ETSI White Paper No. 16

GANA - Generic Autonomic Networking Architecture

Reference Model for Autonomic Networking, Cognitive Networking and Self-Management of Networks and Services

First edition – October 2016

ISBN No. 979-10-92620-10-8

Authors:

Tayeb Ben Meriem, Ranganai Chaparadza, Benoît Radier, Said Soulhi, José-Antonio Lozano-López, Arun Prakash

ETSI
06921 Sophia Antipolis CEDEX, France
Tel +33 4 92 94 42 00
info@etsi.org
www.etsi.org

About the authors

Tayeb Ben Meriem

tayeb.benmeriem@orange.com

Co-editor & Contributor, Orange, Chair ETSI NTECH AFI Working Group

Ranganai Chaparadza

ranganai.chaparadza@altran.com ; ran4chap@yahoo.com

Co-editor & Contributor, Vodafone/consultant

Benoît Radier

benoit.radier@orange.com

Contributor, Orange

Said Soulhi

said.soulhi@verizon.com

Contributor, Verizon

José-Antonio Lozano-López

jal@tid.es

Contributor, Telefonica TID

Arun Prakash

arun.prakash@fokus.fraunhofer.de

Contributor, Fraunhofer Fokus

This White Paper was compiled on the basis of the ETSI GS AFI 002 Specification on the GANA Reference Model and other inputs the authors and editors of the White Paper took into consideration and are cited in the paper. Acknowledgements go to the whole list of contributors to the ETSI GS AFI 002 specification listed at the end of the ETSI GS AFI 002 [3].

Contents

About the authors 2
Contents 3
Executive Summary 4
Introduction 5
The Emerging AMC Paradigm 7
Definition of the AMC paradigm 7
The GANA Reference Model 10
Design principles of the GANA Reference Model 10
GANA Core Concepts 12
Example Use Cases of GANA in a Target Network Architecture 18
To whom is the GANA Model Addressed? 22
Interoperability of Autonomics Software Components, and Compatibility of the Hybrid-SON Model with the GANA Model 25
Implementation Guide for GANA 27
Example Instantiation Case for GANA: GANA Instantiation onto the Backhaul and Core network (EPC) parts of the 3GPP Architecture 32
GANA in the Unified Architecture for AMC, SDN, NFV, E2E Orchestration & Specialized Big Data Analytics 35
Harmonization of Standards & Architectures for Emerging Networking Paradigms 35
Proposed Unified Architecture for AMC, SDN, NFV, E2E Orchestration, and specialized Big Data Analytics 35
Call for GANA Proofs of Concept 39
How to contribute to the Standards for AMC 39
References 40

Executive Summary

The industry consensus is that in the digital services ecosystem, networks evolve to so-called “smart networks of the future”, which are characterized by the need to be operated based on principles of dynamically adaptive Automated and Autonomic Management & Control (AMC) of networks and services (aka autonomics). AMC replaces the increasingly complex and error-prone manual and static management and optimization of networks and services. Telecom operators, service providers, cloud services providers, equipment Providers, Network Functions Virtualization (NFV) product providers, Software Defined Networking (SDN) product providers, Operations and Support Systems (OSS) vendors, ISVs (Independent Software Vendors), application developers, regulators and other significant players in these Digital Services ecosystems now want to clearly understand the achievement and progress in standardization work on AMC as well as the impact of AMC on the roadmap for network and services management evolution, including the roadmap on autonomics standards. This clarity will help them in their decisions to invest time and financial resources in standardization of autonomics and deployment of standards-based self-management related solutions for networks and services. It will also help them understand how to apply AMC to concrete use cases, and how AMC relates to other technologies such as Self Organizing Networks (SON), SDN, and NFV.

The AFI working group in ETSI’s NTECH Technical Committee, as the leading group in the standardization landscape for AMC, has a comprehensive work programme which comprises deliverables on: a reference model for a Generic Autonomic Network Architecture (GANA); an implementation guide for the GANA reference model; and autonomics-enabled implementation-oriented architectures that are a result of GANA instantiations onto various reference architectures defined by standardization organizations such as 3GPP, BBF, IEEE, ITU-T and other Standards Developing Organizations (SDOs). The GANA model defines a generic AMC framework and structure within which to specify and design autonomics-enabling functional blocks (GANA functional blocks) for any network architecture and its management architecture—a process called GANA instantiation onto target architecture.

An example of a use case for autonomics technology that is now being deployed is SON Functions for Radio Access Networks (RANs). As readers study the GANA model, they realize that it shares common principles with the Hybrid SON architectural model, as they both enable combining and interworking centralized and distributed management and control solutions for networks and services. As such, the SON (now being deployed) is compatible with the GANA model design principle. GANA Functional Blocks (FBs) instantiated in network segments outside of RAN (e.g. backhaul and core network segments) can complement SON functions in the RAN. The principles defined by the GANA model are expected to play a role in addressing the emerging requirements for autonomics that go beyond those addressed by the current SON, such as the 5G-SON requirements as well as other autonomics-related requirements that constitute “operations efficiency” requirements, as defined in the NGMN 5G White Paper [17]. Also, industry and the R&D community at large are interested in understanding how the GANA model, as a reference model for the AMC paradigm, articulates with reference models for the other emerging complementary networking paradigms of NFV, SDN, and end to end (E2E) orchestration of resources and services —all of which are viewed together with AMC as complementary software-oriented enablers for 5G. ETSI NTECH AFI is addressing this topic together with other SDOs/Fora involved in the recently launched initiative called *Joint SDOs/Fora Industry Harmonization on Unified Standards for AMC, SDN, NFV, & E2E Service Orchestration* [13].

Introduction

Autonomic networks enable product innovation, network services innovation, operational efficiency for networks and services and smart and intelligent networks that exhibit self-* features such as self-configuration, self-repair/healing, self-protection, self-optimization and self-awareness. The industry consensus is that as networks evolve to future networks, networks and services need to be operated based on principles for dynamically adaptive “automated” and “autonomic” management & control of networks and services. A recent industry workshop organized by IWPC, dubbed “Forging a Path to Autonomous 5G Networks” [29], is another example of the growing calls for evolution of management and control of networks and services through autonomics. GANA and efforts in ETSI NTECH AFI on standards for autonomics were presented at this IWPC workshop.

Therefore, standardization of the technology of Autonomic Management & Control (AMC), that reflects dynamically adaptive “automated” and “autonomic” management & control of networks and services, has to address the following key requirements:

- **Need for generic reference model** that enables developers of Autonomic Functions (AFs) to identify the abstractions at which autonomic functions (referred to as Decision-making-Elements (DEs)) can be designed and interworked within node architectures and the overall network architectures;
- **Need for instantiation cases of the generic model** onto implementation-oriented reference architectures (existing or future architectures);
- **Need for an implementation guide** for the reference model and associated instantiation cases;
- **Need to build trust & confidence in autonomics** by addressing stability of control loops and quality & correctness of the decision-making element algorithms and logic of the embedded autonomic functions;
- **Need for Proofs of Concept (PoCs) of AMC** based on the standardized reference model;
- **Need for interworking & harmonization** of autonomic management & control and other complementary paradigms, e.g. SDN, NFV, E2E Service Orchestration, to move from silos towards harmonized and combined paradigms within a unifying architecture.

There is thus an onus on the autonomics standardization community to provide the answers to those requirements, and this is in fact the principal purpose of the creation of the ETSI NTECH AFI group in 2009. ETSI NTECH AFI Working Group (WG), as the leading SDO in the standardization landscape for AMC, has a comprehensive work programme which comprises deliverables on:

- **a reference model** for a Generic Autonomic Network Architecture (GANA) [3];
- **an implementation guide** for the GANA reference model [10];
- **autonomics-enabled implementation-oriented architectures** that are a result of GANA instantiations onto various reference architectures defined by standardization organizations such as: 3GPP, Broadband Forum (BBF), IEEE, ITU-T, and other SDOs [13], [14];
- **a specification of the Proof-of-Concept (PoC) Framework** aimed at encouraging the industry to set up demonstrations of autonomic management capabilities in services and networks [16].

Therefore, this white paper primarily covers the following topics and objectives:

- Understanding the core concepts of the GANA reference model as the first standard that translates the industry requirements in terms of need for dynamic/adaptive autonomic management & control of networks and services and the need to define the abstraction levels at which closed control-loops can be designed and interworked in a hierarchical and nested fashion;
- Answers to the question: to whom is the GANA Model addressed?
- Understanding how to apply the GANA reference model to well-known architectures (3GPP, BBF, etc.) using the GANA Implementation Guide;
- Insights on the relationship between GANA and SON;
- Insights on how to integrate/interwork the GANA model and complementary paradigms such as SDN, NFV, E2E orchestration of resources and services, Big Data analytics applications for AMC;
- Insights on how the GANA model should play a vital role in the white box networking paradigm

The Emerging AMC Paradigm

Definition of the AMC paradigm

Autonomic Management & Control (AMC)

AMC is about Decision-making-Elements (DEs) as autonomic functions (i.e. control-loops) with cognition introduced in the management plane as well as in the control plane (whether these planes are distributed or centralized).

Cognition (learning and reasoning used to effect advanced adaptation) in DEs, enhances DE logic and enables DEs to manage and handle even the unforeseen situations and events detected in the environment around the DE(s).

Control is about control-logic as the kernel of the DE that realizes a control-loop in order to dynamically adapt network resources and parameters or services in response to changes in network goals/policies, context changes and challenges in the network environment that affect service availability, reliability and quality. Such control-logic is called a Decision-making Element (DE) in GANA terminology.

DEs realize self-* features (self-configuration, self-optimization, etc.) as a result of the decision-making behaviour of a DE that performs dynamic/adaptive management and control of its associated Managed Entities (MEs) and their configurable and controllable parameters. Such a DE can be embedded in a network node (Network Element (NE) in general) or higher at a specific layer of the outer overall network and services management and control architecture. An NE may be physical or virtualized (such as in the case of the NFV paradigm).

From an architecture perspective, a control-loop can be based on a distributed model (for fast control-loops). In this case the DE is embedded in the nodes (physical or virtualized). Whereas in a centralized model (for slow control-loops), the DE is embedded (implemented) outside of the network nodes. Both kinds of control-loops act towards a global goal to ensure a stable state of the network. A DE can negotiate with another DE to realize dynamic adaptation of network resources and parameters, or services, via reference points.

This leads to the notion of global network autonomics, a result of interworking DEs as collaborative manager components that perform AMC of their associated MEs and their parameters.

From an implementation perspective, a DE, as a software module or an executable behavioural specification that enhances intelligence capabilities, may be (re)-loaded or replaced in nodes and in the network's centralized management and control plane. This is directly related to the notion of software-driven networks or software-empowered networks.

Indeed, DEs (software components)are meant to empower the networks and the management and control planes to realize self-* properties: auto-discovery of information/resources/capabilities/services; self-configuration; self-protecting; self-diagnosing; self-repair/healing; self-optimization; self-organization behaviours; as well as self-awareness.

AMC also includes the following aspects for dynamic, intelligent and adaptive management and control of networks and services (even when taking into account the emergence of SDN (Software-Defined Networking)):

- Real-time reactive and proactive network analytics that should be instrumented at various layers of the management and control realms for networks. Network analytics involves strategies and techniques to gather various data (e.g. monitoring data) and analyse the data, so as to infer changes in the state of network resources and deduce any patterns that help build knowledge pertaining to network state transitions, event predictions, and forecasts. The analysis process and the knowledge built is used to decide actions that can be performed to achieve certain objectives;
- Dynamic network policing and dynamic service(s) policing;
- Self-* features such as self-organizing network behaviours, self-configuration; self-protection; self-diagnosis; self-repair; self-healing; self-optimization; self-awareness;
- Autonomic services management (on-demand orchestration and dynamic adaptation/re-programming of services);
- Network applications that provide for network intelligence by controlling the network via the northbound API of an SDN controller (e.g. a hybrid SDN controller—one that exhibits a multi-protocol southbound interface to diverse virtual and physical networks);
- In-network management for aspects requiring in-network reaction and self-adaptation by a thinly instrumented in-network control plane. The in-network control plane could be complemented by an outer and more complex logically centralized control plane that is split from the data plane as in the case of SDN.

In a nutshell , AMC is the key to designing the network and management & control intelligence (software logic) that enables the network and associated management and control operations to dynamically self-adapt to operator's high level business goals and policy changes, challenges to the network (i.e. manifestations of faults, errors, failures, performance degradation) and workload conditions of operation. To achieve AMC, real-time and predictive network analytics (also including predictive and proactive actions) for dynamic network policing and services (re-)programmability as driven by changes in context, workload scenarios, security and services requirements, must be introduced in the network architecture designs and the resulting network infrastructures that get deployed.

Automated Management

Automated management is about workflow reduction and automation i.e. automation of the processes involved in the creation of network configuration input using specialized task automation tools (e.g. scripts, network planning tools, policy generators for conflict-free policies).

Autonomic Management & Control (AMC) vs Automated Management

Autonomic management can be contrasted to automated management. The former emphasizes learning, reasoning, and adaptation, while the latter focuses on efficient workflow implementation and automation of the processes involved in the creation of network configuration and monitoring tasks. Figure 1 illustrates the positioning of both paradigms and highlights the interaction between them.

