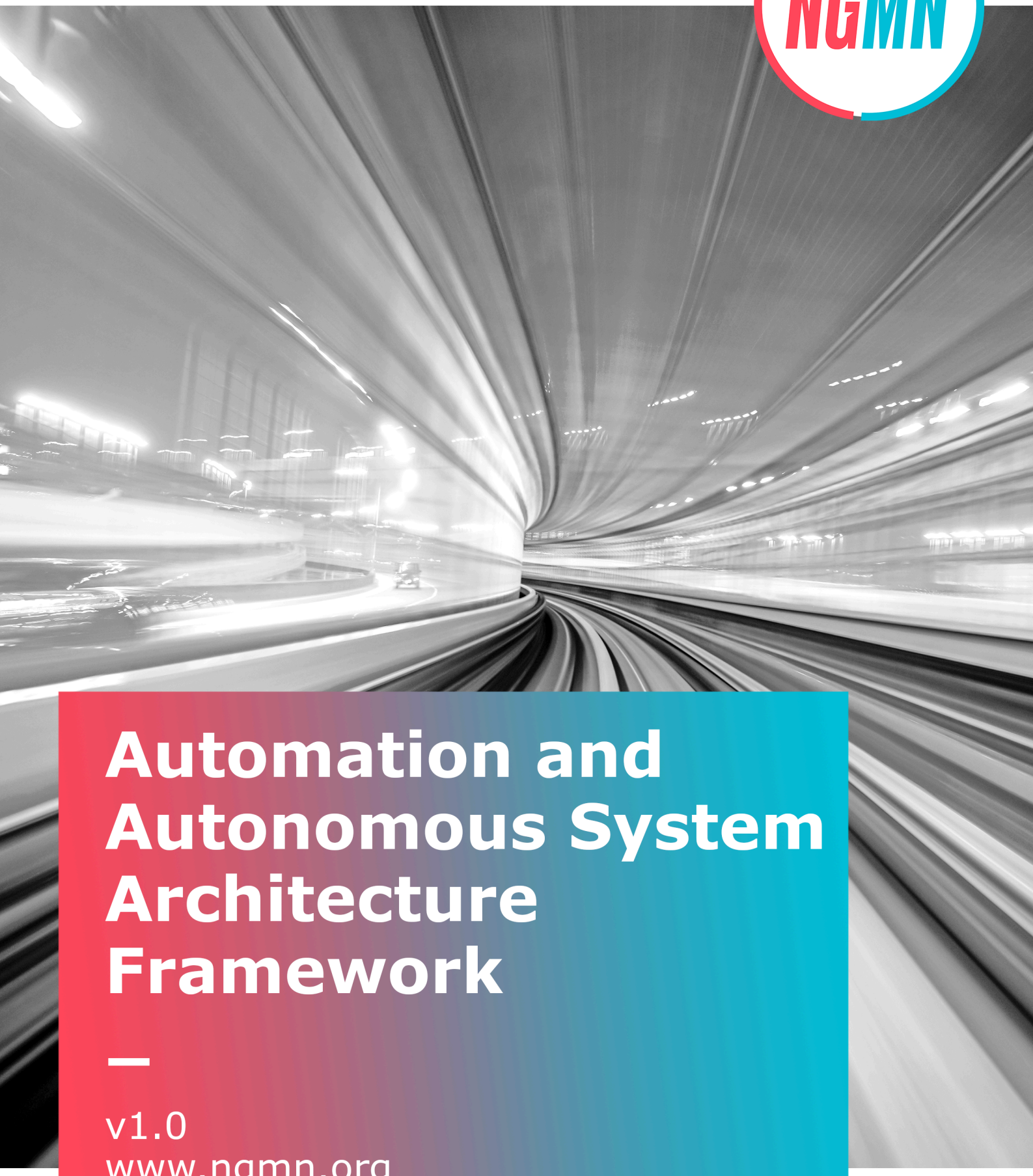


Automation and Autonomous System Architecture Framework

—

v1.0

www.ngmn.org

WE MAKE BETTER CONNECTIONS

Automation and Autonomous system Architecture Framework

by NGMN Alliance

Version:	1.0
Date:	28.10.2022
Document Type:	Final Deliverable (approved)
Confidentiality Class:	P - Public
Authorised Recipients: (for CR documents only)	

Project:	Network Automation and Autonomy Based on AI
Editor / Submitter:	**Sebastian Thalanany (UScellular)**
Contributors:	**Yuhan Zhang (China Mobile Research Institute), Lingli Deng (China Mobile Research Institute), Tony Verspecht (Cisco), Roberta Maglione (Cisco), Andreas Volk (HPE), Andreas Krichel (HPE), Jean Paul Pallois (Huawei), Luigi Licciardi (Huawei), Paolo Volpato (Huawei), Gary Li (Intel), Sebastian Robitzsch (Interdigital), Benoit Radier (Orange), Paul Edward Alvarez (Smart), Sebastian Thalanany (UScellular), Manchang Ju (ZTE), Liya Yuan (ZTE)**
Approved by / Date:	**<NGMN Board / 28 October 2022 >**

Abstract

This document describes a high-level framework, in terms of entities and functions that characterise autonomous system capabilities with an E2E (end-to-end) system perspective. Architectural considerations, associated with autonomous system capabilities, endowed with Artificial Intelligence/Machine Learning (AI/ML) models of cognition and application are delineated as high-level requirements. The term "system" is an abstraction, which generalizes and subsumes details such as specific networks, protocols, and implementations, in terms of high-level requirements, perspectives, and insights.

The E2E system imbued with autonomous system capabilities consists of a virtualized environment, with network slicing as a foundational building-block, for the realization of flexible, granular, and optimized allocation of system resources, such as computing, networking, and storage, for enabling network automation, without human intervention. The application of assorted AI/ML models, facilitate autonomous system behaviours to suit diverse deployment arrangements.

The architectural framework is intended to serve as guidance in the development of inter-operable and market enabling specifications, for a continuing advancement of the 5G ecosystem of heterogeneous access, virtualization, forward-looking service enablers, and emerging usage scenarios.

Contents

1 INTRODUCTION

The purpose of this document is to infer and delineate a high-level framework of architectural principles and requirements. Inferences from a survey of network service providers, on network automation and autonomy based on AI/ML [Annex – Summary of survey], serve as a motivation. The objective is to provide guidance and direction for NGMN Partners and standards development organisations for the articulation of interoperable capabilities, enablers, and services, associated with network automation and autonomous systems. It builds on the architectural concepts and emerging directions in the industry, with respect to the various aspects of autonomous systems for enabling end-to-end network automation, beyond the methods of simple automation, where human intervention is required for system and service configuration, and operation.

The journey from simple automation to an autonomous system rendered zero-touch automation is examined through a system-wide lens. The objective is to advance the promise of a continuously emerging service paradigm, which is enabled through a service-based framework of virtualization, cognitive awareness, and flexible levels of distribution. These aspects facilitate the customization and optimization of operations and capital expenditures, to suit different deployment objectives and business models, while promoting a personalized and experiential service quality.

This document describes the various aspects of an autonomous system, enabled through the use of AI/ML models. The context is an evolving network sliced, distributed, cloud-native, and advancing 5G ecosystem, for realizing an automation of network operations, with limited or no human intervention.

2 REFERENCES

[1] ISO, “Information technology - Artificial intelligence - Artificial intelligence concepts and terminology”, ISO/IEC 22989:2022

[2] RFC 7575, “Autonomic Networking: Definitions and Design Goals”, IETF, June 2015

[3] ETSI, “Experiential Networked Intelligence (ENI); ENI Definition of Categories for AI Application to Networks,” GR ENI 007 V1.1.1, November 2019.

[4] TM Forum, “Autonomous Network Levels Evaluation Methodology,” IG1252, v1.0.0, July 2021

[5] NGMN, “5G End-to-End Architecture Framework”, v4.31, November 2020

[6] TM Forum, “AI Closed Loop Automation – Anomaly Detection and Resolution,” IG1219, v2.1.0, November 2021

[7] ETSI, “GANA - Generic Autonomic Networking Architecture”, White paper No. 16, October, 2016

[8] TR 103.626 V1.1.1, “Autonomic network engineering for the self-managing Future Internet”, ETSI February 2020

[9] O-RAN Alliance, “O-RAN Architecture Description,” February 2020

[10] Fettweis, G., “The Tactile Internet: ITU-T Technology Watch Report”, August 2014. (Link: https://www.itu.int/dms_pub/itu-t/oth/23/01/T23010000230001PDFE.pdf)

[11] IBM “An Architectural blueprint for autonomic computing”, Whitepaper, June 2006.

[12] NGMN, “Description of Network Slicing Concept”, v1.0.8, September, 2016

[13] Ordonez-Lucena, J., Ameigeiras, P., Lopez, D., Ramos-Munoz, J.J., Lorca, J., Folgueira, J., “Network slicing for 5G with SDN/NFV: Concepts, architectures, and challenges,” IEEE Communications Magazine, May 2017

[14] ONF, “Applying SDN architecture to 5G slicing,” Tech. Rep. TR-526, Apr. 2016.

[15] Chen, Z., Wen, J., Geng, Y., “Predicting future traffic using hidden Markov models”, in Proceedings. IEEE Conference on Network Protocols, November 2016

[16] Cao, B., Zhang, L., Li, Y., Feng, D., Cao, W., “Intelligent offloading in multi-access edge computing: A state-of-the-art review and framework,” IEEE Communications Magazine, March 2019.

[17] Zhang, C., Patras, P., Haddadi, H., “Deep learning in mobile and wireless networking: A survey”, IEEE Communications Surveys & Tutorials, September 2019.

[18] Chowdhary, M.Z., Hossan, M.T., Jang, Y.M., "Applying Model-Free Reinforcement Algorithm in Network slicing for 5G", International conference on science, engineering, and robotics technology, 2019.

[19] Coutinho, R.W.L., Boukerche, A., "Transfer Learning for disruptive 5G-enabled Industrial IoT", IEEE Transactions on Industrial Informatics, June 2022.

[20] Zhang, H., et al., "5G wireless network: MyNET and SONAC," IEEE Networks, July 2015.

[21] Benzaid, C., Taleb, T., "AI-Driven Zero Touch Network and Service Management in 5G and Beyond: Challenged and Research Directions", IEEE Network, March/April 2020.

[22] ITU-T, "Distributed ledger technologies: Use cases", Technical paper, HSTP.DLT-UC, October 2019

[23] NGMN, "5G security recommendations package# 2: Network Slicing," v1.0, April 2016

[24] NGMN, "Description of network slicing concept," v1.0.8, October 2016

[25] Kemmoe, V.Y., Stone, W., Kim, J., Kin, D., Son, J., "Recent Advances in smart Contracts: A Technical Overview and State of the Art,", IEEE Access, July 2020

[26] Rivest, R.L.; Adleman, L.; Dertouzos, M.L., "On data banks and privacy homomorphisms," Foundations of Secure Computation," 1978

[27] IETF, "A Reference Model for Autonomic Networking," RFC8993, May 2021

[28] TM Forum, "Autonomous Networks Technical Architecture," IG1230, v1.1.0, January 2021

[29] 3GPP, "Management and orchestration; Management services for communication service assurance; Requirements," TR28.535 V17.5.0

[30] ETSI, "Zero-touch network and Service Management (ZSM); Closed-Loop Automation; Part 1: Enablers," GS ZSM 009-1 V1.1.1, June 2021.

[31] IETF, "Segment Routing over IPv6 (SRv6) Network Programming", RFC 8986, February 2021

[32] ETSI, "IPv6 Enhance Innovation (IPE); Gap Analysis", GR IPE 001 V1.1.1, August, 2021

[33] ETSI, "Autonomic network engineering for the self-managing Future Internet (AFI); Generic Autonomic Network Architecture; Part 2: An Architectural Reference Model for Autonomic Networking, Cognitive Networking and Self-Management." TS 103 195-2 V1.1.1, May 2018

[34] ETSI, "Experiential Networked Intelligence (ENI); Terminology for Main Concepts in ENI,", GR ENI 004 v2.2.1, December 2021

[35] TM Forum, " TAM – Telecom Applications Map by SDN/NFV Suite, " IG1130, v4.0.0, June 2019

[36] ETSI, "Experiential Network Intelligence (ENI); System Architecture," GS ENI 005, December 2021

[37] ETSI, "Federated GANA Knowledge Planes (KPs) for Multi-Domain Autonomic Management & Control (AMC) of Slices in the NGMN 5G End-to-End Architecture Framework," TR 103 747, v1.1.1, November 2021.

[38] RFC8345: "A YANG Data Model for Network Topologies", IETF, March, 2018.

[39] RFC 6020, " YANG – A Data Modeling Language for the Network Configuration Protocol (NETCONF), IETF, October, 2010

[40] RFC 4741, " NETCONF Configuration Protocol", December 2006

[41] NGMN, "A Network Data Layer Concept for the Telco Industry," August 2018.

[42] ETSI, "NFV architecture; Report on the Enhancement of NFV architecture towards Cloud-native and PaaS", GR NFV-IFA 029, v3.3.1, November 2019.

[43] Wiggins, A., "The Twelve Factor App", Online: https://12factor.net/

[44] NGMN, "NGMN Cloud Native Enabling Future Telco Platforms v5.2", May 2021.

[45] TS 28.540, "Management and orchestration; Network; 5G Network Resource Model (NRM); Stage 1", 3GPP, v17.2.0, December, 2021

[46] TS 28.541, "Management and orchestration; Network; 5G Network Resource Model (NRM); Stage 2 and Stage 3", 3GPP, v18.0.0, June 2022

[47] ETSI, "Report on Enabling Autonomous Management in NFV-MANO," ISG NFV-IFA 041 V4.1.1 August 2021.

