<table>
<tr><th colspan="18">NFR-NIP-002</th></tr>
<tr><td colspan="2"><b>Description</b></td><td colspan="16">NIP shall provide openness of interfaces between orchestration/control tiers and NIFs/NISs to mitigate the dependence on specific network operators/vendors/infrastructure providers/service providers</td></tr>
<tr><td colspan="2"><b>Version</b></td><td colspan="16">001M18</td></tr>
<tr><td colspan="2"><b>Owner</b></td><td colspan="16">IMEC</td></tr>
<tr><td colspan="2"><b>Priority</b></td><td colspan="16">Low</td></tr>
<tr><td colspan="2"><b>Risk</b></td><td colspan="16">3</td></tr>
<tr><td colspan="2"><b>Risk Description</b></td><td colspan="16">The vulnerability of open interfaces between management and orchestration tiers, and between NIFs, might impose certain security risks that need to be properly handled.</td></tr>
<tr><td colspan="2"><b>Rationale</b></td><td colspan="16">Distributed edge networks and cloud can be deployed by different vendors/infrastructure providers, belonging to different Mobile Network Operator (MNO) domains. Thus, it is utmost important to provide open interfaces between NIFs and management and orchestration tiers (i.e., edge and cloud) in order to facilitate orchestration operations, and to mitigate the dependence on the vendor-specific configuration of NIFs.</td></tr>
<tr><td><b>K1</b></td><td>X</td><td><b>K2</b></td><td><b>X</b></td><td><b>K3</b></td><td>X</td><td><b>K4</b></td><td>X</td><td><b>K5</b></td><td><b>X</b></td><td><b>K6</b></td><td>X</td><td><b>K7</b></td><td>X</td><td><b>K8</b></td><td><b>X</b></td><td><b>K9</b></td><td>X</td></tr>
<tr><td colspan="2"><b>Parents</b></td><td colspan="16">FR-NIP-002</td></tr>
</table>

<table>
<tr><th colspan="2">NFR-NIP-003</th></tr>
<tr><td><b>Description</b></td><td>NIP shall provide support for multiple virtualization environments for deploying services/applications in distributed domains</td></tr>
<tr><td><b>Version</b></td><td>002M18</td></tr>
<tr><td><b>Owner</b></td><td>IMEC</td></tr>
<tr><td><b>Priority</b></td><td>Low</td></tr>
<tr><td><b>Risk</b></td><td>1</td></tr>
</table>

<table>
<tr><td><b>Risk Description</b></td><td colspan="17">The diversity in virtualization environments needs specific maintenance, and virtualization-specific policies for orchestration operations, which significantly increases complexity of orchestration operations in both tiers within NI-assisted management and orchestration framework.</td></tr>
<tr><td><b>Rationale</b></td><td colspan="17">With regards to the limited resource availability within the edge platforms, comparing to the large and resourceful data-center, the lightweight virtualization, and orchestration solutions for small-size programmable devices are required. Thus, containerization proves to be the suitable candidate to deliver a lightweight deployment of services and applications suitable for network edge deployments.</td></tr>
<tr><td><b>K1</b></td><td>X</td><td><b>K2</b></td><td><b>X</b></td><td><b>K3</b></td><td>X</td><td><b>K4</b></td><td>X</td><td><b>K5</b></td><td>X</td><td><b>K6</b></td><td>X</td><td><b>K7</b></td><td>X</td><td><b>K8</b></td><td>X</td><td><b>K9</b></td><td>X</td></tr>
<tr><td><b>Parents</b></td><td colspan="17">FR-NIP-000</td></tr>
</table>

<table>
<tr><th colspan="18">NFR-NIP-004</th></tr>
<tr><td><b>Description</b></td><td colspan="17">NIP shall provide support for federated multi-domain management and orchestration</td></tr>
<tr><td><b>Version</b></td><td colspan="17">003M18</td></tr>
<tr><td><b>Owner</b></td><td colspan="17">IMEC</td></tr>
<tr><td><b>Priority</b></td><td colspan="17">Low</td></tr>
<tr><td><b>Risk</b></td><td colspan="17">3</td></tr>
<tr><td><b>Risk Description</b></td><td colspan="17">Management level agreements are necessary for establishing collaboration between orchestration and management entities, and NIFs in different edge domains.</td></tr>
<tr><td><b>Rationale</b></td><td colspan="17">Due to the high mobility of users in 5G and beyond 5G ecosystems, applications are deployed in distributed way across different edge platforms. Thus, NI-assisted management and orchestration framework needs to support cross-domain/cross-edge service orchestration for achieving seamless service operation.</td></tr>
<tr><td><b>K1</b></td><td>X</td><td><b>K2</b></td><td><b>X</b></td><td><b>K3</b></td><td>X</td><td><b>K4</b></td><td>X</td><td><b>K5</b></td><td><b>X</b></td><td><b>K6</b></td><td>X</td><td><b>K7</b></td><td>X</td><td><b>K8</b></td><td><b>X</b></td><td><b>K9</b></td><td>X</td></tr>
<tr><td><b>Parents</b></td><td colspan="17">FR-NIP-000</td></tr>
</table>

<table>
<tr><th colspan="2">NFR-NIP-005</th></tr>
<tr><td><b>Description</b></td><td>DAEMON's NIP shall interact with the Network Orchestration Framework aligned with ETSI-NFV-MANO</td></tr>
<tr><td><b>Version</b></td><td>002M18</td></tr>
<tr><td><b>Owner</b></td><td>UC3M</td></tr>
<tr><td><b>Priority</b></td><td>High</td></tr>
<tr><td><b>Risk</b></td><td>1</td></tr>
<tr><td><b>Risk Description</b></td><td>The NIP needs to understand what are the Network Services that are currently running in the system, in order to match the network intelligence to them.</td></tr>
<tr><td><b>Rationale</b></td><td>The Network Orchestrator (either based on an ETSI-NFV-MANO platform or an implementation using ONAP) is the element in the network architecture that keeps track of all the network services (and the network slices implementing them). Therefore, the NIO shall interact with the Network Orchestrator to (we use in the following the ETSI NFV MANO terminology):
• The number and type of network slices/ services that are running (available at the NFV-O)
• The number and extent of subnetwork slices that are running (available at the NFV-O)
• The number and extent of VNFs that are running (available at the NFV-O and VNFM)
• The network topology (available at the NFV-O and VIM)</td></tr>
</table>