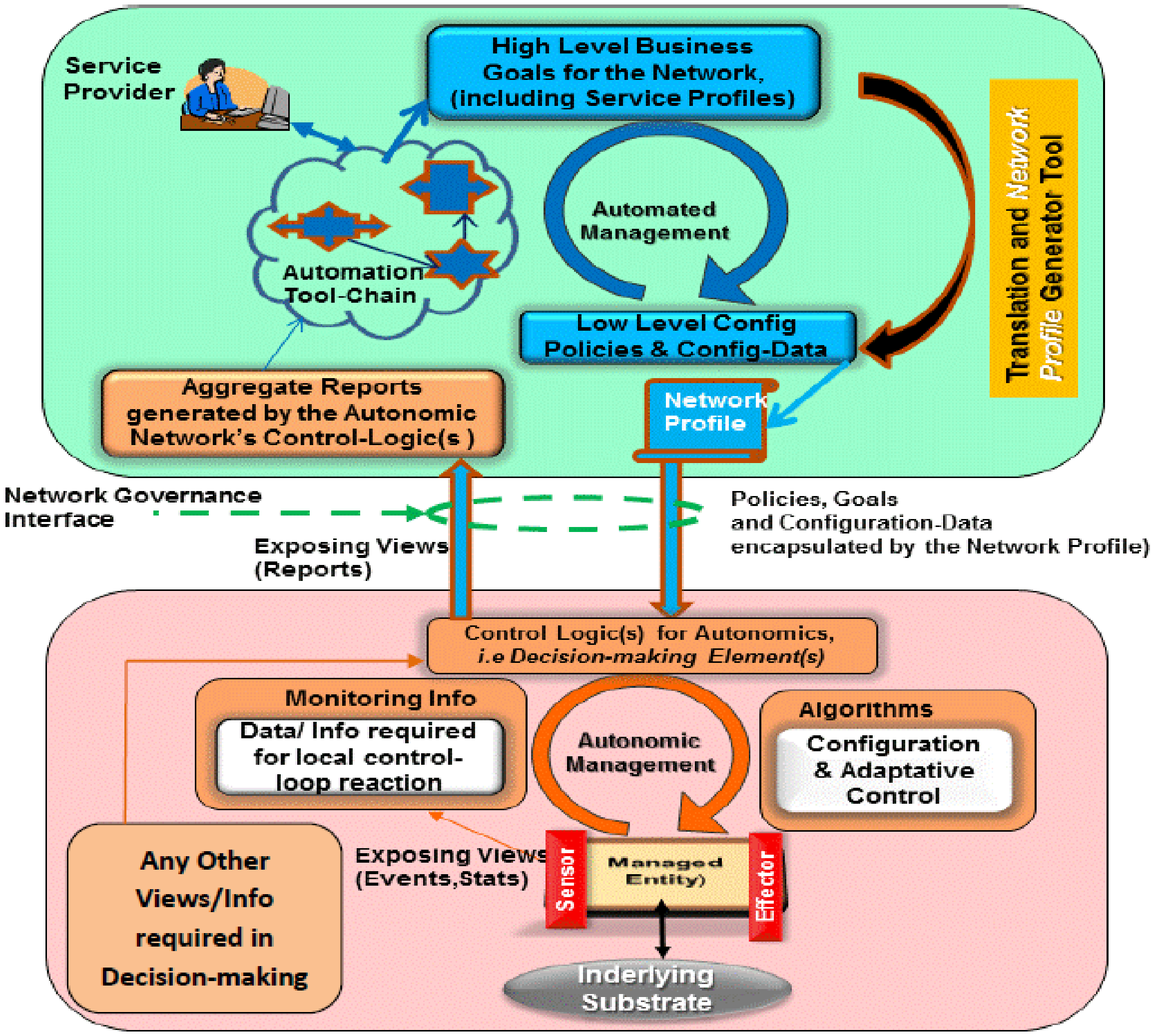

Figure 1: Automated Management vs Autonomic Management illustration (their interaction and complementarity)

Automated management provides input to the AMC. Indeed, autonomic management must exhibit a network governance interface through which the input that governs the configuration of an autonomic network should be provided. Thanks to automation tools and mechanisms (automated management), by using a high level language the operator can define the features of the network services that should be provided by the underlying network infrastructure. Such a business language that can help the operator express high level business goals required of the network may be modelled by the use of an ontology to add semantics and enable machine reasoning on the goals. The human operator defined features relate to business goals, technical goals and some input configuration data that an autonomic network is supposed to use for network resources and parameter configuration.

The GANA Reference Model

Design principles of the GANA Reference Model

Figure 2 presents a snapshot of the Generic Autonomic Network Architecture (GANA) reference model. The GANA reference model [2], [3] defines generic Functional Blocks (FBs) and their associated reference points and characteristic information (messages conveyed through those reference points). All of which are specific to enabling autonomics, cognition and self-management in a target architecture when instantiated onto an implementation-oriented reference architecture such as BBF (Broadband Forum) architecture, NGN/IMS architecture, or 3GPP architecture [28].

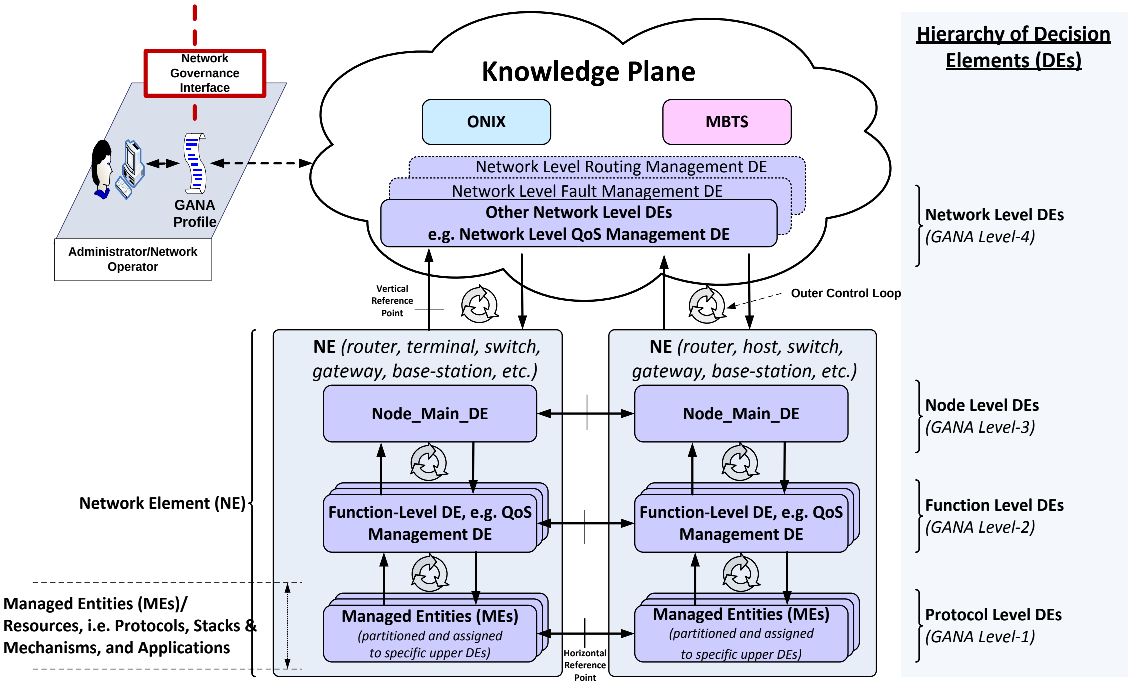

Figure 2: Snapshot of the AFI GANA Reference Model

Distinctively performed GANA instantiations enable to create various types of autonomics-enabled reference architectures, e.g. autonomics-enabled BBF architecture, etc. Note: autonomics is also synonymously referred to as autonomicity in literature.

The GANA reference model can also be applied in designing future network architectures that must exhibit self-management capabilities from the outset of their design. GANA is a generic model in the sense that it defines and separates generic concepts and associated architectural principles for the domain of autonomic networking, cognitive networking and self-management from implementation strategies, details and methods that can be used to implement them. Hence, it is not constrained by any implementation-oriented architecture and to the extent possible avoids inheriting the limitations of today’s technology-specific network architectures.

Concept of GANA DE (autonomic Decision-making-Element)

The GANA model serves as a blueprint model that defines and prescribes the design and operational principles of autonomic Decision-making Elements (as autonomic manager components) called DEs. DEs are also called autonomic functions and are responsible for autonomic management and adaptive control of systems and network resources, parameters, and services. Autonomic behaviours of a DE include secure auto-discovery of the following items: network objectives and policies specified by the human operator, other DEs it requires to collaborate with, and capabilities of the DE's assigned Managed Entities (MEs)—i.e. the information that gets available at run-time. Then after auto-discovery, a DE performs the self-* operations on its assigned MEs (by-design) by orchestrating (launching) the MEs when required, and adaptively (re-)programming the MEs via the effectors of their management interfaces. A DE is designed to perform one or multiple self-* operations such as self-configuration, self-diagnosing, self-healing, self-repair, self-optimization, self-protection. Some specialized DEs may be designed to perform certain self-* operations on a macroscopic level that takes into account wider perspectives needed to complement the same self-* operations intrinsically performed by DEs on the microscopic level. Therefore, in general, a DE is said to realize or implement self-* features. What drives a DE to perform its operations in collaboration with other DEs whenever required are various input information and changes that drive its algorithms, such as changes in the operational state of its ME(s), changes in the governing input policies, context changes, and challenges (e.g. faults, errors, failures) detected in the operation of the MEs and the underlying network substrate.

As discussed in [3], approaches such as FOCALE principles [4], MAPE-K control-loop [7], and other principles validated in various research projects on autonomics can be applied to designing a DE's internal logic and associated control-loops.

Concept of GANA MEs (Managed Entities)

GANA adopts the concept of a Managed Entity (ME) to mean a managed resource, instead of a managed element- a term used in traditional network management terminology and normally intended to mean only a physical Network Element (NE) and not some functional entity within a node/device such as a protocol module or a component (e.g. a monitoring component).

MEs can vary: they can be fundamental MEs at the bottom of the management hierarchy (at the fundamental resources layer, see Figure 2) such as individual protocols or stacks, OSI layer 7 or TCP/IP application layer applications and other types of resources or managed mechanisms hosted in a network node (NE) or in the network in general. MEs can also be composite MEs such as whole NEs themselves (i.e. MEs that embed sub-MEs). It is noted that an NE can be physical or virtual.

Why call the behaviours of DEs self-* features?

The behaviours of DEs are called self-* features because humans are relieved from having to perform the traditionally manual management-oriented tasks, and software, i.e. the DEs, automate the tasks and dynamically perform the tasks based on human specified networking goals and policies, context or state changes, and events detected in network nodes and the network.

A DE should receive as input network goals or governance policies specified by the human operator, and also auto-discover other DEs the DE requires for DE to DE collaboration, and the capabilities of its assigned managed entities (MEs) before the DE starts performing the self-* operations it is designed to perform. Such DE behaviours should be performed by individual DEs embedded in NEs (for self-management that are driven by local reactions within the NE).

Horizontal (distributed) DE-to-DE collaboration: Some DE algorithms may require the collaboration of a DE within an NE with other DEs along an end-to-end (E2E) path in the network, though not necessarily involving hop-by-hop NEs, for a self-* operation (e.g. self-optimization) that may require the collaboration of distributed DEs along a path in the network.

Vertical (Hierarchical) DE-to-DE collaboration: Another possibility is that actions of an NE's DEs may also need to be synchronized by higher level autonomic behaviours coordinated by upper layer DEs (outside the NE) at the GANA Knowledge Plane (KP) level (KP concept is defined below).

Concept of Knowledge Plane (KP) in the GANA Model

The GANA Knowledge Plane (GANA KP) enables advanced management & control intelligence at the Element Management (EM), Network Management (NM) and Operation and Support System (OSS) levels by interworking with them or enhancing and evolving the intelligence of the systems at these levels by way of replaceable and (re)-loadable autonomics modules (DEs) that can be loaded at specific abstraction levels of management and control operations (more details in [13], [3]).

The ANA KP concept is inherited from the Knowledge Plane concept defined in [5] as a pervasive system within the network that builds and maintains high-level models of what the network is supposed to do, in order to provide services and advice to other elements of the network. As illustrated in figure 2, GANA KP's DEs should be complementarily designed to collaborate with DEs at lower layers of the GANA model.

GANA as a holistic unifying model for the well-established models for AMC

GANA accommodates and unifies concepts from the well-established models for autonomics. The ETSI NTECH AFI Group fused a number of leading autonomics efforts/models, including FOCALE [4], IBM-MAPE [7], 4D architecture [6], Knowledge Plane for the Internet [5] and other models, and developed the GANA, as a unified reference model for AMC. This subject, on how concepts from the various models are unified and fused together (accommodated) in GANA is discussed in [3].

GANA Core Concepts

Figure 2 presents the GANA abstraction levels for self-management functionality at which interworking hierarchical/nested control-loops and their associated DEs can be designed. Figure 2 defines the key Functional Blocks (FBs) for enabling and implementing autonomics in target implementation-oriented architectures, as described in the sub-sections below. A table of a summary of all GANA FB reference points and characteristic information descriptions is given in chapter 13 of ETSI GS AFI 002 [3].

Managed Entities (MEs) at the bottom level (the fundamental resources layer in GANA)

At the bottom of the management & control hierarchy (see Figure 2) are the fundamental MEs that are hosted in a network node (NE) and can be employed, orchestrated, configured and adapted to achieve some goals. These form the fundamental resources layer MEs.

Decision-Elements/Engines (DEs) and Decision Plane Hierarchy

GANA DE design is based on the need for hierarchical abstraction levels for self-management (autonomic) functionality. The hierarchy of DEs is meant for implementing AMC of MEs, at various levels of abstractions of self-management functionality, with DEs viewing their assigned lower DEs in the management and control hierarchy as MEs of some sort (since upper DEs manage and control lower level DEs, using policy control for example). The GANA defines four levels of self-management functionality, i.e. levels of abstractions at which DEs and their autonomics control-loops can be designed. The levels are described in more detail in the corresponding sections that follow. Lower GANA level DEs are viewed as MEs by upper GANA level DEs, which inductively control lower level DEs through policies.

