RTG
Internet-Draft
Intended status: Informational
Expires: 4 September 2025

R. Rokui
Ciena
C. Li
Huawei
D. King
Lancaster University

3 March 2025

Artificial Intelligence (AI) for Network Operations
draft-king-rokui-ainetops-usecases-00

Abstract

This document explores the role of the IETF and IRTF in advancing Artificial Intelligence for network operations (AINetOps), focusing on requirements for IETF protocols and architectures. AINetOps applies AI/ML techniques to automate and optimize network operations, enabling use cases such as reactive troubleshooting, proactive assurance, closed-loop optimization, misconfiguration detection, and virtual operator assistance.

The document addresses AINetOps for both single-layer IP or Optical networks and multi-layer IP/Optical networks. It defines the concept of AINetOps for networking and provides its operational benefits such as network assurance, predictive analytics, network optimization, multi-layer planning, and more. It aims to guide the evolution of IETF protocols to support AINetOps-driven network management.

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 4 September 2025.

Table of Contents

1. Introduction 3
1.1. Background 4
2. Conventions and Definitions 4
3. AI, ML, Deep Learning and Gen-AI 5
4. Definition of AINetOps 6
5. Operational Benefits Provided by AINetOps 7
5.1. Operator Network Assistance 8
5.2. Network active and reactive assurance 9
5.2.1. Root Cause Analysis 10
5.3. Predictive Analytics 11
5.3.1. Proactive Network Assurance and Monitoring (Health Check) 12
5.3.2. Anomaly Detection 13
5.3.3. Trending and Forecasting 14
5.3.4. Predictive Maintenance 15
5.3.5. Network Capacity Planning 15
5.3.6. Traffic Optimization 16
5.4. Network Operational Insights 16
5.4.1. Operational Insights Requiring No Further Analysis . 16
5.4.2. Operational Insights Requiring Further Analysis . . . 17
5.5. Network Configuration Management 18
5.6. IP/Optical Multi-layer Planning 19
5.7. Cross-Layer and Multi-Layer Optimization 19
5.8. Traffic Optimization 19
5.9. Closed-Loop Automation 20
5.10. Network Maintenance and Cleanup 20
5.11. Network API Construction 20
5.12. AI-Driven Security Monitoring 21
5.12.1. Threat Detection and Mitigation 21
5.12.2. Intrusion Detection and Prevention 22
5.12.3. Security Policy Automation 22
5.13. Multi Agent Interworking 22

6. AINetOps Scenarios and Use-cases 23
6.1. Network Active and Reactive Assurance 23
6.2. Network Pro-active Assurance 26
6.3. Network Anomaly Detection 28
6.4. Network Predictive Maintenance 33
6.5. Detection of Network Misconfiguration 33
6.6. Generate Node Configuration 33
6.7. Cognitive Search On Internal Operator Data 33
6.8. Network Operator Assistant 35
6.9. Gen-AI based Network Operational Insights 36
6.10. Network Traffic Prediction 36
6.11. Multi-layer Use-case 36
6.12. Multi-layer Network Planning 36
6.13. Causality Discovery 38
6.14. Network Clean Up 38
6.15. Multi Agent Interworking 38
6.16. Network Traffic Management 41
6.16.1. Short term approaches 42
6.16.2. Inference 44
6.16.3. Longer term view 45
6.17. AI-Driven Resilience Testing 47
6.18. Energy Efficiency Optimization 51
6.19. AI-Driven Green Energy Optimization 53
6.20. AI-Driven Policy Enforcement and Compliance Auditing . . 56
6.21. AI-Driven Network Slicing Optimization 58
6.22. Other Use Cases 61
7. Security Considerations 61
8. Normative References 61
Appendix A. IANA Considerations 63
Acknowledgments 63
Contributors 63
Authors' Addresses 64

1. Introduction

The increasing complexity of modern networks has driven the need for innovative approaches to network operations and management. Artificial Intelligence for Network Operations (AINetOps) has emerged as an innovative concept, leveraging artificial intelligence (AI) and machine learning (ML) to automate, enhance, and optimize network management tasks. AINetOps offers the potential to reduce operational costs, improve service reliability, and enhance user experiences by enabling intelligent automation, predictive insights, and efficient decision-making.

The IETF and IRTF play a critical role in defining the protocols, architectures, and standards that underpin global networking. As AINetOps becomes integral to network operations, there is a growing

need to evaluate how existing IETF technologies can support AINetOps use cases and to identify gaps that may require new or extended solutions. This document aims to outline key AINetOps use cases, highlight associated technical challenges, and propose requirements for protocols and architectures to address these challenges effectively.

The use cases considered in this document span multiple aspects of network operations, including reactive troubleshooting, proactive assurance (e.g., anomaly detection, predictive maintenance), closed-loop optimization, and misconfiguration detection. Emerging capabilities, such as generative AI for operational insights and virtual operator assistants, further emphasize the need for a robust framework to support AI-driven network management. Additionally, the multi-layered nature of these use cases, encompassing IP, optical, and cross-layer optimization, underscores the complexity of integrating AINetOps into existing networks.

This document provides a foundation for advancing IETF protocols and architectures to enable AINetOps-driven network operations by exploring these use cases, the requirements, and their implications.

1.1. Background

Efficient and coordinated use of resources is paramount for maintaining optimal performance and reliability of many network environments. The applicability of Artificial Intelligence is well-established, and the use cases are outlined in this document.

Editors note: Future versions of this document will include prior IRTF and IETF work.

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 following terms are used in this document:

* AI: Artificial Intelligence aims to create systems capable of performing tasks that typically require human intelligence, such as understanding natural language, recognizing patterns, and making decisions.

* ML: Machine Learning is a subset of AI that involves training

algorithms on large datasets to enable them to learn patterns and make predictions or decisions without being explicitly programmed.

* Gen-AI: Generative-AI is a subset of ML techniques that creates new content, such as text, images, or audio, by learning from existing data.

* NLP: Natural Language Processing is a field of AI that focuses on the interaction between computers and humans through natural language.

* AINetOps: Artificial Intelligence for Network Operations refers to the application of AI, ML, and generative-AI techniques to enhance and automate network operations.

* Closed-Loop Optimization: Automated feedback-driven processes for continuously improving network performance and reliability.

* Multi-Layer Optimization: Addressing cross-layer dependencies and optimizing resources across different network layers, such as IP and optical layers.

* P-PNC: Packet Provisioning Network Controllers

* O-PNC: Optical Provisioning Network Controllers

3. AI, ML, Deep Learning and Gen-AI

Artificial Intelligence (AI) is the broad field dedicated to creating systems that can perform tasks typically requiring human intelligence, such as reasoning, problem-solving, and understanding language. Within AI, Machine Learning (ML) is a subset that focuses on developing algorithms that enable computers to learn from and make decisions based on data, improving their performance over time without explicit programming. Deep Learning is a further subset of ML that utilizes neural networks with many layers (hence "deep") to analyze various factors of data. This approach is particularly powerful in handling large and complex datasets, making significant advancements in areas such as image and speech recognition, natural language processing, and autonomous systems.