[48] 3GPP, "Study on scenarios for Intent driven management services for mobile networks", TR 28.812 V0.17.0

[49] 3GPP, "Intent driven management services for mobile networks", TS 28.312 V1.0.0

[50] TM Forum, "Intent in Autonomous Networks," IG1253, v1.3.0, May 2022

[51] Truong, A., Walters, A., Goositt, J., Hines, K., Bruss, C.B., Farivar, R., "Towards Automated Machine Learning: Evaluation and Comparison of AutoML Approaches and Tools," IEEE 31st International Conference on Tools with Artificial Intelligence (ICTAI), November 2019

[52] OSM, "Open source MANO VNF onboarding guidelines,", Whitepaper v1.0, June 2019.

[53] Cotroneo, D., Simone, L.D., Natella, R., "Dependability Certification Guidelines for NFVIs through Fault Injection," IEEE International Symposium on Software Reliability Engineering Workshops.

[54] Fisher, M., Collins, E.M., Dennis, L.A., Luckcuck, M., Webster, M., "Verifiable Self-Certifying Autonomous Systems," IEEE International Symposium on Software Reliability Engineering Workshops

[55] TMForum, "Blockchain-based Telecom Infrastructure Marketplace", https://www.tmforum.org/blockchain-based-telecom-infrastructure-marketplace/

[56] TMForum, https://www.tmforum.org/vertical-industry-telcos-federated-dlt-based-marketplace/

[57] Rasheed, A., San, O., Kvamsdal, "Digital Twin: Values, Challenges and Enablers From a Modeling Perspective," IEEE Access, February 2020

[58] Polychronopoulos, D., Dahle, Y., Reuther, K., "Exploring the Core Values of Entrepreneurs: A Comparison to the United Nations 17 Sustainable Development Goals," February 2021.

[59] Hu, Y., Li, D., Sun, P., Wu, J., "Polymorphic Smart Network: An Open, Flexible and Universal Architecture for Future Heterogeneous Networks", IEEE Transactions on Network, Science, and Engineering, Vol. 7, No. 4, December 2020.

3 DEFINITIONS

Autonomic Function A function with intelligent and cognitive attributes, within an autonomous system, which operates through closed-loop feedback of a response for a given stimulus, for an automatic and adaptable behaviour (except for being subject to input governance policies and configuration), and is able to derive all the necessary information, through the discovery of knowledge within its environment.

AI and ML Model A model representing mathematical algorithms that learns using data and input consisting of human expertise to generate an effective and optimized decision, in the presence of dynamic change, when the model is provided with actual information of a corresponding nature for which the model was designed.

Machine Learning Model A model created by a machine through an application of learning techniques on input data. The model may be utilized to generate predictions (e.g., regression, classification, clustering etc.) on untrained or raw input data. Encapsulation of the model may be performed with software (e.g., within a virtual machine or container.). The learning techniques span a broad variety of algorithms (e.g., learning of a function that maps input data into corresponding output data).

Machine Learning Data Model This pertains to a description of the data used for data handling in machine learning applications. The data model may specify the data exchanged between an ML overlay network (e.g., virtualized network) and an ML underlay network (e.g., physical network). The data model includes data structures as well as a semantic description, while collecting data from an ML underlay network, and while applying the output from the ML overlay network to the ML underlay network [1].

4 AUTOMATION AND AUTONOMOUS SYSTEM CONTEXT

This chapter provides a brief overview of the context and background for guiding a consistent understanding of high-level considerations, without reference to specific implementations.

4.1 Overview of Network Automation

Network automation broadly consists of simple and cognitive automation. Simple automation is open-loop, with no adaptive feedback, and is rule based, requiring some human intervention for operation. Cognitive automation leverages adaptive closed-loop feedback, which characterizes an autonomous system, requiring no human intervention for operation. Autonomous systems enable cognitive automation and intent fulfilment, through an intrinsic exhibition of dynamic adaptation consisting broadly of self-Configuration, self-Healing, self-Optimizing, and self-Protecting (self-CHOP) characteristics, using suitable models of Artificial Intelligence/Machine Learning (AI/ML).

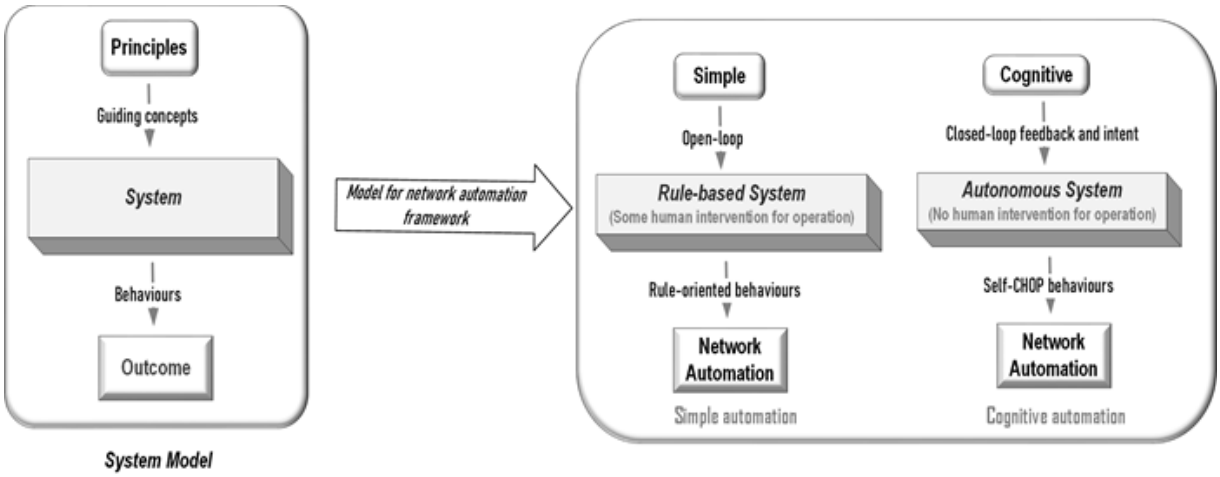


Fig. 1 : Model for Network Automation

The model for the high-level characteristics of simple automation and cognitive network automation, realized through autonomous system constructs, is depicted in Fig. 1.

4.2 Expected Benefits and Commercial Impact

Management of complexity and optimization is realized, through network automation. As a result, it is anticipated that network automation will serve as a catalyst for enabling service innovation, evolution, support for diverse business models, and flexible deployment.

At the same time network automation facilitates a minimization of operation and maintenance expenditures, as well as enabling continuous improvements in configuration, integration, upgrades, service experience, personalization, fault mitigation and management. Commercial beneficiaries of network automation include Network Service Providers (NSPs) (e.g., operators), Service Providers (SPs) (e.g., Verticals), and users (e.g., human and machine interfaces).

Network automation, enabled by the various modalities of simple automation and autonomous system rendered automation provides a holistic framework, for a dynamic adaptation of the system to a given environment, while satisfying diverse KPIs and a personalization of services. An Autonomous system rendered network automation framework provides the requisite operational capabilities to meet the growing system and service demands, which are most likely to exceed human response limits, as a result of the increasing system and service complexity that accrue with continuing technological advances (e.g., virtualization/softwarization, network disaggregation).

5 AUTONOMOUS SYSTEM ARCHITECTURE FOR AUTOMATION

The widespread adoption of virtualization, together with a decoupling of the control plane and the user plane, combined with network disaggregation, and a plurality of access networks consisting of terrestrial and non-terrestrial configurations, facilitate unprecedented levels of customization and flexibility for the Network Service Provider (NSP) and the Service Provider (SP) for 5G and future emerging systems. These evolving directions serve as a catalyst for advancing the service paradigm, with appropriate levels of capacity and coverage that promote a user-centric and personalized service experience.

In this continuing emergence and expansion of the service paradigm, these forward-looking directions permit an enablement of new business opportunities and business model innovation for customizing appropriate levels of capacity and coverage allocation, to suit the KPIs of a variety of emerging 5G and next-generation services. The ever-increasing demand for end-to-end system flexibility, and personalized service experience, require suitable enhancements to the computing, storage, and networking capabilities, while satisfying the attractive attributes of smaller, faster, and cheaper for the end-to-end system. As a consequence of the system-wide advancements, necessary to support an evolving service paradigm, there is a corresponding rise in system complexity, resulting from the interconnected and interdependent constituents of the system, operating within a given environmental context of humans and machines. Hence, the management of complexity is a critical system-wide requirement. With the continuing advancement of interconnected, interdependent, and interdisciplinary services that span the physical and digital worlds, human intervention is both limited and inadequate for managing the continuing rise of end-to-end system complexity.

5.1 Reference Architecture

Autonomic computing principles and constructs, which characterize an autonomous system, are indispensable for yielding sophisticated levels of self-organisation and adaptation to a given dynamic environment, for automating the end-to-end system operation for optimal behaviours, while effectively managing complexity, and satisfying performance. The distinction between “automatic” and “autonomic” is that the former refers to a predefined and programmatic process, while the latter refers to the various aspects associated with self-

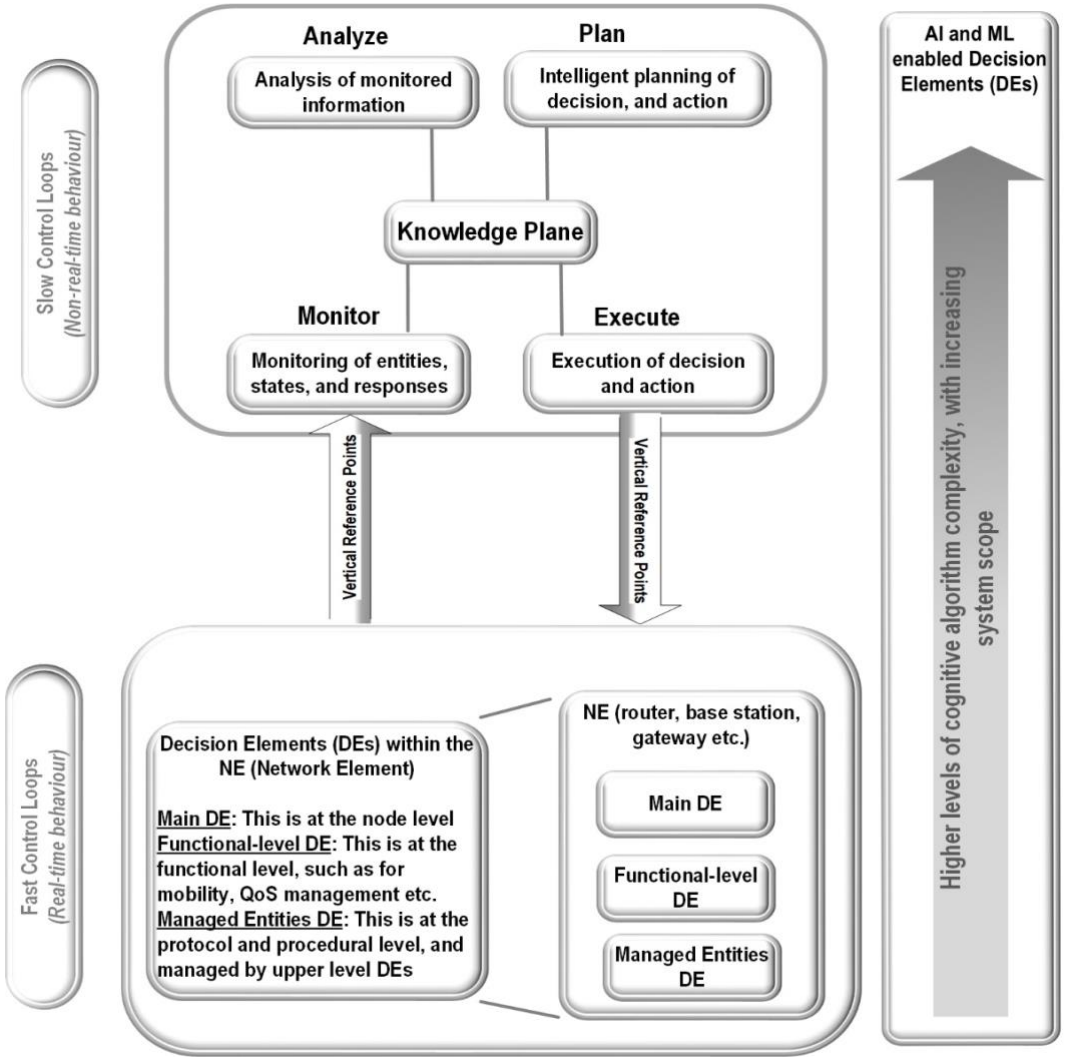
management. Typically, an automatic process operates in a given environment, with no awareness for adaptation without human intervention, if the environment changes. On the other hand, an autonomic process dynamically adapts to a changing environment, without human intervention [2].

The various levels of automation within a system are characterized in terms of the different categories of capabilities, such as those identified as an example below [3] [4]. This reflects an evolution of automation from open-loop models with human intervention to completely autonomous closed-loop models without any operational human intervention:

- Category 0. Manual O&M, which uses legacy interfaces (e.g., logs, alarms) with human intervention
- Category 1. Assisted O&M, which provides scripting level automation for provisioning etc.
- Category 2. Partial automation, which provides automation, with limited decision-making
- Category 3. Conditional automation, with autonomous behaviours within predefined limits of autonomy
- Category 4. High level of automation, which combines categories 2 and 3, across domains
- Category 5. Fully autonomic system, with end-to-end autonomicity for zero-touch automation.

