Design, Development and Orchestration of 5G-Ready Applications over Sliced Programmable Infrastructure

Panagiotis Gouvas, Anastasios Zafeiropoulos, Constantinos Vassilakis, Eleni Fotopoulou, George Tsiolis
Ubitech Ltd, Research Department
Athens, Greece
{pgouvas, azafeiropoulos, cvassilakis, efotopoulou, gtsiolis}@ubitech.eu

Roberto Bruschi, Raffaele Bolla, Franco Davoli
S3ITI Federated National Laboratory,
National Inter-University Consortium for Telecommunications (CNIT)
Genoa, Italy
{roberto.bruschi, raffaele.bolla, franco.davoli}@cnit.it

***Abstract*—5G networks design and evolution is considered as a key to support the introduction of digital technologies in economic and societal processes. Towards this direction, vertical industries' needs should be considered as drivers of 5G networks design and development with high priority. In the current manuscript, MATILDA is presented, as a holistic 5G end-to-end services operational framework tackling the overall lifecycle of design, development and orchestration of 5G-ready applications and 5G network services over programmable infrastructure, following a unified programmability model and a set of control abstractions.**

Keywords: 5G,5G-ready applications, NFV orchestration, network slice.

I. Introduction

5G networks design and evolution is considered as a key to support the introduction of digital technologies in economic and societal processes, leading to the fourth industrial revolution impacting multiple sectors, as stated in the "5G empowering vertical industries" report provided by the 5G-PPP association [1]. The integration of verticals is considered as one of the key differentiators between 4G and 5G systems to open truly global markets for innovative digital business models.

However, to achieve the integration of verticals over a highly evolving and heterogeneous networking and cloud computing ecosystem, vertical industries' needs should be considered as drivers of 5G networks design and development with high priority. Given that 5G networks will be the platform enabling growth in many industries (e.g. manufacturing, media, automotive, IoT, smart cities), the set of services per vertical industry are going to cater for a diverse set of requirements in terms of provisioning and management of infrastructure resources [3]. Such requirements may span from access network requirements (e.g. demand for mobile broadband, massive machine-type communications, low latency in the order of few milliseconds, high reliability) to a set of deployment constraints in the edge, transport and core infrastructure (e.g. trusted computing, increased isolation, end-to-end Quality of Service (QoS) requirements).

To fulfil the set of identified requirements and -in parallel- take into account the peculiarities associated with each vertical industry, 5G networks have to be operated by intelligent orchestration platforms able to support end-to-end applications and services provision over a programmable network, compute and storage infrastructure [2]. By leveraging virtualization and softwarization technologies, developers and operators have to better match needs and capabilities, building application-aware networks and network-aware applications. This joint expressive power is considered to be one of the main drivers of innovations enabled by 5G, as stated at the "View on 5G Architecture" of the 5G PPP Architecture Working Group [2]. Towards this direction, in all the network parts, there is a transition from today's "network of entities" to a "network of (virtual) functions" approach, where a network service can be dynamically composed on an "on-demand", "on-the-fly" basis [2]. Such a composition has to be realized taking into account requirements denoted on behalf of the vertical industries and facilitated by a service-specific grouping of network functions to logical entities and the mapping of logical to physical architecture which is in full accordance with the envisioned ETSI Network Functions Virtualization (NFV) architecture framework [4].

The term network slice is introduced to serve such a diverse ecosystem. A network slice regards the part of the infrastructure (consisting mainly of virtualized resources and virtual network functions without excluding the support of physical network functions where required) that aims to support a set of services and meet the desired Key Performance Indicators (KPIs) of the service providers. A network slice is created on demand based on the available network, compute and storage resources and the requirements imposed on behalf of the services provider. Over a network slice, a set of infrastructure management and application oriented orchestration services are provided [5].

Following the aforementioned notation, a layering of the diverse part of a 5G ecosystem is defined [1], as follows. The Business Service Layer defines and implements the business processes of the verticals along specific value chains. The Business Function Layer contains sets of application-related functions that are defined based on application requirements of the specific vertical sector. The Multi-service Control layer performs the mapping between the business service requirements and the network service topology and configuration. Finally, the infrastructure layer consists of the data communication network spanning all network segments to provide end-to-end connectivity services and a set of cloud computing and storage resources.