<table>
<tr><td></td><td colspan="17">This information is required by the NIO to understand e.g., where to run the NI and to match the already running network services (e.g., an eMBB Network Slice).</td></tr>
<tr><td><b>K1</b></td><td>X</td><td><b>K2</b></td><td>X</td><td><b>K3</b></td><td>X</td><td><b>K4</b></td><td>X</td><td><b>K5</b></td><td>X</td><td><b>K6</b></td><td>X</td><td><b>K7</b></td><td>X</td><td><b>K8</b></td><td>X</td><td><b>K9</b></td><td>X</td></tr>
<tr><td><b>Parents</b></td><td colspan="17">FR-NIP-002</td></tr>
</table>

<table>
<tr><th colspan="18">NFR-NIP-006</th></tr>
<tr><td><b>Description</b></td><td colspan="17">DAEMON's NIP shall interact with the 3GPP Network Analytics System</td></tr>
<tr><td><b>Version</b></td><td colspan="17">002M18</td></tr>
<tr><td><b>Owner</b></td><td colspan="17">UC3M</td></tr>
<tr><td><b>Priority</b></td><td colspan="17">High</td></tr>
<tr><td><b>Risk</b></td><td colspan="17">1</td></tr>
<tr><td><b>Risk Description</b></td><td colspan="17">The NIP needs to interact with the 3GPP Network Data Analytics System, as the network analytics defined by the standard are NIFs.</td></tr>
<tr><td><b>Rationale</b></td><td colspan="17">The network analytics services, as defined by the 3GPP system in [14] are NIFs that need to be orchestrated and managed as the other NIF defined in the project. The producer/consumer NFs in the analytics systems are NIF-C in the DAEMON view. The NWDAF is a particular kind of NIF-C, that implements the model.</td></tr>
<tr><td><b>K1</b></td><td>X</td><td><b>K2</b></td><td>X</td><td><b>K3</b></td><td>X</td><td><b>K4</b></td><td>X</td><td><b>K5</b></td><td>X</td><td><b>K6</b></td><td>X</td><td><b>K7</b></td><td>X</td><td><b>K8</b></td><td>X</td><td><b>K9</b></td><td>X</td></tr>
<tr><td><b>Parents</b></td><td colspan="17">FR-NIP-002</td></tr>
</table>

<table>
<tr><th colspan="18">NFR-NIP-007</th></tr>
<tr><td><b>Description</b></td><td colspan="17">DAEMON's NIP shall interact with the O-RAN on non-RT RIC and near-RT RIC</td></tr>
<tr><td><b>Version</b></td><td colspan="17">002M18</td></tr>
<tr><td><b>Owner</b></td><td colspan="17">NEC</td></tr>
<tr><td><b>Priority</b></td><td colspan="17">High</td></tr>
<tr><td><b>Risk</b></td><td colspan="17">1</td></tr>
<tr><td><b>Risk Description</b></td><td colspan="17">The NIP needs to interact with the O-RAN RIC entities, namely non-RT RIC and near-RT RIC, through standard interfaces defined by O-RAN, e.g., via A1, E2, O2 or O1. The interfaces may need to be extended or new interfaces need to be defined to communicate and interact with DAEMON NIFs.</td></tr>
<tr><td><b>Rationale</b></td><td colspan="17">The Non-RT RIC entities (such as rApps) or Near-RT RIC entities (such as xApps), defined in the O-RAN reference architecture, can be provided by the NIFs in the project. The O-RAN RICs can be considered as the consumer of the NIFs for managing open RAN configuration and RAN related functionalities and resources.</td></tr>
<tr><td><b>K1</b></td><td>X</td><td><b>K2</b></td><td>X</td><td><b>K3</b></td><td>X</td><td><b>K4</b></td><td>X</td><td><b>K5</b></td><td>X</td><td><b>K6</b></td><td>X</td><td><b>K7</b></td><td>X</td><td><b>K8</b></td><td>X</td><td><b>K9</b></td><td>X</td></tr>
<tr><td><b>Parents</b></td><td colspan="17">FR-NIP-002</td></tr>
</table>

<table>
<tr><th colspan="2">NFR-NIP-008</th></tr>
<tr><td><b>Description</b></td><td>The system constraint for NIF selection at the edge are energy, computation, network, and KPIs</td></tr>
<tr><td><b>Version</b></td><td>002M18</td></tr>
<tr><td><b>Owner</b></td><td>IMEC</td></tr>
<tr><td><b>Priority</b></td><td>Low</td></tr>
<tr><td><b>Risk</b></td><td>1</td></tr>
<tr><td><b>Risk Description</b></td><td></td></tr>
<tr><td><b>Rationale</b></td><td>Depending on the available resources and the business goals or SLAs, DAEMON will select the best NIF model that suits the assisted system. For example, in some</td></tr>
</table>

	cases it might be feasible to sacrifice accuracy at the expenses of a lower computational complexity.																
K1	X	**K2**	**X**	**K3**	X	**K4**	X	**K5**	X	**K6**	X	**K7**	X	**K8**	X	**K9**	X
Parents	FR-MTERM-016																

B Appendix: Literature Review – Current status

In this appendix, we present the data obtained from the literature review described in Section 5. We incorporate such information in this deliverable for completeness. For the sake of readability, we show the content in tabular format, where we include the answers to all the research questions that were described in Section 5.1.