The various aspects of autonomous level capabilities are associated with human and machine interfaces, in terms of decision-making depth and scope, collection, analysis, creation of system knowledge, and end-to-end system adaptability to a dynamic environment.

With a continuing evolution of the service paradigm, especially in the IoT and URLLC categories of services, there is significant potential for diverse services scenarios, over human and machine interfaces. In these scenarios, the self-CHOP behaviours of an autonomous system are pivotal for automating and choreographing the semantics, interoperability, flexibility, adaptability of the system to support a user-centric and personalized service experience. The ability of an end-to-end autonomous system to adapt dynamically and automatically to changes within the system or to changes in a given operational environment, characterizes its self-CHOP behaviour.



:

Fig. 2 Aspects of an autonomous system

The aspects of an autonomous system are represented in Fig. 2 in terms autonomic principles, fast and slow feedback control loops, cognitive network functions, and a shared repository of knowledge to facilitate self-CHOP behaviours [5].

The self-CHOP behaviours and the context awareness of an autonomous system, require the use of a knowledge base that serves as a repository of information associated with the end-to-end system and its environment. The ontological arrangement of shared information in a knowledge base is leveraged by the architectural framework, of the end-to-end autonomous system, for realizing system-wide autonomic processes that yield zero-touch automation.

The management of the rising complexity of heterogeneous and flexible end-to-end system configurations, requires feedback control loops that yield self-CHOP behaviours, through for example, an iterative process of Monitoring, Analysis, Planning, and Execution (MAPE) within the system, as well as in interactions with the environment. Conceptually this is similar to Observation, Orientation, Decision, and Action (OODA) [6]. For example, this adaptation is realized as responses to preserve expected end-to-end system operational behaviours, such as in the presence of diverse fault conditions, optimization of end-to-end network slice resources, service KPI assurance, user-centric personalization, trustworthiness, energy efficiency etc.

The use of a shared repository of knowledge, in the knowledge plane, based on the concepts in [7], enables an end-to-end system context awareness, which utilizes information sharing methods, such as Overlay Network Information eXchange (ONIX) [8], which provides the necessary information for adapting to the stochastic nature of wireless mobile networks and their dynamic environment, for flexible, open and scalable arrangement of NSA (Non Stand-Alone) and SA (Stand Alone) deployment configurations, including an integration with open radio access network directions [9]. With respect to open radio access network directions, the RAN Intelligent Controller (RIC) enables closed loop feedback to support near real-time (latency in seconds) and real-time (latency in milliseconds), with ultra-low latencies of the order of one millisecond for transmissions (e.g., tactile internet [9]). The use of ONIX as an information sharing method, facilitates the abstraction of technology and non-technology specific aspects, which enables a generic method for discovering specific information.
The Knowledge Plane (KP) in Fig. 2, is depicted at a high-level in Fig. 3.

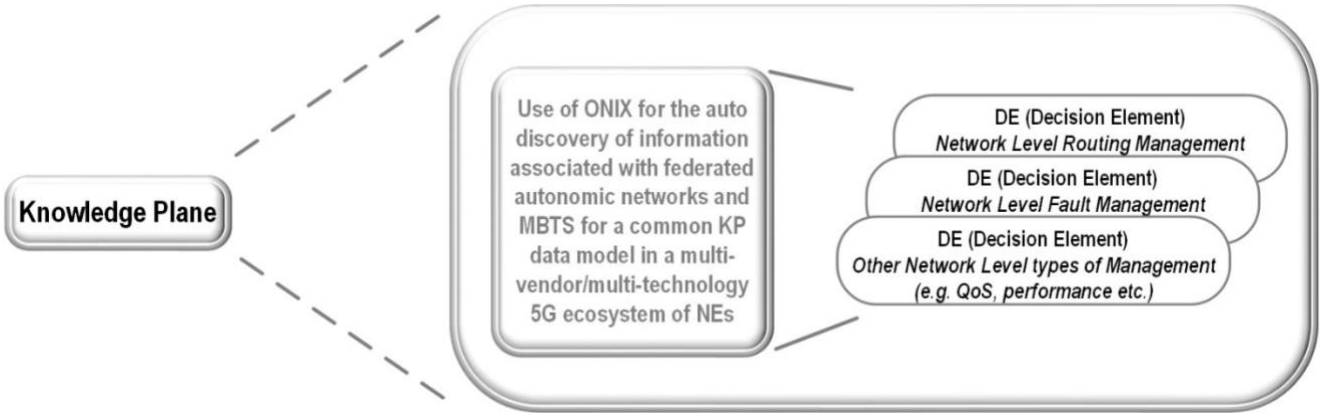


Fig. 3 : Knowledge Plane

5.2 System Characteristics and Context

The complex nature of a virtualized end-to-end system is characterized by divergent service requirements. This complexity is further advanced by distributed and flexible arrangements of networking, computing, and shared resources, while also satisfying diverse business and deployment objectives. An effective management of these growing complexities demands the use of an autonomous system framework for automating operations without human intervention. An autonomous system framework simultaneously manages system complexity, and enables system automation, through the establishment of self-CHOP behaviours.

The self-CHOP behaviours are realized through a harnessing of diverse fields of computing, inspired by the behaviour of the autonomic nervous system in biological systems [11], which are known to exhibit autonomic principles to maintain homeostasis or equilibrium and adaptation to a dynamic and changing environment. These characteristics enable an awareness of the end-to-end system environment as well as its internal state. This is accomplished through the use of cognitive techniques that utilize AI/ML methods, in conjunction with closed-loop feedback control, to suit a given target behaviour of a decentralized, distributed, and virtualized end-to-end system.

A variety of virtualized system-wide characteristics are examined to provide a context that motivates the need for an autonomous system framework architecture to manage the rising system complexity, as well as to optimize the system performance, through self-CHOP enabled automation.

5.2.1 End-to-End (E2E) Network Slicing

End-to-End (E2E) network slicing [12] is a significant system-wide enabling capability, which offers a flexible and virtualized mechanism to customize the allocation of computing, networking, and storage resources to support the demands of emerging services. These characteristics of E2E network slicing that span the core, edge, and radio access networks, together with the user equipment, is a significant aspect of system-wide complexity in a virtualized service-based system architecture. An intelligent framework, realized through the application of autonomic principles is therefore pivotal for optimizing the system performance and cost, while adapting to customized deployment objectives and emerging business models.

The Radio Access Network (RAN) segment of an E2E network slice entails an appropriate level of granularity shaped by diverse considerations, such as different QoS constraints, associated with each different type of service. In other cases, a given service may require a dedicated RAN slice within the service supporting E2E network slice, as a result of unique constraints (e.g., reliability, latency etc.). These supporting attributes of an E2E network slice, may also provide other service-related diverse requirements, within the prominent service categories of eMBB, mMTC, and URLLC. Optimizing the E2E network slice granularity for both service QoE, as well as system resource utilization requires cognitive automation, realized through autonomous system constructs. This naturally entails a closed feedback between the E2E system and its environment, where changes are both dynamic and probabilistic, based on wireless link conditions, and mobility patterns. Hence, these challenges for an optimization of performance, service experience, and resource utilization, cannot be managed by any piecewise or simple automation methods or models. Such diverse system and service demands, require E2E automation enabled through autonomous system constructs that yield requisite levels of cognitive adaptation (e.g., self-CHOP) to dynamic system and environmental changes.

An illustrative contextual diagram for an autonomous management and orchestration of an E2E network slice is depicted in Fig. 4. For enhancing the service experience and the granularity of services (e.g., customization, user-centric personalization etc.), the rising system complexity, ensuing from a continuing evolution of system flexibility, virtualization, heterogeneity, shared resources, and service sophistication, requires to be effectively mitigated and managed. In this context, E2E network slicing serves as an effective mechanism to flexibly isolate, configure, and instantiate the appropriate resources to suit associated services.

The harnessing of diverse resources with scalability, in a virtualized end-to-end system to support emerging and innovative services demands an adoption of autonomic constructs to yield the necessary cognitive automation to imbue a dynamic adaptability and agility to suit service demands. This implies an AI/ML enabled end-to-end network slicing capability in an autonomous system architecture framework. The benefits of an AI/ML enabled E2E network slicing, includes adaptability and automation of system and service aware resource allocation in the following areas:

- Flexible harnessing of radio access network resources through slicing (e.g., configuration as part of an E2E network slice, and instantiation)
- Radio access technology selection (e.g., choice of licensed or unlicensed spectrum, satellite, terrestrial etc.)
- Distributed multi-access edge computing
- Content delivery

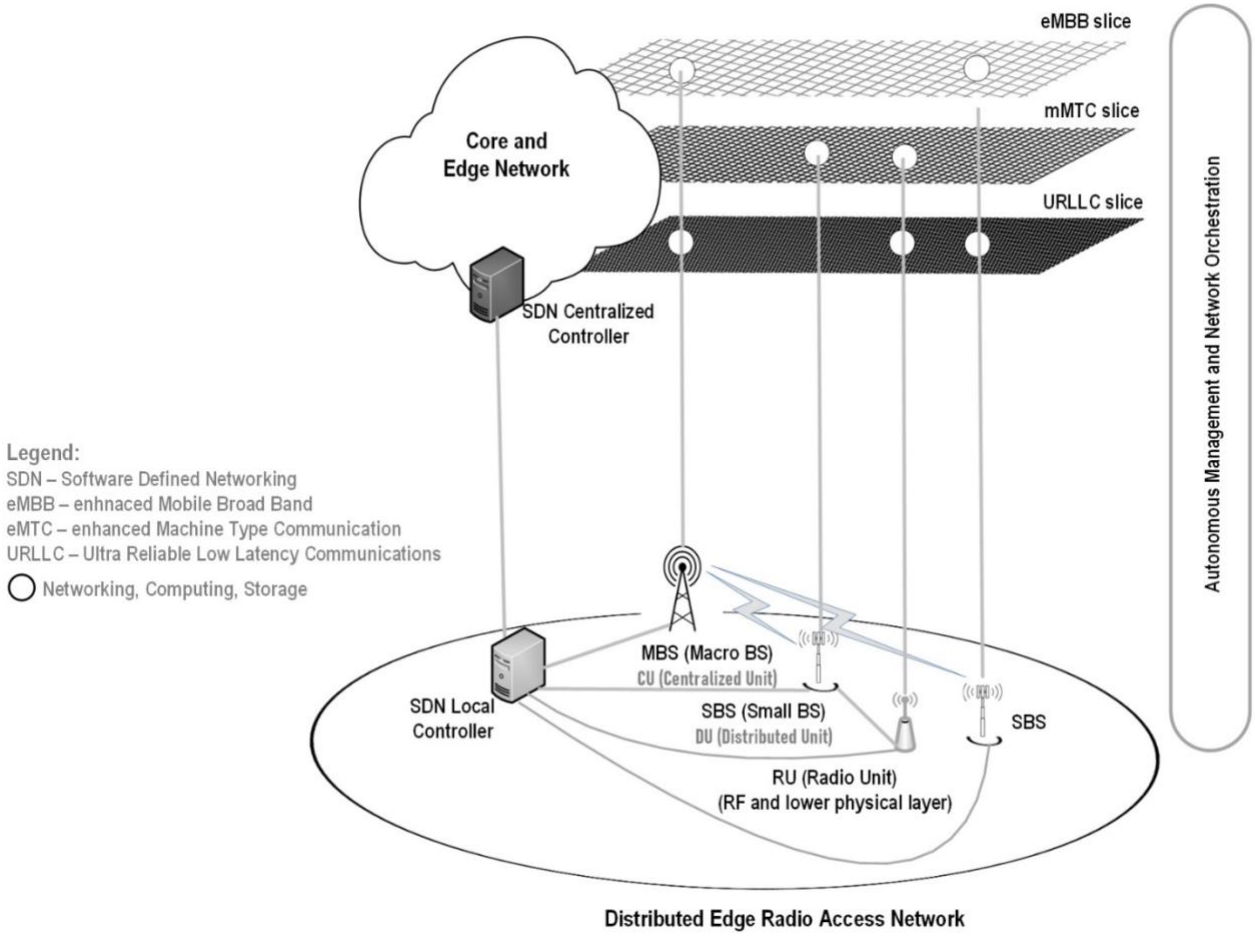


Fig. 4 : Autonomous management and orchestration of E2E network slice types

The elements of a standalone, virtualized end-to-end system consist of a Network Functions Virtualization (NFV) fabric, which includes Virtual Network Functions (VNFs) and is complemented by Software Defined Networking (SDN), which leverages the control plane functions to realize an end-to-end network slice. The NFV fabric configures and manages the lifecycle of an E2E network slice, together with an orchestration of the resources associated with the E2E network slice, through a requisite harnessing of the required VNFs [13]. In

conjunction with the management and network orchestration subsystem, a centralized SDN controller shown in Fig. 4, abstracts and orchestrates network resources, the control logic, the configuring of the blueprint and the instantiation of an E2E network slice [14], while interacting with distributed and local SDN controllers to dynamically steer traffic flows. Fig. 4, shows an example of CU (Centralized Unit) and DU (Distributed Unit) function arrangement across disaggregated radio access network entities consisting of MBS (Macro Base Station), SBS (Small Base Station) and RU (Radio Unit).

