TM Forum Introductory Guide

Autonomous Networks Reference Architecture

IG1251

Maturity Level: General availability(GA)	Team Approved Date: 11-Jul-2022
Release Status: Production	Approval Status: TM Forum Approved
Version: 1.0.1	IPR Mode: RAND

© TM Forum 2022. All Rights Reserved.

Notice

Copyright © TM Forum 2022. All Rights Reserved.

This document and translations of it may be copied and furnished to others, and derivative works that comment on or otherwise explain it or assist in its implementation may be prepared, copied, published, and distributed, in whole or in part, without restriction of any kind, provided that the above copyright notice and this section are included on all such copies and derivative works. However, this document itself may not be modified in any way, including by removing the copyright notice or references to TM FORUM, except as needed for the purpose of developing any document or deliverable produced by a TM FORUM Collaboration Project Team (in which case the rules applicable to copyrights, as set forth in the TM FORUM IPR Policy must be followed) or as required to translate it into languages other than English.

The limited permissions granted above are perpetual and will not be revoked by TM FORUM or its successors or assigns.

This document and the information contained herein is provided on an "AS IS" basis and TM FORUM DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY OWNERSHIP RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

TM FORUM invites any TM FORUM Member or any other party that believes it has patent claims that would necessarily be infringed by implementations of this TM Forum Standards Final Deliverable, to notify the TM FORUM Team Administrator and provide an indication of its willingness to grant patent licenses to such patent claims in a manner consistent with the IPR Mode of the TM FORUM Collaboration Project Team that produced this deliverable.

The TM FORUM invites any party to contact the TM FORUM Team Administrator if it is aware of a claim of ownership of any patent claims that would necessarily be infringed by implementations of this TM FORUM Standards Final Deliverable by a patent holder that is not willing to provide a license to such patent claims in a manner consistent with the IPR Mode of the TM FORUM Collaboration Project Team that produced this TM FORUM Standards Final Deliverable. TM FORUM may include such claims on its website but disclaims any obligation to do so.

TM FORUM takes no position regarding the validity or scope of any intellectual property or other rights that might be claimed to pertain to the implementation or use of the technology described in this TM FORUM Standards Final Deliverable or the extent to which any license under such rights might or might not be available; neither does it represent that it has made any effort to identify any such rights. Information on TM FORUM's procedures with respect to rights in any document or deliverable produced by a TM FORUM Collaboration Project Team can be found on the TM FORUM website. Copies of claims of rights made available for publication and any assurances of licenses to be made available, or the result of an attempt made to obtain a general license or permission for the use of such proprietary rights by implementers or users of this TM FORUM Standards Final Deliverable, can be obtained from the TM FORUM Team Administrator. TM FORUM makes no representation that any information or list of intellectual property rights will at any time be complete, or that any claims in such list are, in fact, Essential Claims.

Direct inquiries to the TM Forum office:

181 New Road, Suite 304
Parsippany, NJ 07054 USA
Tel No. +1 862 227 1648
TM Forum Web Page: www.tmforum.org

© TM Forum 2022. All Rights Reserved.

Table of Contents

© TM Forum 2022. All Rights Reserved.

© TM Forum 2022. All Rights Reserved.

List of Figures

© TM Forum 2022. All Rights Reserved.

List of Tables

© TM Forum 2022. All Rights Reserved.

1. Overview

This guide describes a reference architecture for Autonomous Networks (AN) that can be used to provide guidance on the development of solution architectures for intelligent automation and autonomy within CSP operations. It is not an implementation architecture and as such does not reference specific technologies. This reference architecture builds on the more technical architecture IG1230, which describes the general concepts and principles behind autonomous networks [IG1230]. This guide prescribes the precise AN Framework scope underpinning the principles, the end-to-end requirements of such an industry reference architecture. The reference architecture provides recommended structures and integrations of autonomous platforms, services, and domains to form a mature AN solution. The reference architecture embodies accepted industry best practices (from TM Forum and other SDOs) and recommends the intent and control loop mechanisms for self-management capabilities.

1.1 Guide Structure

Table 1 Guide Structure

#	Section	Description
1	Overview	Describes the scope and terminology used within the AN Reference Architecture. The scope of this doc describes the reference architecture for AN Framework and for the Autonomous Domains that may comprise an entire Autonomous Network.
2	Architectural Principles	Architecture principles define the underlying general rules and guidelines for the use the reference architecture.
3	Architecture Requirements	A holistic look at the different needs of users that the Reference Architecture must address from a purely architectural point of view. Architecture requirements are derived from the scenarios and requirements described in IG1218 and IG1230 and define nonfunctional as well as functional needs that are to be satisfied by the architecture.
4	Reference Architecture	Defines a set of architectural 'building blocks' that form the self-management capabilities underpinning autonomous platforms and systems. Identifies the Reference Points
5	AN Reference Architecture Realizations	Describes how AN Reference Architecture can be realized by other solution architectures. Details how ETSI ZSM's Reference Architecture can be mapped to AN Reference Architecture.
6	AN Integration	The integration fabric of AN Framework. Details of the APIs required to enable interactions across the systems in the 3 operational layers.
7	Summary	A review of the key points of the reference architecture.

© TM Forum 2022. All Rights Reserved.

1.2 Terminology

Table 2 Terminology

Term	Definition / Source
Autonomy	The capability to make decisions free from human control.
Automatic	Able to operate independently of human control
Autonomous Networks	A set of network and software platforms that can sense its environment and adapting its behavior accordingly with little or no human input.
Autonomous Network Levels	(Or simply AN Levels) describe the level of autonomic capability in a given operational workflow or for an autonomous domain (L0 to L5). Autonomous Network Levels identify contextual autonomous capability. Also less formally referred to as 'autonomy levels.
Control loop	Control loops are used to enable autonomous systems to adapt their behavior to respond to changes in user needs, business goals, or environmental conditions.
Intent	Intent is the formal specification of all expectations including requirements, goals, and constraints given to a technical system.
Autonomous Domain	Serves as the basic unit that can fulfill closed-loop automation of specific network operations. An autonomous domain is a set of systems or platforms that is capable of intervention. The autonomous domain does this by realizing self-management capabilities using a closed control loop mechanism, using four key phases: awareness, analysis, decision-making autonomous behavior (e.g., resolve tasks, adhere to objectives) without manual human, and execution. An autonomous domain is a logical construct that provides an administrative governance boundary that defines the scope of the encapsulated autonomous behavior.
Autonomous Platform	An autonomous platform is a system or agent with the ability to complete a task without human intervention, using behaviors resulting from the interaction of software with the external environment. Tasks or functions executed by a platform or distributed between a platform and other parts of the system, may be performed using a variety of behaviors, which may include reasoning and problem identification. In this guide the term 'Platform' is preferred over 'system' notwithstanding the latter's ubiquity. The 'platform' term is not used in the software architecture sense of the term where a platform provides a foundational set of software services for a product.

Table 2 above lists the key terms using the AN technical architecture guide as source [IG1230].

© TM Forum 2022. All Rights Reserved.

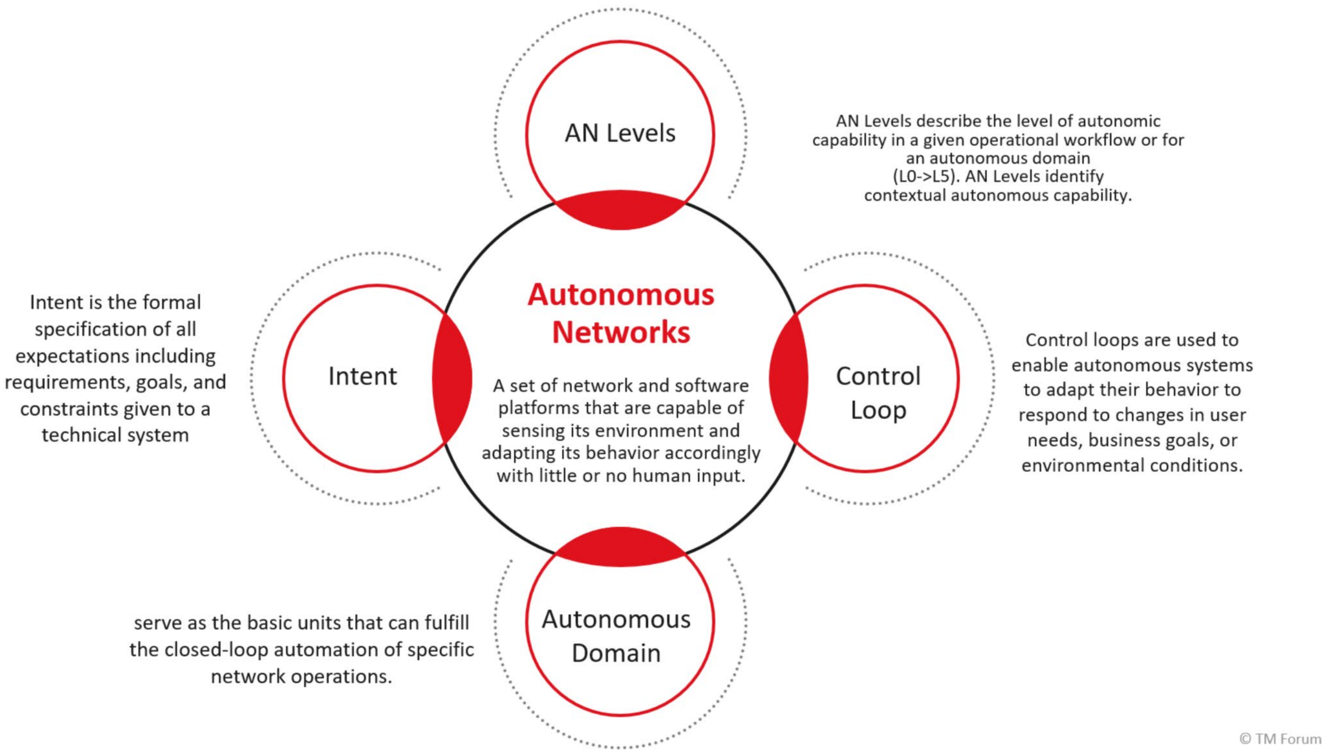

Figure 1-1 AN concept definitions

1.3 Scope

The present document defines and describes the architecture for autonomous networks based on a set of user scenarios and requirements documented in TMF IG1218 and IG1230. This document is intended to describe the autonomous networks architecture framework, which include the three layers of business operations, service operations and resource operations. The framework is the top level design for the autonomous networks, include the design principles, requirements, and architecture diagram. In addition, the present document will establish the relationship between the TM Forum autonomous networks architecture and the autonomous networks architecture of other standards organizations.

