Network Management Operations
Internet-Draft
Intended status: Informational
Expires: 31 August 2026

X. Zhao
CAICT
M. Wang
China Mobile
B. Wu
Huawei
D. Ceccarelli
Cisco
H. Zheng
Huawei
J. Zhou
ZTE

27 February 2026

AI based Network Management Agent(NMA): Concepts and Architecture
draft-zhao-nmop-network-management-agent-04

Abstract

The evolution from Level 3 (assisted automation) to Level 4 (autonomous self-optimization) in Autonomous Networks (AN) introduces requirements for Agentic capabilities, including intent-based reasoning, autonomous planning, and context-aware decision-making, which transcend the static, rule-based logic of traditional SDN Controllers. This document defines the concept of the Network Management Agent (NMA), an AI-driven entity designed to embody these cognitive functions and bridge the gap between service intent and network operations.

This document also specifies how the NMA utilizes the existing capabilities of SDN Controllers—such as topology management, telemetry, and enforcement—to achieve Autonomous L4 without duplicating policy control functions. It further details the architectural integration modes and defines the interface requirements necessary for SDN Controllers to interoperate with NMAs.

Discussion Venues

This note is to be removed before publishing as an RFC.

Discussion of this document takes place on the Network Management Operations Working Group mailing list (nmop@ietf.org), which is archived at https://mailarchive.ietf.org/arch/btrse/nmop/.

Source for this draft and an issue tracker can be found at https://datatracker.ietf.org/doc/draft-zhao-nmop-network-management-agent/.

Status of This Memo

This note is to be removed before publishing as an RFC.

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at https://datatracker.ietf.org/drafts/current/.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on 20 April 2026.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at https://datatracker.ietf.org/drafts/current/.

Copyright Notice

Table of Contents

1. Introduction

1.1. Motivation: The Gap between AN L3 and L4

The Autonomous Networks (AN) framework [TMF-IG1230] defines a series of evolution stages from Level 0 (manual) to Level 5 (fully autonomous) as listed in Appendix I. Current operator networks typically operate at Level 2 or 3, where automation is primarily policy-driven and reactive. Achieving Level 4 (L4) requires evolving from static execution to dynamic assurance.

The initial journey towards L4 could target pragmatic, high-value scenarios, such as automated Root Cause Analysis (RCA), SLA assurance, and service restoration. These use cases allow operators to deploy AI for observability and recommendation, reducing manual toil while maintaining control.

Traditional SDN Controllers excel at deterministic configuration and telemetry collection but lack the analytical depth required for these complex assurance tasks. They execute instructions but cannot autonomously diagnose the 'why' behind a failure or predict SLA violations. Introducing AI-driven logic is necessary to bridge this gap. However, decoupled AI models are insufficient. A new architectural entity— the Network Management Agent (NMA)— is needed to integrate AI-based reasoning with SDN control, starting with assurance use cases and gradually evolving towards full closed-loop autonomy.

While the key issues after the introduction of AI in network management include:

1. The application architecture and deployment methods of AI in network management are still unclear, that is in what form AI can help network management?

2. The relationship between AI and the existing network controllers is not clear.

3. New interface capability requirements after AI is introduced are not clear either.

Therefore, it is necessary to define the general architecture and application form of AI in network management.

2. NMA and SDN Controller: Roles and Collaboration

2.1. Why NMA is Required for Autonomous L4

Achieving L4 autonomy requires a cognitive loop of Intent Interpretation, Perception Analysis, and Dynamic Decision-making—capabilities that extend beyond the native design of traditional SDN Controllers:

* *Intent Translation (The "Why")*: L4 moves beyond simple API commands to handling fuzzy, high-level operational intents. Unlike SDN Controllers, which require precise, low-level technical parameters (e.g., specific bandwidth values or queue IDs), the NMA acts as an Agentic Interpreter. It automatically decomposes abstract goals (e.g., "Ensure optimal experience for VPN users") into concrete, verifiable technical specifications, handling the ambiguity and context that traditional controllers cannot resolve.

* *Perception & Contextual Analysis (The "Sense")*: L4 requires holistic observability not just raw data collection. SDN Controllers excel at gathering telemetry but lack the ability to fuse multi-dimensional data (metrics, logs, traces, alarms) to understand the "state of the network" in a service context. The NMA combines its own knowledge base and memory, using AI models to perform Root Cause Analysis (RCA), detect anomalies, and correlate events across the network to build a comprehensive operational picture.

* *Autonomous Decision & Policy Synthesis (The "Think")*: L4 demands the ability to make non-deterministic decisions in response to unforeseen scenarios. Traditional controllers operate on deterministic, reactive logic (e.g., "If X, then Y"), which cannot handle novel failures or complex optimization trade-offs. The NMA embodies the Decision function, utilizing reasoning capabilities to synthesize new strategies, weigh potential outcomes, and decide on the optimal course of action when standard procedures do not apply, and can be iteratively optimized itself.

Therefore, the NMA serves as the Autonomous Brain (Cognitive Layer) that defines what needs to happen and why, orchestrating the SDN Controllers, which act as the Execution function (Control Layer) that handle how to enforce those decisions on the network infrastructure.

