Network Management
Internet-Draft
Intended status: Informational
Expires: 2 January 2026

Q. Wu
Huawei
C. Zhou
China Mobile
L. M. Contreras
Telefonica
S. Han
China Unicom
L. Tailhardat
Orange Research
Y. Hong
Daejeon University

1 July 2025

Network Digital Twin based Architecture for AI driven Network Operations
draft-wmz-nmrg-agent-ndt-arch-00

Abstract

A Network Digital Twin (NDT) provides a network emulation tool usable for different purposes such as scenario planning, impact analysis, and change management. Integrating a Network Digital Twin into network management together with AI, it allows the network management activities to take user intent or service requirements as input, automatically assess, model, and refine optimization strategies under realistic conditions but in a risk-free environment. Such environment that operates to meet these types of requirements is said to have AI driven Network Operations.

AI driven Network Operations brings together existing technologies such as Network Digital Twin and AI which may be seen as the use of a toolbox of existing components enhanced with a few new elements.

This document describes an architecture for AI driven network operations and shows how these components work together. It provides a cookbook of existing technologies to satisfy the architecture and realize intent-based networking to meet the needs of the network service.

Discussion Venues

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

Discussion of this document takes place on the Network Management mailing list (nmrg@irtf.org), which is archived at https://mailarchive.ietf.org/arch/browse/nmrg.

Source for this draft and an issue tracker can be found at https://github.com/QiufangMa/Agent-architecture.

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/.

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 2 January 2026.

Copyright Notice

Table of Contents

1. Introduction

The rapid expansion of network scale and the increasing demands on these networks necessitate of continuous network reconfiguration to better adapt to ever-changing service requirements.

Since network changes are directly related to service operations, any

successful change needs to not only ensure that new services are provisioned smoothly, but also that existing services are not affected and that no problems are introduced with the new configurations. Network operators are, therefore, increasingly cautious about making network changes, given that they need to review the solution design as well as evaluate all change impacts, before making any change. Then, after the change, they need to perform dialling tests, monitor traffic, and manually check table entries.

The Network Digital Twin (NDT) [I-D.irtf-nmrg-network-digital-twin-arch] has been proposed as a mean to provide a network emulation tool for scenario planning, impact analysis, and change management. Integrating a Network Digital Twin into network management together with AI, it allows network management activities to dynamically adapt to customer needs, network changes, as well as to automatically assess, model, and refine optimization strategies under realistic conditions but in a risk-free environment. An environment that operates to meet these types of requirements is said to have service-oriented AI for network operations.

Service-oriented AI for network operations provide the following capabilities to applications by coordinating the components that operate and manage the network:

* Service intent and service assurance work together to ensure that the network change or network optimization aligns with business goals and that the services provided meet the agreed-upon Service Level Agreements (SLAs).

* Provide Network capacity planning and ensure that the network has sufficient capacity , resources, and infrastructure to meet current and future demands.

* Provide simulation on fault scenarios, formulate recovery plans, and verify whether the plans are applicable and effective so that the service will not be affected during disaster recovery drill.

* Support Fault and risk detection and provide network health check and network risk check.

* Model the network configuration change and use a virtual topology model to test network changes and assess the effect of the network configuration changes on the network.

* Model the protocol operations and interactions among devices in the network and simulate specific networking protocols such as IS-IS, OSPF, BGP, SR, etc to understand how they perform under different conditions.

* Model traffic flow across the network, including traffic generation, flow control, routing, and congestion control and evaluate traffic’s impact on network performance.

* Support generation of rectification solutions for potential network risks and provide verification on the repair solution in seconds, including loop, address conflict, and security policy conflict.

This document describes an architecture for service-oriented AI for network operations, showing how these components work together. It provides a cookbook of existing technologies to satisfy the architecture and realize intent-based networking to meet the needs of applications.

2. Conventions and Definitions

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

The document uses the following definitions and acronyms defined in [I-D.irtf-nmrg-network-digital-twin-arch]:

* Network Digital Twin (NDT)

* Artificial Intelligence (AI)

The following acronyms are used throughout this document:

* Generative Artificial Intelligence (Gen-AI)

* Large Language Model (LLM)

* Retrieval-Augmented Generation (RAG)

Besides, this document defines the following terminology:

Network AI Agent: AI Agent is an autonomous system or entity with awareness of its environment, capable of conducting analysis, making decisions, and executing actions with specific intent based on its knowledge representation to achieve a set of service goals [TMF-1251D].

3. Introduction of Concepts

3.1. Generative AI and AI Agent

The integration of AI into network operations has marked a significant leap forward in the pursuit of network automation and intelligence, while generative AI further enhances the role of AI driven network operations and management. Generative AI is a subfield of AI that uses generative models such as Large Language Models (LLMs) to generate new and original content such as text, images, videos, or other forms of data with the capability to adapt and make decisions to achieve specific goals.