Table 3 Context, goals and design of AN Reference Architecture

Dimension of RA	Value	Remarks
Where will RA be used?	Multiple organizations	All stakeholders in telecom ecosystem
Who defined it?	Multiple organizations	CSPs, CSP partners (Solution Vendors, System Integrators, Network Equipment vendors)
‘When’ is it used?	This RA is appropriate to both existing platforms and newly development systems.	Architecture builds on previous autonomic network architectures but goes further. Developed as a longer-term reference to move from L2 to L3, toward L4/5 Autonomy.
Why is it defined?	Standardization	Define a reference architecture to improve industry collaboration efficiency

© TM Forum 2022. All Rights Reserved.

Dimension of RA	Value	Remarks
What is described?	Functional Blocks, Reference Points, Components, Interfaces.	Contributors form multiple stakeholder organizations with different roles and skillsets have been involved in the architecture definition as standardization requires a consensus in order to be successful
How detailed is architecture described?	Aggregated components; (semi) detailed interfaces	See IG1230 and planned guides IG1252, IG1253 for more detail of AN levels and Intent.
How concretely is it described?	Abstract	RA is technology-, service- , and vendor-agnostic.
How formally is it described?	Semi-formal	UML used in part but not for all description. Formal in style but without use of UML or SysML model.

© TM Forum 2022. All Rights Reserved.

tmforum.org

2 Architectural Principles

2.1 Introduction

Architectural principles define the underlying general rules and guidelines for the use and deployment of resources and assets across the enterprise. They reflect a level of consensus among the various elements of the enterprise and form the basis for making future decisions. The overall design objective of autonomous networks is to enable end-to-end autonomous network operations for telecom services, including business operations, service operations, and resource operations, in a multi-domain, multi-vendor environment. This section defines a set of architectural principles (see Figure 2-1) to guide the autonomous networks reference architecture. These principles guide the design of the system capabilities required to achieve the desired autonomy levels within autonomous domains and autonomous networks.

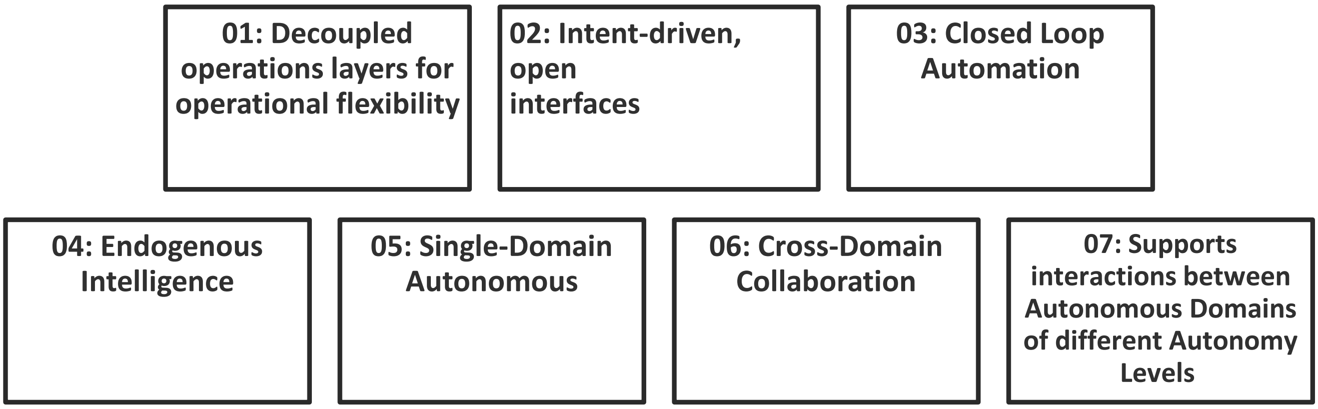

Figure 2-1 Architectural Principles

2.2 Core Architectural Principles

2.2.1 Principle 01: Decoupled Operational Layers supporting Operational Flexibility

The overall architecture should comply with the layered architecture pattern, the autonomous networks architecture is separated into three layers, business operations layer, service operations layer, and resource operations layer. Each layer runs in self-operating mode and hides the details of domain implementation, operations and the functions within the domain to the consumers.

2.2.2 Principle 02: Intent-driven, Open Interfaces

Each autonomous domain (at any of the three operational layers) of Autonomous Networks should expose standardized interfaces. Where appropriate those interfaces should now be intent-driven to provide simplified interaction capabilities for upper-layer services through these open APIs.

© TM Forum 2022. All Rights Reserved.

2.2.3 Principle 03: Closed Loop Automation

Automation uses closed loop mechanisms to assure service experience by completing workflow steps but also continuously adapting to ensure goals and objectives are met. The control loop adjusts and adapts itself (through the decision cycle sages of Awareness, Analysis, Decision, Execution [IG1230]), keeping the system in the desired state without any intervention from outside of a particular loop cycle.

2.3.4 Principle 04: Endogenous Intelligence

Local knowledge, or endogenous ("from within") intelligence, applied locally is a core principle for the internal models of autonomous domains and network elements. More real-time sensing components and AI inference capabilities are introduced to both autonomous domains and network elements to improve observability or digital awareness of resources, services, and surrounding environments. This in turn, enables data sources to have edge intelligence capabilities such as perception analysis and decision execution more locally. For example, AI models could be trained on the cloud and injected into the network through cloud-based collaboration to implement local inference.

2.3.5 Principle 05: Single Domain Autonomy

The autonomy and independence of a domain is a core principle of AN., i.e., each individual or single domain is itself an autonomous domain. It is the independence and self-governing capability of a domain that gives the AN architecture its Lego block appearance. The autonomous domain is an atomic unit with autonomous capabilities in the autonomous network architecture [IG1230]. Individual autonomous domains focus on network technologies, abstract network capabilities, and simplify OSS integration.

2.3.6 Principle 06: Cross Domain Collaboration

In the AN framework, each domain is autonomous, but domains must also collaborate in the following aspects:

i. end-to-end services need to be coordinated between autonomous domains. In the reference architecture practice, administrative domains may have a hierarchical structure, and cross-domain service management may manage end-to-end services across multiple autonomous domains, and coordinate between administrative domains through orchestration.
ii. domains may federate with neighbors, allow upper layer domain only dispatch intent based request without elaborate details which could be coordinated between lower domains by themselves.
iii. aggregation of resource domains may happen for administrative or management purpose. The administrator may merge part of the operations of several domains in a converged spot, arrange unified planning, operation tasks, etc.

© TM Forum 2022. All Rights Reserved.

2.3.7 Principle 07: Supports interactions between Autonomous Domains of different Autonomy Levels

Autonomous Domains of different levels have different network autonomy capabilities and corresponding interfaces, the interactions between domains of different levels should be allowed from architecture perspective, and so is the evolution (L0 to L5) of each domain. The 'runtime' within domains is always dynamic as situations and requirements change during operations, so the autonomy levels supported may also change depending on the capability or competence of the autonomous domain.

© TM Forum 2022. All Rights Reserved.

3 Architecture Requirements

3.1 Introduction

This section defines requirements applicable to the autonomous networks reference architecture framework. Architecture requirements are derived from the scenarios and requirements described in TM Forum IG1230 and IG1230A.

3.2 General Requirements

The following architectural requirements inform the level of detail and scope that a *reference architecture* for AN requires. As such, the specific constraints or goals of the architecture are articulated in the form of requirement objectives (see tables 4 and 5 below).

Table 4 General Architectural Requirements

Req#	Requirement
[ARC-001]	The autonomous networks reference architecture shall separate the system into three layers, business operations layer, service operations layer, and network operations layer.
[ARC-002]	The autonomous networks reference architecture shall support four closed control loops, business operations layer closed control loops, service operations layer closed control loops, and network operations layer closed control loops, and end to end closed control loops.
[ARC-003]	The autonomous networks reference architecture shall support the hierarchical autonomous domain.
[ARC-004]	Each architecture layer should have its own AI capabilities.
[ARC-005]	No waste - intelligent use of basic resources. The world is essentially a closed system regarding resources, and thus their usage should be minimized, and they should be recycled for sustainability.