The establishment of radio access network slice segment within an E2E network slice is indispensable, where the requisite granularity of the associated radio resources are allocated to meet the QoS and KPI requirements of a supported service. The NFV fabric realizes the network functions as VNFs that execute over generic hardware, together with the management and orchestration of the associated network slice resources, through the corresponding VNFs and the network slice lifecycle, while SDN realizes the control plane for enabling the network slicing process. The challenges associated with radio access network slicing segment include the considerations for a unique slice for each service, such as appropriate admission control, energy efficiency, service QoS and KPI alignment, efficient utilization of system-wide resources, and the associated revenue harvest potential. Optimization of diverse objectives in the realization of an E2E network slice requires cognitive decision-making capabilities that are facilitated through an AI/ML enabled autonomous system architecture framework, such as that depicted at a system level in Fig. 4.

Dynamic and self-CHOP configuration and instantiation of an E2E network slice requires the cognitive decision-making capabilities of an autonomous system framework, for an effective utilization of shared system-wide resources to adapt to the diverse QoS and KPI requirements of complex and emerging services. A rapid convergence towards target objectives to suit the demands of a given service, as well as to manage the complexity of an end-to-end system, require a harnessing of appropriate AI/ML models and closed-loop feedback within an autonomous system architecture framework. At the same time the autonomous system architecture framework is expected to ensure end-to-end an automated system operation and adaptation to a dynamic environment, with changing resource demands, across the core, edge, transport, radio access and service realms, For example, supervised learning [15] utilizes labelled data to analyse network information for inferring network characteristics for an estimation of related parameters.

The classification of traffic, millimetre wave beam alignment, handover between FR1 (sub 6GHz) and FR2 (millimetre wave), selection of wireless links to meet QoS targets, smart offloading of traffic [16], among others are some examples of supervised learning. Inferences and identification of patterns without an interpretation of output information, such as anomaly detection, spectrum sensing, prediction of traffic volume [17], leverage unsupervised learning. An a priori model free approach is utilized by Reinforcement Learning (RL). In the case of RL, the system iteratively adapts to its environment to realize a target objective, through a closed-loop feedback process between an agent in the system and its environment, using a selective combination of exploration and exploitation of discovered knowledge. Optimized resource allocation and scheduling of users, beam alignment, handovers etc., in a dynamic wireless environment are among the various examples of use cases for an application of RL [18]. Federated learning (FL) and Transfer Learning (TL) [19] leverage the other types of ML (e.g., supervised, unsupervised, and reinforcement learning) for adaptive, dynamic, optimized, and automated resource allocation for an end-to-end network slice to effectively realize the support for rendering forward-looking and innovative services, while satisfying a sustainable user-centric service experience. In the case of FL, the privacy of information associated with different entities is preserved through the use of corresponding derived information, where identity is concealed. In the case of TL, different AI/ML models are trained and updated in an efficient manner by leveraging information learned from other similar environments or domains (e.g., wireless channel model with similar propagation characteristics etc.).

Closed-loop feedback spans the system aspects of network resource orchestration, network topology, and network protocols, where the end-to-end autonomous network management includes resource scheduling and planning [20], where the planning includes a requisite reservation of resource pools for all system supported network slices, for a given network topology.

5.2.2 Cross-Domain Cooperation

The end-to-end system is composed of multiple constituents, such as the core, edge, transport, radio access, network management, network service orchestration and management, and business management segments. Some of these segments may be within the same administrative domain or across different administrative domains.

Cross-domain collaboration enables interactions among different administrative domains. This is a significant aspect for managing system performance, service provisioning, and service assurance (e.g., cross-domain fault diagnosis), where multiple domains cooperate and coordinate the rendering of a given service. Autonomic capabilities imbued within the system architecture facilitate pivotal self-CHOP characteristics. These characteristics are paramount for an automation of flexible and agile system response behaviours to provide the necessary dynamic capabilities to sustain performance and service objectives, while effectively managing complexity, across diverse cross-domain deployment arrangements.

5.2.3 Security and Privacy

The cognitive capabilities within an autonomous system realized through a combination of feedback control loops in conjunction with AI/ML models of intelligence for self-CHOP behaviours, require both security and privacy aspects to be effectively integrated. The AI/ML techniques and associated data sets require to be protected through the use of zero-trust features [21] that harness Distributed Ledger Technology (DLT) [22] for an unperturbed scaling of resources and functions (e.g., scaling up or scaling down as needed), fulfillment of appropriate Service Level Agreement (SLA), as well as a preservation of privacy requirements through encryption techniques. These considerations are essential for a sustainable system performance, as well as for a protection of investments, while harvesting the enormous benefits of automation realized through an autonomous system.

An intelligent scaling up or scaling down of resources within a virtualized end-to-end system complements network slicing, which facilitates an appropriate and flexible allocation of resources to support the functional and performance demands of a given service. A selective allocation and isolation of the requisite resources within a network slice to suit a given service, requires resilience towards threat scenarios, through an adoption of security considerations [23], including the attributes of confidentiality, integrity, availability, authenticity, and non-repudiation, together with privacy considerations, where the user-specific data is appropriately encrypted.

Availability of a network slice and the accessibility of network functions and the network slice manager, while the integrity of the network slice ensures that changes and updates [24] are restricted to the network slice owner, are among the significant aspects of security. These aspects together with the corresponding authorization allows associated capabilities for resource allocation. A fulfilment of the security aspects associated with a given network slice

and its owner, within an end-to-end system, are bolstered and enhanced through the application of autonomic principles yielding cognitive and zero-touch automation with respect to security.

Autonomic principles embedded within the end-to-end system architecture, are complementary with respect to an application of DLT schemes, for a realization of cross-domain security and an orchestration of trust, in a distributed and multi-stakeholder ecosystem. Enhancements through the use of autonomic principles in an autonomous framework include an optimization of network operations, and a fulfilment of SLA requirements in a cross-domain environment. This is accomplished through automated and adaptive security measures in a zero-trust multi-party environment, where there is no a priori trust establishment to yield support for a dynamic rendering (e.g., smart contracts [25]) of innovative services.

Among the data encryption techniques, homomorphic encryption [26] allows algebraic procedures on ciphertext, without decryption, which facilitates data privacy since it conceals user-specific data, in an emerging and a diverse distributed processing environment of multi-access edge computing that supports low-latency services (e.g., URLLC services etc.), over an E2E network slice. Autonomic and cognitive capabilities are leveraged to optimize and automate the security, privacy and zero-trust procedures across diverse, cross-domain, and distributed environments.

5.2.4 Feedback Control Loop

The complex nature of the system context and characteristics described in section 5.2 provides a strong motivation for the utilization of self-CHOP capabilities within the system to enable an awareness of the dynamic changes within the system, as well as in the environment within which the system operates [27].

Feedback control loops between any entity, its output and its environment serve as a primitive building block for realizing self-CHOP capabilities throughout a virtualized end-to-end system. The essential attribute of an autonomous system consists of feedback control loops. An autonomic node, within an autonomous system, consists of an autonomic function, which is an augmented virtual function in a service-based architecture. An autonomic function exhibits self-CHOP characteristics, which requires no human intervention for operation, and derives any required information through discovery, self-knowledge, and

intent. Such a function may operate at any layer of the protocol stack and includes the following characteristics:

- Abstraction of information associated with a concept or process
- Discoverable (e.g., auto-discovery)
- Distribution and decentralization
- Modularity
- Flexible and domain dependent
- Intent oriented
- Function and layer independent

An autonomic management and orchestration subsystem, utilizes feedback control loops, within an autonomous NSP system to facilitate self-CHOP behaviours, for an automatic adaptation to service, user, and business objectives, without human intervention.
The management of a feedback control loop is managed by a feedback control loop management function, which enables the automatic adaptation characteristic of the autonomous system to adapt to dynamic changes in the system and its environment, while fulfilling business and user objectives. An optimization of resource allocation, management of costs, and operational efficiency are realized through the following types of capabilities, provided by a feedback control loop management function

- **Lifecycle management:** Create, modify, activate/deactivate, delete a feedback control loop.
- **Configuration of objectives:** The feedback control loop for a desired convergence target is set to within a configurable threshold, within which autonomous decisions are applied.
- **Monitoring:** The state of the feedback control loop is monitored automatically and recorded, to detect any anomaly in the decision-making process for triggering appropriate alarms, where human intervention may be needed, depending on the criticality of a given usage scenario.

A feedback control loop management interface facilitates the NSP to provide a convergence target objective for the feedback control loop, where the feedback control loop as a managed construct is expected to autonomously operate to achieve a configured target output objective for a given input. The management interface for a feedback control loop should support the following types of interfaces:

- **Create interface:** Creation of a feedback control loop
- **Modification interface:** Update of a feedback control loop
- **Delete interface:** Deletion of an existing feedback control loop

- **Activation/deactivation interface:** Resumption or suspension of an existing feedback control loop
- **Query interface:** Query of the current status of a given feedback control loop in the process towards a target objective

The intrinsic attribute of closed-loop feedback control within an autonomous system [28], requires assured behaviours [29] for a preservation of communication service quality and automation [30] objectives, from both functional and deployment perspectives.

5.2.5 Bearer Plane Programmability

The continuing emergence of services that unveil corresponding opportunities for service innovation, delivery, and monetization, demand profound architectural shifts in emerging wireless systems as a pivotal building block for E2E network slicing. The complexities of the IP bearer network, within an evolving 5G and next-generation system, are simplified through Segment Routing IPv6 (SRv6) [31], which consists of emerging IPv6 extensions, promoting end-to-end network programmability. Besides being a simplified network protocol for the IP bearer network, the main benefits of SRv6 include compatibility with existing networks, convergence of cloud networks, agility of service provisioning, ubiquitous connectivity, deterministic quality, and improved granularity of resource allocation, through service awareness [32]

With these benefits SRv6 is well suited for an adaptable and efficient resource partitioning, based on differentiated service requirements, to enhance resource allocation in E2E network slicing, where each such slice is unique to an associated service tenant, or a group of service tenants with similar demands. This adaptable level of resource allocation granularity is achieved through the use of SRv6 Segment IDs (SIDs), where each SID has a locator dedicated to identifying the resource segments that constitute an E2E network slice. Each node within a shared physical network, has as many SIDs as there are network slices, while each SID has a locater that identifies a specific E2E network slice. Differentiated network paths may be assigned by an ingress node in the shared physical network, based on the traffic flow associated with an E2E network slice. The SDN controller establishes and distributes the association information between a given service and its traffic flow, across the network nodes that support a corresponding E2E network slice.

Leveraging SRv6 enhances the flexibility of E2E network slicing. This in turn is complementary with respect to enhancing the operation of an autonomous framework, in terms of reducing

the number of protocols in the system, for improved efficiencies in system-wide autonomic behaviours (e.g., intelligent management and orchestration). The simplification of the rising complexity of the IP bearer, through the use of SRv6, ushers a transformational shift towards a network as a computer model, which further advances the efficacy of an autonomous framework for automating the realization of an E2E network slice.

The demands of emerging cloud-native services (e.g., massive IoT, URLLC - Ultra Reliable Low Latency Communications etc.) in the evolving 5G and next-generation system, require the flexible and adaptable characteristics of an SRv6 enabled transport network, where SRv6 contains two primary building blocks, namely, IPv6, and source routing. The programmability of SRv6 promotes an integrated approach to connectivity and service forwarding. In turn this allows for quicker interactions between applications at the service layer and the underlying network layer, thereby assisting the cognitive performance of an autonomous framework to enhance the service experience.

Inter-domain connectivity is facilitated, over an appropriate import of source routes, through a cooperation of associated autonomous frameworks across disparate NSP domains. The benefits that accrue from a business perspective are threefold, namely, a reduction of OPEX relative to traditional IP/MPLS protocols, network slicing flexibility based on resource based traffic steering for an improved service Quality of Experience (QoE), and enhanced granularity of cognitive behaviours in an autonomous framework for automation. These benefits are indispensable for an effective fulfilment of the sophisticated service requirements in the realm of massive IoT, URLLC, and distributed Multi-access Edge Computing (MEC).

The network programming capabilities inherent in SRv6, facilitate an improved utilization of network resources, thereby enhancing the efficiency, and automation of E2E network slicing. SRv6 facilitates the programmability of packet processing and forwarding in the data plane, complementing SDN in the configuration and operation of network nodes. This simplifies the monitoring and enforcement of expected system behaviours, which complements the effectiveness of the autonomous framework, for realizing system-wide automation.

5.3 Knowledge Plane

The availability of patterns of data and information or knowledge, on a system-wide basis that can be leveraged to manage the end-to-end system behaviour, is facilitated by the Knowledge Plane (KP) [33], to enable an adaptation of the system to dynamic changes in its environment.

The KP continuously learns and acquires knowledge through a lifecycle process of exploration and experience of network management, planning, maintenance, operation, and optimization of the autonomous system, such that the intended objectives (e.g., KPIs) are satisfied.

The KP enables the autonomous management of the constituents of an end-to-end system, such as for enhancements as well for evolving the autonomic Decision Elements (DEs) that may require to be replaced or upgraded at appropriate levels of abstraction for management and operations. The KP builds and maintains knowledge on a system-wide basis in its Knowledge Base (KB), which may be distributed across multiple servers, using different functions, such as ONIX, and Model Based Translation Service (MBTS), together with autonomic cognitive functions, with their own local knowledge databases, to collectively build knowledge at the KP. The KP may also retrieve knowledge from entities, such as OSS/BSS,

5.3.1 Knowledge Management

The process of knowledge management consists of the lifecycle of the management of knowledge, which includes knowledge constructs, knowledge processing, knowledge sharing, knowledge applications, knowledge updating etc. Autonomic principles facilitate an autonomous management and application of knowledge for an automatic and dynamic system operation, through an intelligent and adaptive interpretation of system-wide and complex information flows. This facilitates the derivation of knowledge that is applicable for a cognitive decision-making process in the end-to-end system.