Remark: It does not mean that in order to implement Autonomic Functions (AFs), meaning DEs, every hierarchical level in GANA has to be implemented in the target architecture. Because in incremental implementation of autonomics in a network architecture, one particular GANA level or multiple GANA levels and associated DEs may be collectively considered at a time. A full implementation of interworking autonomics at multiple levels (especially Level-2 to Level-4) may simply emerge over time. More discussions on this subject are provided later in the section on the Implementation Guide for GANA.

The value of autonomic behaviour driven by a particular DE instantiated in a particular environment is in the DE's behaviour in monitoring the events/views exposed by its assigned MEs and their execution state, reasoning about the exposed views together with any other input from the environment required by the DE's algorithms. Then deciding on whether to (re)-configure their MEs and their parameters so as to achieve certain objectives of local scope to the DE or requiring collaboration with other DEs. DEs may collaborate horizontally or vertically in the decision plane (see Figure 1 and Figure 2).

The fundamental MEs at the resources layer often have their management interface or Managed Objects (MOs) specified using data modelling structures such as MIBs (Management Information Bases). Such definitions of MOs can be part of the basis upon which DEs that dynamically infer state of the MEs and perform dynamic ME parameter configuration, can be designed to operate. In addition to the MOs other views/info that are external to the MEs are also relevant to be used by the DEs' logic/algorithms (Figure 1).

This value of autonomics will always continue to be the subject of research and innovation. In some network environments there may be certain network and policy control dynamics that require certain DEs and associated control-loops (for purpose of collaboration) to be instrumented. Communication methods (e.g. protocols) for DE-to-DE communications horizontally or vertically in the Decision Plane, outside of NEs, would need to be developed in SDOs such as IETF [18], if existing protocols of today cannot be applied for DE-to-DE communications outside of NEs. The GS AFI002 specification [3] provides guidelines on principles that can be applied for designing DE logic, including external and internal structural models of a DE, interfaces and primitives that should be supported and implemented on DE interfaces.

Therefore, in general, self-manageability of a network and services using GANA is achieved through instrumenting the NEs and an outer realm called the GANA KP with DEs that collaboratively work together in realizing self-* features in nodes and a network as a whole. Nodes' DEs may form peers along a path within the fundamental E2E transport/data-plane architecture (Figure 2).

GANA Knowledge Plane Functional Blocks (FBs)

The **GANA KP** (defined earlier) consists of the following FBs, namely:

- **Network Level DEs,** whose scope of input is network wide. They are designed to operate the outer closed control loops on the basis of network wide views or state as input to the DEs' algorithms and logics for autonomic management.
- **ONIX (Overlay Network for Information eXchange)** (distributed scalable overlay system of information servers).The ONIX is useful for enabling auto-discovery of information/resources of

an autonomic network via “publish/subscribe/query&find” protocols. DEs can make use of ONIX to discover information and entities (e.g. other DEs) in the network to enhance their decision making capability. More details on ONIX are given in the ETSI AFI GANA specification [3]. The ONIX itself does not have network management & control decision logic (as DEs are the ones that exhibit decision logic for AMC).

- **MBTS (Model-Based Translation Service)** which is an intermediation layer between the GANA KP DEs and the NEs (physical or virtual) for translating vendors’ specific raw data onto a common data model for use by network level DEs, based on an accepted and shared information/data model. KP DEs can be programmed to communicate commands and process NE responses in a language that is agnostic to vendor specific management protocols and technology specific management protocols that can be used to manage NEs. The MBTS translates DE commands and NE responses to the appropriate data model and communication methods understood on either side. More details are given in the ETSI AFI GANA specification [3]. The concept of MBTS [4] uses a common model that is also accommodated and unified together with other models into the GANA Model.

Network Governance Interface for Governing the Autonomic Network

The network governance interface is an interface through which humans (through the support of automation tools) generate GANA network profiles (GANA profiles) to be provided as input to the autonomic network through the GANA KP. With the help of automated management tools, the human operator (administrator) creates a GANA profile in form of a data structure that contains the networking objectives/goals, policies and certain types of configuration data that govern the autonomic network. The network profile is then used by the autonomic network to generate lower level policies and configuration data to configure the DEs and the network.

GANA Abstraction Levels for Self-Management (autonomic) Functionality

The GANA reference model defines a hierarchy of DEs. It is organized in four basic hierarchical abstraction levels of self-management (autonomic) functionality (presented below in a bottom-up approach):

- protocol level DE; (lowest level)
- function level DE
- node level DE;
- network level DE (highest level)

Each DE manages one or more lower-level DEs. Table 1 describes the four GANA levels.

GANA reference model hierarchical level	Description
Level 1: Protocol level DE (lowest level)	Relates to any managed entity (ME) such as a protocol or other fundamental mechanisms that may exhibit intrinsic control-loops (DE logic) and associated DE—as is the case for some of today’s protocols such as OSPF, which can be considered an example of the instantiation of a protocol-level DE (though such autonomic-like feature in OSPF is not cognitive (learning and reasoning) in its operation and by design). The GANA Specification puts forward a

	recommendation to primarily focus on the three higher GANA levels of hierarchical control-loops (Level2 to Level4) when introducing autonomics in architectures and considers the protocol level DEs as MEs at the resources layer(along with any other fundamental MEs). The GANA hierarchy emphasizes only the three other levels which should collaboratively work together (Refer to ETSI GS AFI 002 [3]).
Level 2: Function level DE	Relates to a DE for collective AMC of a group of protocols and mechanisms that are abstracted (viewed like a bundle) by a networking or a management/control function. GANA specifies the following six function level DEs: routing management-DE; forwarding management-DE; Quality of Service management-DE; mobility management-DE; monitoring management-DE; service and application management-DE (Refer to ETSI GS AFI 002 [3] for more details on the types of DEs for this level and their associated types of MEs). The control-loop is external to the MEs subscribed to the function (by virtue of abstraction). Multiple DEs at this level are determined by the functions required of the NE.
Level 3: Node level DE	Relates to a DE for AMC of those aspects that cover and restrict the behaviour of the NE as a whole, as well as the orchestration and policing of the function level. GANA Level 3 specifies the following four DEs: Security management DE, fault management DE, auto configuration and discovery DE, resilience and survivability DE. Those four DEs are collectively referred to as the Node-Main-DE. It is because such autonomic management and control functions are the superior ones within a node, as they should operate on the level that globally regulates the node and its composition. (Refer to ETSI GS AFI 002 table 1 [3] for more details on the types of DEs for this GANA level and their associated types of MEs)
Level 4: Network level DE	Relates to a DE for AMC of those aspects that cover network-wide views and the management & control of lower levels e.g. node/device levels, as well as the policing of the lower levels (e.g. node Levels). Such a DE is designed to operate in a logically centralized manner. The control-loops at this level complement lower level control loops by operating on a slower timescale (i.e. they are slower control-loops in contrast to lower level control-loops (the faster control-loops)). The network level DEs constitute the functional blocks of the Knowledge Plane, together with ONIX (Overlay Network for Information eXchange) and MBTS (Model-Based-Translation Service). (Refer to ETSI GS AFI 002 [3] for more details on the types of DEs for this level and their associated types of managed entities).

Table 1: GANA Reference Model's four layers of abstractions for self-management operations

Recommendation to focus on GANA levels 2 to 4 when introducing autonomics in architectures

Though the GANA reference model defines four basic levels of self-management, three levels, level-2 to level-4 are the most important ones when one considers not embedding a control-loop into individual protocols. That means avoiding protocol-intrinsic control-loops (protocol-level-DEs) since they tend to create undesired emergent behaviour in complex protocol interaction scenarios which may be difficult to study and eliminate, as already experienced in many cases today [19], [6]. As such, it has often been noted in both the industry and research communities [19], [6], that embedding intelligence in protocols (often independently developed) through the use of control-loops has made protocols complex and difficult to manage. As a result, Control-loops involving dynamic management of protocols should rather be implemented at an abstraction level (function-level) that is outside of individual protocols.

A DE designed to operate at such a level (function level) aggregates events and makes decisions on when and how to orchestrate or (re)-configure MEs (protocols and parameters) in order to adapt to changes in networking policies, events and faults/errors/failures detected in the NE and network.

Three levels of hierarchical control-loops (GANA level-2 to level-4) demonstrate how AMC can be gracefully (non-disruptively) introduced in today's existing networks and architectures and even in new network architectures that follow the approach of designing and employing protocols to build protocol stacks in which individual protocols are rather simple and do not embed any intrinsic control-loops.

Instantiation of GANA Decision Elements/Engines (DEs) onto target architectures, and fundamental DE behaviours that could be standardized

In the process of the instantiation of DEs onto target implementation-oriented architectures specified by SDOs, such as the Broadband Forum (BBF) reference architecture or 3GPP reference architecture, the DEs that must be instantiated in particular NEs (nodes) must be chosen on the basis of a criterion. This could be the managed networking resources a NE supports and its point of attachment as well as its role in the network topology.

Once this decision has been made the DE behaviours and behaviours of the other DEs and GANA FBs must be further specified in more detail, based on analysing various use case scenarios and requirements for autonomics and self-management in the particular target reference architecture. This also leads to further elaboration of the generic behaviours of the instantiated GANA FBs and their characteristic information exchange on the instantiated reference points (also the protocols used to exchange information and messages are then nailed down). The fundamental DE behaviours that need to be standardized versus those behaviours (e.g. customized DE algorithms) that may not be standardized need to be discussed and agreed in the standardization process. Characteristic information exchanged over the GANA reference points and the protocols used to convey it become more concrete and detailed during the phase of GANA instantiations and autonomics use cases requirements analysis in the target implementation-oriented architecture. ETSI GS AFI002 [3] presents a table on GANA FBs and associated reference points (rfps) and generic characteristic information exchanged through those rfps. The table should be used as a basis for further elaboration required during GANA instantiation and implementation.

DE-to-DE peer communication relationships along an E2E (End-to-End) path in the network

For some distributed algorithms (e.g. for optimization) in the data plane some DE-to-DE communication may be required along the horizontal DE-to-DE reference point scoping the NEs (or nodes) of interest to an algorithm. This is up to innovators of self-optimization algorithms to determine the distributed nature of their algorithms. But alternatively the DE algorithm may be implemented using the centralized approach through the network level DEs. In the centralized approach, information from the NEs of interest would be relayed to the network level DEs that implement the coordination and management of the behaviours of NEs. In an SDN (Software-Defined Networking) environment, such information may be relayed through the SDN controllers to the Network level DEs, which run as applications on top of the SDN controller northbound interface (Figure 6) [9], [13]. Some protocols for DE-to-DE communication along a path could be developed in bodies such as IETF [18].

Stability of Control Loops in GANA, Types of Autonomic Functions Within and External to an NE, and Other GANA Design Principles

Techniques for addressing stability of control-loops and avoiding oscillations are discussed in [8] and are elaborated further in the GANA specification [3]. There is a framework proposed for run-time coordination of GANA DEs towards achieving stability of interacting control-loops across the Hierarchical Levels of Control Loops [8].

The place-holders for internal control-loops (inside an NE [3],[9],[10]) depicted by the GANA reference model enable to design and embed DEs in the NE to include node-local self-management behaviours/algorithms. Node-local self-management such as self-optimization implies some degree of NE intelligence through the internal DEs that realize the internal control-loops. Example NE-scoped DE self-* behaviours that may not necessarily require collaboration/negotiation with other NEs include: plug-n-play through autonomic functions; autonomic security management (self-protection and self-defending behaviour against security attacks and threats); autonomic fault-management and resilience (proactively and reactively), etc.

[9] and [1] note that various validated aspects of the earlier version of the GANA Model (before it was then evolved by the ETSI AFI Industry Specification Group (ISG)) included derivation, prototyping and validating autonomic behaviours of specific DEs in GANA instantiations. Such GANA instantiations sought to introduce various autonomic functions (AFs) in specific implementation-oriented reference network architectures. The validated aspects included various instantiated DEs (autonomic functions) such as DEs for autonomic mobility management, DEs for autonomic QoS management, DEs for autonomic routing, DEs for auto-discovery, DEs for auto-configuration/self-configuration, DEs for autonomic resilience & survivability, etc. [3].

Interworking GANA Nested Hierarchical Autonomic Control-Loops

The basic principle is that fast control loops implemented at lower GANA level(s) (2-3), i.e. intrinsic to a NE, should be complemented by outer slower control loops in the GANA KP (a logically centralized construct). Control loops within NEs are “fast-control-loops” (they operate in shorter time-scale in reaction to events and incidents). As we go up the GANA decision plane hierarchy into the GANA KP, control-loops become “slower” but more sophisticated due to the network-wide scope of the processed information in realizing self-optimization. Coordination of the control-loops can be achieved through the framework proposed for run-time coordination of GANA DEs [8]. Literature refers to two types of hierarchical levels of autonomics space:

- **micro-level autonomics**: DEs in the node (GANA Level-1 to Level-3),
- **macro-level autonomics**: autonomics driven by the GANA KP, i.e. GANA network-Level DE.

From Zero or Very Limited Cognitive Behaviour at Lower GANA Levels to Complex Cognitive Algorithms in the GANA Knowledge Plane (Level-4)

As a fast control loop acts on data sets of limited scope as required for fast reaction by the driving DE in (re)-orchestration and/or (re)-configuration of MEs, such lower level DEs may exhibit zero or rather very limited cognitive behaviour. But the DEs that drive the slower control loops, particularly higher up into the GANA KP, are the ones that must exhibit complex cognitive algorithms for state prediction, forecasting and planning capabilities required for executing changes to the behaviour of the network’s resources. The network-wide and huge data sets and knowledge the network level DEs must operate on mandate the need for dedicated high performance hardware such as COTS (Commercial Off-The-Shelf) servers (or commodity hardware) that could be similar to the type used for NFV (Network Functions Virtualization). COTS hardware for the GANA KP should be well equipped with run-time compute, memory, and storage resources for the KP; as such hardware is required to run complex cognitive algorithms for autonomics. Hence the presence or complexity of cognitive behaviour should increase as we go up the GANA decision plane hierarchy into the GANA KP [3].