Generative AI (Gen-AI) is a specialized branch of ML that involves training models to generate new content, such as text, images, or music, by learning patterns from existing data, thereby enhancing the creative and adaptive capabilities of AI systems. Deep Learning techniques are often employed in Gen-AI to create more sophisticated and realistic outputs, pushing the boundaries of what AI can achieve in terms of creativity and innovation.

Figure 1 shows the relationship between AI, ML, Deep Learning, and Gen-AI.

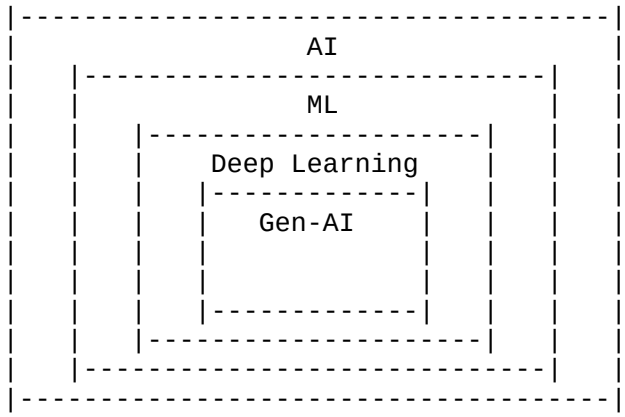


Figure 1: Figure 1: Relationship between AI, ML, Deep Learning, and Gen-AI

4. Definition of AINetOps

Figure 2 illustrates the concept of AI for Network Operations (AINetOps), which leverages AI, ML, Gen-AI techniques and rule-based systems to enhance and automate network operations. By integrating both historical and real-time streaming data, AINetOps employs advanced data analytics to uncover hidden patterns, establish data correlations, and provide trend forecasts and anomaly detection. These insights lead to significant operational benefits, including improved network performance, reduced downtime, and more efficient management of IP optical networks. Additionally, AINetOps enables proactive and predictive analytics, allowing network operators to address potential issues before they impact users, thereby ensuring more resilient and reliable network operations.

This draft introduces the term “Operational Benefit”, which encompasses the comprehensive suite of tools, and methodologies that facilitate the efficient management, debugging, troubleshooting, monitoring, configuration, and optimizing of IP Optical networks.

These operational benefits might include network management systems, automated diagnostic tools, performance monitoring and telemetry systems, configuration management platforms, and optimization algorithms. By leveraging these resources, operators can ensure the robust performance, reliability, and scalability of the network, ultimately enhancing service delivery and reducing operational costs. The integration of these operational benefits is crucial for maintaining seamless network operations and achieving strategic business objectives

Section 5 expands the Operational benefits shown in Figure 2 and provides a detailed explanation of the various operational benefits offered by AINetOp.

Figure 2 shows the relationship between AI, ML, Deep Learning, and Gen-AI.

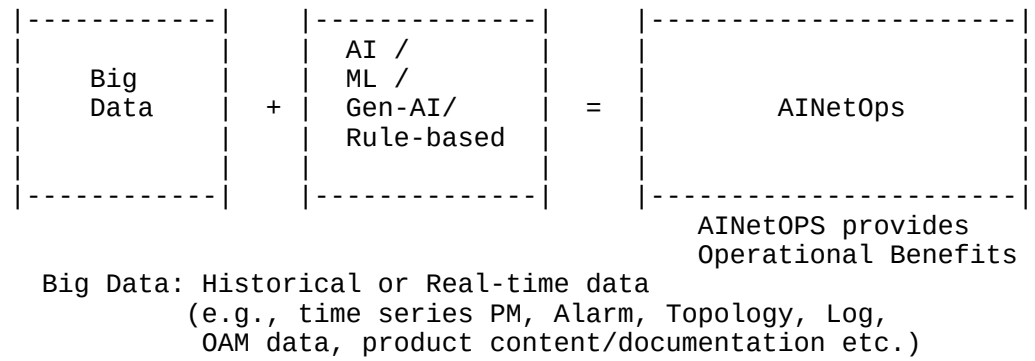


Figure 2: Figure 2: Definition of AINetOp

5. Operational Benefits Provided by AINetOps

AINetOps has the potential to revolutionize network operations by addressing the inherent complexity, scale, and dynamic nature of modern networks. By applying various AI/ML/Gen-AI techniques, network operators can transition from traditional manual or rule-based operations to intelligent, automated systems capable of real-time adaptation, predictive insights, and optimized decision-making.

This section outlines the following key areas where AINetOps can be applied effectively in network operations, leveraging both data-driven models and domain-specific knowledge.

* Section 5.1 "Operator Network Assistance"

* Section 5.2 "Network active and reactive assurance". This area is also related to "Root Cause Analysis" Section 5.2.1

* Section 5.3 "Predictive Analytics" which includes "Proactive Network Assurance and Monitoring" Section 5.3.1, "Anomaly Detection" Section 5.3.2, "Trending and Forecasting" Section 5.3.3, "Predictive Maintenance" Section 5.3.4 and "Network Capacity Planning" Section 5.3.5

* Section 5.4 "Network Operational Insight". This area can be grouped into "Operational Insights Requiring No Further Analysis " Section 5.4.1 and "Operational Insights Requiring Further Analysis " Section 5.4.2

* Section 5.5 "Network Configuration Management"

* Section 5.6 "IP/Optical multi-layer Planning"

* Section 5.7 "Cross-Layer and Multi-Layer Optimization"

* Section 5.8 "Traffic Optimization"

* Section 5.9 "Closed-Loop Automation"

* Section 5.10 "Network Maintenance and Cleanup"

* Section 5.11 "Network API Construction"

* Section 5.12 "AI-Driven Security Monitoring"

* Section 5.13 "Multi Agent Interworking"

5.1. Operator Network Assistance

Powered by Gen-AI, the operator network assistant functions as a virtual network engineer, providing a real-time recommendations, insights, and automated solutions. These systems use NLP for interface interaction, deep learning for anomaly classification, and contextual understanding to enhance operator decision-making.

AI-powered operator assistants function as virtual network engineers, providing real-time recommendations, insights, and automated solutions. These advanced systems leverage the power of natural language processing (NLP) to facilitate seamless and intuitive interactions between operators and the network management interface. By understanding and interpreting human language, these AI assistants can effectively communicate with operators, making it easier for them to manage complex network environments without needing extensive technical expertise.

In addition to NLP, Operator Assistance can integrate other AINetOps functions to solve operators scenarios and use-cases. This capability allows the system to provide timely alerts and recommendations, helping operators to address issues before they escalate into major disruptions. The deep learning models continuously improve over time, becoming more adept at recognizing new types of anomalies and adapting to evolving network conditions.