An AI agent refers to a system or program that Large Language Models (LLM)s to interact with humans (or other AI Agents) for purposes of performing tasks [I-D.rosenberg-ai-protocols]. In the context of network operations and management, Network AI agents are increasingly being designed to interact with physical world and act upon it based on tools [Google-Agents-Whitepaper] and perform network management tasks such as understanding user intent, generating network configurations, diagnosing and resolving network incidents [I-D.ietf-nmop-network-incident-yang]. Meanwhile, other SDOs also try to define terms related to Network AI agent in the context of network operations and management, e.g., TM Forum defines Autonomous Agent in [TMF-1251D] as one of AN (Autonomous Network) Terminologies.

3.2. Network Digital Twin

The Network Digital Twin is a digital representation that is used in the context of network. The concept and architecture of the Network Digital Twin are specified in [I-D.irtf-nmrg-network-digital-twin-arch]. Three core functional components which includes Data Repository component, a Service Mapping Models component, and an NDT Management component are introduced to characterize the Network Digital Twin and its reference architecture.

The Network Digital Twin is widely recognized to be useful as an advanced platform for network emulation, serving as a tool for scenario planning, impact analysis, and change management. By delivering applications requests to the Network Digital Twin through standardized interfaces (see Section 9.4 of [I-D.irtf-nmrg-network-digital-twin-arch]), the Network Digital Twin exposes the various capabilities to network applications.

4. Characteristics of AI driven Network Operations

AIOPS was first defined by Gartner in 2016, combining "artificial intelligence" and "IT operations" to describe the application of AI and machine learning to enhance IT operations. However there is no unified definition for characteristic of "AI driven network operations" within the networking industry. Referring to the characteristics of AIOPS in IT field and the characteristics of networking itself, this document introduces six key elements (i.e., awareness, decision, analysis, execution, intent and knowledge) to characterize the AI driven network operation and its use, as shown in Figure 1. They together form a close-loop of network operation and management.

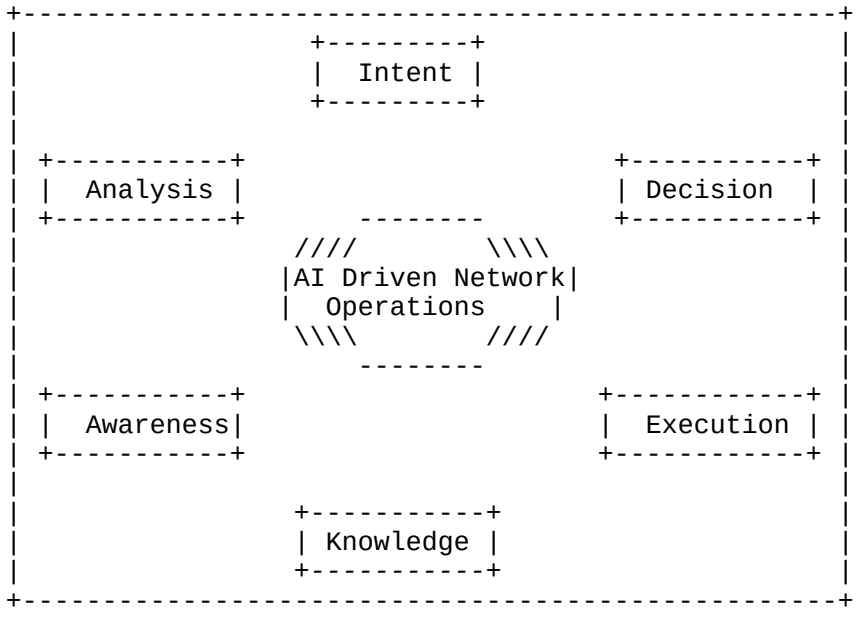


Figure 1: Six Key Elements to Characterize AI driven network operation

* Intent: Intent is defined as a set of operational goals and outcomes defined in a declarative manner without specifying how to achieve or implement them in [RFC9315]. The Network AI Agent must accurately interpret and understand the user’s high-level business or operational objectives, this involves translating declarative requirements into specific network instructions, e.g., configurations.

* Knowledge: The Network AI agent relies on a knowledge base that includes network policies, historical data, expert experience, and best practices in product manual. The knowledge is used to inform its analysis, decision-making, and execution processes. Over time, the Network AI agent can expand its knowledge through machine learning, incorporating new data and experiences to improve its performance. For example, it learns which configurations are optimal for specific scenarios or how to respond most effectively to particular types of network incidents [I-D.ietf-nmop-network-incident-yang].