Some of the research questions have a limited set of possible questions, which we detail in the following. Regarding the operation timescale, we distinguish the following cases:

- Very short timescale (us-ms)
- Short timescale (ms-s)
- Medium timescale (s-min)
- Long timescale (min-h)
- Very long timescale (h-days)

For algorithm location, we distinguish between

- Orchestration Plane
- Control plane
- Data plane

For the micro-domain of operation, we have:

- Subscriber
- Access
- Beyond Edge
- Far Edge
- Edge
- Transport
- Core
- Cross domain ("*Cross*")

With respect to the Application Area, we follow the latest 5GPPP white paper [59] where three major application areas were identified, namely i) Network Planning, ii) Network Diagnostics, and iii) Network Optimization and Control. We include Network Security inside the Network Diagnostics category, as they are correlated.

Next, we present the table. We have split the content in two different tables, due to the size of the content. First, we present the research questions related to the network itself (5.1.1) and the ones related to the data for ML training and testing (5.1.3). Each row includes the information related to one of the questions, whereas each column contains the information of one of the works.

Afterwards, we present an analogous table containing the research questions related to Machine learning (5.1.2) for the same works. In the model description field, we explain inputs and outputs, states/actions for reinforcement learning, etc.), whereas for "optimality gap" we compare the proposed approach versus an optimal solution if it exists or any reference benchmark otherwise

B.1 Research questions related to the network and the data

	Bib key	sotocamelo2021	Zhu2021	ayalagarcia2021	ayalagarcia2020	haeritrajkovic2017	busseGrawitz2019	nakanoyasato2019	xiaozhang2019
Network Related	**Networking problem**	Link Evaluation, Throughput Prediction	Resource management	Energy consumption and performance optimization in vRAN	Radio and computing resource control	Resource management	Traffic classification	Resource management	Resource management
	Application Area	Network Optimization and Control	Network Optimization and Control	Network Optimization and Control	Network Optimization and Control	Network Optimization and Control	Network Diagnostics and Security	Network Optimization and Control	Network Optimization and Control
	Micro-domain	Edge	Edge / Client	Edge	Edge	Cross	Transport	Cross	Cross
	Algorithm Location	Control Plane	Control Plane	Control Plane	Control Plane	Orchestration Plane	Data Plane	Orchestration Plane	Orchestration Plane
	Operation Timescale	ms-s	N/A	s-min	us-ms /ms-s	ms-s	us-ms	N/A	ms-s
Dataset	**Dataset Generation**	Synthetic	Synthetic	Real	Real	Synthetic	Real	Real	Synthetic
	Dataset Generation Setup	600 deployments with random number of APs and STAs (78078 devices in total). APs are fixed at the center of the cell while STAs are randomly placed around the AP's coverage area (10m)	6 vehicles, Computing power of VEC server: 6.3GHz, Computation power of vehicle: 1GHz, data per task: [50, 600]kB, Prices of VEC servers: [0, 1] initial = 0.3, Cost of vehicle: 1, Iterations: 1000, Num. mini batches: 256	2 nodes acting as UE and eNB	2 nodes acting as UE and eNB	Substrate network: 50 nodes and 221 edges, substrate CPU, bandwidth: 50-100, virtual nodes: 3-10, virtual nodes CPU: 2-20, virtual link bandwidth: 0-50	Testbed network with 12 hosts attacked by 2 hosts	SFC1: Proxy - FW - IDS, SFC2: FW - IDS, SFC3: IDS - FW	N/A
	Dataset Availability	Open	N/A	Open	Open	N/A	Open	N/A	Open
	Data Velocity	End-to-end: around KB/seconds Only prediction: around KB/milliseconds	N/A	Second	Second	N/A	Velocity of the data through the switch in the order of MB/seconds	N/A	N/A
	Data Variety	Structured	N/A	Structured	Structured	Structured	Unstructured	Structured	Structured
	Data Volume	Deployment provided in a CSV file with approx. 20KB per file	N/A	3 floating numbers	3x (num. BS) x (num. monitoring samples) floating numbers (T = 100)	N/A	PCAP file of about 8 GB	N/A	N/A
	Data Veracity	validated against ns-3, Bianchi and Markov models. No validation using real data.	N/A	Accurate	Accurate	N/A	Accurate	N/A	N/A

	Bib key	quanghadjadj-aoul2019	peihong2019	zhengtian2019	Solozabal-ceberio2019	bega2019	Foukas-radunovic2021	zhaoliang2019	Zhang2018
Network Related	**Networking problem**	Resource management	Resource management	Resource management	Resource management	Capacity prediction	Execution times prediction	User association and resource allocation	Traffic Prediction
	Application Area	Network Optimization and Control	Network Optimization and Control	Network Optimization and Control	Network Optimization and Control	Network Optimization and Control	Network Diagnostics and Security	Network Planning	Network Optimization and Control
	Micro-domain	Cross	Cross	Edge	Cross	Edge/Core	Edge	Subscriber	Edge/Core
	Algorithm Location	Orchestration Plane	Orchestration Plane	Orchestration Plane	Orchestration Plane	Control Plane/ Orchestration Plane	Control Plane	Control Plane	Control Plane /Orchestration Plane
	Operation Timescale	ms-s	ms-s	N/A	N/A	min-h	ms-s	N/A	min-h
Dataset	**Dataset Generation**	Synthetic	Real	Synthetic	Synthetic	Real	Real	N/A	Real
	Dataset Generation Setup	N/A	N/A	N/A	N/A	Real data from 470 4G eNodeBs of a mobile network deployed in a large metropolitan region of 100 km^2, collected at the gateway of an operational mobile network by monitoring the GPRS Tunneling Protocol (GTP).	Probes at the network deployment	N/A	Real Data from Telecom Italia Dataset for Milan
	Dataset Availability	N/A	Open	N/A	N/A	Private	N/A	N/A	Private
	Data Velocity	N/A	N/A	N/A	N/A	each 5 minutes	N/A	N/A	N/A
	Data Variety	Structured	Structured	Structured	Structured	Structured	N/A	N/A	Structured
	Data Volume	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
	Data Veracity	N/A	N/A	N/A	N/A	Accurate (but used for forecasting, future may not be known from current samples)	N/A	N/A	Accurate (but used for forecasting, future may not be known from current samples)