Interoperable and open knowledge management specifications are pivotal for effectively and consistently processing different types of knowledge in the end-to-end system, to establish a robust Life Cycle Management (LCM) of zero-touch automation through the autonomous framework. This would facilitate a unified knowledge management process, together with appropriate and timely updates, yielding collaborative interactions within the system scope. Through cross-domain knowledge creation, fusion, and transfer, knowledge sharing and collaboration among different domains are realized. The abstraction of voluminous information within an end-to-end system, in terms of knowledge management is an intrinsic aspect of the autonomous framework for directions towards zero-touch automation and operational efficiency, to effectively manage complexity and cost that accrue with system evolution.

The processing of information within the system for assimilating, learning, and understanding knowledge through AI/ML oriented closed-loop, cognitive techniques, while leveraging context-awareness and metadata-based policies, enables the knowledge plane within the system to rapidly adapt new and revised knowledge for decision-making with integrity. This experiential process of harvesting knowledge underscores an evolution of system-wide functionality to enable NSPs to effectively meet the diverse requirements of emerging and innovative services, while optimizing and maintaining system-wide performance [34]

The management of knowledge include of the following types of functions:

- **Knowledge construction:** The transformation process from data processing, information extraction to the creation of knowledge
- **Knowledge processing:** The processing of existing knowledge in a knowledge repository, integrating externally imported knowledge, inferencing or combining, and creating new knowledge through further knowledge inferencing, knowledge mining or other techniques to enhance the integrity and completeness of the knowledge in the system.
- **Knowledge sharing:** The processing of knowledge between a local knowledge repository and an external knowledge repository, for a sharing and leveraging of knowledge, with respect to diverse participating entities within and across different collaborating systems
- **Knowledge application:** The application of knowledge within the system for a realization of relevant functions in the system for cognitive decision-making.
- **Knowledge updating:** The updating and upgrading of knowledge for an alignment with the internal and external environment changes associated with the system.

The knowledge management interface facilitates interactions between the knowledge management system and the application domain knowledge management, as well as between the knowledge management system and an external system. The knowledge management application domain applies knowledge to realize cognitive system functions, such as objects (e.g., networks, hardware/software entities within a system etc.). Specific knowledge management interfaces, within a given knowledge management system, facilitate interactions with external systems.

The knowledge management interfaces include of the following types of operations:

- **Knowledge import:** Interface for the import of knowledge through external systems and human expertise

- **Data collection:** Interface for the knowledge management application domain and external systems for the collection of information to be processed by the knowledge management system.
- **Knowledge application:** Interface for the knowledge management application domain to obtain knowledge from the knowledge repository.
- **Knowledge sharing:** Interface for external systems to obtain knowledge from the knowledge repository.

The preliminary foundations for knowledge management and ongoing research span the basic architectural considerations, challenges, knowledge management, requirements, and definitions [35] [36].

5.4 Management and Orchestration

System-wide intelligence realized through an autonomous architecture framework automatically manages complexity, performance, and adaptability, where the corresponding system response requirements are beyond the limits of human intervention. The continuing advancement of the 5G and next-generation ecosystem is characterized by a diverse service paradigm rendered over a heterogeneous and distributed system infrastructure of networking, computing, and storage resources, which require to be tunable to a variety of deployment objectives, business models, and performance targets. The service lifecycle, from an end-to-end perspective, requires to be maintained in a scalable manner, through the use of autonomous system constructs for automating complex workflows, among the orchestrators, cognitive modules, and controllers (e.g., service and network) within the system, as well across cooperating system domains. The contextual model for the KP within an autonomous system for system-wide management and orchestration [37] with inter-domain, and intra-domain awareness is depicted in Fig. 5. In this context the KPs may also collaborate in a federated manner, in an inter-domain or intra-domain manner based on deployment configurations.

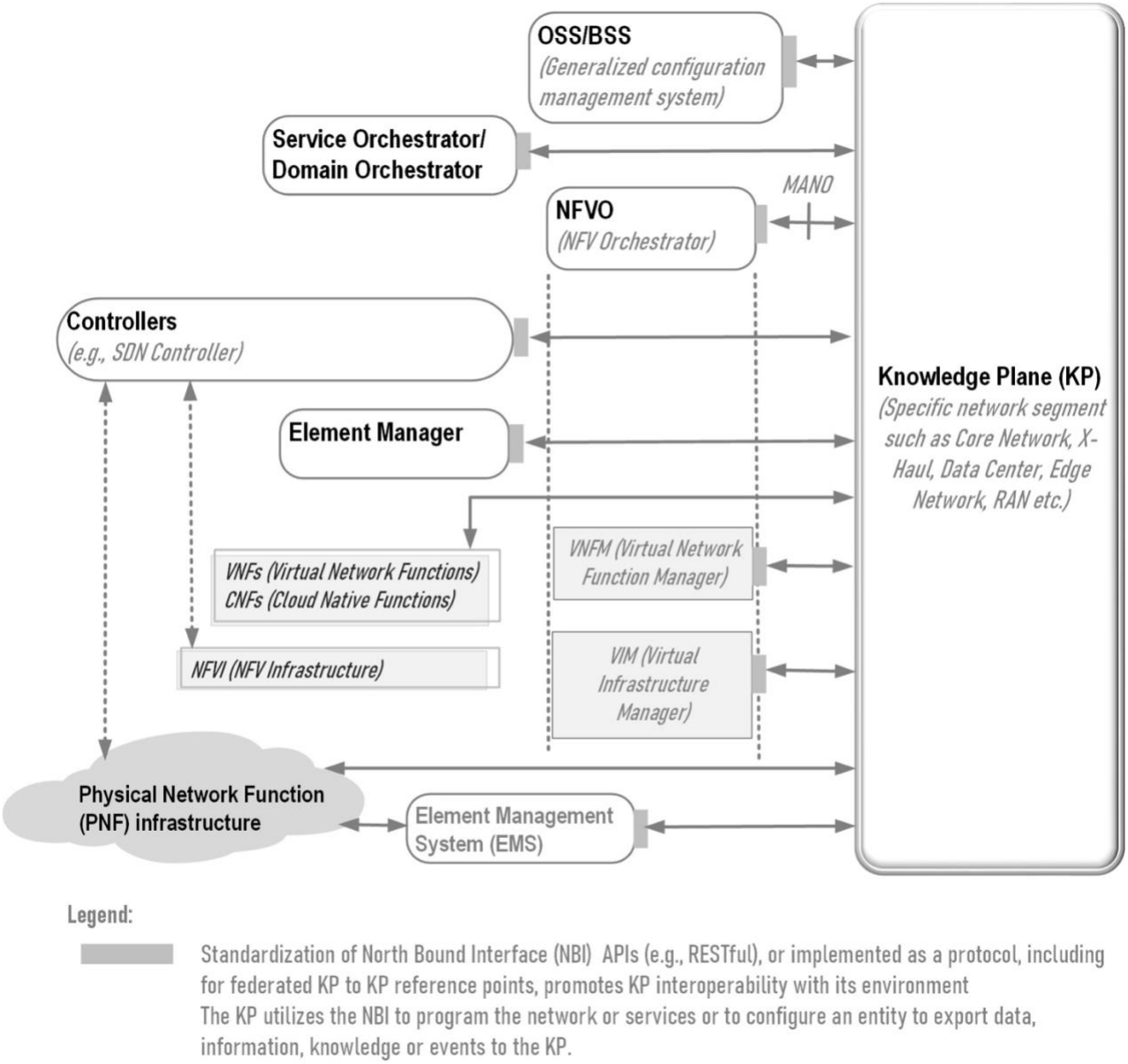


Fig. 5 : Knowledge Plane context within an autonomous system

The objective of management and orchestration is to deliver an optimal quality of service experience. Service orchestration facilitates the cooperation of the different constituents of an end-to-end system in terms of enabling optimized and dynamic workflows, where the *"what is required"* is translated into *"how the requirements are satisfied"*. This translation results in the realization of a corresponding network slice containing the appropriate resource configuration, based on an associated data model abstraction (e.g., YANG [38], representing a tenant, virtual function, analytics etc.). The network slice configuration is enforced through the use of interfaces, such as RESTful or NETCONF [39][40], which are exposed within a cloud-

native Service Based Architecture (SBA) [5] that encompasses the core, edge, transport, and radio-access network. The use of NETCONF interfaces has a broad adoption in the industry as part of service management and orchestration [5]

Within an end-to-end system, the service orchestration process may consist of several sub-system level service orchestrators for an aggregation of the required virtual and physical resources, using a suitable API. The cooperation across KPs may also occur across different administrative domains (e.g., access, backhaul, transport, telco-cloud infrastructure etc.), where the instantiation of a given network slice fulfills the appropriate inter-domain Service Level Agreements (SLAs), which are applied to service quality requirements, lifecycle management of tenants, and the required allocation of virtual and physical resources

An efficient enablement of cloud-native architectural tenets associated with Network Function Virtualization (NFV), embodied within the SBA framework, requires the management of network data within an end-to-end system. This is a prominent aspect of the OSS, where the Network Data Layer (NDL) [41] within an autonomous system framework, promotes flexible deployment choices, while managing complexity and optimizing operational efficiency for diverse usage scenarios. From a federated KP perspective, as depicted in Fig. 5, there would be cooperation between the NDL and the KP, which provides a shared repository of data, collected from various sources within a given system and then converted into cognitive decision-making knowledge.

In the virtualized context of the SBA framework, the application logic associated with data processing, is separated from the data storage, which may be centralized or distributed. This implies that the Virtual Network Functions (VNFs) in the SBA framework can be stateless, where the VNF only renders the service logic, while not managing its own data. In an autonomous system, this separation facilitates an independent scaling of both data processing and data storage requirements, thereby reducing the Total Cost of Ownership (TCO). These aspects are part of the NDL for efficiently managing the end-to-end system data analytics, within the autonomous system architecture framework. Diverse data analytics capabilities can be leveraged for aggregating and discerning the information associated within each segment (e.g., core network, transport network, edge network, radio network etc.), of the end-to-end system, including the management and orchestration sub-system(s). Cross-domain NDL in the autonomous system cooperate to enable data analytics at the network layer.

Networks within an end-to-end system are typically composed of various types of Network Functions (NFs), based on the SBA architectural model, which may belong to different domains or network segments (e.g., core, transport, edge, radio etc.) within the same domain, To harness an efficient cooperation across different domains or different segments within the same domain, while managing complexity, closed-loop and AI/ML oriented data analytics are delegated to the relevant network segments. Intent-based APIs and intelligent orchestration capabilities are leveraged for interaction and harmonized cooperation across the different network segments or domains for accessing data analytics that yield enhancements in system performance and business/deployment objectives.

For cost optimization Network Service Providers (NSPs) and Service Providers (SPs) harness shared resources, in a common system infrastructure, which are virtualised and segmented into appropriate network slices to suit the demands of a given service invocation. Network slices are realized through virtual network functions, which enable support for the demands of an emerging heterogeneous system infrastructure, while accommodating disparate service requirements and objectives. This in turn underscores the need for a management and orchestration subsystem, which is endowed with autonomic capabilities to promote self-CHOP behaviours.

5.4.1 Service Based Architecture (SBA) Context

The SBA context facilitates a flexible and customizable service framework consisting of virtualized functions and resources, to support emerging services and usage scenarios in a 5G and next-generation ecosystem. The network slice construct serves as an enabling vehicle for an appropriate allocation of virtualized functions and resources that are necessary to support the quality of service and experience, based on KPI requirements associated with a given service invocation.

The vast variety of different service requirements are reflective of a growing system complexity, which in turn demands a leveraging of autonomous system behaviours to automate the configuration, instantiation, and maintenance of network slices in the autonomous system framework.

5.4.2 Virtualization and Microservices

Network virtualization is fundamental and complex, since it involves the virtualization of networking computing and storage resources, through a dynamic allocation of these shared physical resources for realizing a given network slice. Cloud computing leverages a migration of shared physical resources from a local environment to a topologically disparate environment, which may be distributed over a network edge or a remote environment. Guarantees of diverse service related KPIs (e.g., quality of service, bandwidth, latency etc.), security, privacy, cross-domain coordination, service composition, device virtualization, flexibility of network function placement etc., are among the attributes that contribute to the complexity of a service based architecture.

The constructs of microservices and containerization facilitate simple and indivisible services, which can be conveniently replicated and instantiated, while scalable with fine granularity. A collection or a specific arrangement of microservices may constitute a network slice, where the associated resource allocation is elastic and customizable to suit the demands of any supported system or user service. The use of microservices in a service based architecture augments the orchestration of resource allocation in a flexible and fine-grained manner for realizing a given network slice, to lower and optimize costs. This complements the efficacy of an autonomous framework, for an automated allocation and clearing of network slice resources.

The evolution of a cloud-native service based architecture, combines the virtualization and the distribution of resources for flexible deployment choices. These cloud-native services, endowed with autonomous system constructs yield cognitive capabilities that enable a dynamic adaptation to change in the system environment (e.g., fault conditions, service changes, congestion etc.), for a realization of end-to-end automation. Along these directions, container based orchestration facilitates the ease of resource configuration, scheduling, load-balancing, security of interactions between containers, and container status monitoring [42].

5.4.3 Cloud-Native and Cognitive Model

The enablement of cognitive capabilities within autonomic functions imbued with a cloud-native approach allows for flexible deployment arrangements, such as for customizing centralized and distributed locations for autonomic functions. This promotes advances in

scalability, reliability, efficiency, and portability to support a diverse and evolving service paradigm.