* Analysis: The Network AI agent continuously analyzes vast amounts of network data from various sources, including network telemetry [RFC9232] and external feeds, and identify the gap between user intent and the existing network status. By integrating Network digital twin [I-D.irtf-nmrg-network-digital-twin-arch] with Network AI agent and leveraging machine learning and other data analytics techniques, it also identifies network fault, problem, incident, anomaly and perform data driven intelligent analysis such as service impact analysis, and so on. Their distinction is further discussed in [I-D.ietf-nmop-terminology].

* Decision: Based on the intent and network analysis, AI makes informed decisions. By integrating network digital twin [I-D.irtf-nmrg-network-digital-twin-arch] and AI, the intelligence decisions making can be realized. These decisions could involve dynamically adjusting network parameters, e.g., rerouting traffic to avoid congestion. The decision-making process is driven by predefined policies, real-time data analysis, and AI models (e.g., LLMs) that enable the Network AI agent to choose the best course of action to meet the specified intent. Network AI agent may also verify the correctness of the decision outcome by performing some network simulation or validation process.

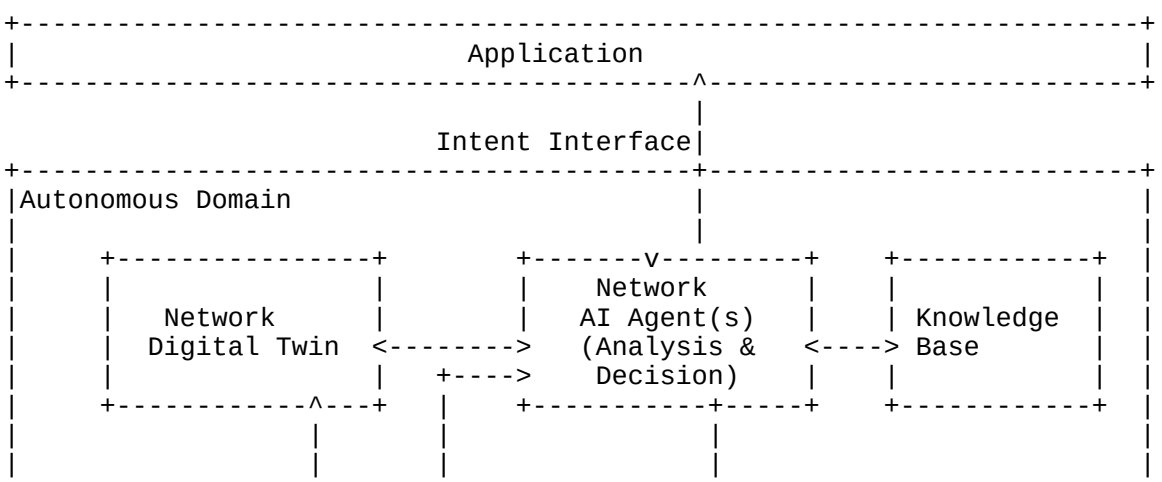
* Awareness: Awareness is achieved through real-time monitoring and data collection. The Network AI agent maintains a comprehensive visibility of the network, enabling it to make context-aware decisions. Network operators can also use the awareness understand the exact cause of specific network issues and achieve closed-loop decision-making.

* Execution: Once a decision is made, the Network AI agent executes the necessary actions to implement it. This could involve, e.g., sending configuration to network controllers or network devices through NETCONF/RESTCONF protocols. The execution is carried out in a controlled and precise manner to ensure that the network behaves as intended without causing disruptions. The Network AI agent also verifies that the executed actions have the desired effect and makes the proper adjustments if needed.

5. Architecture Design

5.1. Overall Architecture

Figure 2 provides the overall architecture for integrating Network Digital Twin and Network AI Agent System.