Following the provided layered approach for an integrated 5G architecture, it is evident the need for the provision of a 5G services end-to-end holistic operational model following a top down approach, where business requirements are translated to on demand creation and management of a network slice over the available programmable infrastructure. Such a model should support the design and development of 5G-ready applications. By 5G-ready applications we refer to network-aware and reconfigurable-by-design applications, able on one hand to denote network/infrastructure-specific requirements that have to be fulfilled and on the other hand being independently orchestratable and self-adaptable according to the deployment environment conditions. The deployment and management aspects of these applications per industry vertical should trigger the provision and configuration of a network slice, allowing in this way the creation of application-aware network slices.

II. CHALLENGES AND MATILDA VISION

Several approaches have been proposed for tackling instantiation and management of network services following ETSI NFV specifications. However, to the best of our knowledge, there is no holistic approach interconnecting this orchestration with the business services (5G-ready applications in MATILDA) deployment and orchestration requirements. This gap is associated with a set of barriers for real enablement of vertical industries acceleration through the exploitation of 5G infrastructures. Continuous evolving business and application-oriented requirements cannot be optimally mapped to the creation of the appropriate network slices in an automated way, negatively impacting in this way the overall quality of the services provision as well as the optimal usage of the allocated resources. To come up with a holistic approach for enhancing 5G with intelligent orchestration platforms able to support end-to-end 5G-ready applications and services provision over programmable infrastructure, a set of challenges that should be tackled have been identified, as follows:

- Define the appropriate abstractions for the design of 5G-ready applications for industry verticals able to take advantage of a 5G programmable infrastructure, rather than the development of generic applications/services that do not fully exploit new network capabilities and services for each vertical.
- Develop an agile programming and verification platform for designing, developing and verifying industry vertical 5G-ready applications and network services (NFV building blocks, VNFs and their managers) tailor made for distinct industry verticals. Such a platform should support also a set of repositories for making the developed software artefacts available for open-source or commercial exploitation.
- Support mechanisms for automated or semi-automated translation of application-specific requirements to programmable infrastructure requirements, triggering the setup of the appropriate application-aware network slices for supporting applications of an industry vertical in an optimal way. Such mechanisms have to follow an Infrastructure-as-a-Code approach and support the enablement of the appropriate infrastructure configuration functions on demand along with a granular customisability of a network slice per vertical.
- Support unified and intelligent orchestration mechanisms for managing the entire lifecycle of 5G-ready applications and network services, along with unified deployment and dynamic runtime policies enforcement schemes defined on behalf of the services providers, as well as a set of network-based machine learning mechanisms for advanced insights.
- Support mechanisms for multi-site network, compute and storage resource management supporting deployment and management of distributed applications' components, as well as network functions.
- Support flexibility in the design, development and deployment of 5G-ready applications and services in the various layers by different stakeholders. These stakeholders should be able to take advantage of Open Application Programming Interfaces (APIs) that should be available at different levels (resources, connectivity and service enablers) for managing applications and infrastructure-oriented services provision.
- Involve key actors of the value chain in the operational model (not only telecom operators, but also vendors, application developers, cloud service providers, infrastructure providers) along with the appropriate definition of their operational business boundaries and collaboration schemes.

The vision of MATILDA is to design and implement a novel holistic 5G end-to-end services operational framework tackling the overall lifecycle of design, development and orchestration of 5G-ready applications and 5G network services over programmable infrastructure, following a unified programmability model and a set of control abstractions. MATILDA aims to devise and realize a radical shift in the development of software for 5G-ready applications, as well as virtual and physical network functions and network services, through the adoption of a unified programmability model, the definition of proper abstractions and the creation of an open development environment that may be used by application as well as network functions developers. Intelligent and unified orchestration mechanisms are going to be applied for the automated placement of the 5G-ready applications and the creation and maintenance of the required network slices. Deployment and runtime policies enforcement is provided through a set of optimisation mechanisms providing deployment plans based on high level objectives and a set of mechanisms supporting runtime adaptation of the application components and/or network functions based on policies defined on behalf of a services provider. Multi-site management of the cloud/edge computing and IoT resources is supported by a multi-site virtualized infrastructure manager, while the lifecycle management of the supported Virtual Network Functions Forwarding Graphs (VNF-FGs), as well as a set of network management activities, are provided by a multi-site NFV Orchestrator (NFVO). Network and application-oriented analytics and profiling mechanisms are supported based on real-time as well as a posteriori processing of the collected data from a set of monitoring streams. The developed 5G-ready application components, applications, virtual network functions and application-aware network services are made available for open-source or commercial purposes, re-use and extension through a 5G marketplace.