Example Use Cases of GANA in a Target Network Architecture

There are various instantiation cases of GANA onto target architectures, including the derivation of autonomic behaviours of instantiated DEs. Some of such GANA instantiation work is being carried out in ETSI NTECH AFI to guide developers of autonomics software in introducing autonomics in the target architectures. The work includes GANA instantiation onto mesh network architecture, GANA instantiation onto the BBF (Broadband Forum Architecture), GANA instantiation onto the 3GPP Core Network and Mobile Backhaul Network, and GANA instantiation onto the IMS architecture.

Interworking Fast Control-Loops in an NE(s) with Slow Control-Loops in the GANA Knowledge Plane

This section focuses on providing an illustration of how a local reaction by a DE that is meant to cover certain management and control aspects within a network node (for fast control-loop(s)) can be complemented by a global reaction by a DE for similar management and control aspects that operates on the Network Level (GANA Knowledge Plane Level) and implements the slower outer control-loop(s).

The case of autonomic management and control of routing and energy saving

Figure 3 illustrates such a use case. Though the figure considers routing in IPv6 networks, the case also applies to other networks such as IPv4 networks. The information/knowledge sharing system in Figure 3 can be realized by the GANA FBs: ONIX and MBTS.

The outer control-loop is slower (operates mainly in longer term time scale) but scopes wider network views (state) that it uses for policing the lower control-loop. It is also used for coordinating those tentative actions the lower level DE needs to synchronize with the upper DE for approval. According to a proposed framework for synchronization of actions and policies in GANA [8], there are actions a node-local DE can be designed to perform without requiring approval (synchronization) by an elected arbiter/coordinator DE on the same level or on an upper level (node level), or even up on the network level. The local fast control loop’s local reaction may be driven by local monitoring data on the device or node level.

Regarding the use of the Complex Event Processing (CEP) paradigm and its role in relation to the interworking of fast control loops and slow control loops: there is a need for coupling CEP capabilities that should be operating in the GANA Knowledge Plane (KP) with policies applied by the KP DEs. CEP in the KP could be realized in two ways: CEP happens in each DE in the GANA Knowledge Plane or a single instance of a CEP module that feeds all the DEs with events that the DEs must act upon could also be used. To facilitate the interworking of fast control loops and slow control loops in CEP, autonomic network nodes should push up into the GANA Knowledge Plane DEs aggregate events concerning local DEs’ actions and other aggregate events/views inferred from monitoring data. GANA nodes should also relay synchronization requests from lower level DEs that require that a DE on the network level coordinates the approval of a tentative action the lower level DE intends to execute.

Case of risk-aware routing being executed by the local Function-Level Routing Management DE algorithm only as a local-reaction

For example [22] presents the case of risk-aware routing in which temperature rise in the NE is mapped to levels of increase in probability of router failure. As the monitored router temperature rises to a certain threshold this triggers the local reaction by the Function-Level Routing Management DE in raising OSPF links weights on interfaces (and advertise high values). The result is to effectively discourage other routers from selecting the router in forwarding traffic (as a consequence, the router is eventually not selected for forwarding anymore). This reduces the risk of packets loss and service disruptions in the case of continued use of such an affected router in traffic forwarding, and the situation can be restored back when the heating in the router goes down.

Such a behaviour can be executed by the local Function-Level Routing Management DE algorithm without requiring the intervention of the upper Node-Main-DE or the outer Network Level Routing Management DE.

Case of certain tentative actions of Function-Level Routing Management DE algorithm requiring the intervention /coordination(synchronization) of its action by an upper DE

Indeed, there are cases when the decision of the Function-Level Routing Management DE (as a tentative action) needs to be coordinated (synchronized) by the Node-Main-DE's security enforcement part before it can be approved for execution or not. Coordination may be required on global node level or up to the network level. For example if the lower level DE intends to create a new instance of OSPF (and advertise on additional interfaces) in response to problems (faults/errors/failure manifestations) detected locally with other instances or in attempts to achieve some load balancing objective. In such a situation, the DE (Function-Level Routing Management DE) may need to synchronize with Node's Security-Management DE (a sub-DE of the Node-Main-DE) for approval of the tentative action on the node level. The Node's Security-Management DE checks the tentative action against security policies of the node and approves or disapproves accordingly. The synchronization of actions and policies in cases such as the creation of new OSPF instances could go up to the Network Level Routing Management DE in the Knowledge Plane for approval (as consequences of a lower level DE action may be known only by an upper level DE). The Network Level Routing Management DE may even communicate new optimization strategies through policies to the lower level Function Level Routing Management DE for enforcement (based on global optimization strategies computed by the outer loop).

Here we presented the case with routing as example, but other DEs such as QoS Management DE, Mobility Management DE, Security Management DE, Fault Management DE, Resilience & Survivability DE, can also be designed to interwork as fast control loops complemented by slower control loops in the GANA Knowledge Plane.

Other cases of fast control-loops at GANA levels 2 and 3 not requiring coordination or action approval by upper DEs

Some cases for which fast control-loops at GANA levels 2 and 3 may react without needing to be coordinated or approved by upper DEs in the Knowledge Plane include the handling of certain types of errors and events reported or inferred about the behavior of the managed protocols (Managed Entities). Other scenarios in this category include the following:

Case of energy savings being driven by lower level control-loops (GANA levels 2 and 3)

An energy saving DE algorithm can be designed to work locally (autonomously) in some cases, or in a distributed fashion across multiple routers of some network scope. A distributed algorithm may be tailored to energy saving from the point of view of selected routers being temporarily switched off for some time. Then they can be brought up again after some time to participate in routing and forwarding traffic. In both cases, the algorithm may be implemented by the Function-Level Routing Management DE or the Node-Main-DE, and the distributed algorithm would use the corresponding DE-to-DE peer horizontal reference points accordingly, to perform the energy-saving negotiations. Local state information may be used by the decision-making algorithm either for autonomous (local) energy saving actions or in DE-to-DE horizontal negotiations on determining the routers that should be switched off at some point.

Other cases of fast control-loops at GANA levels 2 and 3 requiring coordination or action approval by upper DEs in the GANA Knowledge Plane

Work that looks into the following aspects can further inspire the design of the Network-Level Routing Management DE in the GANA Knowledge Plane and its interworking with the Function-Level Routing Management DE:

- Programmatic interfaces for programming routing system (protocols), such as the I2RS (Interface to Routing System) Interface being developed in IETF [23]. The Function-Level Routing Management DE can be viewed as a network application that runs locally on the routing NE, and can be integrated with a local client to dynamically program the routing while registering for events from I2RS as part of its input to its control-loop.
- There is work that advocates for co-existence of distributed and centralized control planes, with coordinated and converged management and programming of the distributed control plane and the centralized control plane. This ensures that no conflicting forwarding behaviour is installed in the network, and that business objectives are met in the hybrid environment [24].

Other cases of fast control-loops requiring coordination with slow control-loops, as reported in literature

In general there are various use cases in literature concerning the need to design and embed fast control-loops in nodes and interwork them with outer slower control loops that operate in a logically centralized management and control plane. The following aspects provide insights to developers of autonomics who intend to interwork fast control-loops and slow control loops by designing and interworking GANA DEs at appropriate levels:

- Splitting management and control aspects between fast control loops incorporated into the network element and slow outer control loops operating in the logically centralized management and control plane, with both types of control loops complementing each other (see for example [25], [26] and many other cases in literature). Interworking the two control-loops can be achieved using principles in [8].
- Some suitable Application Programming Interfaces (APIs) may be available on NE platforms that enable DEs (e.g. Routing Management DEs and other types of Function-Level DEs, possibly from second parties) to be integrated to run in the NE so as to dynamically and adaptively configure (program) the various protocols and mechanisms required to run on the system. If no APIs are available, DE developers may use other means such as SNMP protocols (and MIBs supported on the NE), and CLIs (Command-Line Interfaces).

Cases of fast control-loops at GANA levels 2 and 3 interworking with outer slow control-loops driven by the GANA Knowledge Plane instantiated for the 3GPP EPC Network

The work carried out in ETSI NTECH AFI on Autonomic and Self-Managed 3GPP Backhaul and Core (EPC) Networks has produced use cases that provide a picture on fast control-loops interworking with slower outer control-loops in a GANA instantiation for the 3GPP Core Network (the EPC). A Technical Report (ETSI TR 103 404 [28]), is expected to be published by ETSI within the timeframe of September-October 2016.

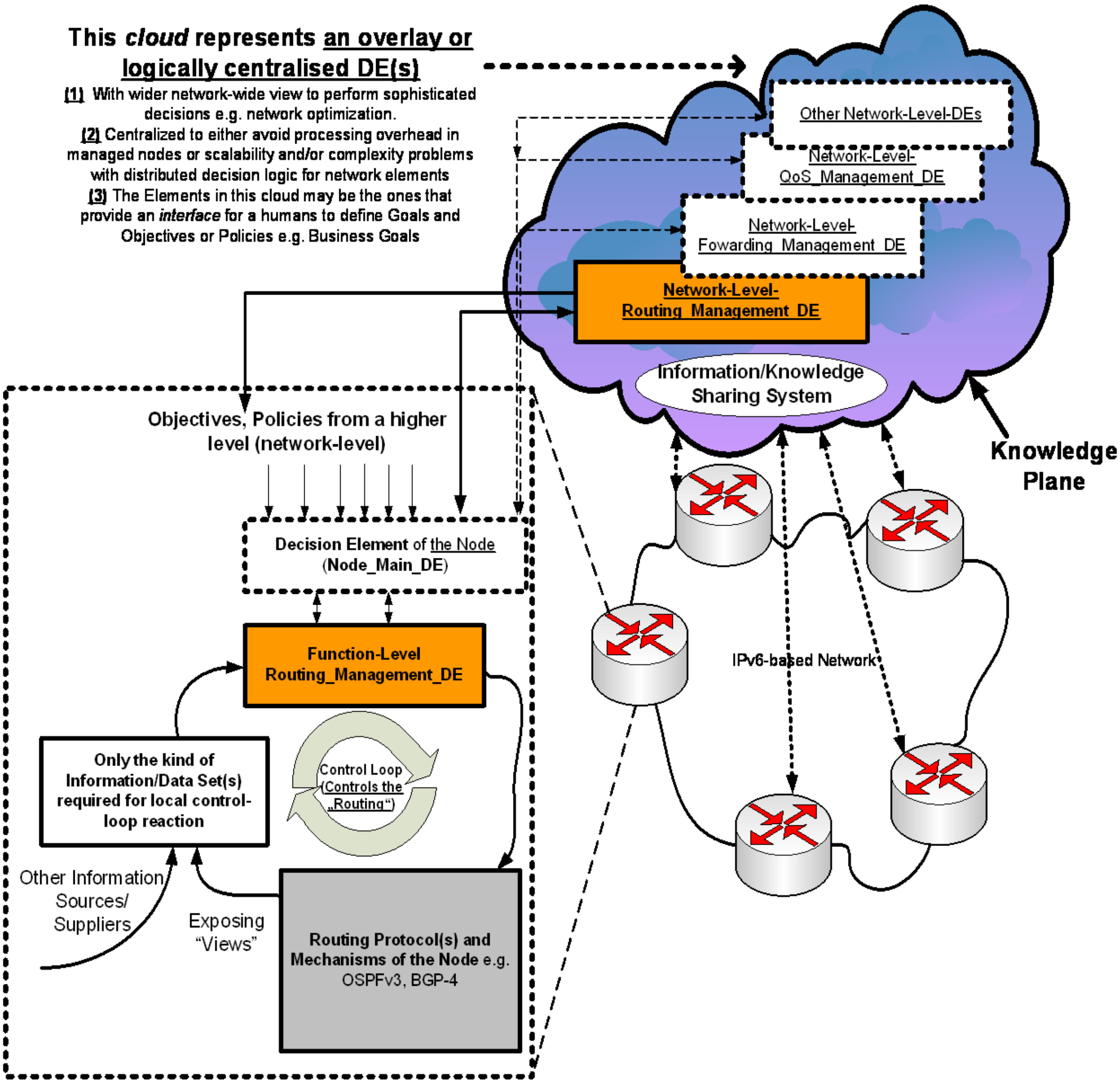

Figure 3: Illustrating the interworking of fast control-loop and slower control-loop for routing management (can be applied both to IPv6 and IPv4 environments)

To whom is the GANA Model Addressed?

The following two categories determine the actors or players the GANA model is addressing:

a) Suppliers (vendors) of GANA Functional Blocks (FBs)

The suppliers can be further categorized as follows, bearing in mind that DE algorithms, just as in the case with SON algorithms, may not be standardized as they should provide the means for DE vendor differentiation so as to facilitate for DE vendor differentiation (and actually there is a need to promote continuous innovation in autonomics algorithms):

- Independent developers of software components and algorithms for autonomics from the research community (research institutes, universities, etc.).
- ISVs (Independent Software Vendors) e.g. OSS (Operations and Support Systems) vendors
- Traditional networking equipment vendors
- Network operators who may have software development capabilities may develop some DEs on their own and load them into nodes (provided this can be supported by the host platform or operating systems) and/or in the Knowledge Plane.

b) Provider of assets required by the developers of GANA Functional Blocks (FBs)

Perspectives on such assets are as follows:

- GANA presents a framework to design autonomic functions (AFs) required at various GANA levels of abstraction for self-management functionality. The section on the implementation guide for GANA and [10] discuss the subject of how to implement, step-by-step, autonomics at various levels of abstractions defined by the GANA model. The GANA specification and other assets described in the section on the implementation guide for GANA and in [10] constitute useful input required by developers. Based on such inputs as discussed in [10], developers should then perform the steps described in the section on the implementation Guide for the GANA, while interacting with ETSI NTECH AFI WG on implementation guidance and helping close gaps in the autonomics standards and the frameworks.