2.2. Utilizing Existing SDN Controller Capabilities

To realize Autonomous L4, the NMA leverages the mature, stable functions already present in SDN Controllers rather than reinventing them. NMA is compatible with the YANG-based automation framework described in [RFC8969], and utilizes the Controller as its primary execution engine:

* *Model-Based Abstraction*: The NMA interacts with the Controller through standard YANG Service and Network Models, bridging the gap between high-level intent and concrete network resources.

* *Telemetry & State Access*: The NMA consumes real-time operational data and topology information provided by the Controller to maintain an accurate perception of the network state.

* *Policy Enforcement*: The NMA invokes the Controller’s configuration interfaces to apply changes, relying on the Controller’s built-in validation and transaction capabilities to ensure stability.

By integrating AI reasoning with this standards-based automation foundation, the NMA elevates the network from L3 (Automated Control) to L4 (Autonomous Management).

3. Terminology

3.1. Acronyms and Abbreviations

AI: Artificial Intelligence

LLM: Large Language Model

NMA: Network Management Agent, refers to AI based network management agent

3.2. Definitions

The document defines the following terms:

Network Management Agent (NMA): A network management entity built based on ML/AI and equipped with the autonomous task processing capabilities. It can automatically carry out network status perception, task intent [RFC9315]interpretation, task planning, decision-making and task execution operations based on user task intentions or preset goals, so as to achieve closed-loop processing of scenarios-oriented network management tasks. For different application scenarios, NMA can be subdivided into multiple scenario-oriented agents.

4. Reference architecture of NMA and Deployment Modes

4.1. Intelligent Network Management and Control Framework Based on NMA

[RFC8969] proposed the framework for automating service and network management with YANG. Building on the architecture proposed in [RFC8969], higher-level intelligent network management and control can be achieved by adding NMA components. Based on the Figure 3 of [RFC8969], the layered architecture of intelligent network management and control after the introduction of NMA is shown in the following figure. NMA can exist at both the Controller and Orchestrator levels; for the device layer, due to the constraints on the computing power of network elements, some end-side AI components may be added on the device side, while it is unlikely to deploy a complete NMA.

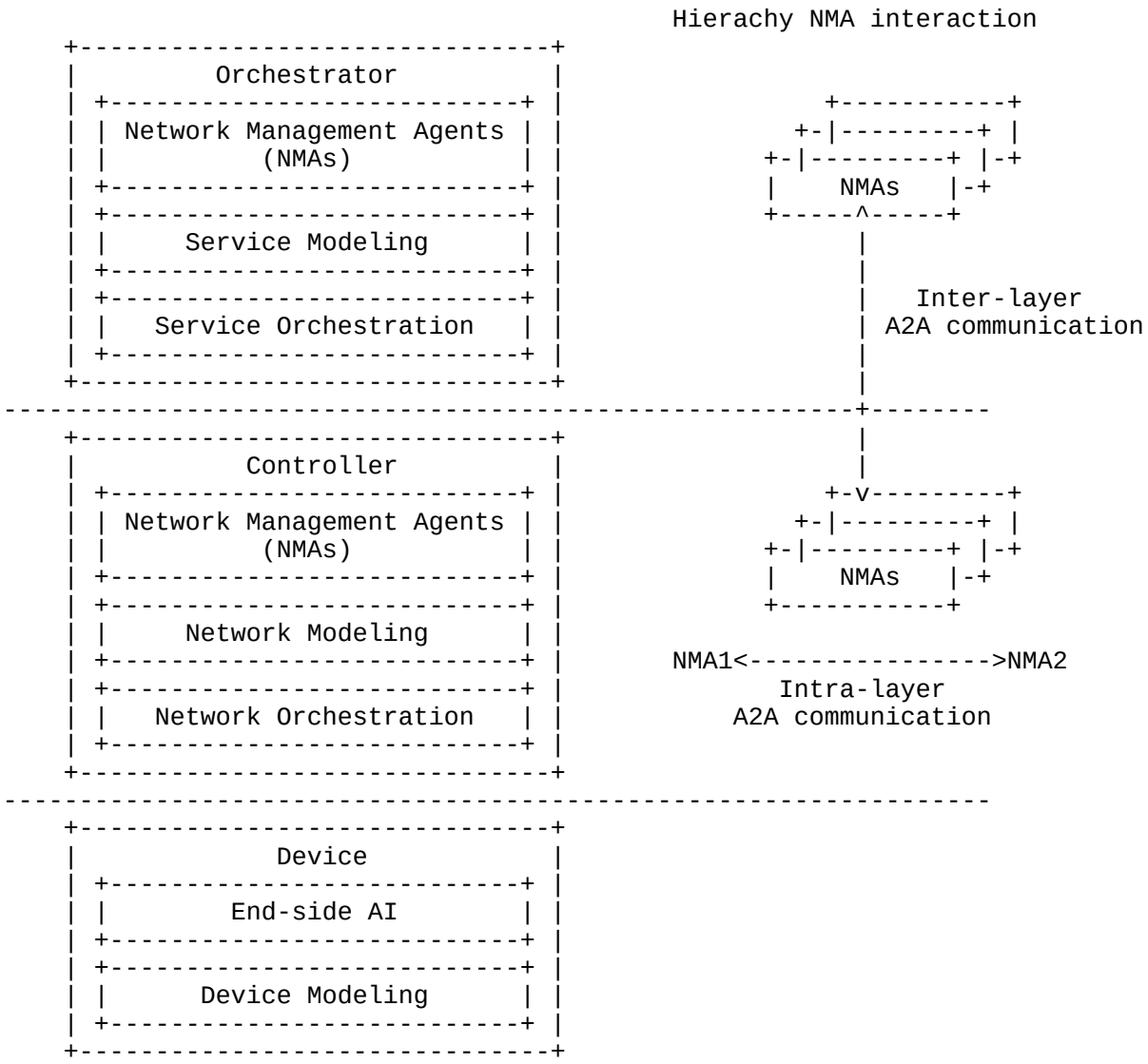

Figure 1: Enhanced intelligent network management and control framework based on NMA

Among them, there may be interaction requirements between NMAs at different layers and between different NMAs at the same layer. Cross-layer NMAs interact through inter-layer Agent-to-Agent (A2A) communication, while different NMAs within the same layer interact through intra-layer A2A communication.