III. MATILDA LAYERED APPROACH

As already mentioned, up to our knowledge, no holistic framework exists that supports the tight interconnection among the development of 5G-ready applications, the creation of the on-demand required networking and computational infrastructure in the form of an application-aware network slice and the activation of the appropriate networking mechanisms for the support of the industry vertical applications. MATILDA follows the layered approach presented in [1]; however, with the consolidation of some functionalities into specific layers. The MATILDA layers along with the main artefacts and key technological concepts comprising the MATILDA framework per layer are depicted in Figure 1.

The Applications layer corresponds to the Business Service and Business Function layer and regards the design and development of the 5G-ready applications per industry vertical along with the specification of the associated networking requirements. The Orchestration Layer regards the support of deployment and optimisation mechanisms of the 5G-ready applications over the available multi-site programmable infrastructure. Orchestration refers to both the application components and the attached virtual network functions and includes a set of intelligent mechanisms for optimal deployment, runtime policies enforcement, data mining and analysis and context awareness support. The Network Functions and Resource Management Layer regards the implementation of the resource management functionalities over the available programmable infrastructure, as well as the lifecycle management of the activated virtual network functions. The Infrastructure Layer consists of the data communication network spanning a set of cloud computing and storage resources.

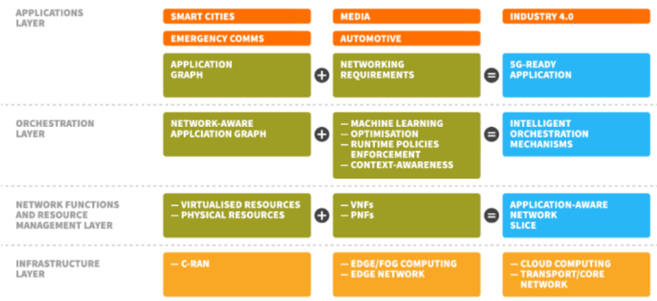

Figure 1. MATILDA 5G High Level Architectural Approach.

IV. MATILDA ARCHITECTURE OVERVIEW

MATILDA's vision will be realized through a consistent architecture that is depicted in Figure 2. The architecture consists of a set of complementary components with fine-grained interfaces that comprise the MATILDA Framework. The architecture follows the layering provided before in section III. The applications layer includes the MATILDA development environment comprising a web-based IDE, a graph composer and a set of metamodels. The orchestration layer includes the MATILDA orchestrator, while the network functions and resources management layer includes a set of mechanisms addressing the lifecycle management of VNF-FGs and the set of multi-site virtualized resources. The infrastructure layer consists of the data communication network and the reserved cloud computing and storage resources.

Prior to delving into the details of the framework's architecture we will clarify the concrete roles/stakeholders that are involved in the framework. The involved roles are five; namely:

- Infrastructure Provider: As infrastructure providers we refer a) to the telecommunication providers that operate a programmable (layer 2 to layer 4) infrastructure that may span from the radio-access to the edge, transport and core network and b) to cloud providers offering compute and storage services in a programmable way.

- Application Developer: As application developers we refer to the software engineers that develop 5G applications. To meet 5G requirements, applications have to be modular and chainable by-design, abstracted from physical and networking resources and reactive-by-design. Therefore, MATILDA introduces the notion of the chainable application component that adheres to a specific metamodel. The combination of several chainable application components formulates an application graph. During the creation of the application graph, the developers will provide, in a formal way, the networking requirements that should be satisfied when the application graph will be instantiated, making in this way the application 5G-ready. These requirements will be satisfied through the usage of specific VNFs and VNF-FGs.