A prominent aspect of cloud-native virtual and autonomic functions is that it utilizes the concept of containers rather than virtual machines, which enables a flexible packaging of microservices with variable granularity across shared resources. This approach leverages a Continuous Integration/Continuous Delivery (CI/CD) agile methodology across a variety of deployment scenarios. The use of containers for autonomic and cloud native functions facilitates a variety of services, while also enabling ease in the on-boarding of constituent components.

The introduction of cloud-native technologies to realize and deploy network functions, combined with streamlining the development and production procedures through CI/CD has prompted an agile softwarization of network functions, without any dependencies on the underlying hardware platforms. Furthermore, a leveraging of state-of-the-art virtualization technologies (e.g., containers, forward-looking function-as-a-service approaches) continues to shape the evolution of NSP systems towards a pervasive adoption of cloud-native principles. This direction is underscored through a realization of a 12-factor application methodology [43], transitioning away from monolithic software towards a flexible and scalable realizations.

The characteristics of different types of network functions in the emerging landscape of diverse virtualization technologies [44], are classified as follows:

- **Physical Network Function (PNF):** This type of network function is a physical entity (e.g., hardware based) and may be vendor-specific, and may consist of a monolithic executable (e.g., single executable or a statically linked set of executables).
- **Softwarized PNF (SPNF):** This type of network function is completely rendered in software, with widespread portability across computing platforms, while it is implemented as a monolithic executable.
- **Virtual Network Function (VNF):** This type of network function leverages virtualization technologies, such as virtual machines, as well as in some cases realized within containers, while the related software remains monolithic.
- **Cloud Native Function (CNF):** This type of network function is implemented following the 12-factor application methodology, which can be orchestrated, using cloud-native principles that leverage lightweight virtualization technologies, such as containers,

which provide scalability, ease of onboarding and maintenance, flexibility of composition and placement in a local edge cloud or a remote cloud environment.

- **cloudified CNF (cVNF):** This type of network function is stateful in nature, such as the User Plane Function (UPF) or a Service Communication Proxy (SCP), which does not conform to a 12-factor application methodology. This type of function, while fully softwarized in terms of handling packets, may be deprecated in favor of fully cloud-native deployments that are capable of scaling on demand.

The small, lightweight nature of containers allows them to be moved easily across bare metal systems as well as public, private, hybrid, and multi-cloud environments. By introducing the lifecycle management of cloud-native network functions and orchestration management capabilities, such as global cross-domain resource scheduling, it helps to speed up the construction of new applications, as well as the optimization and connection methods of existing applications. With these benefits, the system exhibits high flexibility and maintainability, while naturally supporting continuous iteration and automation of operation and maintenance (e.g., DevOps), while optimizing resource utilization, and improving cloud-native network automation and intelligent operation capabilities in concert with an autonomous system framework.

Cognitive services may be instantiated within a NF or the KP, as needed by an NSP or SP, where the Network Data Analytics Function (NWDAF) and the Analytics Data Repository Function (ADRF) could be used to store AI/ML models, which could be requested on demand and deployed within an NF. This service would be subsequently deleted at the end of the lifecycle to avoid the storage and processing of resources for an expired service.

The attributes of management and orchestration associated with cloud-native autonomic functions include:

- Provision of uploading, storage and parsing of a container model package, together with the deployment and configuration of a CNF based on the model
- Orchestration of the CNF-based network service performed by the CNF orchestrator:
    - Parsing of the network service model to support a cloud-native representation
    - Authorization of container resources associated with a cloud-native deployment configuration
    - Establishment of network connections between cloud-native functions
    - Life cycle management through the deployment, update, termination, and deletion of cloud-native functions and other functions

- CNF lifecycle management:
    - Support the conversion of model definition parameters to CNF deployment parameters
    - Support the conversion of CNF deployment model to container resource descriptions
    - Support CNF deployment and configuration based on container resource description
- CNF resource management and configuration management:
    - Resource management and configuration management, through the provision of a runtime environment for containers
    - Scheduling, load-balancing, and status monitoring etc.

The interfaces and models (e.g., the semantics and behaviours of information attributes and relationships that are protocol and technology neutral for management and orchestration) [38] [45] [46] , required to support the attributes of an orchestrator, are to be arranged and specified to suit the operation of the corresponding cloud-native and autonomous functions, associated with the orchestrator. Prominent high-level requirements include:

- Cloud-native function orchestrator should have both a model upload notification interface and a resource authorization interface
- Container lifecycle management interfaces include:
    - **Instantiate Cloud-native function:** Create and deploy container instances
    - **Update Cloud-native function:** Update container resources
    - **Terminate Cloud-native function:** Terminate and delete container resources
    - **Query Cloud-native function:** Query existing container instance information
- Basic container resource management and configuration interfaces, include:
    - Container computing/storage/network resource management interface
    - Container image management interface
    - Container configuration management interface
- The model requirements are as follows:
    - Basic resource description, external connection points and other necessary information for deploying the container.
    - Adaptability to a cloud-native architecture, with parameters to support the completed lifecycle operations, and support for multiple implementation approaches.

- Support the reference network service model within a CNF model.

5.4.4 Intelligent Orchestration

The virtualized end-to-end system spans a diverse array of heterogeneous networks, software functions, and hardware platforms, which implies that the orchestration capabilities harnessed by an autonomous system framework requires intelligent functionality to harmonize and coordinate different interoperable specifications. A foundational set of requirements for management and orchestration [47], is realized in open-source initiatives, such as Open Network Automation Platform (ONAP), for cloud-native orchestration, which is a natural step towards implementations, which are outside the scope of this document.

The effectiveness of cloud-native autonomic functions for intelligent orchestration is complemented by the lightweight nature of containers, which allow portability across heterogeneous hardware platforms (e.g., bare metal), such as in public, private, hybrid, and multi-cloud environments. This strategy, combined with CI/CD process, provides a compatible environment, within the autonomous system framework, where automation is embodied within the larger context of end-to-end self-CHOP system behaviours that yield system-wide automation.

5.4.5 Intent-based Networking

An intent-based networking approach [47], [48], [49], [50] utilizes abstracted high-level requirements to articulate the *"what"* or the objective that is intended, while delegating the *"how"* to an associated technology or implementation.
For example, management of the lifecycle of services rendered over a network slice, which consists of a variety of resources (e.g., virtual functions, physical functions, requisite composition of a chain of autonomic functions etc.). requires an expression of both service and customer requirements in a manner that is independent of specific underlying technologies and implementations.

The combination of intent with service orchestration imbued with autonomic functions provides an automated process for intent translation utilized for the creation of appropriate network slice configurations and instances for service delivery. Intent based end-to-end management of a system facilitates an adaptable response to emerging Verticals, to suit the associated requirements of a given Vertical, in terms of functionality, quality of experience, latency etc., within the scope of the intents declared across the system. These intents may be

abstracted and declared at various levels of granularity to adapt effectively to the various elements and devices within an end-to-end system [27].

The functional capabilities of an intent based end-to-end management of a system broadly consists of the following:

- **Cognitive function:** Analysis and reasoning logic creation to align the current state of the system with a given intent
- **Decision function:** Formulation of a requisite plan of action to move the system state to an intended state
- **Execution function:** Realization of the plan of action formulated by the decision function
- **Verification function:** Examination of whether a user objective is achievable, with respect to a given system intent, in terms of the effectiveness and impact of the intent
- **Decomposition function:** Layering of intent realization in an end-to-end system, across different cooperating administrative domains, where a user objective requires the engagement of multiple domains
- **Knowledge function:** Invocation on-demand of the information associated with end-to-end system intents (e.g., intent knowledge model)

The interfaces associated with an intent based end-to-end system management broadly consists of the following:

- **Interface between the intent owner and the intent handler:** The intent owner is the source of the intent that declares a lifecycle management policy, which is managed by the intent handler within a system or subsystem. Once a specific intent object (policy) is received from an intent owner, the intent handler takes the necessary actions to satisfy the intent as much as possible, based on the resources and solutions available in its management domain, and reports the intent handling status (e.g., success/failure etc.) to the intent owner.

The intent management interface utilizes the following types of interfaces for intent realization:

- *Create interface:* Used by the intent owner to create the intent and send it to the intent handler.
- *Update interface:* Used by intent owner to update the existing intent.
- *Delete interface:* Used by the intent owner to delete the existing intent
- *Query interface:* Used by intent owner to query an existing intent implementation.

Since intent-based network management decouples the NSP objectives for an end-to-end system from specific implementation details, a leveraging of AI/ML assisted data analytics,

within an autonomous system architecture framework promotes a continuing evolution of the service paradigm. This in turn minimizes human intervention for the realization of zero-touch automation, while effectively managing a scaling-up of system complexity, improving the quality of experience, optimizing system performance, energy consumption efficiency, reducing TCO, and enabling new market opportunities.

Further studies are anticipated in the advancement of system-wide network management scenarios, through collaboration and coordination across the industry for establishing the consistency of an end-to-end semantic model, and interoperable behaviours. The existing intent model, management function and interfaces are to be generalized for broad applicability across diverse scenarios.

5.5 AI/ML Models

In an autonomous system framework, the use of closed-loop decision making processes, combined with appropriate AI/ML models, provide a dynamic and adaptive capability for continuous enhancements in the end-to-end lifecycle management of the end-to-end system. The zero-touch automation, ensuing from AI/ML enabled autonomic functions, promotes an augmentation of the system responsiveness to a given environment, through flexibility and agility, while optimizing resource utilization, to suit the user, operational, and business objectives.

5.5.1 Supervised Learning

Supervised Learning refers to the use of artificial intelligence algorithms trained with some input data (features) and outputs associated with that input data (labels), to predict the outcome for a new input data set.

A typical application of Supervised Learning in telecom is to predict traffic patten and volume.

5.5.2 Unsupervised Learning

Unsupervised Learning utilizes artificial intelligence algorithms to identify hidden patterns in data sets containing data points that are neither classified nor labelled.

One example of Unsupervised Learning is fault management, which includes detection, identification, and mitigation of any abnormal status of networks.

5.5.3 Reinforcement Learning

Reinforcement Learning (RL) algorithm is based on a system of rewards and penalties, mathematically learned through a feedback loop of trial and error, with a goal of achieving a maximized reward.

Given that the states of wireless networks are dynamic, examples of applicability include, RL-based effective power control that can reduce inter-user interference, autonomic management, and orchestration, for sustaining and dynamically improving system performance and throughput, in alignment with intended system objectives.

5.5.4 Federated Learning

Federated Learning (FL) is a distributed learning algorithm which enables nodes/devices to collaboratively learn a shared machine learning model, while preserving local data privacy. FL models are located at both nodes/devices (local FL model) and at cloud-oriented nodes (global FL model) with each node/device having its own dataset.

Federated learning, which utilizes locally trained models rather than directly accessing the user data, can be applied to a variety of use cases. For example, an Augmented Reality (AR) user can learn certain popular elements of the augmentations from other users without directly obtaining their privacy-sensitive data. The representative shared information is pre-fetched and stored locally to reduce the latency.

5.5.5 Transfer Learning

Transfer Learning (TL) pertains to the reuse of a previously learned model for a new problem. TL is an effective solution for utilizing the knowledge gained from similar scenarios to achieve highly effective and efficient learning processes. For example, a TL model trained with different types of signals from various devices can be used for an optimization of cellular mobile network channel estimation, where the knowledge learned from the general features may be common across different channels/domains.

5.5.6 Automated ML

Automated Machine Learning (AutoML) [51] enhances the AI/ML capabilities for data scientists, by automating the AI/ML workflows that enable the AI/ML models to learn automatically and to execute with optimized performance. This avoids or minimizes human

intervention, during the entire lifecycle of an AI/ML model. Some examples of AutoML that are applicable within an autonomous system are:

- Automated feature engineering, which automatically constructs features from the data to ensure a better model, thereby yielding an advancement of knowledge, by mining data in the networks that constitute an end-to-end autonomous system.
- Automated Neural Architecture Search (NAS), where corresponding algorithms automate the engineering processes in the autonomous system architecture. Compared with time consuming and error-prone manual designing of models, NAS has the potential to build models more quickly and efficiently, while adapting to an ever-evolving autonomous system.
- Automated AI/ML model compression and development acceleration, where models are optimized and then deployed in heterogeneous network functions, within the autonomous system, for an improved inference performance

5.6 On-boarding and Certification

The onboarding process of a VNF is pivotal for optimizing costs, flexibility, and agile service delivery, through the verification and validation of its expected functionality. This process is applied across the lifecycle stages of a given VNF, for proper behaviour, when leveraged by the management and orchestration subsystem. The lifecycle stages of a VNF, through which expected behaviour is verified and validated include instantiation, service initialization, and runtime operations [52]. The onboarding process includes two parts, namely, the selection and acquisition of a desired VNF, based on related requirements, in a multi-vendor ecosystem, and the VNF operationalization process that incorporates the VNF into the management and orchestration subsystem, with performance testing, before it is deployed in the system.