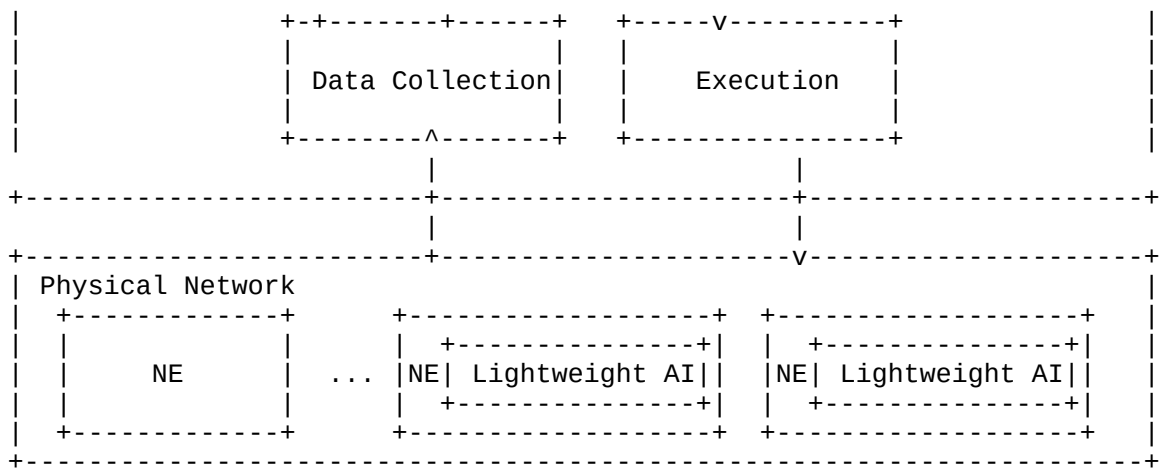


Figure 2: An Architecture for Integrating Network AI Agent with Network Digital Twin

5.2. Functional Components

5.2.1. Application

One of example application is multi-domain orchestrator. Multi-domain orchestrator serves as the top-level coordinator and manages the interactions across different autonomous domains. Multi-domain orchestrator may invoke Network Digital Twin to perform functions such as analyze, diagnose, optimize, control, and emulate as per [I-D.irtf-nmrg-network-digital-twin-arch]. It also provide means to convey user intent to each autonomous domain through a user-facing Graphical User Interface (GUI) or machine-to-machine North Bound Interface (NBI).

5.2.2. Autonomous Domain

An autonomous domain is a self-governing unit that achieves NDT and AI driven network autonomous management.

5.2.2.1. Network Digital Twin

A Network Digital Twin provides an enhanced and optimized solution in the face of increasing network and business types, scale, and complexity. It simulates the behavior, performance, and characteristics of the actual network, which could help in validation and testing scenarios, analyzing and predicting network behavior without affecting the real physical network.

As described in Section 7 of [I-D.irtf-nmrg-network-digital-twin-arch], the core functional components of an Network Digital Twin includes Data Repository, Service Mapping Models, and a Network Digital Twin Management component. The Network Digital Twin collects the real-time operational and instrumentation data from network through the appropriate real network-facing input interfaces, and it delivers NDT services through appropriate application-facing output interfaces, which is the interfaces to Network AI Agent(s) in Figure 2.

5.2.2.2. Network AI Agent(s)

Network AI Agent(s) act(s) as the smart brain of the Autonomous Domain, which is responsible for conducting AI-based analysis and making decisions regarding network operations. It leverages the inference of LLM, the simulation of Network Digital Twin, and the contextual and domain-specific knowledge provided by Knowledge Base to accomplish specific network operation task.

Agents could be scenario-oriented and classified according to the function they perform. It is also possible for multiple Agents to collaborate in some scenarios. Multi-Agents management is needed to handle the agent instance lifecycle (e.g., deployment, update, and retirement of Network AI Agent), Agent registration, Agent discovery, and so on. Some ongoing efforts (MCP [MCP], A2A [A2A]) in the industry may help with multi-agents coordination.

5.2.2.3. Knowledge Base

The Knowledge Base serves as a crucial repository of information within the architecture. It enables the injection of expert knowledge and provides the necessary knowledge and memory that helps AI Agent(s) make more accurate and context-aware decisions. It also helps mitigate the hallucination problems that can arise in large-scale models, which enhances the accuracy of task execution. Additionally, the Knowledge Base plays a key role in providing the data needed for techniques like Retrieval-Augmented Generation (RAG), which further boosts the system’s ability to generate reliable and relevant outputs.

5.2.2.4. Data Collection

Data Collection component is responsible for gathering data from the physical network through various different tools and methods (e.g., IPFIX [RFC7011], YANG-push [RFC8639],[RFC8641], BMP [RFC7854]). It collects various types of network data including configuration data, operational data, network topology, routing data, logs, and trace on management plane, control plane, and forwarding plane as needed. The collected data is fed into the Network Digital Twin and Network AI Agent(s) to provide with up-to-date information about the current state of the physical network.

5.2.2.5. Execution

Once network decisions are made and confirmed, the Execution component performs specific actions to the physical network, e.g., modify specific configuration on network controllers or network devices through protocols like NETCONF [RFC6241] and RESTCONF [RFC8040]. It is the component that makes the planned control and management changes a reality in the real physical network.

5.2.3. Physical Network

This is the actual hardware and infrastructure that makes up the network, which includes a set of network devices and wiring. In a physical network, Network Elements (NEs) with Lightweight AI [I-D.irtf-nmrg-ai-challenges] may also achieve some local close loop without relying on external AI or human intervention. It is also possible for the Leightweight AI to coordinate with AI Agent(s) to enhance the automation and efficiency of network operations. The Network Leightweight AI models could be trained, validated, deployed, and executed on Network Elements, and further refined (e.g., model re-training) through monitoring and continuous optimization based on feedback from LLM.