	Bib key	Bakri2021	Tripathi-pulighеddu2021	mismarchoi2019	ziongZilberman2019	gijon21_longterm	gutterman19	yangcao2020	liuyu2020
Network Related	Networking problem	Network Slice Admission Control	dynamic radio resource allocation in heterogeneous vRANs	Downlink SINR maximization	Traffic classification	Traffic Forecasting	Resource Forecasting	Spectrum Access	Resource management
	Application Area	Network Optimization and Control	Network Optimization and Control	Network Optimization and Control	Network Diagnostics and Security	Network Optimization and Control	Network Optimization and Control	Network Optimization and Control	Network Optimization and Control
	Micro-domain	Core	Edge	Edge	Transport	Edge / Core	Edge/core	Edge	Edge
	Algorithm Location	Control Plane	Control Plane	Control Plane	Data Plane	Control Plane/ Orchestration Plane	Control Plane/ Orchestration Plane	Control Plane	Orchestration Plane
	Operation Timescale	ms-s / s-min	ms-s	ms-s	us-ms	h-days	h-days	ms-s	N/A
Dataset	Dataset Generation	N/A	N/A	N/A	Real	Real	Synthetic	Synthetic	Synthetic
	Dataset Generation Setup	N/A	N/A	N/A	Testbed with 28 different IoT devices (e.g., cameras, sensors, etc)	Data collected from January 2015 to June 2017 (30 months) in a large live LTE network serving an entire country	LTE eNodeB configured with a 10 MHz bandwidth using 700 MHz wireless spectrum	Two USRP2 connected to respective PCs to generate and collect the RF traces. Authors varied the window size, the SNR levels and data payload size. All the experiments were performed in a 4-node star topology.	Task generation is modeled as a Poisson process; the distribution of the IoT devices is modeled by a Poisson cluster process.
	Dataset Availability	Private	N/A	N/A	Open	N/A	N/A	Private	Private
	Data Velocity	N/A	N/A	N/A	Velocity of the data through the switch in the order of MB/seconds	N/A	N/A	N/A	N/A
	Data Variety	N/A	N/A	N/A	Unstructured	N/A	N/A	Unstructured	Structured
	Data Volume	N/A	N/A	N/A	PCAP file	N/A	N/A	N/A	N/A
	Data Veracity	N/A	N/A	N/A	Accurate	N/A	N/A	RF traces are from real devices but might be limited to the ones used during training.	Good amount of considered parameters allowing to obtain a rich dataset

	Bib key	camelomennes2020	yange2020	wangmao2021	jiayang2021	nakashimakamiya2020	xucheng2018	manousis2021	perinoyang2020
Network Related	Networking problem	Spectrum Sharing	Resource management	Resource management	Resource management	Channel Allocation	Spectrum Sensing	Anomaly Detection	Network Failure Management
	Application Area	Network Optimization and Control	Network Optimization and Control	Network Optimization and Control	Network Optimization and Control	Network Optimization and Control	Network Planning	Network Diagnostics and Security	Network Diagnostics and Security
	Micro-domain	Edge	Cross	Cross	Edge	Access	Access	Subscriber	Access
	Algorithm Location	Control Plane/ Orchestration Plane	Orchestration Plane	Orchestration Plane	Orchestration Plane	Control Plane	Control Plane	Data Plane	Control Plane
	Operation Timescale	ms-s	ms-s	ms-s	N/A	N/A	N/A	ms-s	min-h
Dataset	Dataset Generation	Synthetic	Synthetic	Synthetic	Synthetic	Synthetic	Synthetic	Real	Real
	Dataset Generation Setup	Experiments were conducted in Colosseum; five Collaborative Intelligent Radios (CIRs) were sharing the spectrum; the incumbent was a doppler weather radar; radios were sharing 10MHz of bandwidth; the radios were connected to a collaboration network.	N/A	N/A	N/A	Multiple APs were simulated using back-of-the-envelope (BoE) technique, this assumes each AP has a STA and the wireless link is saturated	Multiple PUs (2, 3) and multiple SUs (6, 9) were simulated; the transmission power of each PU is set to 50 mW; transmission signals are assumed to attenuate according to a free-space propagation model with pathloss exponent equal to 4.	N/A	N/A
	Dataset Availability	Private	N/A	N/A	N/A	Private	Private	Private	Private
	Data Velocity	Sample rate of 23.04Mb/s; each scatter voxel contains 35 32-FFT samples	N/A	N/A	N/A	N/A	N/A	N/A	N/A
	Data Variety	Unstructured	Structured	Structured	Structured	Structured	Structured	N/A	Structured
	Data Volume	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
	Data Veracity	N/A	N/A	N/A	N/A	BoE is an easy computation method that produces very accurate results in modest-size networks, however it presents limitations in large-scale networks	The assumptions made might not hold in realistic scenarios (e.g., the coverage area of the PU is a perfect circle)		N/A