Furthermore, the contextual understanding capabilities of AI-powered operator network assistant significantly enhance operator decision-making. By considering the broader context of network operations, including historical data, current network state, and external factors, the AI can offer more relevant and actionable insights. This holistic approach ensures that operators receive comprehensive guidance tailored to the specific circumstances of their network. As a result, operators can make more informed decisions, optimize network performance, and maintain high levels of service reliability and efficiency. In essence, AI-powered operator network assistants are transforming network management by augmenting human capabilities with advanced technology, leading to smarter and more proactive network operations.

5.2. Network active and reactive assurance

Network active and reactive assurance and troubleshooting, both at the single- layer (IP or Optical) and multi-layer (IP over Optical), are critical components in maintaining the health and stability of modern IP, Optical, and IPoDWDM networks. This process involves the identification and resolution of network issues as they arise, ensuring that any disruptions or degradations are promptly addressed. By employing AINetOps techniques, network engineers can quickly pinpoint the root cause of problems, whether they originate in the IP layer, the optical layer, or across both. This reactive approach is essential for minimizing downtime and maintaining the quality of service expected by network users.

In single-layer troubleshooting, the focus is on isolating and resolving issues within a specific layer of the network. For example, in an IP network, this might involve diagnosing routing problems, addressing IP address conflicts, IP layer misconfiguration, hardware failure or resolving issues with network protocols. In an optical network, single-layer troubleshooting could involve identifying fiber cuts, optical signal degradation, or equipment failures

Multi-layer troubleshooting, on the other hand, requires a more integrated approach, as it involves identifying and resolving issues that span across multiple layers of the network. This could include problems where an issue in the optical layer affects the IP layer, such as signal impairments that impact data transmission quality. By effectively managing both single-layer and multi-layer troubleshooting, network engineers can ensure a more robust and resilient network infrastructure.

The importance of assurance and troubleshooting cannot be overstated in today’s high-demand network environments. Rapid response to network issues is crucial to maintaining service continuity and meeting the expectations of end-users. Advanced diagnostic tools and techniques, such as real-time monitoring, automated alerts, and detailed analytics, play a vital role in this process. These tools enable engineers to quickly detect anomalies, assess their impact, and implement corrective actions. Through continuous improvement of assurance and troubleshooting practices, network operators can enhance their ability to maintain network performance, reduce operational risks, and deliver a reliable and high-quality service to their customers.

5.2.1. Root Cause Analysis

In the context of "Network active and reactive assurance," Root Cause Analysis (RCA) is a critical aspect that extends the reactive troubleshooting process to uncover the underlying reasons behind network issues. When an issue is detected in the network, RCA leverages advanced AINetOps techniques to correlate events across different layers of the network, whether it be IP, Optical, or a combination of both. This comprehensive approach ensures that the root cause of an issue is accurately identified, rather than just addressing the symptoms. Techniques such as graph-based analysis enable network engineers to visualize and trace the sequence of events leading to a problem, providing a clear pathway to the source of the issue.

Moreover, natural language processing (NLP) for log analysis plays a significant role in RCA by automating the examination of vast amounts of log data generated by network devices. NLP can sift through logs to identify patterns and anomalies that might be missed by manual inspection. This capability is particularly useful in multi-layer networks where issues in one layer can propagate and manifest in another. By efficiently parsing through logs and correlating data, NLP helps pinpoint the exact cause of disruptions, thereby reducing the mean time to resolution (MTTR). Additionally, knowledge graph representations provide a structured and interconnected view of network components and their relationships, aiding in the rapid identification of fault points and their impact on the network.

By accurately diagnosing the root cause of network issues, network operators can implement targeted corrective actions that address the core problem, preventing recurrence and ensuring long-term stability. This precision in troubleshooting not only minimizes downtime but also enhances the overall reliability and performance of the network. Furthermore, insights gained from RCA can inform proactive measures and optimization strategies, contributing to a more resilient network infrastructure. In essence, RCA empowers network engineers with the tools and knowledge needed to maintain high service quality and meet the demands of modern, high-performance networks.

5.3. Predictive Analytics

Predictive analytics or advanced analytics uses historical and real-time network data, statistical algorithms, and ML techniques to identify the likelihood of future outcomes based on past data. In the context of network operations, predictive analytics involves the use of these methodologies in following areas to anticipate network issues, optimize performance, and improve operational efficiency. By examining patterns and trends in historical network data, predictive analytics can potentially forecast network problems before they occur, allowing for proactive management and maintenance.

The core idea behind predictive analytics is to transform data into actionable insights. For network operations, this means analyzing various metrics such as traffic patterns, latency, performance management (PM) data, and equipment performance to predict future states of the network. For instance, by identifying trends that have historically led to network failures, predictive analytics can alert operators to potential future failures, enabling them to take preventive measures. This proactive approach helps in minimizing downtime, enhancing service reliability, and optimizing resource allocation.

In summary, predictive analytics in network operations is about leveraging historical data and advanced analytical techniques to foresee and address potential issues before they impact the network. This approach leads to more efficient, reliable, and secure network operations, ultimately enhancing the overall performance and user experience. The AINetOps can address the following operator's scenarios.

5.3.1. Proactive Network Assurance and Monitoring (Health Check)

Proactive Network Assurance and Monitoring represents a paradigm shift from the Network active and reactive assurance discussed in Section 5.2. Instead of waiting for issues to arise and then addressing them, proactive network assurance involves anticipating potential problems and implementing measures to prevent them from occurring. This forward-thinking strategy leverages AINetOps to predict and mitigate network issues before they impact service quality.

In single-layer proactive assurance, the focus is on continuously monitoring and analyzing the health of a specific layer IP or Optical layer of the network to identify early warning signs of potential issues. For instance, in an IP network, this might involve analyzing traffic patterns to detect anomalies that could indicate an impending routing problem or hardware failure. ML algorithms can be employed to predict IP address conflicts or protocol misconfigurations before they cause disruptions. Similarly, in an optical network, proactive assurance could involve monitoring signal quality and fiber integrity to detect and address degradations before they lead to significant impairments or outages.

Multi-layer proactive assurance takes this approach a step further by integrating monitoring and analysis across both the IP and optical layers. This holistic view allows for the detection of complex issues that span multiple layers, such as optical signal impairments that could degrade IP data transmission quality. By correlating data from both layers, AINetOps solution can provide insights into how changes in the optical layer might affect IP performance and vice versa. This enables operators to take preemptive actions, such as optimizing signal paths or adjusting routing protocols, to maintain optimal network performance.

The benefits of proactive network assurance and monitoring are substantial. By identifying and addressing potential issues before they escalate, network operators can significantly reduce downtime and improve service reliability. This proactive stance not only enhances the user experience by ensuring consistent network performance but also reduces operational costs associated with

emergency troubleshooting and repairs. Furthermore, the use of advanced analytics and machine learning in AIOps allows for continuous learning and improvement, enabling networks to become more resilient and adaptive over time.