This document can be regarded as an enhancement of the intelligent capabilities of [RFC8969], and subsequent discussions will mainly focus on the NMAs at the controller layer.

4.2. Deployment modes of NMA

It should be noted that although NMA is depicted inside the controller in Figure 1, in practice, NMA can also be deployed as an independent component outside the controller. This document does not impose mandatory restrictions on the deployment location of NMA. The two deployment modes can be called: Independent deployment mode and Integrated deployment mode and are shown in Figure-2, where the NMA can be part of an existing network controller, or can be an independent system deployed separately and interacting both with the controller and the network.

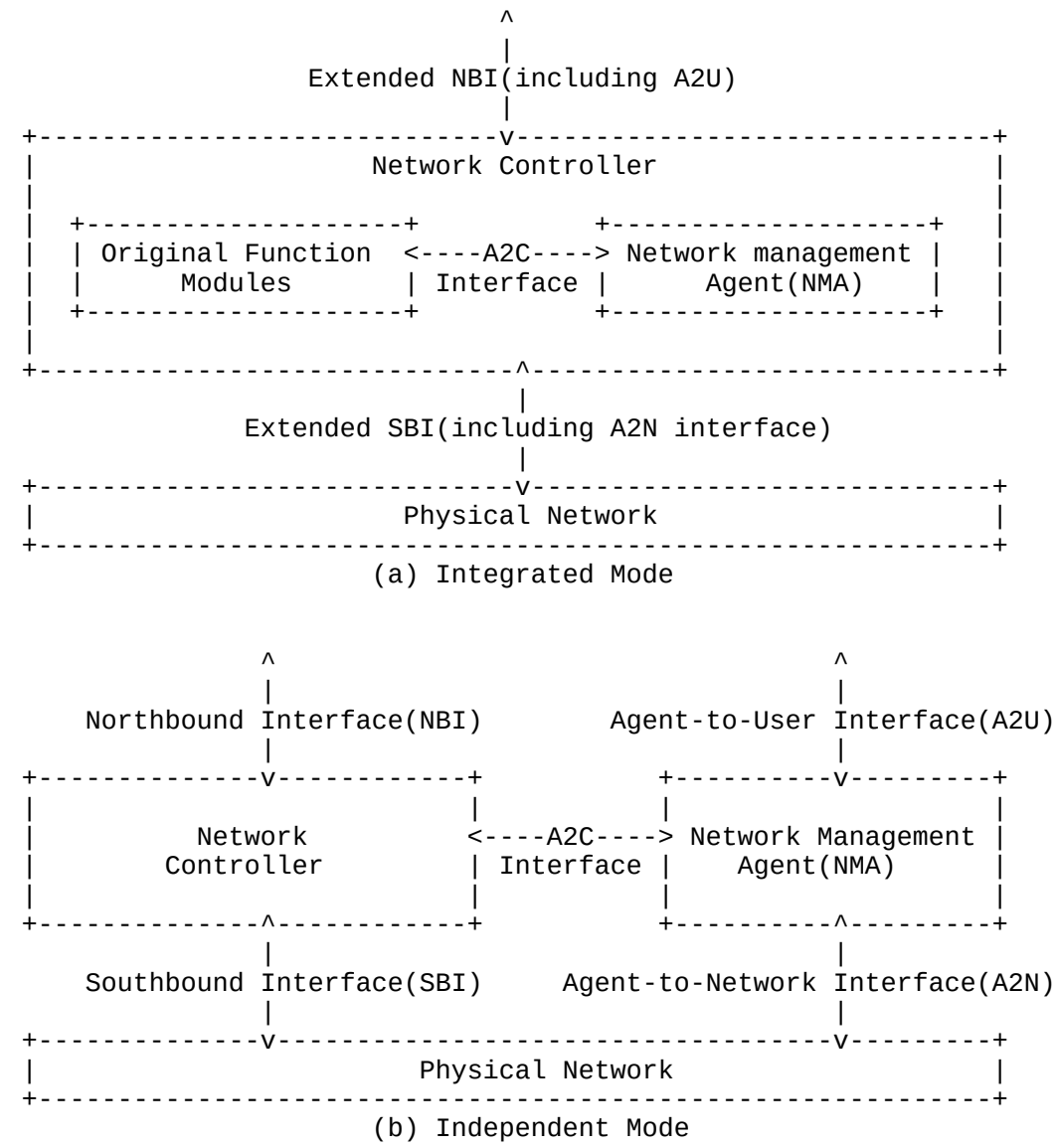

Figure 2: Deployment mode of network management agent (NMA)

Integrated deployment mode: As shown in Figure-2 (a), NMA is integrated and deployed with the original network controller, and the NMA serves as a function of the controller. NMA interacts with original function modules through internal A2C interface. The enhanced controller interacts with the underlay physical network through extended SBI satisfying the A2N interaction

requirements. The specific functional requirements and information model definition of interfaces mentioned above will be discussed in Section 4.4.