	Bib key	iyerli2018	navarrorossi2020	kattadigeraman2021	manglahalepovic2020	lui2019	lossleap2021	tesrnn2021
Network Related	**Networking problem**	RAN Performance Analysis	Anomaly Detection	Traffic classification	Quality of Experience Prediction	Resource management	Resource management	Resource management
	Application Area	Network Diagnostics and Security	Network Diagnostics and Security	Network Optimization and Control	Network Diagnostics and Security	Network Optimization and Control	Network Optimization and Control	Network Optimization and Control
	Micro-domain	Access	Core	Transport	Transport	Edge	Cross-domain	Transport
	Algorithm Location	Control Plane	Control Plane	Control Plane	Control Plane	Control Plane	Control Plane	Control Plane
	Operation Timescale	min-h	min-h	min-h	min-h	ms-s	ms-s	ms-s
Dataset	**Dataset Generation**	Real	Real	Real	Real	N/A	Real	Real
	Dataset Generation Setup	N/A	N/A	N/A	N/A	N/A	QoE pipeline simulated / Collection of traffic monitoring samples in a metropolitan area	Collection of traffic monitoring samples in a metropolitan area
	Dataset Availability	Private	Private	Open	Private	N/A	Private	Private
	Data Velocity	6TB traffic per hour.	minute time scale 70K KPIs per router	Hours	N/A	N/A	GB/seconds	GB/seconds
	Data Variety	Structured	Structured	Structured	Structured	N/A	Structured	Structured
	Data Volume	6TB traffic per hour.	minute time scale 70K KPIs per router	N/A	N/A	N/A	~10GBs	N/A
	Data Veracity	N/A	N/A	N/A	N/A	N/A	Accurate	Accurate

B.2 Research questions related to Machine Learning algorithms

	Bib key	sotocamelo2021	Zhu2021	ayalagarcia2021	ayalagarcia2020	haeritrajkovic2017
Machine-Learning approach	ML Method	SL	RL	RL	RL	RL
	ML Problem	Prediction	Control (resource management policy)	Control	Control	Control
	ML Algorithm	Graph Neural Networks	Deep Deterministic Policy Gradient (DDPG), and Multi-Agent DDPG (MADDPG)	Deep deterministic policy gradient	Neural networks	Monte Carlo Tree Search (MCTS)
	Resource-awareness	No	Yes	Yes	Yes	Yes
	Model description	Input: Node Type, Node positioning, Channel Config., RSSI, SINR, Airtime, Interference among BSSs, Distance between Nodes, Bandwidth per deployment Output: Throughput per Node per deployment	Deep Reinforcement Learning Resource Management (DRLRM)	input: UL and DL channel quality indicator and the "new" bit presence output: configuration policies	input: Encoded representation of the scheduler context output: scheduling policy	State: current node mapping. Action: subsequent node mapping
	Loss function /Reward	Root Mean-Squared Error (RMSE)	Reward of client/server	(i) balance performance and cost (ii) performance maximization	minimize operational cost when CPU capacity is sufficient or meet performance target when there is computing deficit	Embedding Profitability
	Baseline Comparison	Machine Learning (CNN, FNN and GB) Heuristic (Truncated Normal Distribution)	N/A	Only evaluation available	1. CVrain-Rlegacy: Proposed CPU orchestrator and legacy scheduler 2. R-Optimal: knows the required CPU and scheduling policies that maximizes the reward 3. T-Optimal: similar to R-Optimal but maximizing throughput 4. Heuristic: linear model between MCS and CPU load	(Relaxed) MILPs, heuristics
	Limitations of ML vs. benchmark	GNNs are a relatively new learning architecture which require more research when operating with edge features. Optimization methods regularization and normalization do not work out-of-the-box.	sample efficiency and stability	Compute intensive inference	No optimality guarantees	The selection phase deals with the exploration-exploitation dilemma, Monte Carlo state sampling --> inaccuracy, instability, one tree model for each embedding, trained with a fixed flow mapping algorithm
	Advantages of ML vs. benchmark	GNNs exploit the graphs' topological information, independently of how many nodes the graph has, by aggregating neighboring nodes' information.	unstable training of DDPG	Data efficient. Algorithm converges with a small number of samples	Supports non-linear contextual traffic and mobility patterns	Compared to heuristics: no need for complete re-design if the objective changes. Compared to ILPs: can incorporate non-linear terms in the objective function
	Optimality Gap	There is no optimal solution using exact methods. In the best-case experiment GNN outperformed ML methods and the Random guesser by 55% and 64%, respectively.	Not provided	Not provided	2% below optimal	There is no optimal solution available. In terms of profitability, the algorithm at hand is at least 10% better compared to the rest.
	Trade-off ML /benchmark	N/A	N/A	N/A	N/A	N/A

	Bib key	busseGrawitz2019	nakanoyasato2019	xiaozhang2019	quanghadjadj-aoul2019	peihong2019	zhengtian2019
Machine-Learning approach	**ML Method**	SL	RL	RL	RL	RL	RL
	ML Problem	Classification	Control	Control	Control	Control	Control
	ML Algorithm	Random Forest online implementation on a programmable switch	Gradient Boosting Decision-Tree - Monte Carlo Value Iteration	Deep Policy Gradient	Deep Deterministic Policy Gradient based on the Actor-Multiple Critics paradigm	Double deep Q-network (DDQN)	Q-learning (e-greedy)
	Resource-awareness	Yes	No	No	No	No	No
	Model description	INPUT: more than 80 flow features (e.g., size of the packet, inter-arrival time, etc); OUTPUT: classification as malware or benign flow	State: VNF presence at PoPs, allocated CPU and memory per VNF. Action: VNF scaling and migration	State: resource utilization across all servers, links, resource demands of VNs Action: server to host the VNF	State: resource requirements across all VNFs, Vlinks, Action: ranking of node-to-node and link-to-link pairs.	State: average available bandwidth, memory, CPU and #cores on links, nodes, VNFIs across each region. Action: region combination	State: SFC type to be deployed, Action: server to place the SFC
	Loss function /Reward	F1 score, where F1 = 2(precision*recall)/(precision+recall)	Step1: throughput/latency, Step2: utility/cost	minimize opex, maximize throughput	acceptance rate	weighted cost	weighted cost
	Baseline Comparison	1) same model in a floating-point-operation system 2) an offline system (running on a server) that operates over the full flow	Conventional RL (without performance profile)	greedy algorithms and a Bayesian learning method	DDPG	MGSAS (primarily introduced to accelerate embedding solvers by limiting the solution space), Eigendecomposition (operations on adjacency matrices)	ILP, Bicriteria approximation algorithm, greedy
	Limitations of ML vs. benchmark	Loss of accuracy due to compression techniques for memory optimization // no floating point	Additional effort to profile SFCs in terms of performance	neural networks inherently reduce the explainability capacity compared to e.g., a heuristic method, parameter tuning, training	computational complexity (MCN), incorporation of tailored heuristic	repetitive re-training, region selection (not PoP selection) is a strong simplification	optimality gap, memory consumption (conventional Q-learning)
	Advantages of ML vs. benchmark	The classification is performed at line-rate (directly at the switch)	Faster convergence and adaptability	adaptability to varying network environment, fast decision-making	improved performance	improved performance	no need for a-priory knowledge of resource requirements
	Optimality Gap	2% below optimal (in accuracy)	N/A	Not provided	N/A	N/A	32% worse than the optimal
	Trade-off ML vs benchmark w.r.t. performance	N/A	N/A	N/A	N/A	N/A	N/A