In today’s dynamic and high-demand network environments, proactive network assurance and monitoring is one of the operational benefits provided by AINetOps and are essential for staying ahead of potential issues and maintaining a competitive edge. By leveraging the power of AINetOp, network operators can transform their approach from reactive to proactive, ensuring that their networks are not only robust and resilient but also capable of delivering the high-quality service that users expect. This shift towards proactive assurance represents a significant advancement in network management, paving the way for more intelligent, efficient, and reliable network operations.

5.3.2. Anomaly Detection

A critical component of AINetOps in the context of predictive analytics is "Anomaly Detection", which leverages advanced ML algorithms to enhance network reliability and performance. By employing ML techniques such as supervised, unsupervised or reinforcement learning, AINetOps can predict anomalies in real-time by analyzing vast amounts of network telemetry data. Supervised learning models, trained on historical data, recognize known issues, while unsupervised models identify new anomalies by spotting outliers. This comprehensive detection mechanism ensures both familiar and novel network issues are identified promptly. Predictive models, utilizing techniques like time-series forecasting, enable the identification of potential network problems, such as link failures or traffic congestion, before they occur. By forecasting future network states based on historical and current data, these models provide early warnings, allowing for timely interventions to prevent unexpected downtime and maintain optimal performance.

Clustering techniques further enhance anomaly detection by grouping similar data points to identify patterns and trends that signal imminent failures or suboptimal behavior. This method allows ML models to discern subtle changes in network behavior that might otherwise go unnoticed. For example, clustering can reveal traffic congestion patterns under specific conditions, enabling preemptive measures to alleviate potential issues. Additionally, clustering helps identify the root causes of anomalies by correlating various network events and metrics, facilitating a more effective troubleshooting process. By integrating these advanced ML techniques, AINetOps not only improves anomaly detection but also empowers network operators with the insights needed to maintain a high-performing and reliable network infrastructure.

5.3.3. Trending and Forecasting

"Trending and Forecasting" operational benefit is distinct but is related to "Anomaly Detection" Section 5.3.2. Trending and forecasting in the context of single-layer or multi-layer IP optical networks are pivotal components of predictive analytics, providing significant operational benefits through AINetOps. In single-layer networks, such as purely IP or optical networks, trending involves analyzing historical data to identify patterns and behaviors over time. For instance, in an IP network, trends in traffic volume, latency, and packet loss can be monitored to predict future network performance and capacity needs. Similarly, in an optical network, trends in signal quality, attenuation, and equipment performance can be tracked. By leveraging these trends, predictive models can forecast potential issues such as bandwidth bottlenecks or equipment degradation, allowing network operators to proactively optimize resources, plan for upgrades, and prevent service disruptions.

In multi-layer IP optical networks, where both IP and optical layers interact, trending and forecasting become even more powerful. This approach involves correlating data from both layers to gain a comprehensive understanding of network behavior. For example, trends in optical signal impairments can be analyzed alongside IP traffic patterns to predict how physical layer issues might impact data transmission and overall network performance. Forecasting in this multi-layer context can identify potential cross-layer issues, such as how an increase in optical signal noise might lead to higher IP packet error rates. By anticipating these issues, network operators can implement preemptive measures, such as rerouting traffic or adjusting signal parameters, to maintain seamless service. The integration of trending and forecasting through AIOps thus enhances the resilience and efficiency of IP optical networks, ensuring superior performance and reliability.

5.3.4. Predictive Maintenance

Predictive maintenance in the context of single-layer or multi-layer IP optical networks is other aspect of predictive analytics, offering substantial operational benefits through AINeetOps. In single-layer networks, such as purely IP or optical networks, predictive maintenance involves using historical and real-time data to forecast when network components might fail or degrade. For instance, in an IP network, data from routers and switches, such as CPU usage, temperature, and error rates, can be analyzed to predict hardware failures. Similarly, in an optical network, monitoring parameters like signal strength, attenuation, and equipment performance helps predict when optical amplifiers or transceivers might need maintenance. By accurately forecasting these maintenance needs, network operators can schedule interventions before failures occur, reducing unplanned downtime and extending the lifespan of network components.

In multi-layer IP optical networks, predictive maintenance becomes even more effective by considering the interactions between the IP and optical layers. This approach involves analyzing data from both layers to predict maintenance needs that could impact the entire network. For example, if optical layer data indicates a gradual degradation in fiber quality, predictive models can assess how this might affect IP layer performance, such as increased packet loss or latency. By understanding these cross-layer dependencies, network operators can prioritize maintenance activities that have the most significant impact on overall network health. This proactive approach ensures that both layers of the network are maintained optimally, preventing cascading failures and maintaining high service quality. Through the integration of predictive maintenance with AIOps, IP optical networks can achieve greater reliability, efficiency, and cost-effectiveness, ensuring uninterrupted service delivery to end-users.

5.3.5. Network Capacity Planning

Predictive analytics also plays a crucial role in capacity planning and performance management. By forecasting future traffic demands, network operators can ensure that the infrastructure is adequately scaled to meet those demands without over-provisioning. This not only optimizes the use of resources but also ensures that the network can handle peak loads efficiently. Additionally, predictive analytics can help in identifying and mitigating potential security threats by analyzing traffic patterns and detecting anomalies that may indicate malicious activities.

5.3.6. Traffic Optimization

Referring to Section 5.8 for details of AINetOps "Traffic Optimization".

If "Traffic Optimization" is based on prediction of the traffic flows, it can be categorized as one of the areas of "Predictive Analytics".

5.4. Network Operational Insights

"Network Operational Insights" refers to the comprehensive visibility and understanding of an IP optical network's performance and behavior. This concept involves collecting and analyzing detailed data about the network's operations. By providing this valuable insight to network operators, they can gain a holistic view of the network's health and performance. This enables operators to understand their network better and ensure a robust and resilient infrastructure.

By having a detailed understanding of network usage patterns, traffic flows, and performance metrics, operators can make data-driven decisions to optimize resource allocation and improve overall efficiency. This insight is particularly valuable in multi-layer IP/Optical networks, where the interplay between different network layers can be complex. [RFC5557] provides examples of the PCE being using to optimize resource allocation.

By leveraging these insights, operators can ensure that both the IP and optical layers are operating harmoniously, leading to optimal performance and cost efficiency. In essence, Network Operational Insights empower operators with the knowledge needed to maintain a high-performing, resilient, and future-proof network infrastructure.

The network operational insight can be grouped into two categories. By categorizing network operational insights into these two categories, operators can better prioritize their efforts and resources, ensuring both immediate and long-term network health and performance.

5.4.1. Operational Insights Requiring No Further Analysis

Network Operational Insights that fall under this category are those that can be obtained directly from existing data and real-time monitoring without the need for further analysis or simulation. These insights provide immediate, actionable information that can help network operators quickly identify and address issues.