5.3. Architecture Requirements

There are a couple of key requirements of the architecture to integrate Network Digital Twin with service-oriented AI which are crucial in ensuring the proposed architecture can handle the complex and dynamic network scenarios for network operations and management.

5.3.1. Human-in-the-loop

This allows human experts to provide guidance and make critical decisions when necessary. By involving human in the process, the architecture can leverage their insights and experience, ensuring AI actions align with organizational goals.

Human-in-the-loop is also helpful to provide a safeguard for complex or sensitive decisions, where human judgement is essential to avoid potential errors or ethical dilemmas.

5.3.2. Interoperability via Open Standards

Standardized protocols and interfaces facilitate smooth communication and ensures different systems and devices from various vendors can work together seamlessly. The interfaces between Network AI Agent(s) and Network Digital Twin are the application-facing interfaces as defined in [I-D.irtf-nmrg-network-digital-twin-arch]. There are some ongoing efforts that are working on the standardization of Network AI Agent communication [I-D.rosenberg-ai-protocols].

5.3.3. Feedback-driven Improvement

The architecture should incorporate mechanism for continuous improvement based on feedback. This includes collecting data on AI decisions, network performance, and user feedback to identify areas for enhancement. By analyzing the feedback, the system can adapt and optimize its operations over time, leading to better performance and more accurate decision-making. For example, if a Network AI Agent fails to accurately identify the exact cause of a network incident, the relevant records can be submitted as negative samples to the LLM which provides inference services, this allows the LLM to be trained on these negative samples for optimization. Feedback-driven improvement also enables the architecture to evolve with changing network conditions and requirements.

5.3.4. Scalability and Flexibility

The architecture must be designed to scale efficiently to accommodate growing network demands and increasing data volumes. It should also be flexible enough to adapt to new network scenarios and operational requirements. This means that components should be modular, allowing for easy addition or modification of functionality without disrupting the entire system. Scalability and flexibility ensure that the architecture remains effective and relevant in the face of evolving network challenges.

5.4. Collaboration between small AI model and large AI model

The architecture must be designed to support collaboration between small AI model and large AI model.

In the past, we only support AI and machine learning technologies at the network level, e.g., we can use collected various different network data to provide network analysis and generate network insight.

With more intelligence introduced into the network element, more GPU/NPU resource can be allocated for AI inference, this make collaboration between large AI model And small AI model become possible. On one hand, we can use accumulated field engineering expertise to train large AI model into one foundation model for fault management AI agent, On the other hand, we can deploy small AI model, leverage hardware resource or chipset resource in the intelligent network element to collect more fine granularity data or provide local processing for Collected data and summary report generation, Trend prediction, etc. With collaboration between large AI model and small AI model, we can allow Network AI Agent within the Network

controller interact with network element and has more quick response to network change.

6. AI Driven Network Operation: A collection of Use Cases

Network AI Agent could help in the following phases which are usually mentioned in network management:

* Network Planning and Design: includes the understanding of user intent, generation of solutions, and simulation for decision-making.

* Service Deployment: includes the construction of the physical network, as well as intent understanding, pre-deployment simulation, automated configuration, post-deployment validation, and other capabilities to enhance the efficiency and accuracy of network configuration for service deployment.

* Network Monitoring and Troubleshooting: includes intent monitoring, issues identification, solution generation, evaluation and decision-making, solution implementation, and service validation.

* Network Change and Optimization: involves the design, evaluation, decision-making, implementation, and validation of network configuration changes or optimizations to improve network operation efficiency.

In all phases and use cases, after the Agent performs specific action, it always continuously monitors the network by data collection. Based on the result of network running analysis and user explicit feedback, it may adjust and optimize the management strategy if necessary.

6.1. Network Configuration Change

Network configuration changes are needed in scenarios such as optimizing network or service performance, provisioning new network services, or resolving network incidents/faults. Network configuration change leveraging AI and Network Digital Twin may experience the following typical steps:

Step 1: The network operator inputs the intent of network configuration change into the Network AI Agent using natural language. The network operator may simply explain the objectives and requirements of the changes.

Step 2: Network AI Agent first verifies the identity of the user requesting the change and checks the user’s permissions to make certain types of network changes against predefined rules or policies. It then understands and parses the initial intent of the request, and leverages the powerful knowledge and reasoning capabilities of LLM to generate initial suggestions for specific network configuration update, which may include multiple possible network configuration change plans if possible.

Step 3: Network AI Agent communicates with the Network Digital Twin to validate the suggested configuration change, including the syntax and semantics of the configuration, verification of effected application and resources. The network digital Twin may generate a report indicating the validation result, and suggested configuration fix when the validation fails after network simulation leveraging the current physical network operational state.