Table 2 describes in more detail the actors/players and the roles attached to each actor. Indeed, each actor needs to know its related roles, rights, duties and responsibilities.

Actor 1	Suppliers (vendors) of the GANA Functional Blocks (FBs) (software components/modules/libraries, protocols and DE algorithms for autonomics). The Functional Blocks (FBs) defined by the GANA (such as the GANA KP FBs), their associated reference points and characteristic information exchange, and the GANA abstraction DE levels for autonomic components in general (particularly GANA Level-2 up to GANA Level-4 DEs), all determine the types of suppliers for the FBs. The roles described in this table provide a characterization of the types of suppliers of various GANA FB software components/modules/libraries, protocols and DE algorithms for autonomics.
Role 1	GANA KP component suppliers (for network-level DEs, ONIX, MBTS software libraries) and associated algorithms and protocols

Who should fulfil this role?	This role could be fulfilled by ISVs (Independent Software Vendors), e.g. OSS vendors, or traditional networking equipment vendors. Even network operators who may have software development capabilities may develop some DEs on their own.
Role 2	Suppliers of GANA Level-2 and Level-3 Decision Elements/Engines (DEs) and their associated algorithms
Who should fulfil this role?	Could be fulfilled by traditional networking equipment vendors and/or even ISVs. GANA defines an autonomic networking node/device internal reference point along with the structure of a GANA node and visualization of place-holders for control-loops (see in [9], [3]). Such an NE internal reference point could apply in some open networking boxes (i.e. some vendors provide an NE or device-internal interface for control-software agents to be loadable into the device). This would enable either the traditional network equipment vendor or the network operator to load DEs from a third party (e.g. ISV) to empower the device (physical or virtual) with third party developed DEs and algorithms. Also, with the advent of "White Box Networking", ISVs may develop these types of DEs (level 2 and possibly level 3 as well) and associated algorithms that drive the DEs' control loops. More details are included in the section "GANA and White Box Networking". Even network operators who may have software development capabilities may develop such DEs on their own and load them into nodes (provided this can be supported by the host platform or operating systems).
Role 3	Suppliers of GANA Level-1 autonomics (DEs and algorithms)
Who should fulfil this role?	This supplier role could be fulfilled mainly by traditional networking equipment vendors who often provide the protocol stacks that run in the equipment anyway.
Role 4	Providers of Independent DE Algorithms for any of the four GANA levels of abstraction of self-management functionality. This applies especially to GANA levels 2 to 4, as the protocol level (level 1) may not easily allow modifying some of the existing protocols to embed intelligence and control loops. But though this may be possible in certain newly developed protocols, the issue of control-loops in protocols leading potentially to undesired emergent behaviour (as discussed earlier) needs to be considered.
Who should fulfil this role?	This role could be fulfilled by algorithm developers for autonomics from the research sector (research institutes, universities, etc.) as new actors entering this autonomics market.

	As discussed in [9] and earlier, algorithms for autonomics (DE algorithms) may not be standardized — as DE algorithms provide for DE vendor differentiation. However, innovation in autonomics algorithms require the collaboration of industry (traditional network equipment vendors and ISVs) with research organizations (institutes and universities) who are expected to continue advancing the research on developing better algorithms for autonomics (even in the long term). This means research organizations equipped with experimental facilities and expertise on autonomics have the potential to be providers of autonomics algorithms to vendors (e.g. in some partnerships) who can then incorporate the algorithms in their DE software components.
Actor 2	Provider of Assets required by the developers of GANA Functional Blocks (FBs) (software components/modules/libraries, protocols and DE algorithms for autonomics)
Role 1	Provider of the GANA Implementation Guide
Who should fulfil this role?	ETSI NTECH AFI Working Group
Role 2	Provider of autonomics-enabled implementation-oriented architectures
Who should fulfil this role?	The following players fulfil this role: • ETSI NTECH AFI Working Group • Any other SDO (e.g. in ITU, IEEE, etc.) that performs the instantiation of GANA on their reference architecture to create an autonomics-enabled implementation-oriented reference architecture ETSI NTECH AFI WG is performing work on instantiating the GANA onto various reference architectures and producing various autonomics-enabled reference architectures that are required by developers, e.g. autonomics-enabled Broadband Forum (BBF) reference architecture; autonomics-enabled NGN/IMS architecture; autonomics-enabled 3GPP reference architecture; autonomics-enabled ONF SDN architecture; autonomics-enabled (3GPP) EPC and Backhaul architectures [28]; autonomics-enabled wireless ad-hoc/mesh sensor network architectures [21]. What developers can obtain from such GANA instantiation cases are details on what types of DEs and associated control-loops should be implemented in the GANA Knowledge Plane and in specific NEs, as well as the mapping of DEs to specific MEs they should autonomically manage and control.

Table 2: Description of the roles attached to each Player/Actor in implementing GANA Model

Interoperability of Autonomics Software Components, and Compatibility of the Hybrid-SON Model with the GANA Model

The Hybrid Nature of the GANA Model

GANA is a hybrid model that enables developers of autonomics algorithms to combine (interwork) centralized management & control with some limited distributed control within the fundamental NEs and the data plane. This is because there are some problems that are better addressed locally on an NE using fast local control-loops and/or by distributed control algorithms and distributed control-loops that span a certain scope within the data plane. Fast local control-loops in NEs can be contrasted with the outer slower but more complex control-loops at the logically centralized GANA Knowledge Plane. Control-loops introduced in an NE and in the fundamental E2E transport/data-plane network architecture may be required to enable some degree of in-NE intelligence and in-network intelligence i.e. in-system and in-network self-management.

Interoperability of Software Components for Autonomics

There are two aspects to be considered in the space of interoperability of software components for autonomics, according to the GANA model and suppliers for autonomics FBs:

1) **The Functional Blocks (FBs) defined by GANA**

Here we are considering the GANA KP's FBs, their associated Reference Points (Rfp), and characteristic information flow/exchange and the GANA abstraction levels for autonomics components in general (particularly GANA Level-2 up to GANA Level-4 DEs). All these aspects determine the types of autonomic software component suppliers for the FBs as illustrated in Table 2. That means the characteristic information and methods (e.g. protocols) used in conveying the Rfps shall determine interoperability between the components. The completeness of implementation details for each Rfp (characteristic information and communication methods), is a subject that needs to be tracked along with the on-going work in ETSI NTECH AFI Working Group on instantiation of the GANA onto target implementation-oriented reference architectures. This is because the implementation details for each Rfp get completed or even evolve in the process of analysing autonomics requirements and attempting to implement various autonomics use cases in the target architecture. This is because at implementation stage the various use cases require that details of the Rfp's implementation should be nailed down at that stage. Communication means to be employed by the Rfp's FBs and also data flow between the FBs, etc, should become fully detailed at implementation time.

PoC projects to run according to the GANA PoC Framework [16] will also contribute to the detailing of GANA FBs' Rfps implementation.

2) **DE algorithms for autonomics as DE vendor differentiator**

It is assumed that DE algorithms may not be standardized as innovation in DE (autonomics) algorithms should be continuous in the lifecycle of enrichment of networks with advanced self-management capabilities, as already discussed earlier and also in [9], [10]. However, there are autonomic collaborative behaviours, e.g. for E2E network self-optimization that need to span multiple NEs and networks, that can only work if DEs from different vendors (instrumented at various levels and horizontally along a path) can exchange certain types of characteristic information via the DE-to-DE horizontal Rfp and/or via the vertical Rfp defined in GANA (see Figure 2).

Compatibility of Hybrid-SON with the GANA Model Design Principle, and Commonly Shared Aspects

The GANA model shares common principles with the Hybrid-SON architectural model (now being deployed), as both models enable developers of algorithms to combine and interwork centralized and distributed management and control solutions for networks and services. Also, they both are hierarchical management and control frameworks that present levels of abstraction for self-management functionality at which control-loops (DEs) can be designed and interworked. Both models share the common question on interoperability of Autonomic Functions (AFs), and so the following highlight the commonalities:

1) **Interoperability of multi-vendor DEs should be treated similarly to multi-vendor SON functions**

The question of interoperability of multi-vendor DEs on the DE-to-DE Rfps is a question that is similarly applied to interworking/interoperability of multi-vendor SON (Self-Organizing Networks) functions for multi-RAT (Radio Access Technology) Radio Access Networks (RANs). In the hybrid-SON architecture (standardized by 3GPP), a similar hierarchy of AMC functions exists as in the case with DE hierarchical levels in GANA. In the hybrid-SON architecture, Centralized SON (C-SON), which now includes Big Data-SON (B-SON), is supposed to interwork with Distributed SON (D-SON) in RAN equipment, and D-SON functions may also need to interwork across neighbour relationships involving multiple vendors.

2) **Policy control in SON hierarchical architecture is similar to the case of GANA vertical interactions of DEs**

C-SON controls D-SON through policing, i.e. C-SON injects policies to the lower D-SON functions so as to constrain and coordinate their behaviours. D-SON functions in the RAN equipment are normally provided by the different RAN equipment vendors, while C-SON functions may come from ISVs or even RAN equipment vendors too. Therefore, the issue of interoperability of SON functions vertically and horizontally is very similar to multi-vendor GANA DE-interoperability vertically and horizontally. GANA Vertical DE interactions may involve policy control (by upper DE), synchronization of actions and information push up the DE hierarchy.

3) **Recommendations on multi-vendor interoperability of GANA DEs**

To address interoperability of SON functions, NGMN produced a document on Recommended Practices for Multi-vendor SON Deployment [11]. Similar recommendations on multi-vendor interoperability of GANA DEs (autonomic functions) can be derived from the NGMN recommendation [11] and applied by vendors for the case of multi-vendor DE-to-DE interactions/interoperability, e.g. on agreeing the types of information that should be exchanged (possibly without even having to disclose how DE algorithms from the vendors are designed). A given GANA DE manages only its associated MEs. In this respect, a peer-to-peer DE interaction is required if the DE has to indirectly manage through other DEs other MEs not under its responsibility, as this prevents undesirable effects that could occur in the network (e.g. instability in the network, etc.) if DEs were to control MEs in a non-coordinated fashion.

GANA and White Box Networking

The emerging paradigm of white box networking provides more flexibility to the operator's and enterprises' ability to build customized NEs that are based on low cost commodity hardware. A white box is generic, off-the-shelf switching and routing hardware within the forwarding plane of a SDN (definition adopted from SDxCentral [12]). It is noted that the concept of openly customizable low cost commodity hardware can also be applied to other planes outside of the data plane. In contrast to traditional black box switches, white boxes are meant to be basic commodity hardware that offers the possibility and flexibility for operators or enterprises to install an operating system of choice and customize the box with other loadable software modules (off-the-shelf software modules from various suppliers that can be directly integrated into the system) that serve a specific purpose. In the GANA case, those off-the-shelf software modules can be AMC algorithmic software logic (i.e. GANA DEs) purchased separately from ISVs or developed by the operator.

A white box may come with already installed software (e.g. the operating system (OS) and other software) or software can be purchased from a software vendor and loaded separately. With the advent of white box networking, even small innovative software companies can develop control software as well as algorithms for AMC (i.e. AMC enabled by replaceable and (re)-loadable DE logics) and other types of complementary software for dynamic, workload-aware and analytics-driven network services and resources orchestration. Such software can be purchased and loaded into white boxes. Therefore, the on-going work on standardization of AMC in ETSI NTECH AFI Working Group (the main group on standardization of AMC) and also in TM Forum, IEEE, ITU-T and IETF, should offer the opportunity for network operators and enterprises to purchase AMC software separately from hardware. GANA in particular, enables AMC software vendors to identify and develop software that can be purchased as AMC components that add intelligence to white boxes and to the overall network—as made possible by ETSI standards for AMC (i.e. the GANA Model) via the GANA reference points that are meant to support the loading of DEs into NEs.

Implementation Guide for GANA

As presented in Table 2, the Functional Blocks (FBs) defined in the GANA reference model such as the GANA KP FBs, would need to be implemented by ISVs (e.g. OSS (Operations Support Systems) vendors) or by traditional equipment manufacturers, and other FBs by network operators who may have software development capabilities. The GANA KP enhances intelligence in network management by evolving or interworking with EMSs (Element Management Systems) or NMSs(Network Management Systems) or OSSs (see details in [13] in which an architectural perspective is presented on enabling advanced management & control intelligence at various layers of abstraction through Autonomic Management & Control (AMC) software). ETSI GS AFI002 [3] also provides insights on this subject. Though the impact of network virtualization technologies such as NFV is such that the EMS and NMS get transformed into software functions, still the GANA KP would enhance intelligence for such software functions. The reference [10] presents an implementation guide for GANA. A formal description model of GANA (described using a meta-model in an appropriate modelling language such as UML, MOF, etc.) may also be very useful for implementers. Such a meta-model (i.e. GANA meta-model) does not exist yet, and work on the GANA meta-model may be commenced in ETSI NTECH AFI soon.