The performance and integrity of a service rendered by an autonomous system framework, from a virtualized system-wide perspective is pivotal for the service quality rendered by an NSP or SP. The reliability of a cloud-native autonomous system framework hinges on the proper certification and characterization of the system as a whole, in terms of consistent behaviours and quality, such that diverse service demands, and service experience are fulfilled. The self-certification process within an autonomous system, provides an automatic assessment of whether or not the system behaviours satisfy the bounds of operation for which the system was certified, while operating in a dynamic environment. (e.g., wireless mobile and heterogeneous ecosystem) [53] [54]. The extent of a self-certification capability

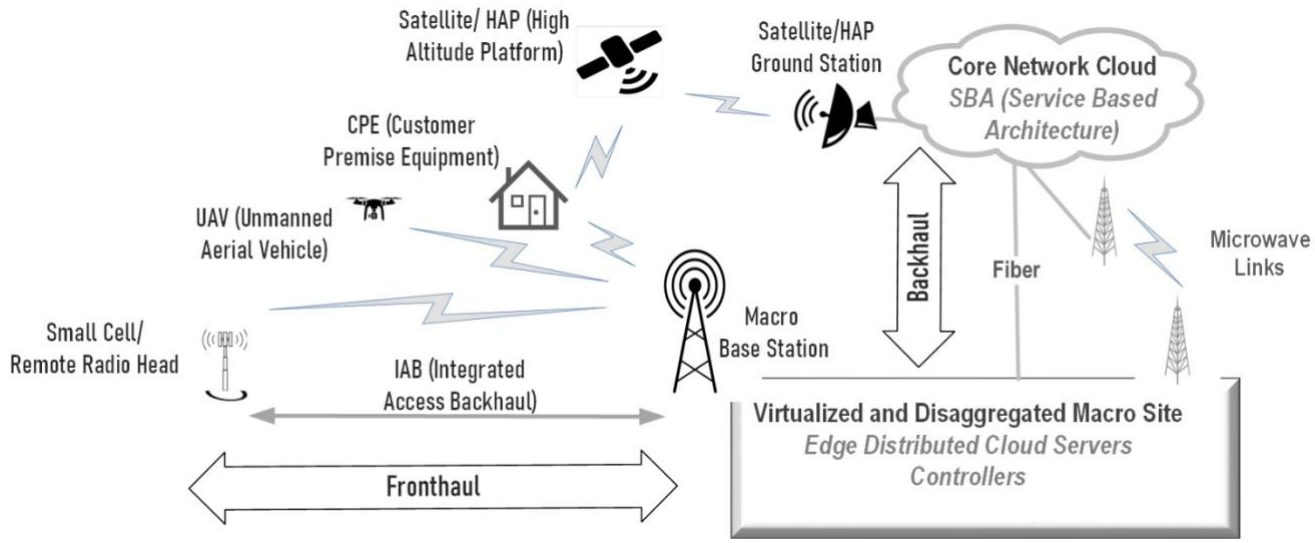
would correspond to, for example, to the degree of autonomic sophistication available, within an autonomous system framework.

A joint DevOps pipeline for a rapid iteration, upgrade, and delivery of software-based network functions (e.g., VNFs) and network management subsystems, promotes their onboarding and certification process. This enhances the adoption, efficiency, and pace of new functions and new business launch cycles, together with an optimization of cost. The functionality of a joint DevOps pipeline includes:

- **Automatic delivery:** This is an automated delivery of software from the supplier pipeline to the NSP (operator) pipeline to ensure that the NSP pipeline can obtain the latest software products on time.
- **Acceptance test:** This is automated testing conducted by the NSP's pipeline after receiving the software product to verify whether the software delivered by the supplier meets the operator's expectations. This includes trust verification, functional test, performance test etc.
- **Production deployment:** This is an automatic deployment of the software that passes the acceptance test for the production environment to provide external services.
- **Operation monitoring:** This is an automatic monitoring of operational data, associated with the supplier software in the NSP's production environment, to provide information feedback for the supplier.
- **Information feedback:** This is timely feedback provided by the NSP for the supplier, in terms of testing and operational status, together with any necessary auxiliary information of the newly released software, for a continuous optimization of the supplier's software.

6 SERVICE SCENARIOS

The various aspects of an autonomous system framework, leverage feedback control loops to intelligently adapt a network sliced end-to-end system, such that the objectives of diverse usage scenarios (e.g., eMBB, mIoT, URLLC) are fulfilled. A few usage scenarios are identified, among several emerging directions.



:

Fig. 6 Emerging service context

The enablement of cognitive end-to-end network slicing is foundational for the realization of new capabilities and services being envisioned for advanced 5G and next-generation systems. In this context, edge computing requires not only a convergence of terrestrial and non-terrestrial access technologies with heterogeneous coverage footprints, but also a cognitive awareness of the system and the environment within which deployments are operationalized. An exemplification of the context of a network edge everywhere in a continuing advancement of the service paradigm is shown in Fig. 6.

System-wide cognitive awareness is facilitated by emerging machine learning capabilities that are tuned to optimize the system-wide behaviour to suit a given value-added service, adapted to satisfy appropriate levels of a personalized service experience. This sets the stage for the requisite levels of optimization and architectural arrangements to suit the performance and

experiential demands of diverse KPIs associated with advanced services imagined in the advanced 5G and next-generation systems.

6.1 Common Marketplace

The use of DLT [22] in a cross-domain context provides a secure distributed environment for engaging multi-domain asset provider (e.g., NSPs, SPs, regulators, spectrum providers etc.) partnerships, collaborating across autonomous systems for service innovation (e.g., services over MEC, emerging Verticals etc.). The benefits that accrue from the complementary functionalities of an autonomous system and DLT include an adaptive, zero-trust, immutable, self-CHOP and self-sovereign transaction of information and value to suit business and deployment objectives, based on a decentralized consensus protocol within DLT.

The use of DLT (e.g., permissioned blockchain) facilitates collaboration in a cross-domain environment with multiple stakeholders, without the need for intermediaries or brokers, which optimizes OPEX, while enabling NSPs and SPs to offer services with zero-CAPEX on infrastructure, through autonomous system framework guided automated cooperation, across multiple stakeholders [55][56].

6.2 Cyber-Physical Interface

A Digital Twin provides a digital representation of a physical entity [57], through a corresponding arrangement of information and simulation for examining, managing, and predicting the behaviours of the physical entity, which can be leveraged for enhancements in the decision-making process within an autonomous system.

Various usage scenarios, associated with an integration of cyber-physical interfaces can be envisioned in the context of a digital twin, with respect to service augmentation (e.g., Industry 4.0, Verticals etc.), where the output of a digital twin provides enhancements to the performance and behaviours associated with a corresponding physical entity. Among other scenarios, the learnings from a digital twin are applicable for improvements in energy efficiency, digital maps of diverse environments for hazard prevention/prediction etc. Within an end-to-end system, consisting of the core, edge, transport, and radio networks, the outputs from a given digital twin, associated with a physical entity can be used to augment the autonomic decision-making processes, in terms of network element configurations, lifecycle management etc.

6.3 Energy Efficiency

The design of an energy efficient system from an end-to-end perspective is of paramount significance, with respect to optimizing operational expenditures, sustainability of a reduced carbon footprint, while system capabilities, complexity of features, and innovative services continue to evolve.

For a realization of energy efficiency, as systems continue to evolve in an interdependent manner, the strategies for advancing a sustainable energy efficiency paradigm, require a cognitive and continuous process to adapt effectively and efficiently to the demands of a dynamic environment within which a wireless system operates. This entails an effective and autonomous management of energy efficiency, associated with the various components (e.g., hardware and software), sub-systems (e.g., core network, transport network, edge network, distributed radio network elements etc.), resource allocation for network slices, and across cooperating domains, while supporting the performance, reliability, and quality of service demands.

Feedback control loop enabled cognitive functions that cooperate within an end-to-end autonomous system is intrinsic aspect of an autonomous system for realizing a dynamic and continuous process that resolves conflicting objectives associated with optimizing both energy efficiency and resource allocation, within an end-to-end system. Techniques such as transfer learning can be embodied within the system to minimize the energy associated with the training of AI/ML assisted cognitive functions, using AI/ML models that have already been trained for similar scenarios and cognitive functions within a given system. For example, AI assisted Control Loops (ACLs) are harnessed in transfer learning [19], where cognitive functions can learn from knowledge shared by other cognitive functions, in terms of enhancing energy efficiency. ACLs are realizable through the use of the knowledge plane, consisting of a feedback loop sequence of Monitor, Analyze, Plan and Execute (MAPE) for enabling system-wide self-CHOP behaviours, as depicted in Fig. 2.

Energy efficiency directions are a part of the global UN sustainable development goals [58] that include devices and wireless systems for a reduction in carbon emissions, as well as for a harnessing of renewable energy technologies, yielding a cleaner environment for human wellbeing. The consumption of energy associated with the various components, functions, and sub-systems within an end-to-end system (e.g., power amplifiers, baseband circuit

boards, various hardware/software elements etc.), are among the prominent entities for an automated and dynamic improvement of system-wide energy efficiency, and related cost optimization, through the use of autonomous system constructs. These directions enable an optimized conservation of energy, for distributed and flexible network deployment topologies, with an automated allocation of resources that are adaptable to the demands of a personalized service experience.

To align and optimize the energy consumption with the traffic demands in an end-to-end system (e.g., correlating zero-bit traffic with a zero-watt power consumption), a self-CHOP adaptation is pivotal for augmenting the energy efficiency automatically by leveraging autonomous system constructs embedded in the management and orchestration of the end-to-end system. Enhancements in energy consumption savings entail a system aware and automatic turning off of unloaded network areas, based on a learning and prediction of dynamic traffic flow patterns, on a temporal and spatial basis (e.g., tuning of base station sleep times etc.). The autonomous level 4 capabilities [3], where autonomic principles embedded with AI/ML models facilitate a real-time adjustment of power, and hence energy consumption over time, aligned with the traffic volume and predictions of traffic volume to improve energy utilization efficiency at the cell level, at the carrier level, at the channel level, and even at the symbol level. The use of RL enables a gradual and adaptive deactivation of one or more base stations, based on traffic load conditions, where a fine granularity of deactivation may vary, for example, from microseconds to seconds. The distribution of AI/ML based cognitive intelligence across the end-to-end system, encompassing high-levels of decentralization and distribution, is pivotal for advancing the energy efficiency with higher levels of granularity, across the core, edge, transport, and radio access networks. The higher levels of distribution of the edge and radio access network promote, higher levels of localization, which reduces the signalling overhead across the end-to-end system, to thereby realize a further improvement of energy efficiency. Intelligent collaboration across different levels of granularity within a system, such as module-level, site-level, network-level, and service-level entities provides an automated advancement of energy efficiency for the entire system,

6.4 Trustworthiness

The establishment of trust mechanisms within an autonomous system architecture framework facilitates the advancement of distributed and decentralized connectivity across

disparate domains and users, while also enabling the enhancements of user-centric services, in a scalable and efficient manner, embodying both security and privacy.

The use of AI/ML algorithms, corresponding to appropriate AI/ML models, is intrinsic for the realization of autonomous systems for an automated adaptation to the probabilistic environment of wireless systems, from an end-to-end perspective. These cognitive models facilitate dynamic and adaptive decision-making within an autonomous wireless system, while consistently satisfying system-wide performance targets.

From a trustworthiness perspective, it is essential that the transparency of the AI/ML algorithms is preserved in terms of enabling access to logs, associated with the behaviour of the AI/ML algorithms, during closed-loop operation. This transparency promotes the trustworthiness of deployed AI/ML algorithms, within the autonomous wireless system, for both NSPs and SPs. The autonomous system framework should be capable of providing information pertaining to on-boarded cognitive functions, as well as operational cognitive functions, where a cognitive function is AI/ML enabled. Information on the type of AI/ML model (e.g., supervised learning, unsupervised learning, reinforcement learning etc.) that is operational within a cognitive function must be easily accessible, including when and how the AI/ML algorithm is applied within the wireless autonomous system.

The actions and decisions taken by AI/ML algorithms must be explainable and traceable to bolster the trustworthiness of a corresponding cognitive function. The resilience and robustness of underlying AI/ML algorithms, within corresponding cognitive functions, is a pivotal characteristic of trustworthiness, within a given arrangement of a wireless autonomous system. The transparency of an AI/ML algorithm is an attribute of trustworthiness, which reveals an understanding of how the AI/ML algorithm makes decisions, for the realization of a cognitive function. The transparency and accountability in conjunction with the robustness and resiliency of an AI/ML algorithm are essential ingredients of trustworthiness within an end-to-end autonomous wireless system.

The verification and validation of AI/ML algorithms is required for a consistent and fault-tolerant behaviour of cognitive functions, especially with the increased attack surface of the code associated with AI/ML algorithms. The selection of appropriate AI/ML algorithms (e.g., linear regression, Support Vector Machine (SVM), K-means clustering, Q-learning etc.,) is essential for corresponding cognitive function behaviours, across different subsystems within

an end-to-end autonomous system framework. Robustness of cognitive function behaviours that match desired objectives within each subsystem (e.g., KP, management and orchestration, NWDAF, adaptive beam correspondence etc.), within an end-to-end autonomous system framework contributes to continuing enhancements of trustworthiness of the system, through corresponding AI/ML model training and updates. These aspects of the autonomous architectural framework underscore the fulfilment of expected KPIs in a consistent manner in terms of a variety of performance parameters (e.g., latency, mixed-media, traffic volume etc.), associated with the delivery of emerging and innovative services in an emerging and next-generation Verticals ecosystem (e.g., Industry 4.0, autonomous vehicles etc.). Fault-tolerance rendered by self-CHOP behaviours of the autonomous system framework improves the resilience of the end-to-end system from cyberattacks or human errors to advance the trustworthiness of the system.