	Bib key	solozabalceberio2019	bega2019	foukasradunovic2021	zhaoliang2019	Zhang2018	Bakri2021
Machine-Learning approach	ML Method	RL	SL	SL	RL	SL	RL
	ML Problem	Control	Forecasting / Prediction	Prediction	Prediction	Forecasting / Prediction	Classification /Control
	ML Algorithm	Neural combinatorial optimization	Deep Neural Network	Decision Tree	dueling-double DQN	Deep Neural Network	Deep Q-Learning / Regret Matching
	Resource-awareness	No	No	yes	no	No	No
	Model description	State: sequence of VNFs (SFC), Action: mapping of VNFs to servers	Input: Previous traffic measurements (But it allows other inputs, e.g. signal quality, occupied resource blocks...) Output: forecast of the capacity for future demands for a specific network slice	input: Set of features describing the state of the base station (number of scheduled UEs and their transport block sizes, number of layers)	input: List of allowed actions to be taken by all Ues output: Optimal sequence of actions to achieve QoS requirements of all UEs	Input: Traffic volume Output: Traffic Volume	Input: Slices
	Loss function /Reward	minimize energy consumption with constraint violation penalization	Tailored loss function for capacity forecasting (asymmetric cost between overprovisioning and underprovisioning (= SLA violation)	N/A	The reward function focus on guaranteeing the quality of service (QoS) requirement of UEs.	Least Square error (L2 Loss function)	N/A
	Baseline Comparison	MILP, First-Fit heuristic	- Same ML architecture without tailored loss function (MAE) - Naïve (Replicate last) - Infocom17 (SoA) - MobiHoc18 (SoA) - - Overprovisioning	Benchmark for the prediction model: 1. linear regression 2. (non-linear) gradient boosting model	1. Optimal policy	Machine Learning, ARIMA HW-ExpS	QL- DQL-RM
	Limitations of ML vs. benchmark	benchmark performs better on small sized problems; constraint satisfaction is not guaranteed - but the probability of occurrence is reduced	Loss function (single) parameter needs to be tuned (also advantage)	Not provided	Not provided	Not provided	Not provided
	Advantages of ML vs. benchmark	informative guidance to heuristic for improved performance on large scale networks	- Loss function parameters can be tuned - Tailored Loss function allows to obtain much better performance because it adapts to the problem	Ability to predict task execution times	near-optimal solution with a small number of iterations	More accuracy than with other methodologies	Not provided
	Optimality Gap	within 10%	N/A	Not provided	Near optimal results	Near optimal results	Not provided
	Trade-off ML vs benchmark w.r.t. performance	N/A	N/A	N/A	N/A	N/A	N/A

	Bib key	tripathipuligheddu2021	mismarchoi2019	ziongZilberman2019	gijon21_longterm	gutterman19
Machine-Learning approach	**ML Method**	RL	RL	SL	SL	SL
	ML Problem	Prediction	Prediction	Classification	Forecasting	Forecasting
	ML Algorithm	SARSA	Deep neural networks / Deep Q-learning	Decision Tree, SVM, Naive Bayes, K-means	Decision Trees; SARIMA,AHW,Random Forest,ANN,ANN–LSTM,SVR	Hybrid (X_LSTM: ARIMA+LSTM)
	Resource-awareness	Yes	No	No	No	No
	Model description	input: SNR, buffer state, and also the status of aggregate traffic load already hosted on the available links output: policy to select the best available link and transmission parameters for packet transfer	Input: Initial downlink SINR and target SINR for the voLTE downlink closed loop power control and a set of handling actions in a network for the Fault management solution	INPUT: 11 flow features (e.g., size of the packet, source and destination ports, etc); OUTPUT: classification of the type of device (e.g., sensor, video, etc)	Input: Past Traffic volume Output: Predicted Traffic Volume	Input: Past Traffic volume Output: Predicted Traffic Volume
	Loss function /Reward	The reward function focus on accomplish the packet loss and latency requirements, being as close as possible to the optimal value	VoLTE PC: Ensures that the base station radio link power is constantly tuned to meet the target SINR. Fault management: focuses on solve the impact of impairments on DL throughput as experienced by Ues	F1 score, where F1 = 2(precision*recall)/(precision+recall)	MAE	REVA: combining the amount of average Physical Resource Blocks with individual channel bearer conditions
	Baseline Comparison	Modified version of the proposed solution where the reward is evaluated by averaging the reward over all the RL agents	1. Fixed power allocation 2. Maximum SINR	Decision Tree with tree depth = 11 implemented in a bmv2 software switch is compared against: 1) the same implementation with a smaller tree depth; 2) a hardware implementation in a NetFPGA with 5 levels.	Among themselves	simple LSTMs
	Limitations of ML vs. benchmark	No optimality guarantees	Not provided	Loss of accuracy due to a reduced three depth in both cases	longer to converge/train	Not provided
	Advantages of ML vs. benchmark	Effectively addresses the need for a solution that can swiftly adapt to the underlying channel network dynamics for context-aware radio resource allocation in heterogeneous vRANs	Effectively tunes downlink SINR and number of active faults through exploration and exploitation without the interaction of the UE	Better accuracy	Better MAE/MAPE	degree of prediction accuracy with a MAPE
	Optimality Gap	Not provided	Not provided	Loss of accuracy of: 1) 1-2% per each level of the tree; 2) about 9%.	Not provided	Not provided
	Trade-off ML vs benchmark w.r.t. performance	N/A	N/A	N/A	N/A	N/A