In short, the goals of AN are to affect the 'top and bottom lines' of an operator, namely, to enable simplification and achieve operator opex cost reduction while at the same time providing a significantly better service 'fit' for end-consumers of AN and thereby increase operator revenues. The architectural requirements in Table 4 list the former, i.e., operational requirements while Table 5 below lists the latter, i.e., the end-customer experience requirements of the vertical ecosystem.

© TM Forum 2022. All Rights Reserved.

tmforum.org

Table 5 AN Consumer Requirements

[ANC-001]	Autonomous Networks should include customer-facing Interfaces. Expose interfaces (APIs) to customers, allowing them to control services at a much lower cost
[ANC-002]	Autonomous Networks service capabilities should be Self-Service where customer has direct control of services. Services autonomously adapt to customer intent. Customers can adapt services themselves, without sending orders / tickets / or having phone calls. This also applies to Partnering in operations themselves. This allows multiple autonomous domains and platforms to work together to deliver overall business operations.
[ANC-003]	Autonomous Networks should provide the ability to have 'On-demand' delivery – that is where services when required, can be flexibly scaled-up or scaled-down. Services can be autonomously turned on and off as needed.
[ANC-004]	Autonomous Networks should expose customized end-user Services. Services delivering the correct mix of features for the specific vertical use case
[ANC-005]	Ecosystem offerings: Autonomous Networks should provide CSPs need business platforms that can facilitate better interworking with ecosystem partners to deliver packaged offerings to the 'Vertical Consumer' or 'AN Consumers'.
[ANC-006]	The ease of consumption and integration into end-customers' business must be a requirement for an autonomous network and for the touchpoints within the business operations layer. See Service 'Consumability' in IG1229 Guiding Principles for more on this term.
[ANC-007]	The Autonomous Network should include the ability to operate with external regulation

3.3 Autonomous Domain Requirements

Table 6 Autonomous Domain Requirements

Req#	Requirement
[DOM-001]	The autonomous domain should support operation in an independent and self-operating mode.
[DOM-002]	The autonomous domain should support Intent-driven API, hides internal functions and implementation details, and providing intent verification and reporting capabilities.
[DOM-003]	Autonomous Domains should be able to operate in 'Isolation' mode when other systems are not resilient to failure
[DOM-004]	Autonomous Domains should be designed for uncertainty or unexpected situations. Decision-making is still required in scenarios that are unforeseen.
[DOM-005]	Autonomous Domains should be clear on what levels of autonomy are supported, what capabilities are supported and in what situations.
[DOM-006]	The knowledge base used by Local Intelligence in an Autonomous Domain should be deployed locally.
[DOM-007]	Autonomous Domains can be either *atomic* or *composite*. Atomic here refers to its ability to operate internally as an independent functional block. Composite domains are the result of coupling particular domains together to form a functional composite where there are dependencies on the contained domains within the composite.

© TM Forum 2022. All Rights Reserved.

Req#	Requirement
[DOM-008]	Autonomous Domains are ‘federatable’ – i.e., can enable operations management through negotiations composed of multiple autonomous domains. It should be possible to connect any implementation of the autonomous domain to all others that were independently developed, in a federated structure, to allow the sharing of appropriate information and workflows between them.

© TM Forum 2022. All Rights Reserved.

tmforum.org

4 AN Reference Architecture

4.1 Architecture Overview

4.1.1 Introduction

This section defines the reference architecture for Autonomous Networks as

i. a capability-based layered architecture
ii. that is project, market and vendor-agnostic, and does not prescribe any specific tools or technologies
iii. an architecture that enables modular and incremental implementations

The autonomous networks reference architecture is based on the layered-architecture design pattern. Functional blocks within the layered architecture are organized into three operational layers. The layers are decoupled and evolved independently and interact with each other through prescribed reference points that detail the logical communications required between two layers. For the subsystem architecture of each layer, a service-based architecture is used to support architecture flexibility and scalability. The autonomous networks architecture defines a group of logical building blocks, called autonomous domains that construct the overall logical architecture of the autonomous network. This service-based architecture provides the requisite modularity of the framework using autonomous domains where domains can be deployed from a variety of sources and suppliers. Domains can be interconnected using interfaces defined at the relevant reference points.

4.2 Reference Architecture

4.2.1 AN Reference Architecture Diagram

Figure 4-1 depicts the TMF AN Reference Architecture that describes a layered yet ‘composable’ framework for autonomous operations. It emphasizes and echoes all of core architectural principles outlined in Chapter 2. Operational layers are fully decoupled (Reference points I1,I2,I3). Open interfaces are used at all integration points between domains with intent-driven interfaces being recommended at I and F reference points. Intelligence is also layered and distributed closest to the ‘doer’ the functional block that executes an action, i.e., Intelligence is decentralized and localized *within* its own domain. This is the approach that best suits fast[1] decision-making adaptive control loops. Autonomy is maintained within these logical units of management, namely Autonomous Domains. Complexity is encapsulated with these constructs to compartmentalize the capabilities so that the AN can be assembled from multiple federated domains horizontally and vertically.

[1] What ETSI GANA terms ‘fast control loops’

© TM Forum 2022. All Rights Reserved.

tmforum.org

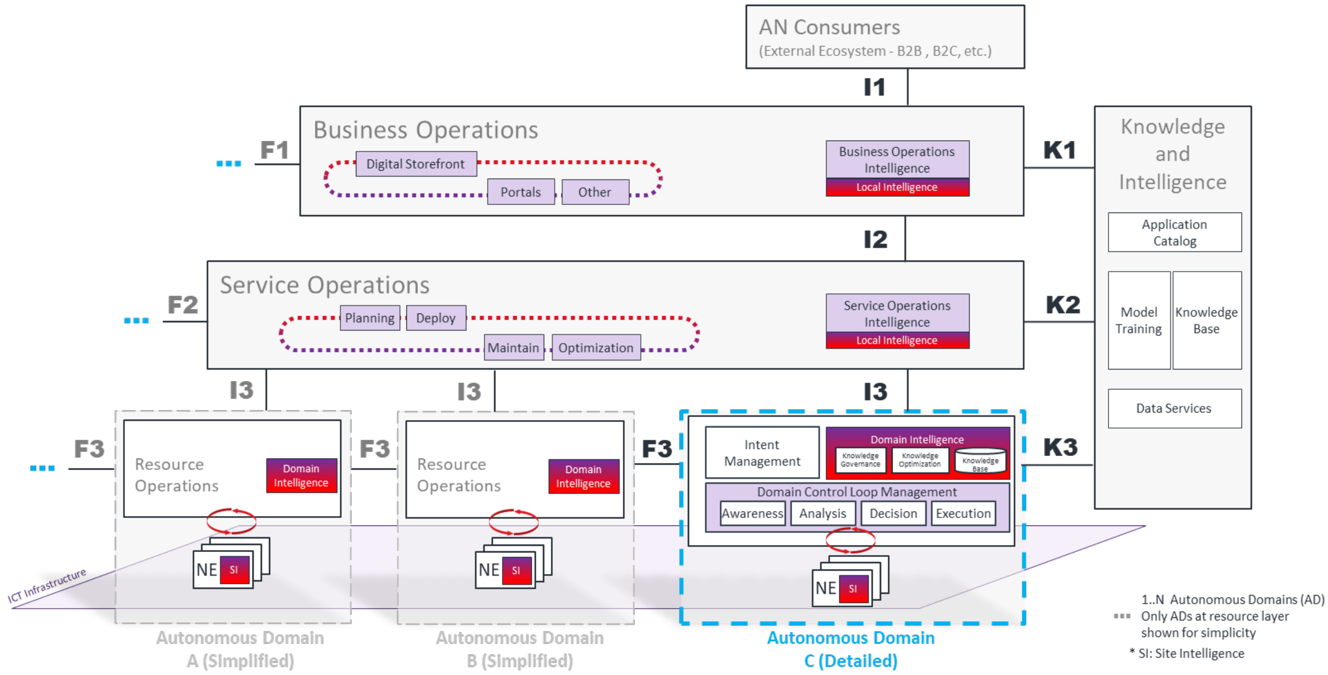

Figure 4-1 AN Reference Architecture

Note that Figure 4-1**Error! Reference source not found.** shows domains at a single layer for simplicity but these building blocks repeat at all three layers. Autonomous Domain C is shown as an exploded view of the autonomous domain, illustrating some of the inner functions that comprise a domain. Domain C and Domain B and Domain A all cooperate in a federated manner (using F3 in the figure above) but federation applies to all three operational layers. The Service Operations layer achieves a cross-domain collaboration of the Domains A, B, C to provide end-to-end coordination of all of those resource operation domains. The blue dashed rectangle for Autonomous Domain C (Detailed) shows the mechanisms and functions that embody an Autonomous Domain.

1. *Operational Layers* are a logical construct that organizes platforms along three distinct horizontal axes namely Business, Service, and Resource Operations.
2. *Autonomous Domains* are also a logical construct that group particular platforms[2] into a coherent domain of autonomy. We call these platforms *autonomous platforms*.
3. An Operational Layer can encompass the operations provided by *multiple* autonomous platforms.
4. An Autonomous Domain can comprise of multiple autonomous platforms.
5. An Autonomous Domain can comprise of autonomous platforms from multiple operational layers.