Integrated mode is targeted at network scenarios with single-vendor SDN infrastructure and high requirements for service real-time performance. This mode features deep coupling between the NMA and the SDN Controller, low decision-making and execution latency, and simple deployment and operation & maintenance (O&M), making it suitable for autonomous network management in single-vendor domains. At the same time, since it is extended on the basis of an existing SDN controller, the changes and impacts on the live network are also smaller, which facilitates the application and evolution of NMA in the live network.

Independent deployment mode: As shown in Figure 2 (b), NMA is independently deployed from the original network controller. NMA and controller are independent systems. A new east-west interface needs to be added between the NMA and the controller to achieve capability calling and result feedback operations. This interface can be called “Agent-to-Controller Interface” (A2C). In this deployment mode, controller uses southbound interface (SBI) to interact with physical network, while an Agent-to-Network interface (abbreviated as “A2N”) needs to be added between NMA and the underlying physical network.

Independent mode is applicable to multi-domain, multi-vendor heterogeneous network environments. Boasting high flexibility and scalability, this mode enables the NMA to act as a centralized cognitive brain that orchestrates multiple SDN Controllers to achieve closed-loop execution of end-to-end service intents.

While the independent deployment mode brings significant flexibility to the management of large-scale and complex networks, its decoupled architecture between the NMA and SDN Controllers introduces a series of potential issues in practical deployment, including management and O & M conflicts between the two entities, which are mainly reflected in the following aspects:

Configuration and policy conflicts: Concurrent delivery of configurations to network devices by the NMA and the Controller may result in configuration conflicts on the devices. In addition, the NMA generates dynamic control policies based on AI-driven intent reasoning and real-time network context analysis, whereas SDN Controllers maintain pre-configured static rule sets and traditional deterministic automation policies. Inconsistencies between these two types of policies may lead to policy execution failures and even service interruptions.

Inconsistent network state synchronization: The autonomous decision-making of the NMA relies on real-time and accurate network state data (telemetry, alarms, topology) provided by SDN Controllers. In the independent mode, network transmission latency and data processing delays between the NMA and Controllers may compromise the accuracy of the NMA's decision-making.

This document does not mandate a specific deployment mode for the NMA. When the independent deployment mode is adopted, it is advised to follow the principle of separation of cognitive decision-making and execution enforcement: the NMA is responsible for intent interpretation, context analysis and autonomous decision-making, while SDN Controllers retain the authorities of policy validation, resource enforcement and network state management. This ensures the consistency and effectiveness of the collaborative operation between the NMA and SDN Controllers.

4.3. Reference Functional Architecture of NMA

In order to achieve above capabilities, by referring to the common AI agent framework, this document presents the reference functional architecture of NMA as shown in Figure 3.

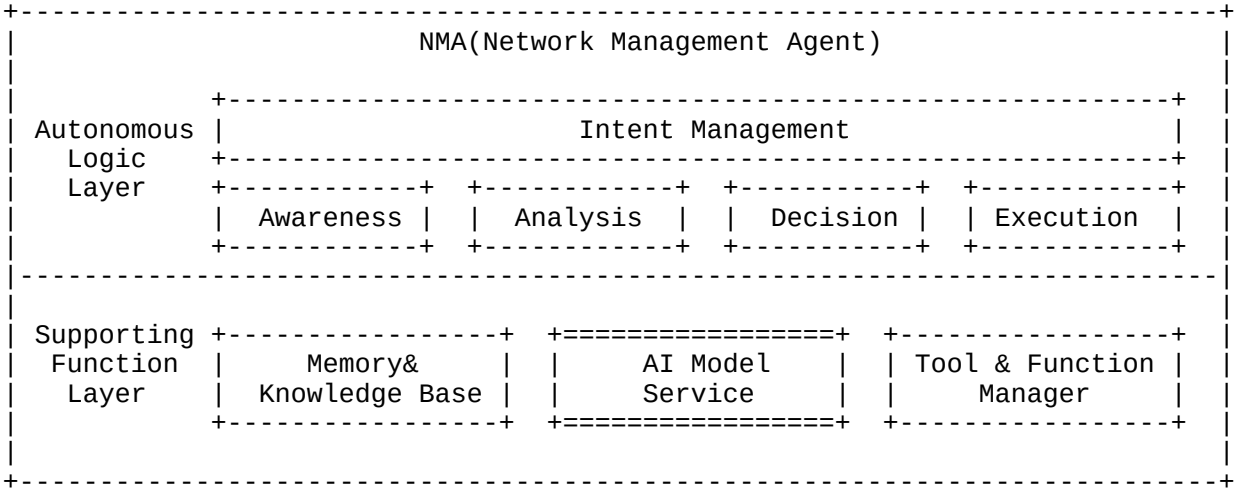

Figure 3: Reference function architecture of NMA

The NMA is structured into two primary layers: the Autonomous Logic Layer, which embodies the autonomous closed-loop from intention to perception, analysis, decision-making, and execution, and the Supporting Function Layer, which provides foundational capabilities to enable autonomous operations.