	Bib key	yangcao2020	liuyu2020	camelomennes2020	yange2020
Machine-Learning approach	**ML Method**	SL	UL/RL	SSL/SL	RL
	ML Problem	Classification	Clustering; Decision Making	Recognition; Forecasting	Control
	ML Algorithm	Convolutional Neural Network (CNN)	K-Means; DQN	Convolutional Neural Network (CNN); Context Tree Weighting (CTW)	Asynchronous Advantage Actor-Critic and Graph Convolutional Neural Network
	Resource-awareness	No	No	No	No
	Model description	Master-CNN Input: IQ samples of RF traces Output: Number of colliding STAs Slave-CNN Input: IQ samples of RF traces Output: ID of the colliding STAs	K-Means Input: User Priority (distance, computation offloading probability) Output: Which devices must compute their tasks locally, at the edge or which of them can decide. DRL Actions: transmission power of the device (P=0 indicates local computation) States: channel gain, task queue, remaining computation capacity of each device.	*Technology Recognition:* Input: RF traces of different radio technologies and idle noise. Output: if a given technology is present in a given spectrum voxel *Spectrum Usage Pattern Predictor*: Input: The transmission pattern of an incumbent. Output: Forecast the pattern of future incumbent transmission	State: (max CPU and bw on each node, residual CPU and bw on each node, virtual node CPU and bw requirements, remaining virtual nodes to be placed), Action: physical node to embed current virtual node
	Loss function /Reward	Cross-entropy	Silhouette Coefficient and the sum of squared error (SSE) for selecting the number of clusters. --- Reward of the DRL- The system cost (weighted sum of energy consumption and task execution latency) of computing the tasks locally or at the edge	Technology Recognition: Binary Cross-Entropy Spectrum Usage Pattern Predictor: N/A	Shaped reward that combines acceptance ratio, revenue, cost, load balancing, and eligibility traces
	Baseline Comparison	IEEE802.11 DCF implemented in ns-2 simulator with basic access and with four-way handshake RTS/CTS/DATA/ACK	Heuristics 1. Completely local computation 2. Completely edge computation 3. Greedy (minimize the system cost)	Not provided	MCTS (MCVNE), relaxed MILPs (R-Vine, D-Vine), NodeRank, GRC
	Limitations of ML vs. benchmark	performance gain increases with an increase of the number of STAs (N) while the inference accuracy decreases with an increase of N.	Not provided	No optimality guaranteed	The proposed algorithm requires lots of computational resources for its parallel implementation (24 actor-critic network pairs), trained with a fixed flow mapping method (shortest path or path selection)
	Advantages of ML vs. benchmark	The colliding transmissions can be rescheduled, improving the overall throughput (performance gain w.r.t. standard IEEE802.11 DCF)	adaptability to varying network environment; independent decision-making	Technology Recognition - To work with Fast Fourier Transform (FFT) instead of raw IQ samples reduces the number of samples to be processed in the TR - Spectrum Usage Pattern Predictor - The CTW has low time and space complexities with theoretical performance guarantees; the algorithm does not require offline training	Improved performance and adaptability w.r.t. varying VN request types
	Optimality Gap	Not provided	Not provided	Not provided	Not provided
	Trade-off ML vs benchmark w.r.t. performance	N/A	the DRL optimizes the system cost but does not outperform baselines w.r.t. task execution latency and energy consumption.	The execution time of the two-step approach can be easily implemented in a RAN Intelligent Controller (RIC).	N/A