6.5 Cognitive Polymorphic Network Behaviour

Among the adaptive self-CHOP behaviours of an autonomous system, a capability to support emerging and innovative services in the Vertical ecosystem is to facilitate an efficient and intelligent utilization of a variety of resources, consisting of networking, computing, and storage resources, in an end-to-end system. A cognitive polymorphic network structure depicted in Fig. 7, leverages a flexible and definable structure through the different layers of the stack to realize cognitive software functions that utilize fine grained and adaptable combinations of resources and network configurations [59]. This facilitates autonomic function customization and personalization or polymorphism, to suit different users, within the same network.

The dynamic concepts of AI/ML enabled cognitive management and orchestration together with programmable hardware, forwarding, protocols, and interconnectivity, allow for a flexible autonomous framework architecture.

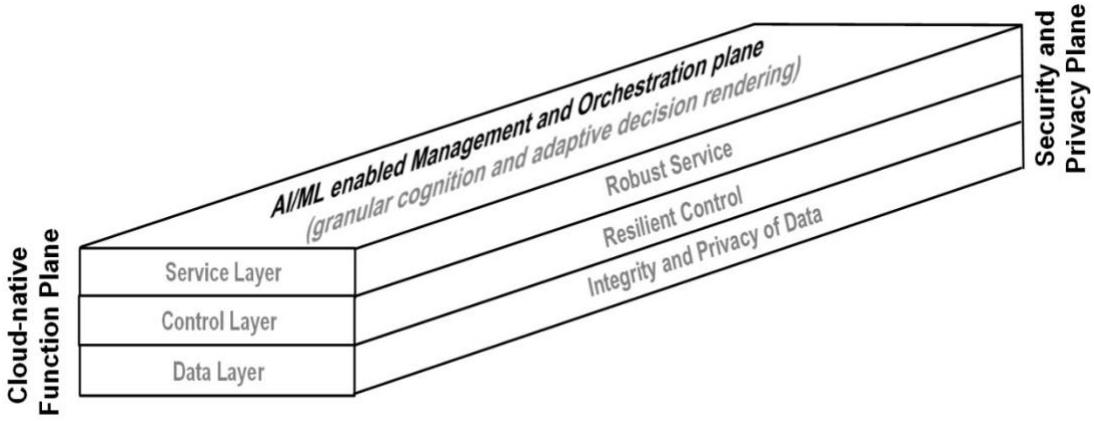


Fig. 7 : Polymorphic structural context within an autonomous system

A prominent benefit of dynamic structural flexibility is to simultaneously satisfy a given service KPI and a user-centric service experience, while optimizing the utilization of resources within a given end-to-end system. A cognitive AI/ML enabled management and orchestration subsystem realizes dynamic system adaptability through feedback control loops, hinging on a sequence of perception, adaptation and decision-making to realize the state of an end-to-end wireless system to autonomously satisfy an intended target system behaviour.

7 INDUSTRY GAPS, COOPERATION AND STANDARDIZATION

The challenges and gaps in the industry include the arena of identifying, studying, and developing consensus on interoperable enablers, associated with emerging technologies, which include autonomic systems that are pivotal for network and system-wide automation.

7.1 Industry Gaps

The scarcity of requisite resources, in terms of knowledgeable and skilled personnel, together with appropriate levels of time commitments are impediments for a smooth progress followed by implementation investments and market adoption. A dominant challenge is to harness technology knowhow from various research communities that imagine and create emerging ideas to fill these gaps in the advancement of technologies. These directions are promoted through information dissemination, requirements development, and standardization, followed by design, implementation, and integration for an alignment of adaptability to versatile business models and objectives.

Coordination among various research initiatives, industry forums, and standards development organisations is pivotal for leveraging the knowhow, pertaining to an emerging autonomic system framework, for efficient and effective collaboration, while avoiding duplication and overlap of content. Clarifying requirements together with an advancement of perspectives and understanding is essential, while referencing other research endeavours without replication.

The distinct and complementary aspects of content, delineated in different research communities, industry forums and standards organisations must be acknowledged and recognized, while harmonizing and advancing the various distinct perspectives.

In this context the knowhow associated with emerging autonomic systems, may be broadly categorized in terms of the following considerations:

- Onboarding of knowhow through investments in AI/ML centres of excellence with the support of investments and commitments from key stakeholders in the business for advancing research (e.g., Centres of Excellence (CoE)) in autonomic systems that promote network automation.

- AI/ML CoE provides definitions and priorities for functional areas in terms of AI/ML initiatives, associated with short-term, mid-term, and long-term (e.g., multiple years) business benefits that delineate an executable plan for transformation towards an adoption of autonomic systems.

- Strategic directions provided by AI/ML CoE initiatives are pivotal for influencing the adoption of autonomic systems for network automation, beyond the legacy piecewise automation and open-loop systems that are insufficient for managing the rising system complexity, while enabling a zero-touch (zero human intervention operation) network operation and system-wide automation.

- Guidance provided by AI/ML CoE, and data scientists, are expected to enable the existing network operations and maintenance personnel, in terms of AI/ML models, utilization, model training, and integration related to model upgrades for network stability, service quality, system availability, and system evolution.

- Broad categories of logical considerations, in terms of features, capabilities and functions that could be envisioned as part of a transformation towards an adoption of autonomic systems are depicted in Fig. 8 for guiding customizable organisational structures, suitable for a given business or deployment model.

New features and capabilities	**Service Functions** (e.g., data Scientists, operational and tactical personnel)	**Strategic Functions** (e.g., business, cross-functional, stakeholders)
AI/ML modeling and simulation	x	
CI/CD for AI/ML	x	x
AI/ML embedded in the network and system	x	x
Network and system digital twin	x	x
Advanced analytics for system resilience, prevision, and evolution	x	x

Fig. 8 Logical considerations for an adoption of autonomic systems

7.2 Industry Cooperation and Standardization

Cooperation and coordination across the industry, are significant across the ecosystem for continuing advancements towards system-wide automation, realized through an autonomous system framework. These directions require engagements, partnerships, and recommendations that span a multitude of industry fora (e.g., ETSI, IETF, ITU, GSMA, TMForum etc.), system specifications in 3GPP, and open-source communities, such as ONAP within the Linux Foundation etc. These initiatives (e.g., M-SDO (Multi-SDO) Autonomous Networks facilitated by TM Forum etc.) serve as a catalyst for the advancement and harmonization of an AI/ML enabled autonomous system framework that includes management and orchestration with cognitive cloud-native functions. These considerations are essential for harmonization, widespread interoperability, and consistent behaviours, across a multi-vendor ecosystem.

Research and study on the closed-loop feedback control characteristic of an autonomous system require to be extended to the definition and specification of the closed-loop feedback control management functions and their interfaces. In the arena of knowledge management, continuing advances are anticipated towards a unified specification of the process and structure of knowledge management, embedded with autonomic behaviours for realizing an

automation of knowledge creation, derived from system-wide information. These enhancements require cross-industry cooperation and collaboration in the various stages of specification development and commercialization.

The advancements of cloud-native container orchestration embedded with autonomic principles, hinging on the foundations in [47], are anticipated to continue through industry wide cooperation and coordination, including open-source communities (e.g., ONAP etc.). In this regard, reference implementations, promoted by open-source communities are expected to serve as a valuable catalyst towards further understanding and corresponding enhancements. The complementary nature of collaboration across these communities, in a complex ecosystem, is pivotal for an adoption of cognitive cloud-native autonomous functions that are essential for an evolution towards end-to-end self-CHOP behaviours rendered by an AI/ML oriented autonomous framework.

Cooperation across a muti-vendor ecosystem is pivotal for establishing harmonized technical requirements, where the intrinsic benefits of virtualization and containerization of network functions at the microservice level complement CI/CD methodologies. Along these directions, open-source communities serve as catalyst for realization, through a leveraging of the relevant, and emerging, tools and technologies

8 ABBREVIATIONS AND GLOSSARY

Term	Description
AI/ML	Artificial Intelligence/Machine Learning
Autonomous (Autonomic)	Self-management characterized by self-CHOP (Configuring, Healing, Optimizing, and Protecting) for cognitively adapting to environmental changes to suit a given behavioural objective or intention, realized through the principle of closed-loop feedback. (adjective)
Autonomy	Condition or state of being autonomous, where the associated entity operates independently (noun)
Autonomous	Attribute of any entity, such as a network, system, or a sub-system, characterized by autonomic capabilities that render autonomy for the entity, implying independent of human intervention. (adjective)
Automatic	Attribute of an autonomous entity which is dynamically adaptive, or an entity that is not autonomous, while being programmatic with limited adaptability. (adjective)
Automation	Process that embodies automatic behaviour. (noun)
BSS	Business Support System
CI/CD	Continuous Integration/Continuous Delivery
CNF	Cloud Native Function
DE	Decision Element
eMBB	enhanced Mobile Broadband
DLT	Distributed Ledger Technology

FL	Federated Learning
Intent	This refers to an abstract, prescriptive, and adaptive high-level expression of policy for system-wide (end-to-end network) operation, based on autonomous systems.
IoT	Internet of Things
KB	Knowledge Base, which is a KP database
KP	Knowledge Plane
KPI	Key Performance Indicator
MEC	Multi-access Edge Computing
mMTC	massive Machine Type Communications
NETCONF	Network Configuration Protocol
NF	Network Function
NWDAF	Network Data Analytics Function
OAM	Operation Administration and Maintenance
OSC	Open Source communities
OSS	Operations Support System
ONIX	Open radio Network Information eXchange
QoE	Quality of Experience
R&D	Research and Development
REST	Representational State Transfer, which embodies an architectural style for APIs, with the principles of platform independence, statelessness between a client and server etc.
RL	Reinforcement Learning
Self-CHOP	Self-(Configuring, Healing, Optimizing, and Protecting)
SDO	Standards Development Organisation
TCO	Total Cost of Ownership
TL	Transfer Learning
URLLC	Ultra-Reliable Low-Latency Communications
YANG	Yet Another Next Generation

9 ANNEX – SUMMARY OF SURVEY

9.1 Survey Inference

The inferences from the survey are summarized in the following sections, with respect to network automation and growing interest in this area, with respect to its relevance for managing complexity, through self-organizing capabilities that can be realized through an end-to-end application of autonomic principles leveraging AI/ML, as system evolution continues, in order to enable emerging and sophisticated services.

9.2 Background

An anonymous survey on network automation and autonomy based on AI was completed in December 2021. The survey was based on related technology and industrial aspects to gauge the current status of a development and application of network intelligence, together with potential trends, among operators (NSPs), for realizing network automation. The objective of the survey was to gather the results from such a survey to both infer the status of the adoption of network intelligence for automation, as well as to promote the relevant research to harness autonomy for automation.

The survey was distributed to all 24 operator members of NGMN. The response to the survey were received from 11 operator members for analysis and further study.

9.3 Methodology

The methodology used in the network intelligence survey consisted of the related overall progress, scenarios, R&D strategy, and ecosystem strategy.
Using combined classification, correlation, and benchmarking techniques, an analysis of the gathered survey results from operators was conducted in two dimensions:

- Overall analysis, through queries from a holistic perspective
- Correlation analysis to derive trends, through a correlation of the responses, across various aspects of the survey

9.4 Survey Analysis

The different facets of an analysis of the survey are delineated, as an inference of the trends associated with the use of network intelligence for an advancement of network automation.

9.4.1 Overall Industry Progress

From an overall industry perspective, all operators are on a journey towards the use of network intelligence for simple forms of network automation. Small scale pilot deployments have adopted AI, while about 18% of the NSPs, have indicated long-term plans for large-scale AI enabled automation for their deployments.

9.4.2 Development Strategy

An observation of the R&D directions of operators in the different phases of an adoption of network intelligence, revealed that in-house R&D was a preferred approach, beyond a sole reliance on integrators after an initial launch phase, and for advancement strategies towards larger scale deployments.

Unified platform directions were found to be an attractive strategy from a forward-looking and state-of-the-art perspective for a collaborative management of network intelligence for coordinating the R&D, deployment, and operations for AI oriented network automation. In this regard, collaboration is anticipated to include and leverage cross-domain intelligent applications and capabilities, in a multi-vendor ecosystem. This approach of collaborative management is expected to yield an attractive advancement of network intelligence from PoC (Proof of Concept) to a large-scale deployment, over a unified platform.

9.4.3 Application Scenarios

An arena of broad interest expression among operators is network intelligence with respect to Operations Administration and Maintenance (OAM), with respect to fault management, recovery, and isolation. Intent-based services is an emerging aspect of network intelligence for operators (NSPs), in terms of PoC and a preliminary small-scale deployment phase. Operators involved with network intelligence for a large-scale deployment phase appear to have an interest in supporting emerging Verticals.

9.4.4 Challenges and Opportunities

The trustworthiness of network intelligence, and a shortage of AI talent and related technology resources are among common challenges expressed by operators, for advancing towards sophisticated levels of network automation with autonomy.

The introduction of an end-to-end autonomous architectural framework for automation, with inter-operability and consistent behaviours with standardized ingredients, is a gap that requires to be closed to fulfil an overall operator demand to realize automation with autonomy, without human intervention for system operation.

9.4.5 Industrial Ecosystem

Network service and network related AI algorithms appear to have been given a higher priority for an evaluation and certification of these algorithms in an ad hoc manner, relative to an incorporation of AI within network functions for network intelligence, which is a building block towards an autonomous system architecture framework for advanced automation.

For the construction of an evaluation and certification framework of applications to support network intelligence, the operators in the survey believe that the effectiveness of the evaluation of applications is a foundational consideration. Subsequently, there should be consideration on whether or not the application should be introduced to support network intelligence and further transformation towards an autonomous system for advanced automation.

Finally, the operators in the survey believe that there should be comprehensive evaluation of a full-stack architecture and any related impacts on the operator's organisation.