4.3 Autonomous Domain

4.3.1 Autonomous Domain Definition

An *Autonomous Domain* is an operational management domain that defines the scope of encapsulated autonomous behavior [IG1230]. It is the ‘building block’ (i.e., unit) of autonomous

[2] *** Platform and Domain are specific terms in TM Forum described in TR255. The semantics of these terms have been respected in the ANF definitions.*

© TM Forum 2022. All Rights Reserved.

behavior that when federated together form a complete Autonomous Network. It serves as the basic unit that can fulfill closed-loop automation of specific network operations. An autonomous domain is a set of systems or platforms that is capable of intervention. The autonomous domain does this by realizing self-management capabilities using closed control loop mechanisms, using four key phases: awareness, analysis, decision-making autonomous behavior (e.g., resolve tasks, adhere to objectives) without manual human, and execution. An autonomous domain is a logical construct that provides an administrative governance boundary that defines the scope of the encapsulated autonomous behavior.

4.3.2 Autonomous Domains Building Blocks

The dashed red and blue rectangular boxes in Figure 4-2 illustrate how ADs can be assembled to form a hierarchy of platforms using Intent-driven interactions at the vertical (governance) and horizontal (federation) levels. Note that the I1, I2, I3 reference points are 'reused' for communication between two different domains but the reference point is still specific to the operational layer. For example, if you take the scenario of two ADs at the business operations layer, then both ADs are federated using the I1 reference point as I1 is specific to business operations. Similarly, I2 is for service operation interfaces and I3 for resource operations interfaces. The next section describes the federation of domains in more detail.

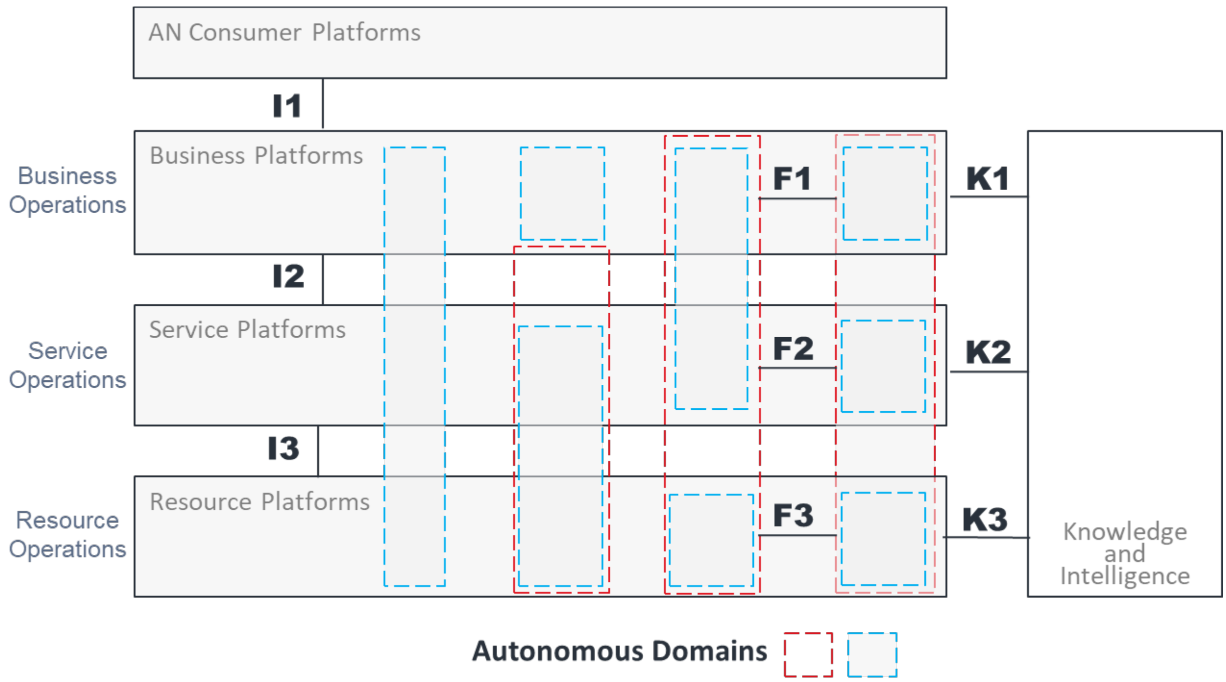

Figure 4-2 Autonomous Domains Building Blocks within the Reference Architecture

© TM Forum 2022. All Rights Reserved.

4.3.3 Composite Autonomous Domains

Autonomous Domains can be either *atomic* or *composite (see* Figure 4-3*)*. As stated in an earlier section on Autonomous Domain Requirements, 'Atomic' here refers to its ability to operate internally as an independent functional block. Composite domains are the result of coupling particular domains together to form a functional composite where there are dependencies on the contained domains within the composite.

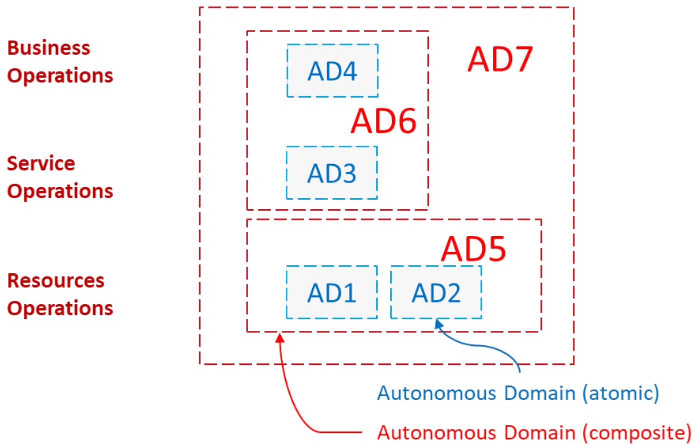

Figure 4-3 Autonomous Domains are composable

1. Autonomous Domains (AD) have well-defined boundaries (dashed[3] rectangles).
2. Domains may be atomic (AD1) or composite[4] (AD5) (i.e., composed of two or more AD's)
3. An Autonomous Domain can support multiple "layers" of the Autonomous network (e.g., AD6)
4. An entire Autonomous Network could therefore be bounded as a single Autonomous Domain (e.g., AD7)

4.3.4 Federated Autonomous Domains

The previous sections emphasize that Autonomous Domains are decoupled vertically (top-to-bottom) using I1, I2, I3 reference points and also how these domains can be federated (horizontally) using the same reference points. **Error! Reference source not found.** shows that the same interfaces are used to realize both the layer decoupling and the federation, i.e., the "vertical" and "horizontal" interactions of Autonomous Domains. That is, the same interfaces are reused at the I and F Reference Points.

[3] *Any dashed or dotted line or shape in ANF means that the construct is logical and conceptual rather than a physical construct.*
[4] *Refer to IG1230 Section 5.2.1 Autonomous Domain is a composite structure.*

© TM Forum 2022. All Rights Reserved.

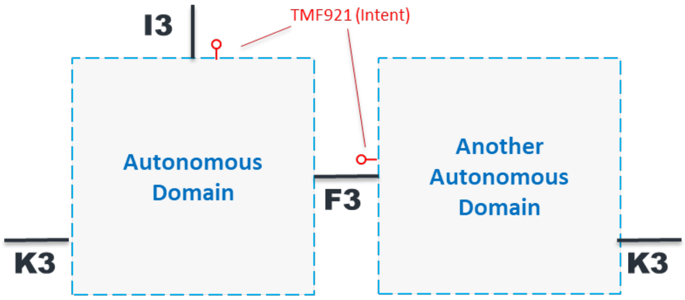

Figure 4-4 Domains can be Federated using same Intent Interfaces at I and F Reference Points

For completeness, the figure below shows the same approach but for all three operational layers.

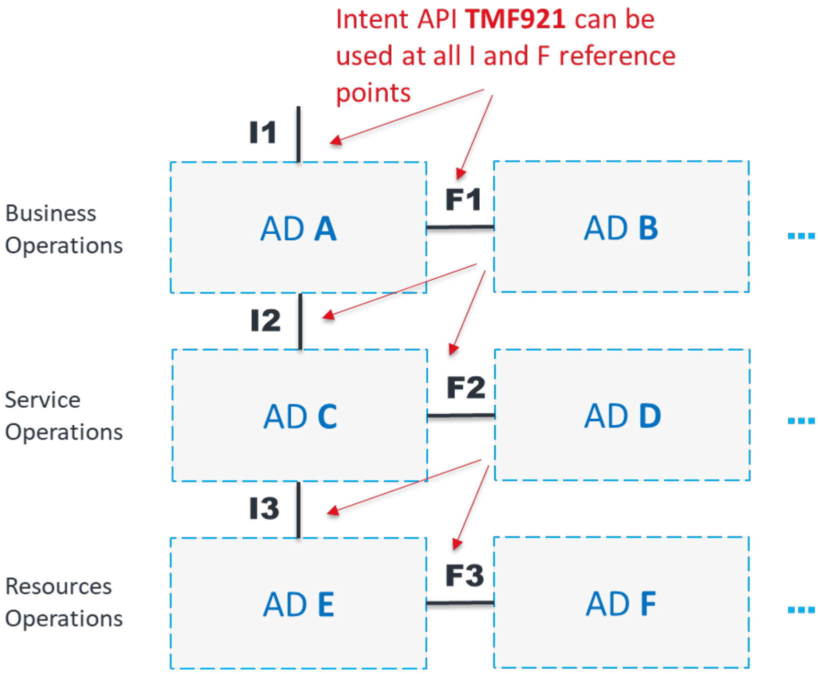

Figure 4-5 Domains can be federated at all operational layers

Federation of domains means autonomous collaboration of platforms across multiple self-governing domains. A federation is a peer-to-peer relationship between two or more parties involved in order to achieve a common goal [FED]. For example, in a resource operations layer, multiple ADs may be interfaced in a layered, peering, or orthogonal way to combine operational contexts and promote a virtualization of an autonomous network over the physical infrastructure. The benefit of this flexible network architecture is that it scales over time and in a functional way; that is, the network and its operations can extend both horizontally as well as vertically and change over time. This architecture is functionally adaptive and scalable and is the basis for an evolving network that incorporates various self-X capabilities such as self-optimization, self-healing, and self-protection [IG1218]. These self-X capabilities are realized in individual control loop mechanisms defined within domains and across domains.