- VNF/PNF Developer: As VNF/PNF Developers we refer to the software engineers that either implement from scratch or wrap existing network functions. These functions, when combined in the frame of VNF Forwarding Graphs, provide seamless layer 2 to layer 4 functionalities such as tunneling, flow prioritization, traffic shaping, L3 firewalling, etc. Such functions can be totally virtual (i.e. VNFs) or physical (i.e. PNFs) in the case of programmable networking equipment (e.g. cloud-RAN, openflow switches). Although the VNF ecosystem has already evolved, MATILDA will propose an augmented VNF/PNF metamodel -fully compatible with the ETSI NFV VNF and VNF-FG descriptors- which will be able to meet the application graph networking requirements mentioned above.

- Service Provider: As service provider we refer to the organizational entity that selects one of the available developed applications and instantiates it on top of reserved programmable infrastructural resources.

- Service Consumer: As service consumer we refer to the entity that is benefited by the operation of the instantiated application graph.

The top-down view of the roles provides also a time-based sequential ordering of the usage of the MATILDA framework. Infrastructure providers expose their programmable resources (physical and virtual), VNF/PNF developers implement and publish (to the MATILDA marketplace) their artefacts, application developers publish their applications in the form of application graphs that consist of chainable application components, and service providers select an application graph and concretize it through the selection of appropriate VNFs which can meet the specified networking requirements. The concretized graph, which is addressed as network-aware application graph, is deployed on top of the reserved resources using one of the multiple policies that may have been defined.

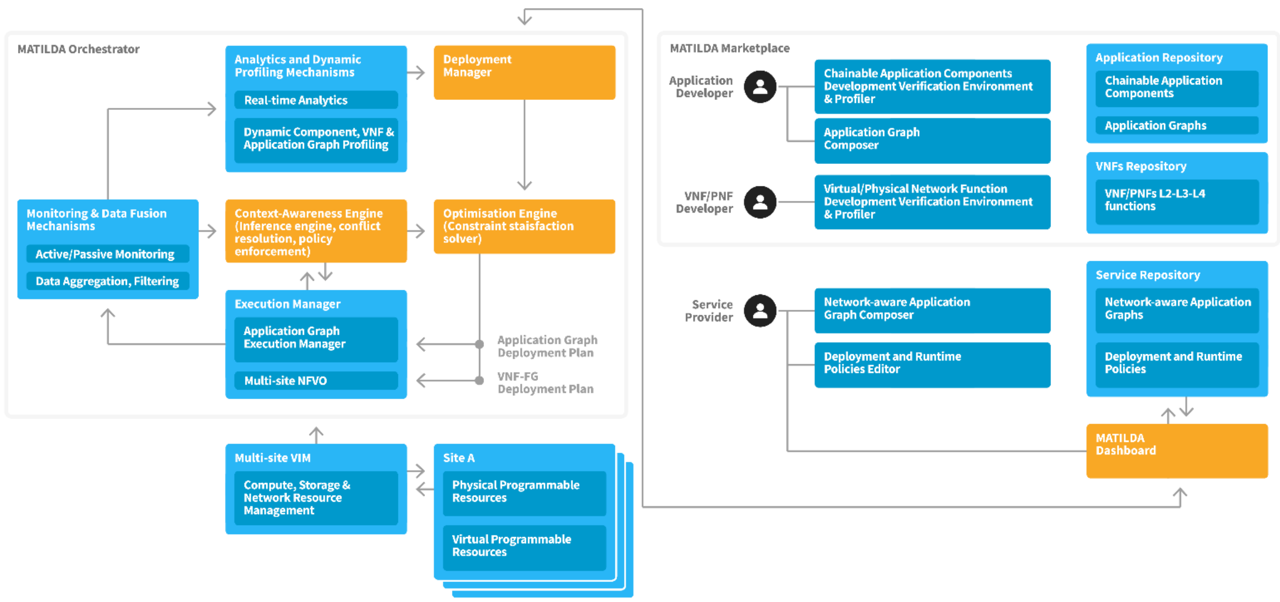

Figure 2. MATILDA 5G High Level Architectural Approach.

A. *MATILDA Metamodels and Development Environment*

MATILDA is going to provide a web-based development environment (open source Eclipse Che [6]) for developing chainable application components, virtual network functions and 5G-ready applications following a microservices-based development paradigm. The design and development of the diverse software components is going to be compatible with a set of defined MATILDA metamodels, namely the Chainable Application Components and 5G-ready Application Graphs metamodel, the VNF/PNF Metamodel and the Network-aware Application Graph Policy Metamodel.