These insights are typically derived from real-time monitoring systems that continuously track network performance and health metrics. For example, showing the Network Element (NE) with the highest alarms or displaying the current alarm table for a specific NE (e.g., NE 1.1.1.1) can provide immediate visibility into potential issues. Similarly, identifying the NEs with the highest problems during the last hour or plotting the Bit Error Rate (BER) for the 10 worst modems in a specific region (e.g., Northeast) allows operators to quickly pinpoint areas that require attention. These insights are crucial for maintaining network stability and ensuring prompt resolution of emerging issues.

These insights also include detailed information about network components and their performance. For instance, identifying which photonic services cross a specific fiber (e.g., OTS1) or determining which modems are in use for a particular optical service (e.g., SVC-1) can help operators understand the current network configuration and its operational status. Additionally, insights such as the average time to failure for similar equipment in the network or identifying geographic regions with higher rates of network issues provide valuable context for proactive maintenance and resource planning. By leveraging these direct insights, operators can maintain a well-functioning network with minimal downtime and optimal performance.

5.4.2. Operational Insights Requiring Further Analysis

Network Operational Insights in this category require deeper analysis and possibly simulation to derive meaningful conclusions. These insights often involve complex scenarios where simple monitoring data is insufficient, and further investigation is needed to understand the underlying causes or to predict future behavior.

Insights that require investigation and simulation often involve predictive analytics and scenario planning. For example, determining whether an L0 optical service can be created between two cities (e.g., city A and Y) involves analyzing the current network topology, available resources, and potential constraints. Similarly, understanding why an IP TE-tunnel cannot be established between two points (e.g., point A and B) may require simulation of different routing scenarios and examination of network policies. These investigations help operators to not only troubleshoot current issues but also to plan and optimize future network expansions and configurations.

These insights are crucial for long-term network health and performance optimization. Identifying the most common failure points in the network or detecting signs of degradation in wireless network

performance requires a combination of historical data analysis and predictive modeling. By simulating different maintenance activities based on current network health, operators can prioritize tasks that will have the most significant impact. For instance, understanding what maintenance activities are needed based on the current network health can help in scheduling proactive maintenance that prevents future outages. These insights enable operators to take a strategic approach to network management, ensuring sustained performance and reliability over time.

5.5. Network Configuration Management

AI can assist in automating the generation and enforcement of network configurations, significantly enhancing network reliability and performance. By leveraging AI/Gen-AI algorithms, network operators can automate the creation of configuration templates that are precisely tailored to specific network requirements. These templates can encompass a wide range of settings, such as Quality of Service (QoS) parameters, Access Control Lists (ACLs), tunnel configurations, and service configuration ensuring that each network segment is optimized for its intended purpose. This automation not only speeds up the deployment process but also reduces the likelihood of human errors that can occur during manual configuration, leading to a more robust and efficient network infrastructure.

Furthermore, AINetOps can play a role on validation of network configuration, i.e., "network configuration audit". AINetOps plays a crucial role in validating configurations against predefined network configuration, ensuring that all network setups comply with intent configuration. By continuously monitoring network configurations, AINetOps can detect and flag any deviations or misconfigurations that could pose security risks or operational inefficiencies. For example, an AI system can identify inconsistencies in ACLs that might allow unauthorized access or detect suboptimal QoS settings that could degrade service quality. By proactively addressing these issues, AINetOps helps maintain the integrity and performance of the network, enabling operators to focus on strategic initiatives rather than troubleshooting configuration errors. This proactive approach to configuration management not only enhances network security and efficiency but also supports the dynamic and scalable nature of modern network environments.

5.6. IP/Optical Multi-layer Planning

Multi-layer planning is an approach that integrates the planning of IP and optical networks based on traffic patterns, network simulations, and capacity planning. By analyzing these factors, IP optical network can be designed to optimize resource allocation, enhance network efficiency, and ensure the network can handle current and future demands, resulting in a more resilient and scalable infrastructure.

5.7. Cross-Layer and Multi-Layer Optimization

AI can address the dependencies between different network layers, such as IP and optical layers, by integrating data and decision-making across these layers. Multi-layer optimization algorithms ensure resource efficiency and performance by aligning the goals of individual layers, such as minimizing power consumption at the physical layer while maintaining SLA guarantees at the application layer.

Moreover, Network Operational Insights facilitate informed decision-making for network optimization and capacity planning. By having a detailed understanding of network usage patterns, traffic flows, and performance metrics, operators can make data-driven decisions to optimize resource allocation and improve overall efficiency. This insight is particularly valuable in multi-layer IP/Optical networks, where the interplay between different network layers can be complex. By leveraging these insights, operators can ensure that both the IP and optical layers are operating harmoniously, leading to optimal performance and cost efficiency. In essence, Network Operational Insights empower operators with the knowledge needed to maintain a high-performing, resilient, and future-proof network infrastructure

5.8. Traffic Optimization

Another AINetOps operational benefits is "Traffic Optimization" where IP/Optical network traffic flows can be monitored and appropriate adjustments to network protocols, network topology, network configuration, load balancing, bandwidth allocation and so on can be dynamically initiated. AINetOps traffic optimization considers multiple factors such as latency, packet loss, and link utilization, enabling networks to adapt to changing conditions in real time.

Expanding on this, AINetOps traffic optimization leverages advanced algorithms to continuously monitor network conditions and predict potential congestion points before they impact service quality. By analyzing historical data and real-time metrics, machine learning models can forecast traffic patterns and proactively adjust routing

decisions to ensure optimal performance. For instance, AINetOps can reroute traffic through less congested paths when high utilization is detected, balancing the load and enhancing overall network efficiency. This intelligent management reduces latency and packet loss while maximizing bandwidth utilization.

Furthermore, traffic optimization enhances the network's ability to respond to sudden changes in demand, such as peak usage times or unexpected traffic spikes. Traditional static configurations may struggle with such fluctuations, leading to bottlenecks and degraded performance. With AI, the network can dynamically reconfigure itself in real-time, redistributing traffic loads and reallocating bandwidth as needed. This adaptability reduces the need for manual interventions and allows network operators to focus on strategic initiatives. In essence, AI-driven traffic optimization enables networks to be more resilient, responsive, and capable of delivering consistent high-quality service.

Note that "Traffic Optimization" AINetOps operational benefits is closely related to "Predictive Analytics" covered in Section 5.3.

5.9. Closed-Loop Automation

Closed-loop automation systems use AI to adjust network configurations based on real-time data dynamically. Reinforcement learning (RL) algorithms and policy-based decision frameworks can automate traffic engineering, resource allocation, and fault remediation tasks. AI-driven systems ensure optimal network performance without human intervention by continually monitoring network state and applying corrective actions.

5.10. Network Maintenance and Cleanup

AI can automate cleanup operations by identifying and resolving transient issues, removing redundant configurations, and optimizing resource utilization. These tasks may involve the identification of "stale" network states or unused resources, enabling networks to operate more efficiently.