Step 4: Network AI Agent may generate a configuration change plan

and submit to the network operator for approval. Based on the feedback from the operator, Network AI Agent then further decides whether to optimize the change plan or deliver the plan to the Execution component to conduct the physical network configuration change.

6.2. Network Troubleshooting

Network AI Agent could assist in network troubleshooting in the following significant aspects:

* Fault Identification: Network AI Agent continuously monitors and aggregates data from various sources, the comprehensive data collection provides a holistic view of the network operational state. By analyzing the real-time data, Network AI Agent could detect network anomalies swiftly, which enables the prompt identification of potential issues before they escalate into major faults, minimizing downtime or service disruptions. In some cases, the Leightweight AI located in the Network Element may handle some simple fault identification tasks (e.g., optical module fault automatic identification) to enhance the awareness, while the Network AI Agent and LLM could leverage their powerful processing capabilities to analyze the time-domain data collected from the optical module.

* Fault Diagnosis: Once a fault is identified, Network AI Agent delves into diagnosing the exact cause, it may also invoke some existing operations such as "incident-diagnose" RPC defined in [I-D.ietf-nmop-network-incident-yang]. By correlating symptoms and/or applying AI models trained on historical data, it can narrow down the potential causes and pinpoint the exact cause, which accelerates the diagnosis process and reduces the time needed to address the issue.

* Fault Repair: After diagnosing the fault, Network AI Agent can generate targeted repair solutions. These solutions range from specific configuration adjustments to more complex fixes (e.g., hardware replacement). Network AI Agent would also communicate with the Network Digital Twin to simulate the proposed repair solutions and get feedback from the Network Digital Twin. In advanced setups, Network AI Agent may automatically execute these repairs, ensuring quick restoration of normal operations and enhancing the overall reliability and efficiency of network management. But it may also first present the fault details and repair advice to the network operator for review, and proceed to carry out the repair task once it is confirmed.

* Fault Prediction As an advanced enhancement of fault management capabilities, fault prediction aims to reduce network risks through proactive management that prevents problems before they occur. Before a fault actually occurs, the NDT constructs a dynamic simulation model by collecting real-time multi-dimensional operational state data, including network topology, traffic load, and device performance indicators. Based on the network data, AI Agent uses large models and machine learning algorithms (such as time-series prediction models and anomaly detection models) to reason and analyze potential faults—for example, predicting the risk of physical link interruption based on optical cable signal attenuation data. Furthermore, the AI Agent generates recommended operations to avoid faults and validates them through simulation in the NDT, thereby achieving predictive maintenance of the network.

6.3. Network Optimization

Network optimization is often introduced due to the Network AI

Agent’s awareness of some potential network faults or anomalies through continuously monitoring of network operational state, e.g., AI models may predicts network congestion by analyzing historical and real-time network traffic data. It may also be triggered by the network operator actively inputting the network optimization intent.

Based on the analysis of network data and user’s intent (if any), AI Agent proposes network optimization strategies. For instance, once the network congestion sometime in the future is predicted, it may proactively optimize the network configuration, or suggest scaling up to meet specific demands.

Before the network optimization is conducted, Network AI Agent implements and evaluates the optimization solution using the Network Digital Twin. This may need repeated trials and validations based on specific evaluation criteria, before the optimal strategy could be selected. Network AI Agent may also first present the suggested network optimization solution to the network operator for review, and apply it to the physical network after obtaining approval from the network operator.

6.4. Network level Energy Efficiency Management

Network level Energy Efficiency refer to a set of processes used to discover a inventory of capabilities, use specific metrics to monitor and assess energy consumption of the network , operate, and control the use of available energy in an optimized manner while achieving the network’ s functional and performance requirements by improving overall network utilization.

Network level Energy Efficiency allows network operators not only see real time energy consumption in the network devices of large scale network through interaction with the GREEN Network AI Agent, but also allow them see

o which network devices enable energy saving, which devices not,which are legacy ones,

o The total energy consumption changing trend over the time of the day, for all network devices,

o Energy efficiency changing trend over the time of the day for the whole network.

On the other hand, With the better observability to energy consumption statistics data and energy efficiency statistics data, the Network AI Agent can know which part of the network need to be adjusted or optimized based on network status change.

6.5. Network Security Drills

The AI Agent can help construct a dynamic attack-defense verification system in network security drills through NDT and AI reasoning capabilities. It uses generative AI to automatically generate diversified attack paths, models network topologies with graph neural networks, covers attack stages such as reconnaissance and penetration, and dynamically adjusts strategies via reinforcement learning to simulate the adaptive characteristics of network attacks. The virtual range built based on the NDT can 1:1 map the production environment, supporting simulations of composite scenarios like ransomware chain attacks and supply chain attacks—such as simulating the entire process of Conti virus laterally penetrating to domain controllers through weak passwords.