As already mentioned, a key MATILDA concept regarding 5G applications is the chainable application component. MATILDA will provide a concrete metamodel of a 5G chainable component, which covers all characteristics that should be taken under consideration. These characteristics are grouped in facets (i.e. logical groups) that include: configurability, chainability, QoS awareness, programmability, scalability, networking and profiling aspects. Given that these components comply with the same metamodel, chaining can be easily performed based on the published model instances.

As already explained, the application graph consists of chainable components that adhere to a strict model. Yet the application graph per se cannot be deployed on top of virtualized resources since all the networking aspects related to the chainable application components are not taken under consideration. The 8-faceted rule-of-thumb that was defined above in the frame of the chainable application component metamodel is highly valid also in the case of VNF development. The existence of a formal chainability profile, the need of a clear configuration layer regarding mutable configuration parameters, the need of horizontal/vertical scalability by design and estimated resource consumption profile are totally valid as far as VNFs/PNFs are concerned, leading to the specification of the relevant VNF/PNF metamodel.

A network-aware application graph can be deployed on top of reserved resources in multiple ways, based on the defined policies on behalf of a services provider. For supporting this policies' definition, MATILDA entails a strict metamodel that will allow the creation of multiple rules that will be evaluated continuously. A policy may consist of multiple expressions and each expression combines several conditions that can trigger multiple actions. The conditions and the actions are bound to the models that have been analyzed above.

B. *Deployment Manager & Optimization Engine*

The Deployment Manager is responsible for the complex task of "translating" a deployment model into an optimal deployment plan taking under consideration: a) the available programmable resources, b) the current situation in the infrastructures where these resources reside, and c) the selected policy. The actual goal of the Deployment Manager is to select the proper compute resource which will host the virtual image (in the case of a chainable application component or a VNF) or the physical resource which will be configured according to a PNF. Beyond the proper compute resource selection, the Deployment Manager will infer, in the case of VNFs and chainable application components, the proper parameters, i.e. amount of VCPUs, memory and volume that should be allocated during instantiation. Finally, the Deployment Manager will select the type of virtualized execution environment that is appropriate for the instantiation.

The Deployment Manager consults the Optimization Engine for getting the optimal deployment plan. The latter is responsible for producing results in terms of optimized deployment plans to support both pro-active adjustment of the running configuration as well as re-active re-configurations of deployments, based on measurements that derive from the monitoring mechanisms. Thus, the ultimate goal of the Optimization Engine is to produce deployment plans to satisfy: i) zero-service disruption and ii) optimal configuration across time.

In MATILDA, the Deployment Manager and Optimization Engine will regard extensions of the components made available through the ARCADIA Orchestrator [9] for supporting optimal placement of distributed applications -denoted in the form of a service graph- over programmable infrastructure.

C. Multi-site Virtual Infrastructure Manager

The Multi-site Virtual Infrastructure Manager (hereinafter multi-site VIM) exposes a specific interface where programmable resources are registered and managed. Such resources include compute nodes (with given vCPU and memory), volumes, OpenFlow switches, Edge Devices (e.g. CPEs), etc. It relies on an abstraction layer where many infrastructure providers can plug their proprietary resource management interface through an adapter to this abstraction layer. The multi-site VIM will rely on the registered adapters in order on one hand to request/have a synchronized view of the programmable resources offered and their availability, which will feed to the MATILDA Orchestrator, in order to concretize the deployment plan and, on the other hand, to reserve the required resources for an application according to the plan, or release them when no longer needed.

For the sake of implementation, the multi-site VIM will provide de-facto interfacing with Openstack, OpenNebula, Google Cloud Compute and Amazon WS. The interfacing will rely on the abstraction layer that is already developed in the ARCADIA H2020 project [9]. However, one critical extension is the interfacing with edge/fog computing devices, which will be offered by the OpenVolcano platform [8], that is the open-source software platform under development in the INPUT H2020 Project [7] for fog computing [10].