© TM Forum 2022. All Rights Reserved.

4.4 Functional Blocks

The overall architecture is composed of a set of functional blocks defines across the three operational layers and the framework. The functional blocks realize the behavior required to realize initially Autonomous Domains as 'building blocks' and secondly the other blocks required by platforms to create a coherent reference architecture. The most fundamental abstraction used is the nested, composable nature of domains that when chained together or assembled form an autonomous network.

4.4.1 AN Consumer

AN Consumer Platforms are the systems where the end-customer interacts with the autonomous network(s). AN Consumer platforms comprise of:

1. CSP Consumers Services (B2C)
2. CSP Services to Enterprises (B2B)
3. CSP Services to Enterprises' Customers (B2B2X)
4. CSP Internal Applications

4.4.2 Functional Blocks within Autonomous Domain

Table 7 Functional Blocks of Autonomous Domain

Functional Block	Abbr.	Description
Intent Management	IM	Intent management unit, which implements external intent API interaction. (Including System Class (Intent Handler Registration) (Capability Release), other intent entity discovery and connection establishment) and intention interaction class (receiving, negotiation, and reporting). Intent Instance and Lifecycle Management (Intent realization and intent assurance, including sub-intention generation and management, are completed through intent closure.)
Domain Intelligence	DI	Provides local knowledge governance capabilities, provides local AI inference based on the local knowledge base and optimizes local parameters.
Knowledge Governance	KG	Provide local knowledge governance capabilities, including local inference (including local AI inference and knowledge inference) and continuous local model training.
Knowledge Optimization	KO	Local AI parameter optimization and knowledge optimization. Provide on-site model development and model retraining capabilities. On-site model development provides lightweight development services to quickly obtain personalized models applicable to sites. Model retraining is performed periodically based on live-network samples to obtain new models to ensure model accuracy.
Knowledge Base	KB	Local Knowledge Base. A dynamic repository for facts and knowledge. To provides Local Knowledge Base and AI asset management capabilities, Assets include models and data, etc. E.g., collecting and releasing AI assets, demonstrating model application result (include intent report), and model interpretation.
Domain Control Loop Management	CLM	Provides the closed loop processing capability in an AD, processes external intents by sensing, analyzing, making decisions, and executing, and maintains intents in an AD.
Awareness	Aw	Provides AD state awareness and information collection for intent execution such as environment information, faults, events,

© TM Forum 2022. All Rights Reserved.

Functional Block	Abbr.	Description
		logs, and performance data. Analyze service impacts by correlating collected multi-dimensional original data, identify incidents that affect services, and predict risks.
Analysis	An	By analyzing the data or context information in incidents or risks generated in the sensing phase, the model inference and analysis techniques are used to further predict the future change trend of the network state, and the decision-making suggestions are proposed.
Decision	D	Based on the decision-making suggestions provided in the analysis phase, use a specific rule or man-machine collaboration, interface with local knowledge inference, and provide required management operations, such as network configuration or parameter adjustment.
Execution	E	Performs management actions and responds with results/feedback to the intent sender (intent owner).

As described in IG1230, the autonomous domains embody an adaption loop that consists of several phases or processes (*Awareness, Analysis, Decision, Execution*) as well as sensors and effectors on the managed entity (e.g., Network Element).

Table 8 shows the specific network automation control loop that takes action on the network element (NE) target of the resource control loop.

Table 8 Network Element in Resource Control Loop

Functional Block	Abbr.	Description
Network Element	NE	A Network Element represented the device that is under management.
Site Intelligence	SI	‘Site Intelligence’ refers to an AI inference unit of the Network Element. This block can deployed/applied to a broad range of Network Elements across multiple network domains and infrastructures

4.2.3 Knowledge and Intelligence

To achieve the vision of AN, CSPs will also need an intelligence platform, powered by AI, to enable cross-domain and complete closed-loop service automation. We term this platform “Knowledge and Intelligence” within the reference architecture. This platform is responsible for aggregating and federating data sources from the operational layers, generating AI/ML models and supporting cross-domain automation with end-to-end insights, predictions, and recommendations.

‘Knowledge and Intelligence’ provides both human and machine intelligence and analytics services to the AN operational layers.

© TM Forum 2022. All Rights Reserved.

Table 9 Knowledge and Intelligence

Functional Block	Abbr.	Description
Knowledge and Intelligence	K&I	Provide Intelligence services for all three operational layers. Includes (AI/ML) model training, data services, knowledge base, and AI/Analytics application marketplace. Works with Localized Intelligence (see Local Intelligence functions block) where local data and training services are provided by the layered platforms. It does this using 'local-cloud linkages' defined at the K1, K2, K3 reference points.
Application Catalog (i.e., app market)	AC	Application Catalog manages the published AI applications/AI models in a secure catalog. The App Catalogs contain detailed information on ownership and execution requirements of individual models to simplified model selection.
Model Training	MT	This service provides an integrated development environment for a one-stop training design environment, model development services, Domain model service, federated learning, and knowledge graph. The Training Platform delivers models to the localized intelligence in the operational platforms and (optionally) receives offline data to tune its training algorithms (using K1, K2, K3 ref points as appropriate depending on the operational layer in question).
Knowledge Base	KB	A repository system that represents knowledge explicitly. The KB can also use tools enabling tacit knowledge exploitation also – e.g., a reasoning system that allows it to derive new knowledge and facts used to perform decision-making and reasoning within K&I.
Data Services	DS	Services may include Unified data modeling, dataset development and platform and data security dataset enhancement, digital network insight, and simulation

4.5 Reference Points

The reference architecture defines nine reference points to identify the communication that must occur between the functional blocks previously defined in the AN framework architecture [IG1230]. Three of the reference points identify the interactions between the three operational layers of the AN framework, namely business, service, and resource operations. These reference points are vertical in orientation and are termed I1, I2, and I3. Domains within the operational layers can all communicate in a horizontal orientation to achieve a federation, and these are termed F1, F2, and F3. And finally, three reference points in a horizontal orientation connect the operational layers to a Knowledge and Intelligence platform. These are termed K1, K2, and K3.

Table 10 AN Reference Points

Reference Point	Description
I1	I1 identifies the reference point for interactions between AN Consumers platform and the Business Operations layer. Intent-driven interactions are required at this reference point. Intent delivery interface between the business layer systems and the consumers' application systems. Intent report interface between the Business layer systems and its customer application systems.

© TM Forum 2022. All Rights Reserved.

Reference Point	Description
I2	I2 identifies the reference point for interactions between Business Operations layer and the Service Operations Layer. Intent-driven interactions are required at this reference point. Intent delivery interface between the service layer systems and the consumers' application systems. Intent report interface between the service layer systems and the consumers' application systems.
I3	I3 identifies the reference point for interactions between Services Operations layer and the Resource Operations Layer. Intent-driven interactions are required at this reference point. Intent delivery interface between the resource layer systems and the consumers' application systems. Intent report interface between the resource-layer systems and the consumers' application systems
F1	F1 identifies the reference point for interactions between autonomous domains at the Business Operations layer. Intent-driven interactions can be used to assemble federated autonomous domains. Intent delivery interface between the resource layer systems and the consumers' application systems. Intent report interface between the resource-layer systems and the consumers' application systems
F2	F2 identifies the reference point for interactions between autonomous domains at the Service Operations layer. Intent-driven interactions can be used to assemble federated autonomous domains. Intent delivery interface between the resource layer systems and the consumers' application systems. Intent report interface between the resource-layer systems and the consumers' application systems
F3	F3 identifies the reference point for interactions between autonomous domains at the Resource Operations layer. Intent-driven interactions can be used to assemble federated autonomous domains. Intent delivery interface between the resource layer systems and the consumers' application systems. Intent report interface between the resource-layer systems and the consumers' application systems
K1	Training sample data collection interface between the Business layer systems and the offline K&I training systems. AI inference model delivery interface between the AI offline training systems and the business layer systems
K2	Training sample data collection interface between the Service layer systems and the offline K&I training systems. AI inference model delivery interface between the offline K&I training systems and the Service layer systems.
K3	Training sample data collection interface between the Resource-layer systems and the offline K&I training systems. AI inference model delivery interface between the offline K&I training systems and the Resource layer systems

4.5.1 Interfaces at Reference Points

The nine reference points identified will be realized by a set of interfaces, either service-based interfaces or more peer-to-peer interfaces. These interfaces or APIs will become control points between two platforms (system) as an implementation of the functional blocks.

Table 11 Interface at Reference Points

Interface	Reference Point	TM Forum Reference	Payloads
Intent Management API Suite	I1,I2,I3 F1,F2,F3	***TMF921*** (Planning stage)	Intent Expression is passed in the interface message body (payload) in

© TM Forum 2022. All Rights Reserved.