First Steps in the Implementation of GANA

- **Instantiation of GANA onto a particular target architecture**

The first process involved in the implementation of GANA FBs for autonomics in a target implementation-oriented architecture is the instantiation of GANA onto the particular target architecture to create an autonomics-enabled reference architecture. For illustration, later in this section, an example instantiation case for GANA onto a target architecture is given based on ETSI TR 103 404 [28] that addresses GANA Instantiation onto the Backhaul and Core network (EPC) parts of the 3GPP Architecture.

How is the instantiation done? GANA Instantiation is done in the following manner ([10] provides more details on the procedures on GANA instantiation):

1. Superimposing the required GANA FBs such as GANA KP FBs (ONIX, MBTS, Network-Level-DEs) onto the logically centralized management and control planes of the target implementation-oriented reference architecture (integration with OSS's/NMS's and EMS may be needed or the GANA Knowledge Plane can be implemented in a standalone manner (more details in [3])),
2. Instantiating the DEs that are required in specific NEs' architecture and the overall network architecture (i.e. the DEs for the Knowledge Plane).By doing so, the DEs and their associated control-loops can be further designed to perform autonomic management and control of the specific resources (fundamental Managed Entities) in the target architecture.

- **Recommendations on fundamental behaviours of instantiated GANA FBs**

Recommendations on the basic behaviours required of the GANA FBs in specific environments are being developed in ETSI NTECH AFI Working Group. The impact of virtualization and dynamic versus static instantiation of GANA DEs is a subject discussed in [10], [9].

Currently ETSI NTECH AFI Working Group is working on various cases of instantiation of GANA onto target architectures to create autonomics-enabled reference architectures on the basis of which FBs and DE algorithms can be further elaborated and implemented based on autonomic requirements and use case scenarios in the individual target architectures.

- **The input required by Developers**

Developers need to work with such instantiated GANA FBs in target architectures to then perform DE algorithm simulations and software implementation. ETSI NTECH AFI Working Group is working on GANA instantiation cases listed below (readers may keep track of the work programme in NTECH on this subject [14]):

- Autonomics-enabled SDN reference architecture based on mappings/instantiation of the GANA Model onto a particular SDN facilitating Architecture;
- Autonomicity and Self-Management in the Backhaul and Core network (EPC) parts of the 3GPP Architecture. ETSI NTECH AFI WG has completed addressing this particular topic and corresponding documents will be available to the public in 2016;
- Autonomics-enabled Broadband Forum (BBF) reference architecture. ETSI will commence work in September 2016 on GANA instantiation onto the BBF architecture that incorporates SDN and NFV (readers are encouraged to follow this activity closely to access the intermediate documents and also the final output in 2017).
- Autonomics-enabled IMS architecture;
- Autonomics-enabled CDN (Content-Delivery Network) reference architecture;

- Autonomics-enabled wireless ad-hoc/mesh sensor network architectures (readers may refer to [21] for an instantiation of GANA for mesh networks that was validated and published).

Second steps in the implementation of GANA

- **Requirements analysis for use cases for autonomics in a particular target architecture**

The next step after having an autonomics-enabled architecture at hand is the process of requirements analysis for use cases for autonomics in a particular target architecture. The implementer needs to analyse a particular requirement for autonomics in a target network environment and derive behaviour of the GANA autonomics FBs instantiated in the target architecture, their communication flow and data exchange on reference points /interfaces that enable implementation of the requirement. Examples of such requirements for autonomics can be found in ETSI TS 103 194 [20], but there are many other requirements implementers may want to consider [3], [16].

- **Further details implementers should obtain from GANA instantiations cases**

Details implementers (i.e. software developers for autonomics) can obtain from GANA instantiation cases in ETSI NTECH AFI Working Group are:

- Details on the types of DEs and associated control-loops that should be implemented in specific NEs and in the GANA Knowledge Plane of an autonomics-enabled architecture;
- The mapping of DEs to specific MEs they should autonomically manage and control. The GS AFI002 specification [3] also provides guidelines on principles that can be applied for designing DE logic, including external and internal structural models of a DE, interfaces (reference points) and primitives that should be supported and implemented on DE interfaces.

What then comes next for Developers?

Based on all such inputs, as discussed in [10], developers should then perform the following steps:

- Use the instantiated GANA FBs and Reference Points (Rfps) for enabling autonomicity (self-management) in a target architecture, to specify autonomic behaviours of the FBs within the logically centralized management and control plane architecture and within the E2E (end-to-end) transport and data plane architectures.
- Specify behaviours of instantiated GANA FBs (including DEs and their control-loops)
- Develop and simulate the GANA DE algorithms for autonomics, and then implement them.

In order to help developers, we propose in Table 3, five complementary perspectives on GANA implementation approaches, their descriptions and suggested recommendations.

GANA Implementation Perspective	Description & Recommendations
Using existing R&D results on autonomics/AMC to implement GANA FBs	Developers of the GANA FBs instantiated in a target architecture as well as autonomics algorithms for the required DEs, should exploit the results obtained in various autonomic networking research/R&D projects that are applicable to the target architecture, to further derive behaviours desired of the instantiated GANA FBs and also the data exchange required on their associated reference points. The same applies to the process of deriving the types of required DE algorithms, i.e. using algorithms developed for the components usually referred to as autonomic manager components in the various applicable projects. All such derivations of GANA FB behaviours, data exchange of reference points, and applicable DE algorithms, assume that the research results being exploited were based on the same architecture as the one under consideration. Hierarchical autonomic manager components map to DE hierarchies (as DEs are autonomic managers). The autonomic manager components (developed in various projects on autonomics), which are often not modular by design can be decomposed into multiple interacting DEs (for modularity). Every abstraction level of a control-loop for AMC prototyped in various projects can be mapped to a particular abstraction level in GANA. Overall, this implies that developers of GANA FBs can help evolve the standards and frameworks for inter-operable autonomics in ETSI NTECH AFI and other groups that are liaising with ETSI NTECH AFI, such as BBF, ITU-T SG13 & SG2 & SG15, 3GPP, TMForum, IEEE NGSON, NSF-CAC (Cloud and Autonomic Computing Center), MEF, IETF, etc. In particular, developers can provide very useful contributions to the work in ETSI NTECH AFI on instantiation of GANA onto various reference architectures to enable autonomics in the target architectures.
Development of GANA Decision Element(DE) logic as Run-Time Loadable Modules	GANA Decision Element (DE) logic should be viewed as software components or modules that can be (re)-loaded into nodes (to introduce or enhance a node's self-* features, such as self-adaptation) and into the GANA KP (thereby enabling software-empowered networks and enabling both distributed control using distributed algorithms and centralized control for aspects that require centralized control). As discussed in [9], [13], DE algorithms are a subject of continuous innovation, meaning that design principles for modular, (re)-loadable, evolvable or replaceable GANA DEs are desirable (particularly by network operators); [3], [10], [13] discuss this subject.

Choosing between Top-down approach versus Bottom-up approach in implementing and deploying autonomic functions (AF) in a target autonomics-enabled network architecture	As discussed in [10], the implementation of GANA may need to take a top-down approach in some cases, meaning that node-external control-loops in the GANA KP (i.e. Network-Level-DEs) and the other GANA KP FBs (MBTS and ONIX) may be prioritized in the implementation and deployment of autonomic functions in a target autonomics-enabled network architecture. This means that starting with autonomics at a higher level could be desirable in some cases, before then incrementally moving on to implement the GANA lower level autonomics (lower GANA levels and the fast control-loops intrinsic to NEs). In other cases, it may be desirable to take a bottom-up approach, whereby the lower level GANA autonomics is prioritized, i.e. DEs within NEs and their collaboration in a distributed fashion (through GANA levels 1 to 3). As discussed earlier (in the section on core concepts of GANA), within NEs (nodes), GANA levels 2 and 3 may be the important levels to consider. However, in both bases (top-down approach or bottom-up approach to implementing the GANA), the point is that not every DE and level in GANA needs to be implemented in all cases.
Choosing between implementing DEs as standalone processes or combined together as a single process or as executable behavioural models	Regarding the implementation of DEs in NEs: While DE algorithms and their containing modules meant to be implemented and run in a NE (node) can be designed and simulated individually for each DE and its interactions with its MEs and with other DEs (particularly its upper DE), the actual final implementation of the run-time DE instances can take different approaches. For example: (1) DE modules and their algorithmic logic could run as standalone processes or threads, just as they may have been simulated as standalone processes; (2) DE modules and their algorithmic logic could be combined together, e.g. as runtime loadable libraries, to run as a single process or thread; (3) DE modules and their algorithmic logic could be implemented to run as executable behavioural models which require an interpreter to execute them (e.g. similarly to how scripting languages are interpreted in execution of a script).
Instantiation of lower level autonomics (e.g. GANA levels 3 and below) in environments involving SDN (Software-Defined Networking), and implementing the GANA KP as	In environments involving SDN (Software-Defined Networking) the instantiation of lower level autonomics (e.g. GANA levels 3 and below) in NEs in the data plane should be limited to those management and control aspects that need local fast reactions within an NE(s) without relying on centralized decisions of a centralized control-plane. Such lower level autonomics in the data plane elements may also involve the collaboration of DEs within certain NEs along a data plane path of some scope (the collaboration of the NEs' DEs needs not necessarily be on a hop-by-hop basis, but may involve NEs at network domain borders). Such lower level autonomics is then complemented by the GANA Knowledge Plane DEs

Northbound SDN Application(s) for AMC	that may run as network intelligence applications on top of the SDN controller. The types of fundamental MEs (networking resources such as protocols and stacks) present in the data plane elements, together with a thin control-plane left to run in the data plane (complemented by the outer centralized control plane) would determine the types of DEs that can be instantiated in the NEs of the data plane. More details on this subject can be found in [9] and are to be further elaborated in the on-going work in ETSI NTECH AFI on the integration of GANA with paradigms such as SDN, NFV and E2E Services and Resources Orchestration [15].

Table 3: Complementary Perspectives on GANA implementation approaches/use cases, their descriptions and suggested recommendations

Example Instantiation Case for GANA: GANA Instantiation onto the Backhaul and Core network (EPC) parts of the 3GPP Architecture

To provide an example of a concrete GANA instantiation, ETSI NTECH AFI has recently completed work on GANA instantiation for Autonomicity and Self-Management in the Backhaul and Core network parts of the 3GPP Architecture, with consideration for interworking GANA autonomics with C-SON for the RAN (Radio Access Network). Readers should refer to the work described in ETSI TR 103 404 [28]. The figures below are extracts from ETSI TR 103 404 [28]. GANA autonomics introduced in the backhaul and core network segments by the work in ETSI TR 103 404, complement SON for the RAN, since SON functions were specifically defined and designed for automation and intelligence in managing RAN. Bearing in mind that an “Autonomic Function (AF)” in GANA is a decision-making software logic that drives a closed control-loop over managed parameters/resources, it means that autonomics in its broad sense includes SON and other automation & cognitive functions for enabling self-management and control of network resources in general.

As illustrated in the figures below (**Figure** 4 and **Figure** 5), the work in ETSI TR 103 404 also identified the need for introducing a reference point (rfp) between C-SON and GANA Knowledge Plane (KP) for the Core Network (EPC). Such a reference point is meant to enable the exchange of information (e.g. KPIs (Key Performance Indicators)) and decisions between C-SON and EPC’s Knowledge Plane, for the purpose of implementing cross domain self-optimization behaviours for resource utilization in the network segments. Readers should refer to ETSI TR 103 404 [28] for more details.

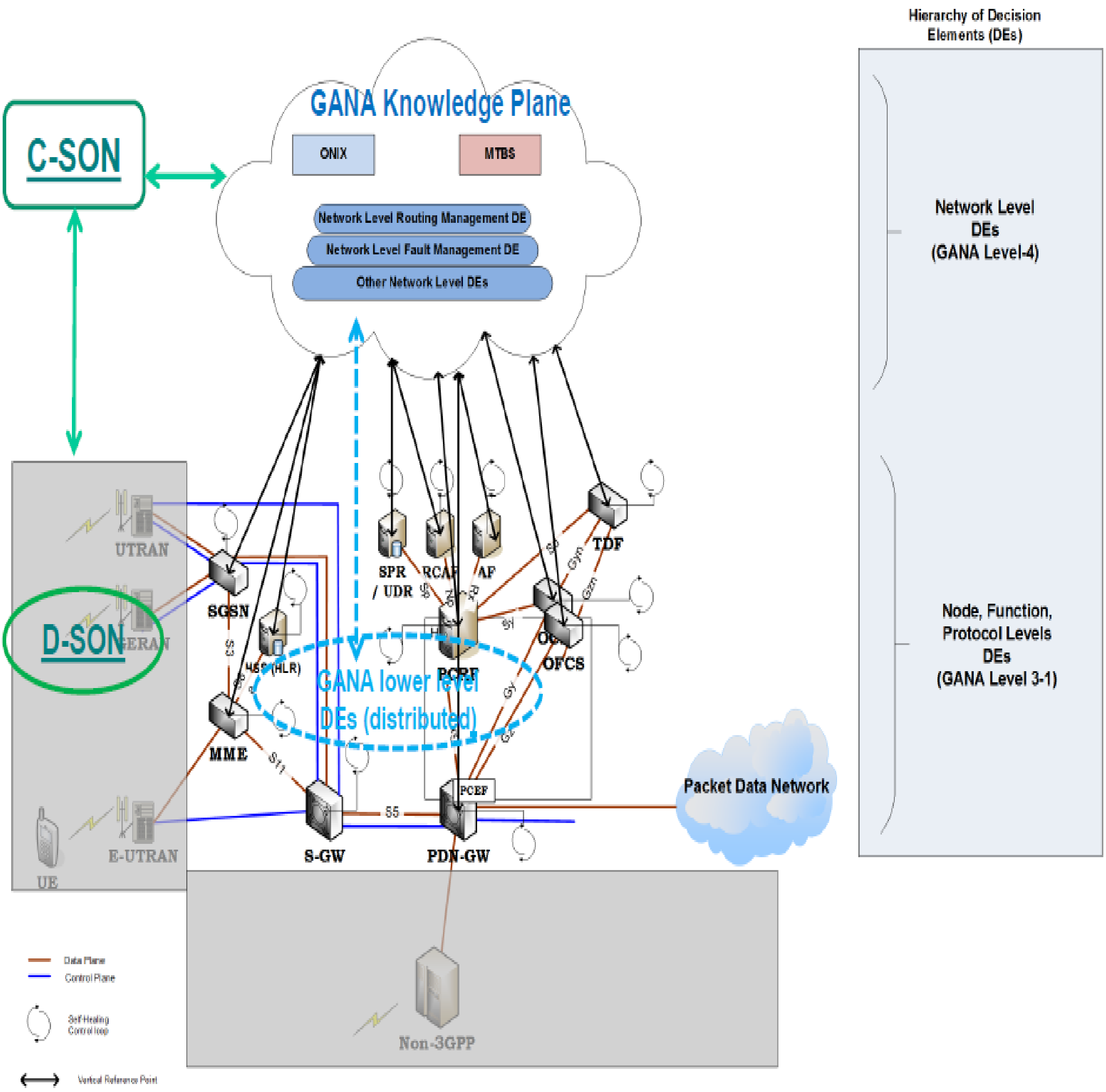

Figure 4: Instantiation of the GANA Knowledge Plane for the 3GPP EPC Core Network, and the need for EPC KP to interwork with 3GPP C-SON (figure extracted from [28])

The figure below (Figure 5), illustrates that for E2E Self-Optimization objective across domains (RAN, Backhaul and Core Network), there is a need to interwork C-SON with GANA Components instantiated in the Backhaul and Core Network.