D. Application Graph Execution Manager

As already discussed, the actual running service consists of instantiated VNFs/PNFs and instantiated chainable application components. From a technical point of view, both of these ‘elements’ are independent graphs that have different orchestration needs. However, the end-to-end graph has to remain consistent in terms of lifecycle management. Moreover, the entire service should have an observable state, i.e. in each lifecycle state each node of the end-to-end graph should have a consistent state. The component that maintains the global state of the network-aware application Graph is the Execution Manager. As such, it handles the dependencies of the entire graph. Therefore, it acts as a coordination entity between the multi-site NFVO that handles VNF Forwarding Graphs and another component, which is addressed as Application Graph Execution Manager and manages the lifecycle of the chainable application components.

For the sake of clarity, it should be noted that, upon an initial deployment, the Application Graph Execution Manager is in a blocking mode until the application-aware network slice is created. This means that the multi-site NFVO has to realize (successfully) the deployment of all the required VNF-FGs in order for the chainable application components to be placed. Finally, during policy enforcement the interfacing with a chainable application component is performed through the Application Graph Execution Manager, while the interfacing with the VNFs/PNFs is performed through the Multi-site NFVO.

E. Multi-Site NFV Orchestrator

The scope of the multi-site NFV Orchestrator (NFVO) is to perform basic management operations on the instantiated VNFs by communicating with a respective VNF Manager (VNFM). In parallel, the multi-site NFVO interacts with the multi-site VIM in order to make accessible the physical and virtual resources of the network infrastructures belonging to different infrastructure providers. The main VNF management functionality performed by the multi-site NFVO consists of basic life-cycle management of VNFs (create/update/delete), enhanced network-slice aware placement of high-performance NFV workloads (so as to make VMs run on the most suitable hardware platform and gain the maximum advantages from hardware features built into the system), health monitoring of deployed VNFs and, finally, VNF auto-healing/auto-scaling based on MATILDA policy metamodel.

The multi-site NFVO component will be developed on a basis of existing open source solutions that have to be clearly extended to meet the goals of MATILDA. In particular, VNF management functions will be inherited from Tacker [11], an official OpenStack project building NFV Orchestrator and Generic VNF Manager using standard based architectures and compliant with ETSI MANO requirements.

F. Monitoring, Analytics & Dynamic Profiling Mechanisms

After the deployment of a service, there are multiple chainable application components and VNFs/PNFs that are running. To this end, a set of data monitoring streams may be activated on demand per network-aware application graph. Such monitoring streams represent data regarding VNF Forwarding graph metrics (e.g. available bandwidth, end-to-end delay, download and upload speed, physical radio parameters), resources usage metrics from Chainable Components or VNFs (e.g. average CPU usage, memory usage) and chainable application component specific metrics (e.g. average response time, http requests).

Monitoring streams are going to be collected based on a) dedicated monitoring VNFs b) on the multi-site VIM interface (which provides execution-ware metrics) and c) on the chainable application component’s management interface. All the extracted measurements will feed the analytics toolkit. Prior to feeding such data to the analytics toolkit, the profiling component or the context awareness engine, a data aggregation and filtering component is responsible for building real-time streaming data pipelines that reliably get data and apply a set of transformations (e.g. estimation of rate of change of a metric, calculation of average values in specific time windows) or filtering functionalities. The Kafka distributed streaming platform [12] is envisaged to be used for this purpose.

In the analytics toolkit, a set of machine learning algorithms supporting predictive analytics, prescriptive analytics as well as clustering and classification algorithms will be deployed building upon big data analytics frameworks such as Spark [13]. Based on the portfolio of the supported algorithms, real-time profiling of the deployed application components and VNFs/PNFs will be also provided.

The extracted profile will be used by the Deployment Manager during the calculation of the placement options for the associated VNFs and chainable application components. This will significantly leverage the stability of the running service, since based on the extracted profiles the amount of mitigation actions, in case of SLA violation or performance degradation, will be minimized. For example, assume that a chainable application component (or VNF) is classified as 'almost linear' in terms of resource consumption to the served workload, and according to a high-level requirement it has to be deployed near a services consumption point. However, the installed VIM that exposes the resources that are near the edge reports a specific amount of resources, which are capable of temporarily hosting the chainable application component given the estimated or predicted future workload; yet, they are not enough to support scaling to meet the future demand. If the profile was not a-priori known the option to place it near the edge would be the de-facto decision. However, given the profile, the most appropriate placement decision may be taken.