Interface	Reference Point	TM Forum Reference	Payloads
	**		the request with Intent Reporting as response(s)
AI Closed Loop API Component Suite	K1,K2,K3	**TMF919** (Planning stage)	Training Data Data for Federated Learning (i.e., for both *Site Intelligence* and *Local Intelligence*)
Note: More Interfaces will be identified as part of future project work			

** Caution: TMF921 is not the only interface needed at I and F Reference Points

It is important to note that we expect many more interfaces will need to be specified at these nine reference points in order to realize a fully operational autonomous network. The TM Forum AN Project team working with the AIOps Initiative team and the TM Forum's API, and AI project teams will collaborate to build out a map of the required interfaces at these reference points.

4.5.2 Intent-driven Interfaces

Operations automation is needed to support the fast deployment requirements of the 5G new services. Operators will need to be more efficient to offer the best possible user experience, with full lifecycle automation of the network and maximum utilization of the network resources. The networks' automation relies on an automated closed loop driven by the provided intent. As such, closed loop automation needs to be in each operational layer including the network elements (managed entities) at the resource operations layer, and the cross-domain service operations layer to achieve the goal of network autonomy.

Cross-domain collaborative closed loops and single-domain autonomous closed loops coordinate and exchange information with each other by open interfaces. To reduce the integration complexity between the layers, the AN reference architecture requires a simplified open interface between the platforms at the lower layer and its upper layer consumer platforms. The information exchanged through these interfaces will change from the current data-centric and parameter-heavy payloads to an exchange of *intents (see [1230] and [IG1253]*. The simplification of the open interfaces, in turn, relies on autonomous network capability in each autonomous domain regardless of what operational layer the domain encompasses.

A key capability of intent-driven autonomous systems is to infer the intent of the user, rather than require the end-user to provide detailed instructions. These systems should be able to adapt to the needs of the user and understand their language and learn how to *interact* with users. An intent interface may be an API with an explicit "grammar" for intents (other synonyms demands, requirements, expectations, objectives). It may also use a "dialog" style of interaction, i.e., a *conversational API* with intelligent 'back-and-forth' between the owners and handlers of the intent. The nature of this dialog could use an "interview" style interface in which the handling system requests additional information from the user as needed, to provide clarifications, to provide alternatives, to recommend changes to intent ("Intent Negotiation").

The I1,I2,I3 reference points include many different communication styles, some traditional mechanisms but also new intent-driven interactions. TMF921 Intent Management API Suite will provide a set of interfaces to support Intent-driven interactions.

© TM Forum 2022. All Rights Reserved.

tmforum.org

5 AN Reference Architecture Realizations

5.1 ZSM Reference Architecture

5.1.1 Service and Resource Operations Realization

The TM Forum Autonomous Network Framework has been broadly embraced by many industry standard developing organizations. These SDOs have positioned the TM Forum Autonomous Networks architecture as the top-level architecture for reference and instantiation. The ETSI ZSM group was formed with the goal to accelerate the definition of the end-to-end service management architecture, spanning both legacy and virtualized network infrastructure, to enable automatic execution of operational processes and tasks.

ZSM instantiates the *services and resource operations layer* of autonomous networks architecture as shown in Figure 5-1.

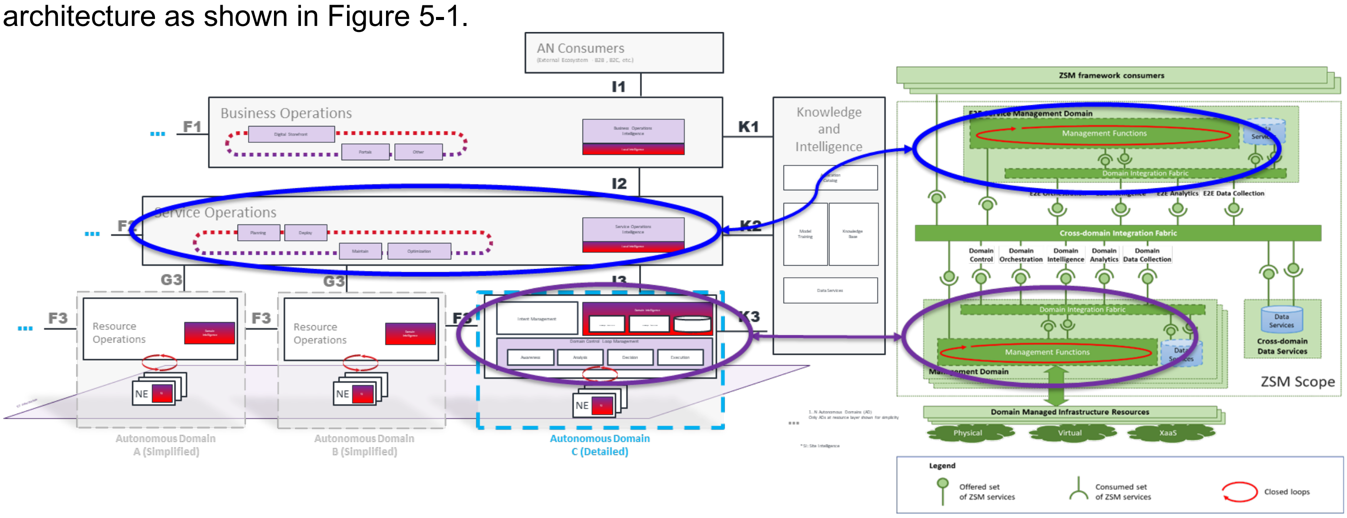

Figure 5-1 ETSI ZSM Architecture as the realization of AN RA

The E2E Service Management Domains contain management functions and management services that align to the autonomous platforms of TM Forum's AN Frameworks Service Operations layer. The ZSM Management Domains contain management functions and management services that align to the autonomous platforms of TM Forum's AN Frameworks Resource Operations layer. There inner service closed loop within ZSM architecture conform to the respective inner closed loops of the TM Forum's AN Framework. There is a correlation or alignment between both architectures.

© TM Forum 2022. All Rights Reserved.

tmforum.org

6 AN Integration

6.1 Example of Intent-driven AN Reference Architecture Call Flows

Figure 6-1 below shows an overview of the message flows between the key functional blocks of the reference architecture. Intent-driven interactions at the I1, I2, I3 reference points drive the closed control loops within the layers which are informed by localized knowledge with the AN Framework.

The detailed flows within the intent management function and the control loop management functions are omitted for clarity and brevity. The important messages are 1, 4 and 7 where message cross the operational layer boundaries.

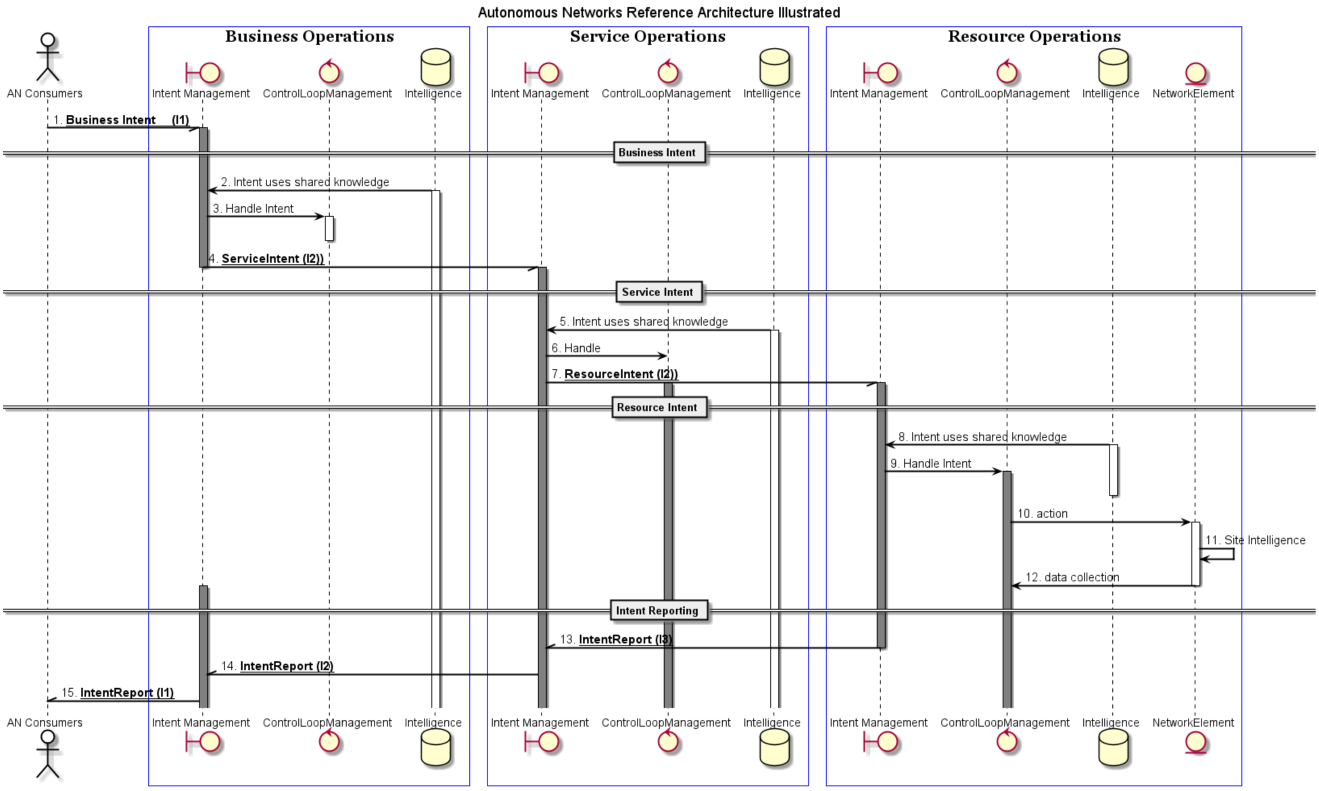

Figure 6-1 E2E Intent-driven Autonomous Network

Note that the three Intent interactions (see messages 1, 4, and 7 above) are realized by the forthcoming TMF921 API specification.