4.3.1. Autonomous Logic Layer

This layer embodies the intelligent loop of L4 autonomy, translating service goals into network actions. It mainly includes the following logical functional modules which are fully consistent with the IAADE-closed loop of autonomous network defined in TMF (see detailed in Section 8):

Intent Management: This module serves as the entry point for Intent. It is responsible for receiving high-level goals from users or orchestration systems, interpreting natural language or policy objectives, and normalizing them into structured, verifiable intents that the agent can pursue. It ensures that the autonomous operations remain aligned with service KPIs. After interpreting the target intent and reasoning through the necessary steps to achieve it, this module can orchestrate the sequence of operations required to progress toward that goal. It breaks down complex objectives into a sequence of executable sub-tasks (e.g., awareness -> analysis -> decision -> execution) and handles dynamic planning under uncertainty, ensuring that the chosen course of action aligns with the desired intent.

Awareness: This module acts as the intent-driven selective sensing hub of the NMA, responsible for orchestrating the targeted query and perception of task-relevant network data. It proactively initiates data acquisition operations across heterogeneous sources such as controllers, physical/virtual network devices, etc., with a core focus on filtering out irrelevant information to collect only the network data pertinent to the current intent. Covering critical dimensions including device operational status, link performance metrics, service traffic statistics, and configuration parameters, this module lays a precise foundational data base for the subsequent analysis, decision-making, and execution processes.

Analysis: This module serves as the intelligent analytics core, leveraging the reasoning capabilities of the AI Model Service in the Supporting Function Layer. It orchestrates advanced analytical tasks tailored to the specific task intent, including anomaly detection, root cause analysis (RCA), event correlation, and impact quantification, etc. By combining real-time perceived data with historical insights retrieved from the Memory&Knowledge Base, it transforms raw data into actionable, context-rich network

insights and diagnostic conclusions. It can clearly identify the root causes of network issues, evaluates the impact of abnormal states on service objectives, and outputs structured analytical results that directly guide the strategic decision-making.

Decision: This module functions as the strategic decision-making core of the NMA, responsible for formulating optimal and feasible operation strategies based on the analytical insights from the Analysis Management module and the constraints of the original user intent. It employs AI reasoning capabilities and draws on the Memory&Knowledge Base to evaluate multiple potential action paths, selecting the strategy that best aligns with service-level objectives and network operation rules. It decomposes complex strategic decisions into a hierarchical, ordered sequence of executable sub-tasks, defines clear trigger conditions and task dependencies for each step, and maps these sub-tasks to specific tools or functions managed by the Tool&Function Manager. This process ensures that the generated decisions are not only logically sound but also fully operationalized for subsequent execution.

Execution: This module acts as the intent-closed-loop operational execution core, tasked with translating the structured sub-tasks from the Decision Management module into concrete, reliable network operations. It orchestrates the invocation of appropriate network interfaces, management tools, and operational functions via the Tool&Function Manager, executing tasks such as configuration adjustment, fault remediation, resource scheduling, and service provisioning in a sequential and controlled manner. It real-time monitors the execution status of each sub-task, handles execution exceptions and retries according to pre-defined rules, and conducts rigorous result validation against the original user intent and decision criteria. Finally, it feeds back the execution outcomes, status, and validation results to the Memory&Knowledge Base and upper-layer modules, forming a complete closed-loop of autonomous network management driven by intent.

NMA enables the cognitive capabilities on task lifecycle management procedure described in [RFC8969].

4.3.2. Supporting Function Layer

This layer provides the foundational capabilities and resources necessary for the Autonomous logic Layer to function effectively.

Memory & Knowledge Base: This module serves as the long-term and

short-term memory of the NMA, storing historical operational data, network topology snapshots, and a comprehensive repository of expert knowledge including technical documents, troubleshooting guidelines, and past incident resolution cases, etc. It provides unified search capabilities across multi-type knowledge sources such as vector knowledge bases, system online help documentation, and operation and maintenance data logs. Based on accurate domain-specific information, this module improves the accuracy and reliability of NMA's reasoning and decision-making, enables the agent to reuse historical experience and expert logic, and ensures the consistency and effectiveness of autonomous operations.

AI Model Service: This module acts as the cognitive engine of the NMA, providing unified upward exposure of diversified AI capabilities. It supports not only Large Language Models (LLM) and other generative AI models, but also classic AI algorithms and lightweight dedicated models, enabling natural language understanding, logical inference, time-series analysis and other intelligent capabilities. It supplies the comprehensive general and domain-specific intelligence required to drive the core processes of intent management, perception and analysis, reasoning and planning, and decision and execution.

It should be noted that the AI Model Service is not limited to being deployed inside the NMA; it can also be located outside the NMA, and the NMA can invoke AI model capabilities in real time to complete relevant reasoning operations.

Tool & Function Manager: This module serves as the Gateway to Reality. It manages the connection between the NMA and external systems, primarily the SDN Controllers via the A2C (Agent-to-Controller) interface. It abstracts network functions (e.g., configuration, telemetry, simulation, etc.) as invocable "Tools." This module ensures that the decisions made by the upper layer are translated into concrete, standard-compliant network operations (e.g., YANG data manipulation).