G. *Context Awareness Engine*

The Context Awareness Engine (CAE) regards the entity responsible for managing the collected information/context by the various monitoring streams, the extraction of advanced information and insights upon reasoning over them, and the suggestion of actions based on the active policies per network-aware application graph. The CAE in MATILDA is providing policies enforcement over the deployed graphs following a continuous match-resolve-act approach. Specifically, the match phase regards the mapping of the set of applied rules, which are satisfied based on the data streams coming from the monitoring infrastructure, the resolve phase regards the process of conflict resolution -if any- among the satisfied rules taking into account the pre-defined salience of each rule, while the act phase regards the provision of a set of suggested actions by the CAE to the orchestration components; the Deployment Manager and the Execution Manager of the MATILDA Orchestrator, are responsible for network-aware application graphs placement and management, respectively. Policies enforcement is realized through a rule-based framework that attempts to derive execution instructions based on the current set of data and the active rules; rules are associated with the deployed service graphs at each point in time. The CAE consists of three basic functional components: a) the working memory, where facts based on the provided data are inserted, b) the production memory, where predefined-static rules that are bound to the policy exist, and c) an inference engine that supports reasoning and conflict resolution over the provided set of facts and rules, along with triggering of the appropriate actions. Policies enforcement mechanisms are going to be deployed based on Drools [14], which is an open-source, highly scalable Rules Management System (BRMS) solution.

H. *MATILDA Marketplace*

The MATILDA marketplace is offering a two-fold objective. On one hand, it acts as a centralised repository to provide access to the set of developed application components, graphs and VNFs/PNFs. Open-source releases of software and continuous update processes are going to be promoted, leading to a critical mass of popular chainable application components and VNFs/PNFs that may be used by application developers and service providers/telecom operators. On the other hand, the marketplace acts as a means to support service providers, network operators and software houses to commercialize new virtualized products and network-aware applications. Establishment of collaboration among specific stakeholders may be initiated through the marketplace, leading to customised software covering a service provider's needs.

V. Conclusions

In the current manuscript, a novel a holistic 5G end-to-end services operational framework is presented. The framework aims to tackle the overall lifecycle of design, development and orchestration of 5G-ready applications and 5G network services over programmable infrastructure, following a unified programmability model and a set of control abstractions.

Acknowledgment

This work is supported by the European Commission Research Programs through the MATILDA Project under Contract H2020-761898.

References

[1] 5G PPP Architecture Working Group , 5G empowering vertical industries, February 2016, Online: https://5g-ppp.eu/wp-content/uploads/2016/02/BROCHURE_5PPP_BAT2_PL.pdf

[2] 5G PPP Architecture Working Group, View on 5G Architecture, Online: https://5g-ppp.eu/wp-content/uploads/2014/02/5G-PPP-5G-Architecture-WP-For-public-consultation.pdf

[3] 5G Vision, The 5G Infrastructure Public Private Partnership: the next generation of communication networks and services, Online: https://5g-ppp.eu/wp-content/uploads/2015/02/5G-Vision-Brochure-v1.pdf

[4] ETSI Network Function Virtualization (NFV), Architectural Framework, Online: http://www.etsi.org/deliver/etsi_gs/nfv/001_099/002/01.01.01_60/gs_nfv002v010101p.pdf

[5] NGMN Alliance, Description of Network Slicing Concept, January 2016, Online: https://www.ngmn.org/uploads/media/160113_Network_Slicing_v1_0.pdf

[6] Eclipse Che, Online: http://www.eclipse.org/che/

[7] EU project INPUT, Online: http://www.input-project.eu/

[8] R. Bruschi, P. Lago, C. Lombardo, and S. Mangialardi. OpenVolcano: An Open-Source Software Platform for Fog Computing, Programmability for Cloud Networks and Applications 2016 (PROCON 2016), Würzburg, Germany, (September 2016)

[9] EU project ARCADIA, Online: http://www.arcadia-framework.eu/

[10] F. Bonomi, R. Milito, J. Zhu, S. Addepalli. Fog Computing and Its Role in the Internet of Things, Proc. of ACM MCC 2012, Helsinki, Finland, August 2012.

[11] Tacker, Online: https://wiki.openstack.org/wiki/Tacker

[12] Apache kafka, Online: https://kafka.apache.org/

[13] Apache Spark, Online: http://spark.apache.org/

[14] Drools Business Rules Management System, Online: https://www.drools.org