© TM Forum 2022. All Rights Reserved.

6.2 Example of Call Flows between Autonomous Domains

The figure below shows an example of the message flows between the autonomous domains of the reference architecture. Intent-driven interactions at the I1, I2, I3 reference points connect the autonomous domains and create the integration fabric.

Many variants of the intent negotiation flow exist and beyond the scope of this guide but described in the Intent in Autonomous Networks guides [IG1253]. One variant is where message 6) below is sent to the third Intent Management function and it then communicates with Intent Management in fourth AD. Here the Intent Management of the third and fourth Autonomous Domain federate the domains together and each 'resource operations' domain negotiates with the other and then communicates to the cross domain service AD.

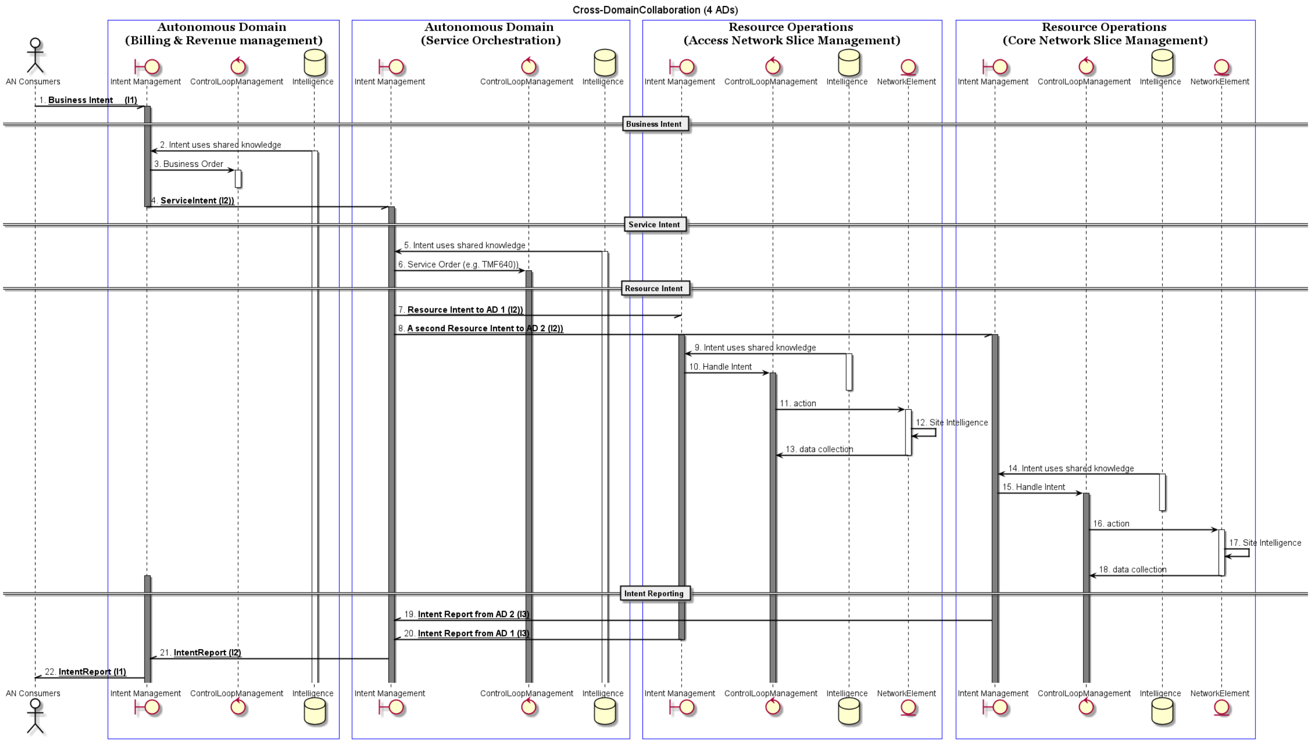

Figure 6-2 AN Reference Architecture Call Flows (Multiple ADs within single layer)

The following section details the main interactions between the functional blocks within the overall reference architecture. A 'thin slice' is examined in the form of the I1,I2,I3 reference points that go from the consumer to network element and back. As such this set of interactions describes the outer *customer loop* from the AN framework itself.

The procedure, as illustrated in Figure 6-3 Autonomous Networks Reference Architecture Illustrated (UML Sequence), consists of the following steps:

© TM Forum 2022. All Rights Reserved.

Business Intent

1. AN Consumer sends a Business Intent to Intent Management (IM) function at the Business Operations layer
2. The Intent Management function requests contextual knowledge based on part of the intent expression content.
3. The Intent Management function handles the intent request by translating the Intent Expression into a directive for a Control Loop Management (CLM) system.
4. The CLM starts at the *analysis* stage by determining affected contextual data requirements and mines contextual data from the knowledge base.
    a. Details of the workflow loop connection creation, interactions with the knowledge base, and health checks are omitted.
5. The *decision* stage then compares characteristics of results with policies and constraints defined for that capability
6. The *execute* stage sends actions to the managed entity in this layer.
7. An awareness step that monitors the outcome of the action taken completes and the CLM reports back on the outcome
8. The Intent Management function decides to issue a newly generated intent southbound and sends it to a downstream intent management function of the service operations layer (I2)

Service Intent

9. Steps 10 to 16 are similar to the above steps 2 to 8
17. The Intent Management function sends a resource intent to the Intent Management function defined for an autonomous domain at the resource layer.

Resource Intent

18. (Steps 18 to 22) The CLM of the Resource Operations layers executes the AADE loop mechanism as steps 2 to 8 above
23. The CLM executes an action on the Network Element (NE)
24. (optionally) Site Intelligence at the Network Element is performed.
25. Data collected from the NE is provided as part fo Awareness stage of control loop
26. Awareness stage is completed and the Intent reporting phase begins
27. The CLM functions responds to the IM function , and
28. The IM sends a intent report to its ‘owner’ intent caller.
29. This IM at the service operations layers also reports to its ‘owner’ Busines Intent caller
30. The IM at the business operations layer then reports on teh Intent based on what was the outcome was at the service layer.

Note 1: The Awareness, Analysis, Decision-making, Execution (AADE) control loop pattern defined in IG1230 is based on the well-known MAPE-K pattern. For clarity, the individual components and the messages between the four stages are not shown as individual actors in the UML sequence diagrams. Rather, the generalized steps of AADE are shown as part of the CLM lifecycle.

Note 2: Many more illustrative UML sequence flows across AN are possible, such as more detailed call flows detailing the use of F reference point interaction to federate domains together, as well as call flows showing the use of K reference points from the local domains to the

© TM Forum 2022. All Rights Reserved.

tmforum.org

Knowledge and Intelligence platform, but are not included here for reasons of brevity and also so that we do not advocate particular *implementation* centric views of this RA.