5.11. Network API Construction

Another significant operational benefit of implementing AINetOps in single-layer or multi-layer IP/Optical networks is the generation of various Network Controller APIs. These APIs are essential for the seamless integration of network controllers (whether IP, Optical, or multi-layer) with Operational Support Systems (OSS) or other network controllers. A key advantage of this operational benefit is that operators do not need to possess in-depth knowledge of the APIs.

Typically, network operators spend considerable time creating and verifying APIs to integrate IP or Optical network elements with the broader management layer, including OSS/BSS. By developing robust and versatile APIs, network operators can ensure smooth communication and coordination between different network management systems, thereby enhancing overall network efficiency and performance

The APIs developed for network controllers serve as a bridge, enabling the OSS to interact with the underlying network infrastructure in a more dynamic and automated manner. This integration allows for real-time data exchange, automated provisioning, and efficient fault management, which are essential for maintaining optimal network performance. Moreover, these APIs facilitate the orchestration of network resources across different layers, whether it be IP or Optical, ensuring that the network can adapt to varying demands and conditions with minimal manual intervention.

AINetOps leverages the power of Generative AI (Gen-AI) to further enhance this integration process. By translating the operator’s intent into precise network controller APIs, Gen-AI enables a more intuitive and user-friendly approach to network management. This translation capability ensures that even complex operational requirements can be seamlessly converted into actionable commands for the network controllers. This not only reduces the operational burden on network engineers but also significantly enhances the agility and responsiveness of the network to changing conditions and user demands.

5.12. AI-Driven Security Monitoring

AI is becoming a cornerstone of modern network security, enabling proactive, adaptive, and intelligent measures to safeguard network operations against a rapidly evolving threats. By leveraging AI, network operators can enhance their ability to detect, prevent, and respond to threats in real-time while automating complex security processes. This section details the key areas where AI drives security enhancements in network operations.

5.12.1. Threat Detection and Mitigation

AI significantly enhances threat detection and mitigation through ML and deep learning. By analyzing vast amounts of network traffic data, AI models identify unusual patterns and behaviors indicative of malicious activity. This includes detecting anomalies that signal threats like zero-day attacks or insider threats, generating real-time alerts, and incorporating external threat intelligence to recognize known attack signatures. Together, these capabilities

enable faster response times and improved threat recognition.

5.12.2. Intrusion Detection and Prevention

AI improves intrusion detection systems (IDS) and intrusion prevention systems (IPS) by enhancing accuracy and reducing false positives. It achieves this through behavioral analysis, which identifies unauthorized access or suspicious activities, and automated responses that isolate compromised devices or block malicious IP addresses. Additionally, AI's adaptive learning capabilities ensure continuous updates to address new threats in dynamic environments.

5.12.3. Security Policy Automation

Using AI would simplify the creation and enforcement of security policies by automating configurations and adjustments, reducing the potential for human error. It dynamically updates firewall rules and access controls based on real-time threat intelligence, assigns risk scores to network devices and applications to prioritize enforcement, and ensures compliance with regulatory standards by monitoring for deviations and recommending corrective actions.

5.13. Multi Agent Interworking

As seen in the use cases above, the usage of agents introduces various challenges, spanning from the definition of APIs that can be used by the various agent to the interworking with already existing components of the Network Management and Control stack. New challenges arise when we move from a single agent to a multi-agent architecture. When multiple agents are deployed we need to consider how they discover each other, how they interwork with the discovered agents and how they are kept in synch.

The discovery aspect could be relatively simple in the short term, when few agents will be deployed in the network and it could be possible to manually configure each agent with the identifiers and capabilities of the other agents to interact with. With the evolution of AI based architectures with more and more agents being part of the architecture, mechanisms to advertise their presence and more important their capabilities will be required.

The second aspect to consider is the interworking between them. As of today the way we interact with agents is mostly based on LLM, but would that be the best way for interacting between them as well? Probably a more machine oriented type of language, encoding and protocols would have better performances.

6. AINetOps Scenarios and Use-cases

{Editor’s note: This is a work in progress. More use cases will be added, and existing ones will be revised.}

This section further expands Section 5 by exploring scenarios and use cases for applying AINetOps in network operations, focusing on their architectural, procedural, and protocol-level requirements. Each use case highlights how AINetOps can be leveraged to address challenges in network management and optimization, while identifying the relevant IETF protocols, interfaces, and data models that are evolved or need enhancement.

For every use case described, the following dimensions are examined to provide a comprehensive understanding of its implications and requirements.

* Architecture: The high-level architecture necessary to support the use case, including control-plane and data-plane interactions, as well as integration points for AI-driven systems

* Interfaces and APIs: The key interfaces between AI systems and network elements, including management APIs (e.g., NETCONF, RESTCONF, gNMI) and telemetry interfaces

* Protocols: IETF protocols involved in enabling the use case, and potential extensions to existing protocols to accommodate AI-driven operations.

* Data Models: The data models required to represent network state, telemetry, policies, and configurations

* Processes and Procedures: Workflow considerations for integrating AI systems into existing operational practices, including training, validation, and deployment.

* Alignment with IETF Standards: Analysis of how existing IETF standards can be leveraged or extended to support the use case.

6.1. Network Active and Reactive Assurance

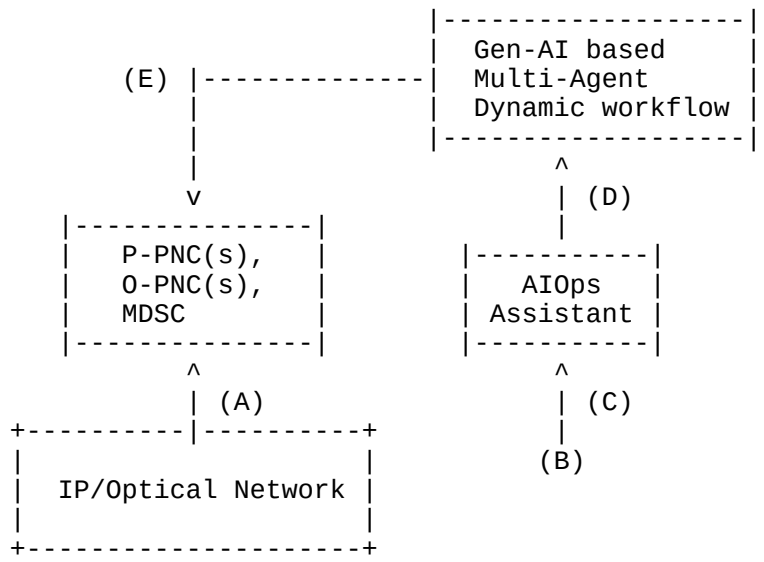
Network active and reactive assurance, both at the single- layer (IP or Optical) and multi-layer (IP over Optical), are critical components in maintaining the health and stability of modern IP, Optical, and IPoDWDM networks. This process involves the identification and resolution of network issues as they arise, ensuring that any disruptions or degradations are promptly addressed. By employing AINetOps techniques, network engineers can quickly

pinpoint the root cause of problems, whether they originate in the IP layer, the optical layer, or across both. This reactive approach is essential for minimizing downtime and maintaining the quality of service expected by network users.