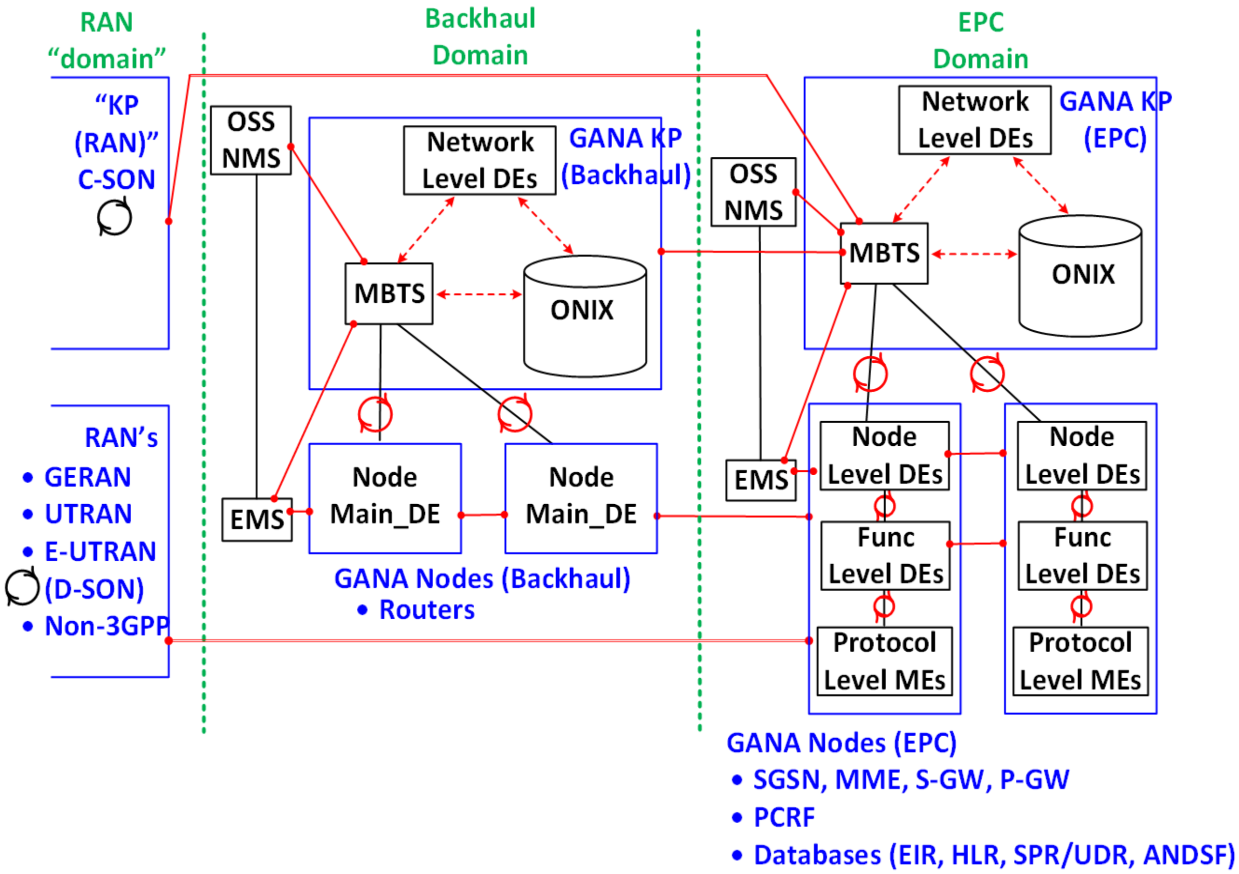

Figure 5: Interactions involving C-SON, and GANA entities for 3GPP Core Network and backhaul, to enable E2E Self-Optimization across network segments (figure extracted from [28])

GANA in the Unified Architecture for AMC, SDN, NFV, E2E Orchestration & Specialized Big Data Analytics

Harmonization of Standards & Architectures for Emerging Networking Paradigms

AMC, SDN, NFV, E2E orchestration of services and resources, and Big Data analytics applications for network management & control constitute the emerging complementary networking paradigms for evolving and future networks. The industry is now faced with a number of problems linked to standards for these paradigms. For example, the lack of a unifying standardized architectural framework for all the paradigms put together, which have so far been developed in “silos” that lead to unnecessarily high costs for the industry’s R&D activities. Another issue is the problem of standards collision and duplication in some SDOs/Fora [13].

Need for harmonization across multiple SDOs/Fora

The industry is now seeing that standards gaps and overlaps are being exposed when merging these emerging paradigms to interwork in a common architecture. Hence, it implies that there is a critical need for industry harmonization of the related standards across multiple SDOs/Fora that are working on these emerging paradigms. In order to address these challenges the industry (through SDOs/Fora) recently launched an initiative: Joint SDOs/Fora Industry Harmonization on Unified Standards for AMC (Autonomic Management & Control of Networks and Services), SDN, NFV, E2E Orchestration of Services and Resources—as Software-Oriented Enablers for 5G [13]. The initiative is an informal collaboration of SDOs/Fora interested in sharing information with others on what is in the scope of their work-programs regarding the emerging paradigms, learn what others are doing in the areas, discuss with others on what standardization gaps are being identified in various groups, and indicate to others the areas for potential collaborations through their established formal channels. The initiative is driven by a recurring Joint SDOs/Fora workshop that helps to learn on a multi-groups level what to do to avoid collisions in standards, avoid duplication of work, and support industry harmonization efforts for unified standards (to reduce silos).

Proposed Unified Architecture for AMC, SDN, NFV, E2E Orchestration, and specialized Big Data Analytics

The Joint SDOs/Fora industry harmonization initiative has started some work on a unified architecture for AMC, SDN, NFV, E2E orchestration, and Big-Data analytics applications for network services management & control [13] (see Figure 6), in which the GANA Functional Blocks (e.g. the Knowledge Plane) are integrated into the architecture with FBs for SDN, NFV and E2E service orchestration, OSSs, EMSs, etc. Figure 6 illustrates the proposed architectural framework. The GANA KP is the centralized point in which complex algorithms for autonomics are to be implemented through DEs and their closed-loops, which use information or knowledge supplied from the various sources. The GANA KP analytics (autonomics Decision-making Elements/Engines (DEs)) implement algorithms that reason on what needs to be dynamically and adaptively triggered at the Universal E2E Orchestrator, at the SDN Controller(s), at the OSS, and at the NFV Orchestrator(s), for the purpose of adaptive service delivery as may be required at any point (management & control point and time) in response to challenges in conditions of network operations (faults, errors, failures, threats, service performance degradations, etc.), and changes in workloads, and also for achieving autonomic service assurance. Information or knowledge required to be acted upon by the KP DEs in real-time is directly sent or streamed to the DEs by the various sources. The information or knowledge should be additionally pushed (recorded) into the ONIX servers for historical information/knowledge trace storage. Analytics by DEs in the GANA Knowledge Plane (KP) differs from analytics by certain Big Data analytics applications in the sense that analytics in the GANA KP is driven by information/data sources that are required purely for network and services management & control (dynamic network and service policing) while some Big Data applications may be driven by information/data from end user terminals as well. Big Data analytics applications that are specialized for network management & control or analytics-driven orchestration should be implemented as DEs in the Knowledge Plane (provided that the information/data sources they operate on are those required purely for network and services management & control), and like all DEs in the KP, the DEs should coordinate (whenever necessary) with other DEs against any optimization objectives conflicts and resolve the conflicts. To ensure that decision-making by Big Data analytics applications should complement and not conflict with decision-making by DEs in the Knowledge Plane, Big Data applications for analytics-driven service or resource orchestration need to coordinate their operations with the Knowledge Plane. Because KP DEs need to know (understand) intended state changes on the underlying resources to be either orchestrated or re-configured to provision the service that is to be orchestrated by the E2E universal service orchestrator, and verify that such changes/operations would not conflict with the optimization objectives of the DEs in the KP (before the universal E2E service orchestrator is then allowed to perform the operation). Interfaces that feed information/knowledge to the DEs in the KP, such as the interface exposed by the MBTS to the DEs and the interface exposed by the data collectors to the DEs, should be similar, as they are expected to translate information into the representation expected by DEs for their real-time consumption and operations. Regarding the integration of the GANA KP with an SDN controller, this subject is addressed in [9], including insights on how GANA KP DEs can be implemented as network-intelligence enhancing applications that drive an SDN controller via its northbound interface.

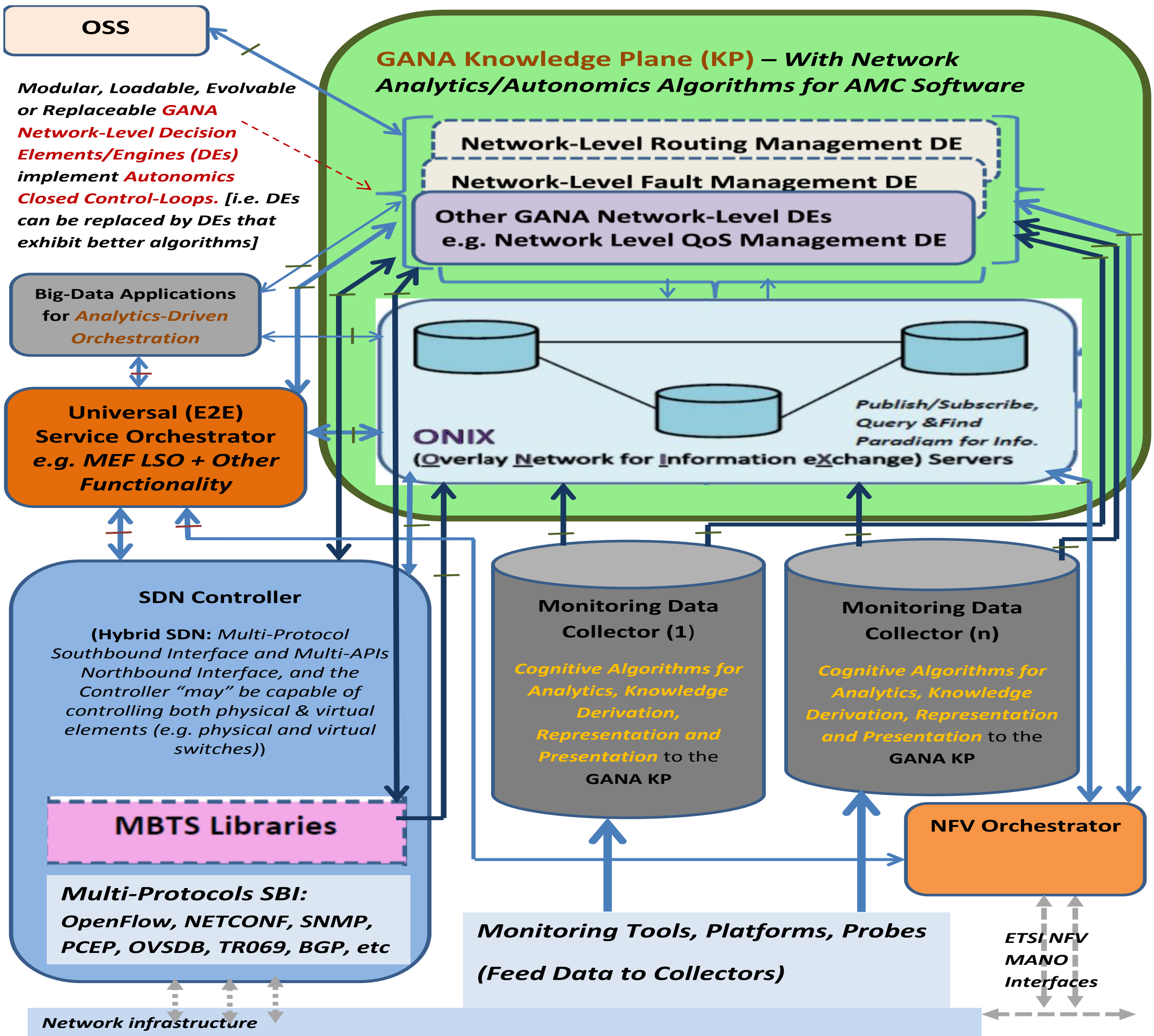

Figure 6: Core part of the unified framework for AMC, SDN NFV, e2e Orchestration, Big-Data driven analytics [13]

This unified architecture presents an architectural perspective on how to enable advanced management & control intelligence" at various layers of abstraction through Autonomic Management & Control (AMC) Software (i.e. GANA Functional Blocks). This unified architecture enables holistically viewing the interworking of the complementary paradigms together within a complete picture of network architecture(s). What this unified architecture for the paradigms implies is that the reference points between the GANA Knowledge Plane and E2E Service Orchestrator, SDN Controller, Data Collector, NFV Orchestrator, OSS, and Big-Data driven analytics applications need to be detailed by the community (the industry and research circles). Therefore, technical experts are welcome and encouraged to provide contributions on detailing the reference points. Such inputs should flow through ETSI NTECH AFI and/or other groups participating in the initiative on Joint SDOs/Fora Industry Harmonization for Unified Standards [13].