During drills, the AI Agent automatically deploys virtual environments with vulnerabilities, collects defense response data in

real time through NDT, and generates attack path heatmaps and repair suggestions. This capability can further verify emergency response processes, inject real-time threat intelligence to dynamically update drill scenarios, and simulate end-to-end automated deployment, vulnerability injection, and real-time analysis of security drills, enhancing the proactive verification ability of defense systems against real-world threats.

7. Challenges of Integrating Service-oriented AI into Network Management

In addition to the research challenges in coupling AI and network management specified in [I-D.irtf-nmrg-ai-challenges], this document also identifies some challenges that need to be considered when integrating service-oriented AI into network management.

7.1. Hallucination

Hallucination refers to the generation of AI responses that are incorrect, irrelevant, or even nonsensical in relation to the input or context provided. Although Gen-AI can produce seemingly impressive results at first glance, there’s a risk of them being completely wrong at times. These hallucinations can lead to incorrect decisions and actions in network management. For example, if the AI generates inaccurate network configurations or diagnoses faults incorrectly, it may cause network disruptions or security vulnerabilities. The challenge lies in identifying and correcting these hallucinations to ensure the reliability of AI-driven network management actions.

7.2. Security

Integrating AI into network management introduces new security challenges. Large volumes of network data needs to be accessed to learn network behaviors and make accurate decisions. Protecting sensitive network data and ensuring the integrity of AI-generated decisions are crucial. Besides, AI systems can become targets for attacks aimed at compromising network security. For instance, malicious actors could attempt to manipulate AI models to make them generate harmful network configurations or to disclose confidential network information. Additionally, the integration of AI Agents from different vendors may create new vulnerabilities that need to be addressed, e.g., lack of effective authentication and authorization among different Agents. In summary, ensuring robust security measures throughout the entire AI-based network management architecture is essential to prevent unauthorized access and maintain the security of the network infrastructure.

7.3. Data Quality and Consistency

The performance of AI models heavily relies on the quality and consistency of the data they’re trained on. In network management area, data sources can be diverse and heterogeneous, leading to potential issues such as data inconsistencies, missing, or outdated data. Poor-quality data may result in inaccurate AI predictions and decisions. For example, if incorrect or outdated network configuration data is provided, the model may provide incorrect repair advice when diagnosing network incidents or faults, it may suggest checking an non-existing interface. Ensuring that data is properly cleaned, validated, and maintained is a significant challenge in providing reliable inputs for AI-driven network management.

7.4. Interpretability and Explainability

AI-generated decisions can sometimes be difficult to interpret and

explain, as the AI model structure and the parameter settings make it hard to track its internal decision-making logic. Network operators need to understand the reasoning behind AI-driven decisions to trust and effectively utilize them. For example, if an AI system recommends a particular configuration change to optimize the network performance, operators may wonder why that specific change is being suggested. The lack of interpretability can hinder the adoption of AI Driven Network Management and make it challenging to identify potential issues with AI-generated recommendations.

7.5. Fast Decision-making

In network operation and maintenance scenarios with high real-time requirements, such as scheduling strategy optimization and critical fault repair, the rapid generation of network optimization decisions is crucial. However, AI Agents based on large models adopt a "Token-based" generation and reasoning approach, which is limited by computing power and algorithms, resulting in generally slow reasoning speeds. In addition, the simulation and verification process of Network Digital Twin (NDT) further increases decision latency, which leads to long end-to-end decision-making time in complex scenarios and is difficult to meet the real-time requirements of services. To improve decision efficiency, continuous efforts are needed in lightweight NDT modeling algorithms, optimizing large model reasoning frameworks (such as quantization technology and parallel computing), and deploying high-performance AI acceleration hardware.

8. Security Considerations

TODO Security

9. IANA Considerations

This document has no requests to IANA.

10. References

10.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/rfc/rfc2119>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/rfc/rfc8174>.

10.2. Informative References

[A2A] "Agent2Agent (A2A) protocol", April 2025, <https://google-a2a.github.io/A2A/#/documentation?id=agent2agent-protocol-a2a>.

[Google-Agents-Whitepaper] "Agents", 2024, <https://www.kaggle.com/whitepaper-agents>.

[I-D.ietf-nmop-network-incident-yang] Hu, T., Contreras, L. M., Wu, Q., Davis, N., and C. Feng, "A YANG Data Model for Network Incident Management", Work in Progress, Internet-Draft, draft-ietf-nmop-network-incident-yang-04, 14 June 2025, <https://datatracker.ietf.org/doc/html/draft-ietf-nmop-network-incident-yang-04>.