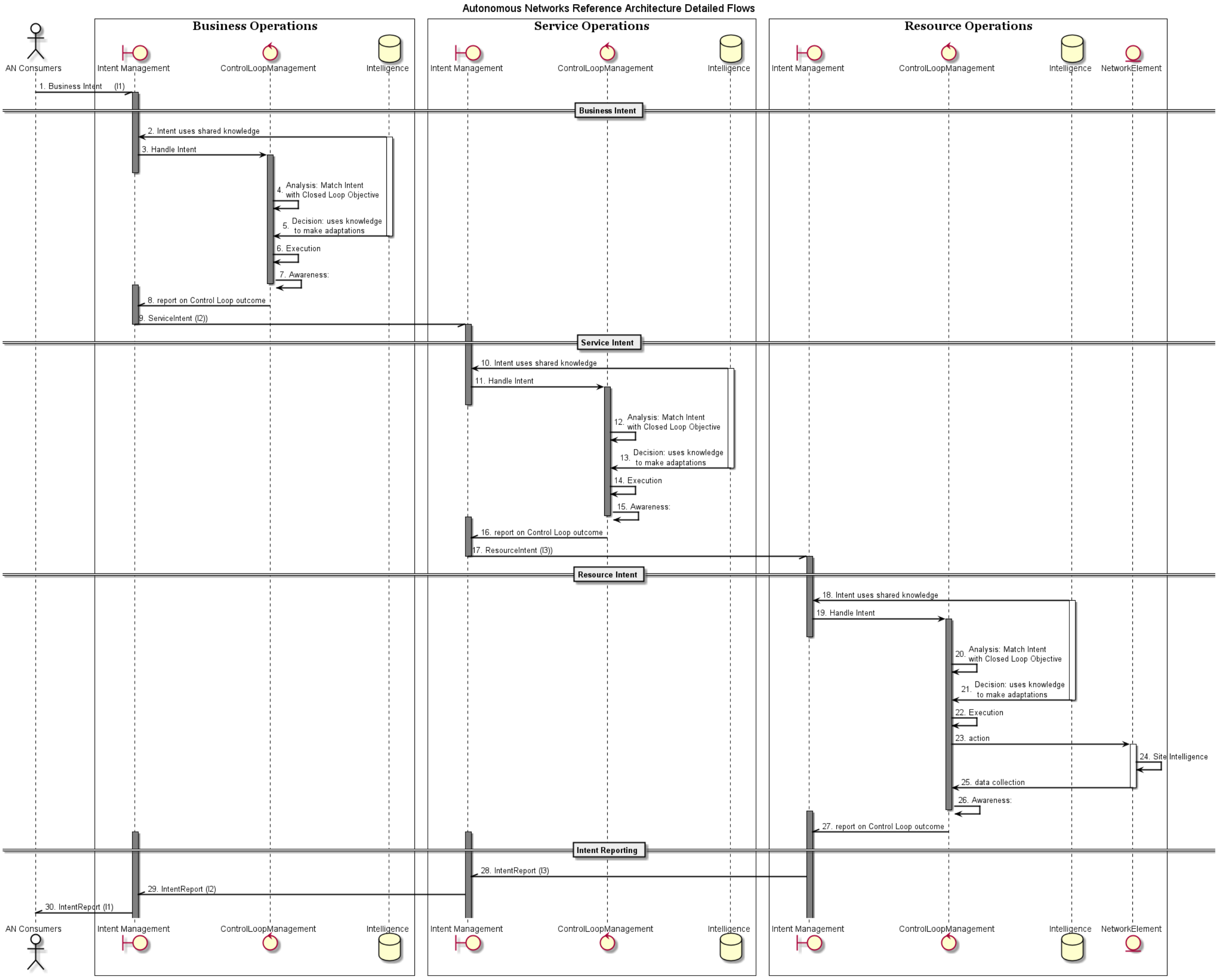

Figure 6-3 Autonomous Networks Reference Architecture Illustrated (UML Sequence)

© TM Forum 2022. All Rights Reserved.

tmforum.org

7 Summary

Autonomous does not mean “unmanned”. Humans are still needed in the operations of Autonomous Networks. However, humans are now ‘on the loop’ operational supervisors rather than ‘in the loop’ bottlenecks to automation. Humans are good at providing input to systems – much like a teacher to a student – this is where expert knowledge will be accumulated over time to build human intelligence into autonomous platforms. As more and more systems take on complex tasks autonomously, we can still rely on humans to be there for the tasks that the system cannot handle – this will work to also build the necessary *trust* we need in autonomous operations in Autonomous Networks.

This guide introduces the AN Reference Architecture and provides realization scenarios instantiating the architecture as described. The architecture is defined at a functional level with the identification of external reference points.

Three classes of reference points are defined, these are:
I interactions between operational layers
F interactions between autonomous domains within layers
K interactions between automation domains at a given layer and a knowledge and intelligence platform

The important interfaces required at each reference point have been defined, with standardization work on intent-driven interfaces and AI closed loop management already underway. A complete implementation of the reference architecture is not simply achieved in one project or iteration but rather can be **achieved step-by-step** - starting with a single domain at a time and then building cross-domain coordination until you ultimately achieve a full Autonomous Network.

The Autonomous Network will be intent-driven, context-aware, and guided by knowledge and intelligence. Autonomous Domains provide the modular building blocks for CSPs to assemble these hierarchical self-governing networks.

The development of Self-X capabilities within domains will be expedited by the **use of standard reference points/interfaces** and standard data models. Hierarchical and programmable closed loop automation architectures with **distributed decision-making** elements will allow for faster and more accurate automated operations. **Federation** will allow cross-domain sharing of information among decision elements that will provide better insights on the end-to-end customer experience. More work is planned within the TM Forum to identify and further develop the interface specifications at all the reference points that comprise the AN reference architecture.

Real-world CSP uses cases will make use of **multiple autonomous domains** across the operational layers. AI-enabled software and traditional software **will coexist with human operations**. The ability to have both single-domain autonomy and cross-domain collaboration will be essential to providing autonomous closed loop management of complex networks and services.

© TM Forum 2022. All Rights Reserved.

8 Administrative Appendix

8.1 References

#	Title	Organization
IG1253	Intent in Autonomous Networks	TM Forum
ENI005	Experiential Networked Intelligence System Architecture GS ENI 005 V1.1.1.	ETSI
IG1167	ODA Functional Architecture TM Forum	
IG1190	AIOps Service Management - service management operations processes to handle and govern AI-enabled software at scale	TM Forum
IG1218	Autonomous Networks Business Requirements and Framework v1.1	TM Forum
IG1230	Autonomous Networks Technical Architecture v1.1	TM Forum
IG1230A	Autonomous Networks Scenarios Realizations v1.1	TM Forum
IG1230B	Autonomous Networks Industry Standards v1.1	TM Forum
IG1260	Autonomous Networks Project Deliverable Guide	TM Forum
ZSM002	Zero-touch network and Service Management Reference Architecture. ETSI GS ZSM 002 V1.1.1	ETSI
FED	Bakker, J.H.L. Pattenier, F.J. "The layer network federation reference point-definition and implementation"	Bell Labs
GANA	Autonomic network engineering for the self-managing Future Internet (AFI); Generic Autonomic Network Architecture 2013, ETSI GS AFI 002.	ETSI

8.2 Document History

8.2.1 Version History

Version Number	Date Modified	Modified by:	Description of changes
0.1	17-May-2021	Kevin McDonnell	Initial Draft for Team Review
1.0.0	28-May-2021	Kevin McDonnell	Incorporated Team review changes
1.0.1	30-May-2022	Zheng Guangying	Incorporated Team review changes

8.2.2 Release History

Release Status	Date Modified	Created by:	Description of changes
Pre-production	28- May-2021	Kevin McDonnell	Final edits before publication
Production	26-Jul-2021	Adrienne Walcott	Updated to reflect TM Forum Approved Status
Pre-production	11-Jul-2022	Alan Pope	Final edits and re-branding prior to publication
Production	23-Sep-2022	Adrienne Walcott	Updated to reflect TM Forum Approved Status

© TM Forum 2022. All Rights Reserved.

8.3 Acknowledgements

This document was prepared by members of the TM Forum Autonomous Networks project.

8.3.1 Guide Editors and Authors

Member	Title	Company
Kevin McDonnell	Senior Director, Intelligent Automation	Huawei
Zheng Guangying	Autonomous Driving Network Standard Expert	Huawei

8.3.2 Main Contributors (Alphabetically)

Member	Title	Company
Azahar Machwe	Enterprise Architect OSS Automation	BT
Dave Milham	Chief Architect	TM Forum
Jiachen Zhang	Research Engineer	China Mobile Research Institute
Jörg Niemöller	Expert of Analytics and Customer Experience	Ericsson
Lingli Deng	Lead Researcher and Technical Manager	China Mobile Research Institute
Mohammed Fahim Momen	General Manager OSS	Robi Axiata Limited
Tayeb Ben Meriem	Senior Standardization Manager	Orange

8.3.3 Additional Inputs (Alphabetically)

Member	Title	Company
Abdul Majid Hussain	Solutions Architect	Telstra
Ali Mozaffar	Director	Clarity Global
Andy Corston-Petrie	Senior Research Manager	BT Group plc
Ankur Goyal	Lead Solutions Consultant	Infosys
Brad Peters	Architect	NBN Co Ltd
Cao Xi	Senior Researcher of AI and Intelligent Operation	China Mobile Research Institute
Dong Sun	Chief Business Strategist, Digital Transformation	Futurewei Technologies
Emmanuel A. Otchere	Chief Technical Expert VP, Standards & Industry Development	Huawei
Feng Chengcheng	Researcher	China Mobile Research Institute
Huang Zhihua	Autonomous Driving Network Engine Design Expert	Huawei
Knut Johannessen	Senior Advisor	Telenor
Liu Hongbo	Deputy General Manager, Intelligent Network Center	China Unicom
Luca Franco Varvello	ICT Senior Consultant, Senior Advisor	Huawei

© TM Forum 2022. All Rights Reserved.

Member	Title	Company
Luigi Licciardi	Consultant, Executive Advisor	Huawei
Manoj Nair	Senior Solutions Architect, CTO Office	Netcracker
Min He	Chief Architect	Futurewei Technologies
Qiao Zizhi	Senior Engineer, Intelligent Network Center	China Unicom
Rati Mehrotra	Solution Design Lead	Telstra
Razieh Mosayebi	Technical Manager	Clarity
Steve Iatropoulos	Client & Industry CTO	Microsoft
Takuya Kato	Research Engineer	KDDI Research, Inc
Thierry Reynard	OSS Consulting Manager	ETIYA
Uma Lakshman	Senior IT Consultant	Telstra
Vance Shipley	CEO	Sigscale
Vinay Devadatta	Practice Head (Innovation & Industry Relations)	Wipro Technologies
Wang Lei	Principal Engineer	Huawei
Weng Zaixin	Principal Engineer	Huawei
Xie Yuan	Systems Expert	Huawei
Yuval Stein	AVP of Technologies	TEOCO
Zou Lan	OSS Standard Prime	Huawei

8.3.4 TM Forum Staff

TM Forum	Title	Company
Aaron Boasman-Patel	Vice President, AI, Customer Experience & Data	TM Forum
Alan Pope	Collaboration Manager	TM Forum
Dave Milham	Chief Architect	TM Forum

© TM Forum 2022. All Rights Reserved.