4.4. Interface Requirements for NMA Integration

As shown in Figure 2, the interfaces related to NMA include three types:

1. *Agent-to-User interface (A2U):*the interface between the NMA and the user, where the user can be upper-layer NMA, controllers or orchestrators. This interface is used to receive call requests from users and return task processing results. It should support both structured and natural language modes. The natural language interface is mainly used for interaction with humans, while the

structured interface is used for interaction with other upper-layer systems or other Agents. The Agent-to-Agent (A2A) interface between NMAs is included in the scope of this interface. In the independent mode, this interface is a separate one provided by the NMA to the outside; in the integrated mode, it is included in the northbound interface of the controller.

Since this interface bridges the NMA with human operators or higher-level orchestrators. It must support dual-mode interaction:

Natural Language Interaction: For human operators, the interface must support conversational inputs (e.g., text) and return structured responses or execution confirmations.

Structured Intent Interface: For upper-layer orchestrators or peer agents, the interface must support structured intent definitions (e.g., based on YANG models or JSON/GNMI). It requires:

* Intent Submission: Accepting high-level goals with constraints (e.g., latency, cost).

* Status Reporting: Providing real-time feedback on intent fulfillment progress, including intermediate states (e.g., "Analyzing", "Planning", "Executing").

2. *Agent-to-Controller interface (A2C):* the interface between NMA and the controller or the original functional components of the controller. In the independent mode, this interface is an east-west interface between the controller and NMA; in the integrated mode, this interface is an internal interface of the controller and is not within the scope of this document.

3. *Agent-to-Network (A2N):*the interface between NMA and the physical network. In the independent mode, this interface is a southbound interface between the Agent and the network; in the integrated mode, it is included in the original southbound interface of the controller.

To elaborate in more detail, when NMAs are deployed in integration with the controller, as shown in Figure 4, the related interface to be extended includes:

1. *Extended SBI of the controller:* The southbound interface between the controller and devices, including the aforementioned A2N interface function. Theoretically, NMAs will not directly configure or operate devices; instead, they will call the

original functional modules of the controller for device-related configuration and management. Therefore, the need for standard extension of this interface is minimal, and it is not within the scope of this draft.

2. *Extended NBI:* The northbound interface of the controller. As a key interface for collaboration between upper and lower layer systems, this interface needs to realize functions such as capability discovery and invocation between upper and lower layer NMAs. Hence, there is a strong demand for its standardization, and it is necessary to consider the extension of the northbound interface of the controller oriented to the communication needs between NMAs. NBI must be augmented to expose the NMA's cognitive capabilities as Intent-Based RPCs. Unlike standard configuration RPCs that set specific parameters (e.g., set-bandwidth), these Intent-Based RPCs accept high-level operational goals (e.g., optimize-performance or diagnose-incident). This distinction allows upper-layer systems to invoke autonomous behaviors that require reasoning and planning—capabilities that native controller interfaces lack.

   In terms of communication channels, the orchestrator and the controller communicate one-to-one through the northbound interface. When there is a need for direct communication between NMAs in the upper-layer orchestrator and those in the lower-layer controller (A2A Communication), it will manifest as a single communication channel physically but multiple communication processes logically (i,e.including multiple A2A communication processes).

   To sum up, entended NBI should handle logical multi-process multiplexing. Current protocols typically handle a single request-response session.The extended NBI must support multiple independent A2A communication processes over a single physical channel. It must maintain strict context isolation between different agent tasks (e.g., one diagnosing a fault, another optimizing QoS) to prevent state interference—a requirement not addressed in standard HTTP/RPC models.

Besides, there are several internal interfaces within the controller, which include the interaction interfaces between NMAs within the controller and the original functional modules of the controller, as well as the interaction interfaces between multiple NMAs within the controller. Since all the above are internal implementations of the controller, there is no need for standardization.

The specific implementation methods, related protocols, etc. of each interface are to be defined subsequently in other documents.