In single-layer troubleshooting, the focus is on isolating and resolving issues within a specific layer of the network. Multi-layer troubleshooting, on the other hand, requires a more integrated approach, as it involves identifying and resolving issues that span across multiple layers of the network. This could include problems where an issue in the optical layer affects the IP layer.

In both reactive and active assurance, network faults have already occurred. These faults may include impairments such as optical fiber cuts, IP packet drops, IP link latency issues, or Threshold Crossing Alarms (TCA), among others.

As illustrated in Figure 3, reactive assurance assumes that a fault occurs in the IP/Optical network (Step A) and is subsequently detected by the operator through various means (Step B). Detection methods may include alarm monitoring, performance telemetry data analysis, or customer reports indicating service disruptions. To initiate troubleshooting, the operator can launch the AIOps-Assistant, which acts as the front-end interface for AINetOps (Step C). The assistant then utilizes the backend assurance and troubleshooting mechanisms, leveraging a Gen-AI multi-agent framework. In Step D, a dynamic workflow is executed to diagnose the issue and identify potential root causes. Optionally, at Step E, the Gen-AI dynamic workflow can recommend remedial actions to resolve the issue and implement these actions in a closed-loop fashion, ensuring automated network recovery.

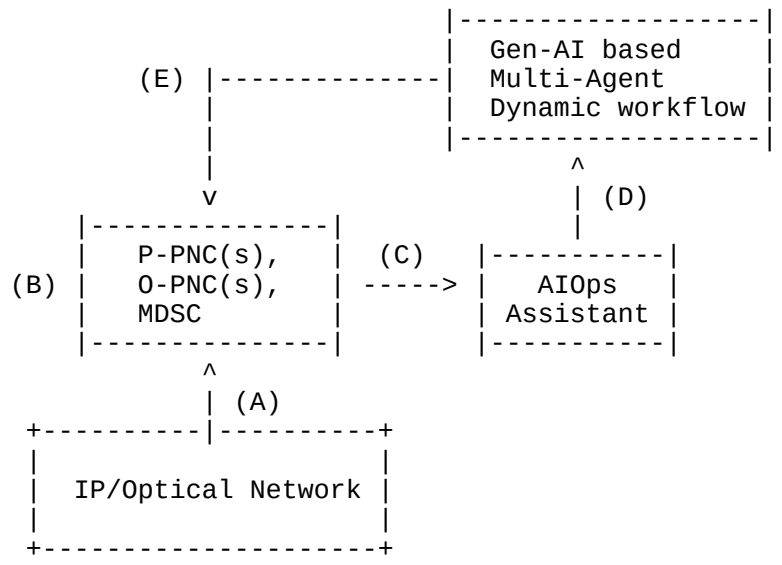


Legend:
(A) A fault happened in the network
(e.g., Fiber cut, IP packet drop, TCA crossing etc.)
(B) Operator is aware of the network issue
(C) To start troubleshooting, Operator starts AIOps-Assistant
(D) Start troubleshooting using Gen-AI multi-agent dynamic workflow
(E) Optional remedial actions

Figure 3: Multi-layer Reactive Assurance Using Gen-AI

In both reactive and active assurance, network faults have already occurred. These faults may include impairments such as optical fiber cuts, IP packet drops, IP link latency issues, or Threshold Crossing Alarms (TCA), among others.

The active assurance and troubleshooting process is illustrated in Figure 4. In contrast to Figure 3, active assurance assumes that a fault occurs in the IP/Optical network (Step A) and is subsequently detected automatically by higher-layer controllers (Step B). These controllers may employ detection methods that include monitoring alarms, analyzing performance telemetry data, or processing customer reports indicating service disruptions. To initiate troubleshooting, the detection logic launches the AIOps-Assistant, which serves as the front-end interface for AINetOps (Step C). Steps D and E are identical to those depicted in Figure 3.



Legend:
(A) A fault happened in the network
(e.g., Fiber cut, IP packet drop, TCA crossing etc.)
(B) The higher layer Controller notifies Operator
(C) To start troubleshooting, AIOps-Assistant starts automatically
(D) Start troubleshooting using Gen-AI multi-agent dynamic workflow
(E) Optional remedial actions

Figure 4: Multi-layer Active Assurance Using Gen-AI

More to be added.

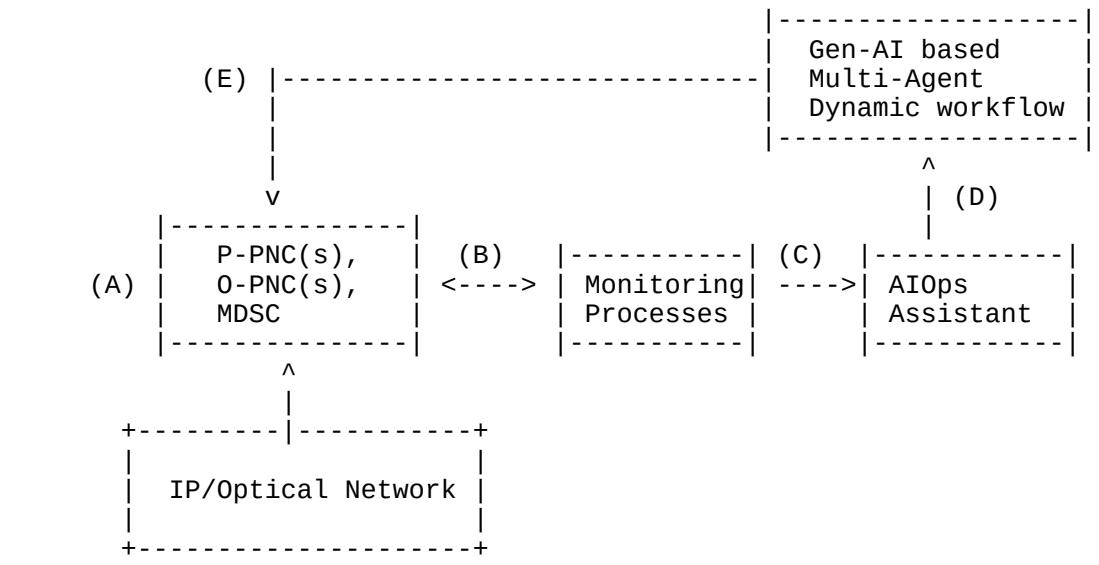
6.2. Network Pro-active Assurance

Unlike reactive and active assurance, proactive assurance does not wait for a fault to occur in the IP/Optical network. Instead, the network is continuously monitored through a series of trending and forecasting processes designed to detect early signs of deterioration that may eventually lead to faults.