Universal E2E Service Orchestration: what it is?

The E2E service orchestrator interprets service definitions and determines from a network service node's attributes if a service of a network function (node) is to be implemented via a VNF (Virtualized Network Function) or PNF (Physical Network Function). Then the E2E service orchestrator accordingly triggers the SDN controller or a legacy NMS or EMS to configure the required PNFs for those nodes in the service definition that explicitly relate to PNFs according to service designer indications. Finally, for those service nodes that relate to VNFs, the E2E orchestrator triggers the NFV-O (NFV-Orchestrator) to perform orchestration of VNFs required by the E2Eservice instantiation. The E2E service orchestrator is therefore aware of configurations that are required in the physical network domain versus those required in the virtualized environments (NFV). Also, the GUI (Graphical User Interface) for service definition should enable the service designer to invoke the E2E orchestrator to instantiate the completed service definitions, and also some interactions between the GUI and E2E service orchestrator can be required in cases where state information maintained in the E2E orchestrator can be used in service definitions through the GUI.

In the unified architecture (Figure 6), Metro Ethernet Forum Life cycle Orchestration (MEF LSO) is simply listed as an example of an E2E service orchestrator of some sort, and it is indicated that LSO as it is specified/designed currently by MEF would need to be extended with additional/other functionality in order to fully fulfil the desired features of an E2E service orchestrator.

How to position this E2E Service Orchestrator with respect to OSS?

From a high level view, a universal E2E orchestrator is a kind of OSS that potentially could replace some old legacy OSSs over time. It is important to note that co-existence of an E2E service orchestrator with legacy OSSs, some of which have been evolved to now manage SDN and NFV components, is likely to last for some time, before designs for modular and real-time OSS or E2E orchestrators would fully take over.

It is possible to characterize the interface between the E2E service orchestrator and the OSS in two possible scenarios that are captured by the unified architecture diagram of Figure 6:

- Legacy NMS (which may be called an OSS) being directly below the E2E orchestrator;
- OSS (some of which have been evolved to now manage SDN and NFV components and are integrated with BSS functions) placed above the E2E orchestrator.

Interfacing the GANA Knowledge Plane with other Functional Blocks of the Unified Architecture

As shown in Figure 6, new reference points are introduced between GANA FBs and other functional blocks (e.g. SDN Controller, OSS, ETSI /NFV / MANO NFV-O, universal E2Eservice orchestrator, monitoring data collectors, Big-Data applications for analytics-driven orchestration). All of them need now to be characterized and detailed by the collaborating SDOs/Fora involved in the initiative on Joint SDOs/Fora Industry Harmonization for Unified Standards [13]. The work on further elaboration of the unified architecture for AMC, SDN, NFV, E2E orchestration and Big-Data applications for network management & control will progress mainly through the groups involved in the initiative [13], but all interested technical experts are welcome to contribute to this work in general through any of the collaborating SDOs/Fora. ETSI NTECH AFI Working Group is working on a Technical Specification (TS) linked to this unified architecture for AMC, SDN, NFV and E2E orchestration ([15], and interested experts are welcome to join the activity).

Call for GANA Proofs of Concept

ETSI NTECH AFI has produced a GANA Proof of Concept (PoC) Framework [16] and is now calling for PoC projects: GANA PoCs aimed at assessing the implementation of the GANA reference model (as a reference model for the paradigm of AMC) by demonstrating autonomics at a single GANA level or multiple levels of self-management (autonomic) functionality. PoCs aimed at showcasing GANA articulation with reference models for the other emerging complementary networking paradigms of NFV, SDN, and E2E orchestration, are also welcome. Through this GANA PoC Framework, use cases for AMC, as well as use cases that combine the paradigms of AMC, SDN, NFV, and E2E orchestration, can also be showcased so as to contribute to further detailing the harmonized/unifying architectural framework that combines the paradigms [13].

How to contribute to the Standards for AMC

Interested parties are invited to contribute to the following aspects of standards on AMC in ETSI NTECH AFI Working Group:

- On-going work on various cases of instantiation of GANA onto target architectures to create (autonomics)-enabled reference architectures on the basis of which GANA autonomic functional blocks and DE algorithms can be further elaborated and implemented based on autonomics requirements and use case scenarios in the individual target architectures.
- The work on further elaboration of the Reference Points and APIs required in the unified architecture for AMC, SDN, NFV, E2E orchestration and Big-Data applications for network management & control—the architecture emerging from the Joint SDOs/Fora Industry Harmonization Initiative on Unified Standards for AMC (Autonomic Management & Control of Networks and Services), SDN, NFV, E2E orchestration of services and resources—all seen together as Software-Oriented Enablers for 5G [13].
- Submission of GANA PoC proposals (and PoC contributions) following the process described in the GANA PoC Framework [16] and NTECH WiKi: http://ntechwiki.etsi.org.

References

[1] R. Chaparadza, S. Papavassiliou, T. Kastrinogiannis, M. Vigoureux , E. Dotaro, A. Davy, K. Quinn, M. Wodczak, A. Toth, A. Liakopoulos, M. Wilson: Creating a viable Evolution Path towards Self-Managing Future Internet via a Standardizable Reference Model for Autonomic Network Engineering. Published in the book by the Future Internet Assembly (FIA) in Europe: Towards the future internet - A European research perspective. Amsterdam: IOS Press, 2009, pp. 136-147.

[2] Michal Wodczak, Tayeb Ben Meriem, Benoit Radier, Ranganai Chaparadza, Kevin Quinn, Jesse Kielthy, Brian A. Lee, Laurent Ciavaglia, Kostas Tsagkaris, Szymon Szott, Anastasios Zafeiropoulos, Athanassios Liakopoulos, Apostolos Kousaridas, Maurice Duault: Standardizing a reference model and autonomic network architectures for the self-managing future internet. IEEE Network 25(6): 50-56 (2011)

[3] ETSI GS AFI 002: Autonomic network engineering for the self-managing Future Internet (AFI): GANA Architectural Reference Model for Autonomic Networking, Cognitive Networking and Self-Management.

[4] John C Strassner, Nazim Agoulmine, and Elyes Lehtihet. FOCALE – A Novel Autonomic Networking Architecture. In LAACS, Campo Grande, MS, Brazil, 2006.

[5] David D. Clark, Craig Partridge, and J. Christopher Ramming. A knowledge plane for the Internet. In SIGCOMM, pages 3–10, 2003.

[6] Albert Greenberg, et al: A clean slate 4D approach to network control and management : In ACM SIGCOMM Computer Communication Review Homepage archive Volume 35 Issue 5, October 2005, Pages 41-54

[7] IBM White paper: An architectural blueprint for autonomic computing: June 2005

[8] Nikolay Tcholtchev, Ranganai Chaparadza, and Arun Prakash. Addressing Stability of Control-Loops in the Context of the GANA Architecture: Synchronization of Actions and Policies. In IWSOS '09: Proceedings of the 4th IFIP TC 6 International Workshop on Self-Organizing Systems, pages 262–268, Berlin, Heidelberg, 2009. Springer

[9] R. Chaparadza, Tayeb Ben Meriem, Benoit Radier, Szymon Szott, Michal Wodczak, Arun Prakash, Jianguo Ding, Said Soulhi, Andrej Mihailovic: SDN Enablers in the ETSI AFI GANA Reference Model for Autonomic Management & Control (emerging standard), and Virtualization Impact: In the proceedings of the 5th IEEE MENS Workshop at IEEE Globecom 2013, December, Atlanta, Georgia, USA

[10] R. Chaparadza, Tayeb Ben Meriem, Benoit Radier, Szymon Szott, Michal Wodczak, Arun Prakash, Jianguo Ding, Said Soulhi, Andrej Mihailovic: Implementation Guide for the ETSI AFI GANA Model: a Standardized Reference Model for Autonomic Networking, Cognitive Networking and Self-Management: In the proceedings of the 5th IEEE MENS Workshop at IEEE Globecom 2013, December, Atlanta, Georgia, USA

[11] Recommended Practices for Multi-vendor SON Deployment: Deliverable D2 by NGMN Alliance, version 1.0, 28 January 2014.

[12] SDx Central: https://www.sdxcentral.com/flow/white-box-switching/

[13] Workshop Report from the Initiative: Joint SDOs/Fora Industry Harmonization for Unified Standards on AMC, SDN, NFV, E2E Orchestration, Software-oriented enablers for 5G: Joint SDOs/Fora Workshop Report from the workshop held on 4th June 2015 at the TMForum Live 2015 can be downloaded from here: http://projects.sigma-orionis.com/eciao/wp-content/uploads/2015/07/Report-on-Joint-SDOs-Industry-Harmonization-for-Unified-Standards-on-AMC_SDN_NFV_E2E-Orchestration_ver3.01.compressed.pdf

[14] ETSI NTECH AFI WG Work Programme: http://webapp.etsi.org/WorkProgram/Report_WorkItem.asp?WKI_ID=42951

[15] Ongoing work in ETSI on the impact and relationships between GANA and other emerging paradigms such as SDN, NFV, Cloud Networking, etc.: https://portal.etsi.org/webapp/WorkProgram/Report_WorkItem.asp?WKI_ID=42950: DTR/NTECH-AFI-0020-GANA-impact

[16] ETSI TS 103 371 NTECH AFI Proofs of Concept Framework: http://ntechwiki.etsi.org/

[17] NGMN Alliance 5G White Paper version V1.0 March 2015: https://www.ngmn.org/fileadmin/ngmn/content/downloads/Technical/2015/NGMN_5G_White_Paper_V1_0.pdf

[18] Autonomic Networking Integrated Model and Approach (anima): https://datatracker.ietf.org/wg/anima/charter/

[19] Gabor Retvari, Felician Nemeth, Ranganai Chaparadza and Robert Szabo: "OSPF for Implementing Self-adaptive Routing in Autonomic Networks: a Case Study". In proceedings of the 4th IEEE International Workshop on Modelling Autonomic Communication Environments (MACE 2009), October 26-27 2009, Venice, Italy.

[20] ETSI TS 103.194 "Network Technologies (NTECH); Autonomic network engineering for the self-managing Future Internet (AFI); Scenarios, Use Cases and Requirements for Autonomic/Self-Managing Future Internet"

[21] S. Szott, M. Wódczak ; R. Chaparadza ; T. Ben Meriem ; K. Tsagkaris ; A. Kousaridas ; B. Radier; A. Mihailovic ; M. Natkaniec ; K. Łoziak ; K. Kosek-Szott ; M. Wągrowski: Standardization of an autonomicity-enabled mesh architecture framework, from ETSI-AFI group perspective (2 parts paper): In proceeding of IEEE Globecom Workshops (GC Wkshps), MENS Workshop 2012: DOI: 10.1109/GLOCOMW.2012.6477687

[22] Cong-Vinh, Phan: Formal and Practical Aspects of Autonomic Computing and Networking: Specification, Development and Verification: A book published by IGI global disseminator of knowledge: DOI: 10.4018/978-1-60960-845-3

[23] A. Atlas, J. Halpern, et.al: An Architecture for the Interface to the Routing System: draft-ietf-i2rs-architecture-13: February 20, 2016

[24] S. Vissicchio, L. Cittadini, O. Bonaventure, G. G. Xie, L. Vanbever: On the Co-Existence of Distributed and Centralized Routing Control-Planes: In proceedings of IEEE Infocom 2015, 26 April - 1 May 2015, Hong Kong.

[25] Ali Tizghadam, and Alberto Leon-Garcia: Autonomic traffic engineering for network robustness: In IEEE Journal on Selected Areas in Communications, Volume 28 Issue 1, January 2010, Pages 39-50: doi>10.1109/JSAC.2010.100105

[26] Alexej Starschenko et al: Auto-Configuration of OSPFv3 Routing in Fixed IPv6 Networks: In the proceedings of the ICUMT 2015 – The 7th International Congress on Ultra Modern Telecommunications and Control Systems: 6 – 8 October, 2015, Brno, Czech Republic

[27] ETSI TR 103 404: Autonomicity and Self-Management in the Backhaul and Core network parts of the 3GPP Architecture: expected date of publication by ETSI: September-October 2016 timeframe.

[28] Industry Workshop: "Forging a Path to Autonomous 5G Networks": Organized by IWPC - The International Wireless Industry Consortium: Hosted by Deutsche Telekom: June 14-16 2016: Cologne, Germany: http://www.iwpc.org/workshops/2016/2016-06-DT/agenda.html

ETSI
06921 Sophia Antipolis CEDEX, France
Tel +33 4 92 94 42 00
info@etsi.org
www.etsi.org