	Bib key	wangmao2021	jiayang2021	nakashimakamiya2020	xucheng2018	manousis2021
Machine-Learning approach	**ML Method**	RL	RL	RL	UL	UL
	ML Problem	Control	Control	Decision Making	Clustering	Prediction
	ML Algorithm	Double Deep Q-network	A3C	Deep Q-Learning; Graph Convolutional Neural Network	Non-parametric Bayesian Model	Gaussian Processes
	Resource-awareness	No	No	No	No	No
	Model description	State: (initial, occupied, reserved resources of each DC, similar state of each link, features of current SFC), Action: (a,b) pair where a is the DC for actual deployment and b is the DC for the standby SFC instance	State: (VNF type, number of VNFs remaining in the chain, length of SFC chain, VNF computation load, remaining length of the chain, remaining time to deadline, VNF types that nodes maintain, advanced time on each node), Action:(defer rate, i.e., the probability that a VNF scheduling will be deferred for the next scheduling event)	State: adjacency matrix and current channel allocation. Actions: new channel allocation	Input: Timeseries of spectrum sensing data of different Secondary Users (SUs). Output: Number of spectrum states	Not provided
	Loss function /Reward	{1, if SFC is placed, -1 otherwise}	based on whether the SFC execution occurred prior to its deadline	The reward function is the average throughput of the lower 40% APs	N/A	N/A
	Baseline Comparison	Random greedy, best-fit greedy, near optimal sorting greedy, deep q network	Earliest Finish First, Earliest Start First, DQN	1. Random Allocation 2. DQN + CNN 3. Potential game-based	1. Energy Detection 2. Gaussian Mixture Model - Expectation Maximization 3. Gaussian Mixture Model - Bayesian Information Criterion 4. Mean Shift	N/A
	Limitations of ML vs. benchmark	offline training, neural networks reduce explainability	A3C typically trains multiple agent workers to improve stability, which requires many computational resources. In addition, the proposed method uses RL not to determine the placement decision, rather to assign a probability for deferring the execution of a VNF	Not provided	The spectrum sensing performance degrades when more Primary Users (PUs) are present for all proposed methods	N/A
	Advantages of ML vs. benchmark	fast decision making	faster convergence compared to the DQN, improved acceptance rate	Using GCN instead of CNN the learning performance improves; the proposed method allocates channel in shorter timesteps with larger cumulative rewards.	proposed method is more robust: it takes advantages of the spatial-temporal characteristics of the timeseries data	N/A
	Optimality Gap	slightly worse than a near-optimal method	at least 5% better in terms of reliability	No optimal results are available	Not provided	N/A
	Trade-off ML vs benchmark w.r.t. performance	N/A	N/A	It achieves better reward than the immediate reward maximization method achieving better allocations	N/A	N/A

	Bib key	perinoyang2020	iyerli2018	navarrorossi2020	kattadigeraman2021	manglahalepovic2020
Machine-Learning approach	ML Method	SL	SL	UL	SL	SL
	ML Problem	Classification	Classification and regression	Classification	Classification	Classification
	ML Algorithm	XGBOOST	Multi-task learning with ensembles	Anomaly detection algorithms (based on distance, density, clustering, subspace) both for batching and stream and feature scoring algorithms. It then builds a pipeline composed of AD->FS->expert knowledge->human labeling->ticketing	XGBOOST	SVM, k-NN, XGBoost, Random Forest, and Multilayer Perceptron
	Resource-awareness	No	No	No	No	No
	Model description	Input: features built on alarms from cell site and additional information (e.g., weather, location, power supply type) Output: failure permanent or temporal.	Input: features built on bearer records, signaling records, TCP flow level statistics, network elements records. Output cell drop rate classification or throughput prediction (potentially applicable to other problems)	Input: router KPIs. Output: tickets for anomalies	Input: features derived from packet level traces or TCP flow level information and video session level. Output: is the flow a 360-video streaming or regular streaming	Input: features derived from TLS flow level information and video session level. Output: target QoE metric (i.e., video quality, rebuffering ratio, combined QoE)
	Loss function /Reward	N/A	Custom loss function to consider a weighted several parameters (defined by the PCA). It allows for a mixed online-offline approach.	N/A	N/A	N/A
	Baseline Comparison	1. Waiting a fixed 24h time threshold to define a failure permanent. 2. LSTM. 3. changing the time threshold based on experience derived from the model (i.e., 6h, 12h, 18h). 4. probability to define a threshold permanent proportional to failure past duration	1. per base station modelling 2. different spatial grouping methods for cells 3. grouping only	The paper compares the 10 AD algorithms and the 10 FS algorithms.	1. Heuristics based on threshold on input fields. 2. Different ML models as CNN, Multi-layer Perceptron, KNN, naive Bayes	1. Comparison of the different ML methods 2. comparison of the proposed technique to ones based on packet level traces (still based on ML techniques)
	Limitations of ML vs. benchmark	The ML approach is more expensive than heuristics	It requires a more complex design than the benchmarks	N/A	The ML approach is more expensive than heuristics based on thresholds on input fields	The approach based on packet level is 5-7% superior. But it is also based on ML.
	Advantages of ML vs. benchmark	The helped deriving the heuristics and still provides benefits w.r.t. them. The approach is simpler and less computational expensive than LSTM while providing the same benefits.	Superior performance with limited data available. Best delay/performance tade-offs	N/A	The model based on XGBOOST is superior than heuristics, and is the one performing the best among different other models tested. Similar performances are only provided by CNN but VNN is more complex.	The approach based on flow level information is amenable to actual usage, while packet level traces one does not scale.
	Optimality Gap	Not provided	Not provided	Not provided	Not provided	Not provided
	Trade-off ML vs benchmark w.r.t. performance	N/A	N/A	N/A	N/A	N/A

	Bib key	lui2019	lossleap2021	tesrnn2021
Machine-Learning approach	**ML Method**	RL	SL	SL
	ML Problem	Control	Loss-Metric Mismatch	Control
	ML Algorithm	Deep Q-Learning	Deep Neural Networks	Recurrent Neural Networks with LSTM
	Resource-awareness	No	No	No
	Model description	N/A	Input: Past states and decisions Output: Next action	Input: Past traffic Output: Traffic / Capacity / resources forecast
	Loss function /Reward	Weighted reward representing the monetary revenue of the Edge operator (accounting for spectrum leasing, computation resources, rate, client cost, etc)	Self-learned	MSE of forecasted traffic or SLA violations
	Baseline Comparison	N/A	ML with Standard loss functions (MAE, MSE,) ML with Expert-defined loss function	N/A
	Limitations of ML vs. benchmark	N/A	Slightly more complexity	N/A
	Advantages of ML vs. benchmark	N/A	It is able to characterize a complex or even unknown loss function during training. This knowledge can be transferred to other cases	Adapts to data with high variation and close-to-zero values. Stability
	Optimality Gap	Not provided	No optimal. It can perform better than the "trainer" for standard loss functions (i.e., the case where the predictor is trained with MSE or MAE)	Not provided
	Trade-off ML vs benchmark w.r.t. performance	N/A	For complex or unknown loss functions, a small increase of complexity provides significant gains	N/A