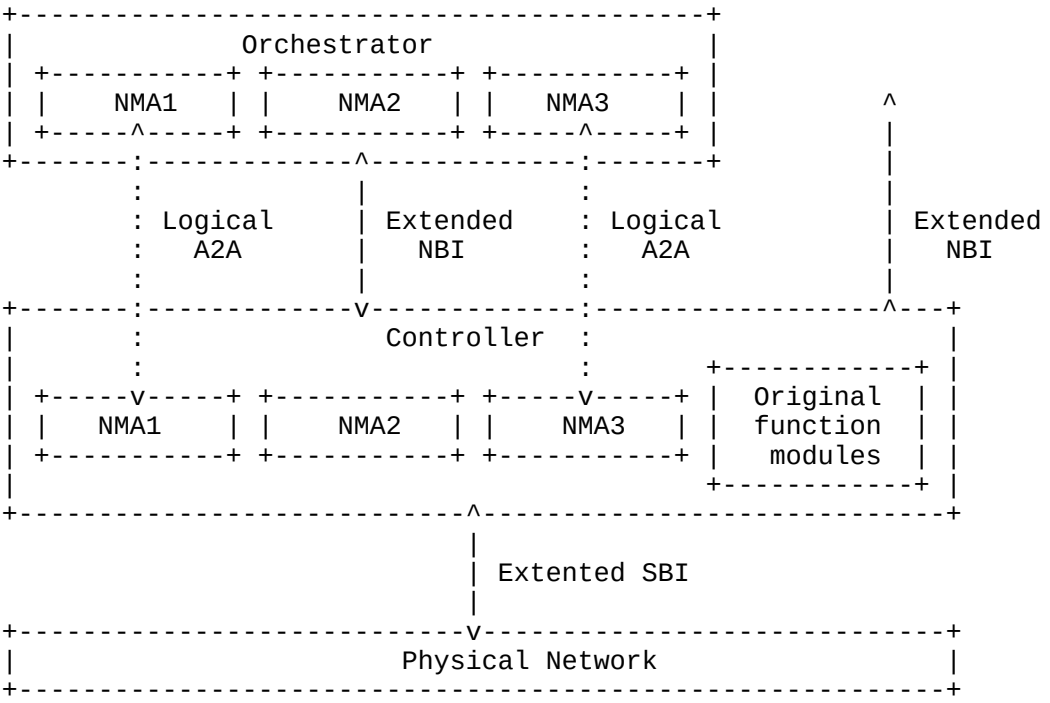

Figure 4: Interfaces to be extended on the controller

5. Operational Agent Example

To address specific operational needs, the NMA architecture supports multiple specialized agents. These agents function as modular entities, with the Intelligent Assistant Agent serving as the primary entry point for interaction, followed by specialized agents such as Fault management Agent and Optimization Agent:

* *Intelligent Assistant Agent*: Serving as the primary interface for human operators, this agent leverages LLMs to provide natural language Q&A and conversational capabilities. It enables users to perform "one-click" queries for fault descriptions or resource status. By automatically translating human intent into precise data retrieval commands, it significantly enhances the efficiency of knowledge retrieval and daily maintenance support.

* *Network Fault Management Agent*: Focused on service assurance, this agent leverages comprehensive troubleshooting guides and expert knowledge bases to support intelligent fault handling. It implements automated root cause analysis (RCA) and fault impact analysis. In addition to fault diagnosis, it orchestrates control

plane APIs to execute self-healing operations, and integrates with external work order systems to achieve closed-loop incident resolution.or self-healing actions and integrates with external work order systems to enable closed-loop incident resolution.

* *Network Optimization Agent*: Focused on performance and efficiency, this agent translates high-level optimization goals into technical constraints, such as load thresholds or routing policies. Leveraging traffic prediction models, it anticipates network congestion and proactively generates strategies for traffic engineering (e.g., pre-diversion) and dynamic energy saving. It operates in a closed-loop manner to autonomously execute decisions that maintain optimal network performance.

6. Security Considerations

Since networks are critical infrastructure, misoperations can have a significant impact on them. Therefore, NMAs shall meet the following security and reliability requirements:

1. Support multi-factor authentication mechanism for sensitive operations. For operations involving network configuration changes or those that pose significant risks to network operation security, a manual confirmation mechanism must be introduced, and multiple authentication methods such as passwords and dynamic tokens shall be used to ensure operation security.

2. Support circuit breaker mechanism. When abnormal results occur during the execution of an NMA task, it shall provide error prompts and transfer the task directly to manual control for handling.

3. Support rollback mechanism. After the execution of an NMA task is completed, it shall support operation rollback to restore the network configuration.

4. Support data security and privacy protection mechanism. It shall support the encryption of sensitive data such as network configurations and user behavior logs; support user permission division, and set differentiated data access permissions for different users.

5. Support operation permission control mechanism. For different application scenarios, the minimum permissions required to perform tasks in the scenario shall be set. For example, a fault handling NMA may query data such as topology resources and performance, but shall not have permission to perform service configuration operations.

7. IANA Considerations

This document has no requests for IANA action.

8. Appendix：Definition of L0~L5 levels in Autonomous Network

Table 1 summarizes the Autonomous Network (AN) levels defined in TM Forum IG1230 [TMF-IG1230]. It illustrates that current IETF automation frameworks, such as [RFC8969], primarily enable Level 3 (Partial Autonomy) by utilizing data models (YANG) to enforce pre-defined policies.

LEVEL	NAME	DESCRIPTION (CORE CHARACTERISTICS)	HUMAN VS. MACHINE ROLE
L0	Manual	Fully manual processes. No automation.	Human does everytding.
L1	Assisted	System provides tools (dashboards, alarms).	Human makes all decisions; tools assist.
L2	System-assisted	Automation of single tasks/ scripts witdin a specific domain.	Human initiates tasks; system executes.
L3	Partial Autonomy	Closed-loop automation based on pre-defined policies/models witdin a domain.	"Human-in-the-Loop": Humans define rules/ models and monitor; system executes and reports exceptions.
L4	High Autonomy	Cross-domain/ cross-layer context analysis and closed-loop optimization based on Intents.	"Human-on-the-Loop": Humans define high-level intents; system self-configures and heals. Human only intervenes on system failure.
L5	Full Autonomy	Self-evolving, self-optimizing, fully driverless operations.	"Human-out-of-the-Loop": System requires no human intervention for business goals.

Table 1: Autonomous Network Levels (L0-L5)

Figure 5 depicts the ‘Intent-Awareness-Analysis-Decision-Execution (IAADE)’ control loop AN architecture, highlighting the evolution from the rule-based automation of Level 3 to the intent-driven, AI-powered autonomy of Level 4, which is the focus of this document. Network Management Agent can serve as an augmentation layer, enhancing network management automation and orchestration capabilities through natural language intent translation, cross-vendor semantic bridging, and knowledge codification. In this context, Agents focus on decision support and workflow orchestration,

while critical configuration changes continue to follow manual approval and transactional execution mechanisms via existing deterministic protocols (e.g., NETCONF), striking a balance between automation efficiency and operational certainty.