[I-D.ietf-nmop-terminology]
Davis, N., Farrel, A., Graf, T., Wu, Q., and C. Yu, "Some Key Terms for Network Fault and Problem Management", Work in Progress, Internet-Draft, draft-ietf-nmop-terminology-19, 18 June 2025, <https://datatracker.ietf.org/doc/html/draft-ietf-nmop-terminology-19>.

[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-05, 18 March 2025, <https://datatracker.ietf.org/doc/html/draft-irtf-nmrg-ai-challenges-05>.

[I-D.irtf-nmrg-network-digital-twin-arch]
Zhou, C., Yang, H., Duan, X., Lopez, D., Pastor, A., Wu, Q., Boucadair, M., and C. Jacquenet, "Network Digital Twin: Concepts and Reference Architecture", Work in Progress, Internet-Draft, draft-irtf-nmrg-network-digital-twin-arch-10, 28 February 2025, <https://datatracker.ietf.org/doc/html/draft-irtf-nmrg-network-digital-twin-arch-10>.

[I-D.rosenberg-ai-protocols]
Rosenberg, J. and C. F. Jennings, "Framework, Use Cases and Requirements for AI Agent Protocols", Work in Progress, Internet-Draft, draft-rosenberg-ai-protocols-00, 5 May 2025, <https://datatracker.ietf.org/doc/html/draft-rosenberg-ai-protocols-00>.

[MCP] "Model Context Protocol", November 2024, <https://modelcontextprotocol.io/>.

[RFC6241] Enns, R., Ed., Bjorklund, M., Ed., Schoenwaelder, J., Ed., and A. Bierman, Ed., "Network Configuration Protocol (NETCONF)", RFC 6241, DOI 10.17487/RFC6241, June 2011, <https://www.rfc-editor.org/rfc/rfc6241>.

[RFC7011] Claise, B., Ed., Trammell, B., Ed., and P. Aitken, "Specification of the IP Flow Information Export (IPFIX) Protocol for the Exchange of Flow Information", STD 77, RFC 7011, DOI 10.17487/RFC7011, September 2013, <https://www.rfc-editor.org/rfc/rfc7011>.

[RFC7854] Scudder, J., Ed., Fernando, R., and S. Stuart, "BGP Monitoring Protocol (BMP)", RFC 7854, DOI 10.17487/RFC7854, June 2016, <https://www.rfc-editor.org/rfc/rfc7854>.

[RFC8040] Bierman, A., Bjorklund, M., and K. Watsen, "RESTCONF Protocol", RFC 8040, DOI 10.17487/RFC8040, January 2017, <https://www.rfc-editor.org/rfc/rfc8040>.

[RFC8639] Voit, E., Clemm, A., Gonzalez Prieto, A., Nilsen-Nygaard, E., and A. Tripathy, "Subscription to YANG Notifications", RFC 8639, DOI 10.17487/RFC8639, September 2019, <https://www.rfc-editor.org/rfc/rfc8639>.

[RFC8641] Clemm, A. and E. Voit, "Subscription to YANG Notifications for Datastore Updates", RFC 8641, DOI 10.17487/RFC8641, September 2019, <https://www.rfc-editor.org/rfc/rfc8641>.

[RFC9232] Song, H., Qin, F., Martinez-Julia, P., Ciavaglia, L., and A. Wang, "Network Telemetry Framework", RFC 9232, DOI 10.17487/RFC9232, May 2022,

            <https://www.rfc-editor.org/rfc/rfc9232>.

   [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-1251D]
              "AN Agent Architecture v1.0.0", May 2025,
              <https://www.tmforum.org/resources/introductory-guide/
              ig1251d-an-agent-architecture-v1-0-0/>.

   [TMF-1258] "Autonomous Networks Glossary v1.2.0", May 2025,
              <https://projects.tmforum.org/wiki/display/PUB/
              IG1258+Autonomous+Networks+Glossary+v1.2.0>.

Acknowledgements

Contributors

   Qiufang Ma
   Huawei
   Email: maqiufang1@huawei.com

Authors' Addresses

   Qin Wu
   Huawei
   China
   Email: bill.wu@huawei.com

   Cheng Zhou
   China Mobile
   China
   Email: zhouchengyjy@chinamobile.com

   Luis M. Contreras
   Telefonica
   Email: luismiguel.contrerasmurillo@telefonica.com

   Sai Han
   China Unicom
   China
   Email: hans29@chinaunicom.cn

   Lionel Tailhardat
   Orange Research
   Email: lionel.tailhardat@orange.com

   Yong-Geun Hong
   Daejeon University
   Email: yonggeun.hong@gmail.com