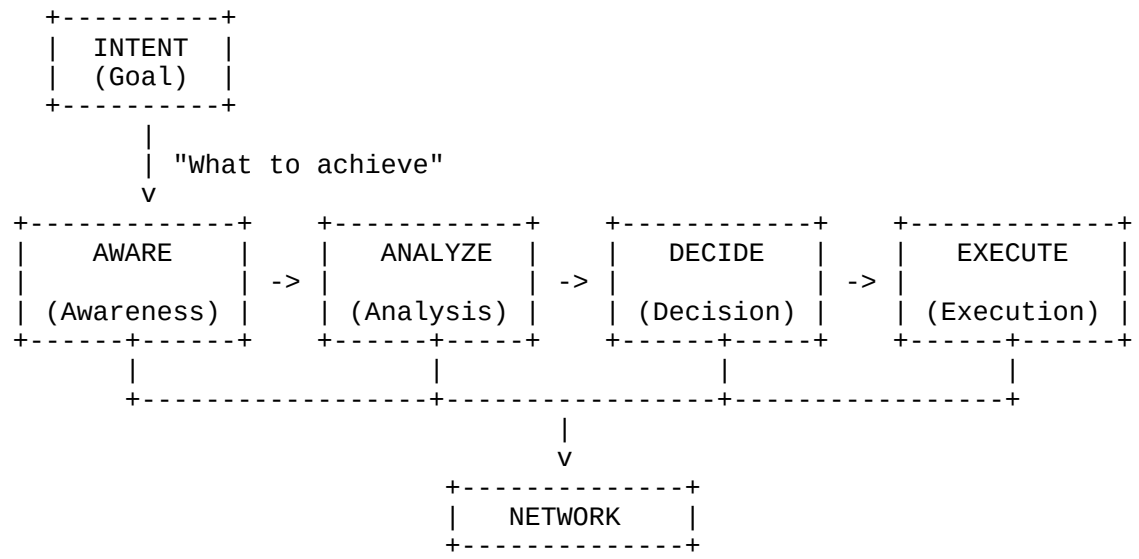

Figure 5: IAADE Control Loop for Autonomous Networks

9. References

9.1. Normative References

9.2. Informative References

[I-D.irtf-nmrg-ai-challenges]
Franois, J., Clemm, A., Papadimitriou, D., Fernandes, S., and S. Schneider, "Research Challenges in Coupling Artificial Intelligence and Network Management", Work in Progress, Internet-Draft, draft-irtf-nmrg-ai-challenges-03, 4 March 2024, <https://datatracker.ietf.org/doc/html/draft-irtf-nmrg-ai-challenges-03>.

[I-D.kdj-nmrg-ibn-usecases]
Yao, K., Chen, D., Jeong, J., Wu, Q., Yang, C., and L. Contreras, "Use Cases and Practices for Intent-Based Networking", Work in Progress, Internet-Draft, draft-kdj-nmrg-ibn-usecases-01, 8 July 2024, <https://datatracker.ietf.org/doc/html/draft-kdj-nmrg-ibn-usecases-01>.

[LLM-powered-autonomous-agents]
Weng, L., "LLM Powered Autonomous Agents", 23 June 2023.

[RFC7575] Behringer, M., Pritikin, M., Bjarnason, S., Clemm, A., Carpenter, B., Jiang, S., and L. Ciavaglia, "Autonomic Networking: Definitions and Design Goals", RFC 7575, DOI 10.17487/RFC7575, June 2015, <https://www.rfc-editor.org/rfc/rfc7575>.

[RFC7576] Jiang, S., Carpenter, B., and M. Behringer, "General Gap Analysis for Autonomic Networking", RFC 7576, DOI 10.17487/RFC7576, June 2015, <https://www.rfc-editor.org/rfc/rfc7576>.

[RFC8969] Wu, Q., Boucadair, M., Lopez, D., Xie, C., and L. Geng, "A Framework for Automating Service and Network Management with YANG", RFC 8969, DOI 10.17487/RFC8969, January 2021, <https://www.rfc-editor.org/rfc/rfc8969>.

[RFC9315] Clemm, A., Ciavaglia, L., Granville, L. Z., and J. Tantsura, "Intent-Based Networking - Concepts and Definitions", RFC 9315, DOI 10.17487/RFC9315, October 2022, <https://www.rfc-editor.org/rfc/rfc9315>.

[TMF-AN-journey-guide]
Tansuthepverawongse, Boonchoung., "AN Journey Guide Autonomous Networks L4 industry blueprint-high-value scenarios", June 2024.

[TMF-IG1230]
McDonnell, K., Machwe, A., Milham, D., O' Sullivan, J., Clemm, A., and J. Niemller, "Autonomous Networks Technical Architecture", TMF IG1230, December 2022.

Authors' Addresses

Xing Zhao
CAICT
Beijing
China
Email: zhaoxing@caict.ac.cn

Minxue Wang
China Mobile
Beijing
China
Email: wangminxue@chinamobile.com

Bo Wu
Huawei
China
Email: lana.wubo@huawei.com

Daniele Ceccarelli
Cisco
Email: dceccare@cisco.com

Haomian Zheng
Huawei
China
Email: zhenghaomian@huawei.com

Jin Zhou
ZTE
China
Email: zhou.jin6@zte.com.cn