As illustrated in Figure 5, achieving proactive assurance involves running multiple processes that continuously monitor network performance. These processes collect and analyze a wide array of network telemetry data, including performance monitoring (PM) data, alarms, logs, network topology, and inventory details (Step A). By employing various techniques including advanced AI/ML algorithms, these processes provide real-time trending and forecasting insights, identifying patterns and anomalies that could indicate potential degradation (Step B).

When these background processes detect any signs of deterioration or anomalous behavior, they trigger the AIOps-Assistant for further investigation (Step C). The AIOps-Assistant then leverages a Gen-AI multi-agent framework to initiate the assurance and troubleshooting procedures. In Step D, a dynamic workflow is executed to thoroughly diagnose the emerging issue and identify potential root causes. Optionally, at Step E, the Gen-AI dynamic workflow can recommend remedial actions to resolve the identified issues. These recommendations can be implemented in a closed-loop fashion, ensuring automated network recovery and continuous improvement of network performance. This proactive approach not only mitigates the risk of unexpected network faults but also optimizes operational efficiency by addressing issues before they escalate into service-impacting events.

Furthermore, by integrating advanced analytics with automated corrective measures, proactive assurance enhances overall network resilience. It enables network operators to maintain a high quality of service and reliability, even in complex and dynamic network environments.



Legend:
(A) Collect the IP/Optical telemetry data, inventory, logs etc.
(B) Processes which monitor the network
(C) Upon detection of potential issue, start AIOps-Assistant
(D) Start troubleshooting using Gen-AI multi-agent dynamic workflow
(E) Optional remedial actions

Figure 5: Multi-layer Pro-active Assurance Using Gen-AI

More to be added.

6.3. Network Anomaly Detection

Network anomaly detection is a critical component of modern network security and management, aimed at identifying deviations from normal network behavior that may indicate potential threats or operational issues. With the increasing complexity of networks and the growing sophistication of cyber threats, traditional rule-based detection methods are often insufficient. The integration of Artificial Intelligence (AI) and Machine Learning (ML) techniques offers a more dynamic and adaptive approach to detecting anomalies in real-time. This section outlines the architecture, interfaces, protocols, data models, and alignment with IETF standards necessary to implement an effective AI-driven network anomaly detection system. The design and implementation of such systems may use some relevant technologies, such as RFC 8345 (YANG Data Model for Network Topologies), RFC 6241 (NETCONF Protocol), and RFC 8529 (YANG Schema Mount).

Machine learning would provide a key function in network anomaly detection as it can be seamlessly integrated into the architecture, via the “Analysis Layer” described in the figure above. By leveraging ML techniques, it would be possible to identify deviations from normal behavior, uncovering anomalies that might be imperceptible to human network engineers.

An ML technique using unsupervised learning is particularly well-suited for network anomaly detection, as the network infrastructure is typically dynamic and evolving by nature. While machine learning requires large volumes of high-quality data and substantial computational resources for training, its benefits outweigh these challenges. Machine learning models offer generalizability, robustness, and reduced dependence on manual fine-tuning. More importantly, they enable the detection of complex and previously unseen anomaly patterns, enhancing network security, reliability, and operational efficiency.

* Architecture

  The architecture for network anomaly detection using AI typically involves a distributed system where data collection, analysis, and response mechanisms are decoupled but interconnected. The system comprises the following key components:

  o Data Collection Layer: Responsible for gathering network traffic data from various sources such as routers, switches, and endpoints. This layer may leverage protocols like IPFIX (RFC 7011) for flow data export.

  o Analysis Layer: Utilizes machine learning (ML) models to detect anomalies in the collected data. This layer may include both real-time and batch processing capabilities.

  o Response Layer: Executes predefined actions based on the analysis results, such as alerting administrators, blocking malicious traffic,or reconfiguring network devices. This layer may integrate with DOTS (RFC 8811) to mitigate DDoS attacks.

  The architecture should be scalable to handle large volumes of data and adaptable to incorporate new AI models as they evolve.

  Figure 6 illustrates the high-level architecture of an AI-based network anomaly detection system:

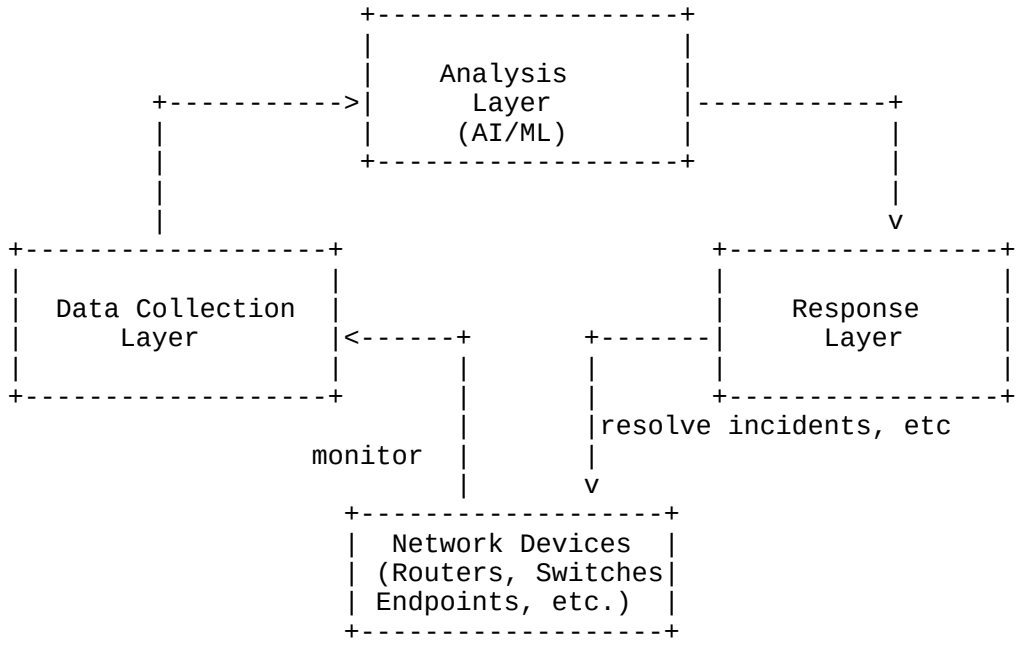


Figure 6: Architecture of network anomaly detection system

* Interfaces and APIs

  To facilitate interoperability and integration with existing network management systems, the following interfaces and APIs are recommended:

  - Northbound API: Provides a standardized interface for external systems to query anomaly detection results and receive alerts.This API should align with RESTCONF [RFC8040] for consistency with IETF standards.

  - Southbound API: Allows the anomaly detection system to interact with network devices for data collection and response actions. This API may use NETCONF [RFC6241] or RESTCONF [RFC8040] for device management.

  - Model Management API: Enables the deployment, updating, and monitoring of AI models used in the analysis layer. This API should support secure communication as defined in [RFC8446] (TLS 1.3).

  These APIs should adhere to RESTful principles or other widely adopted standards to ensure ease of integration.

Figure 7 illustrates the interaction between the anomaly detection system and external components via the defined interfaces:

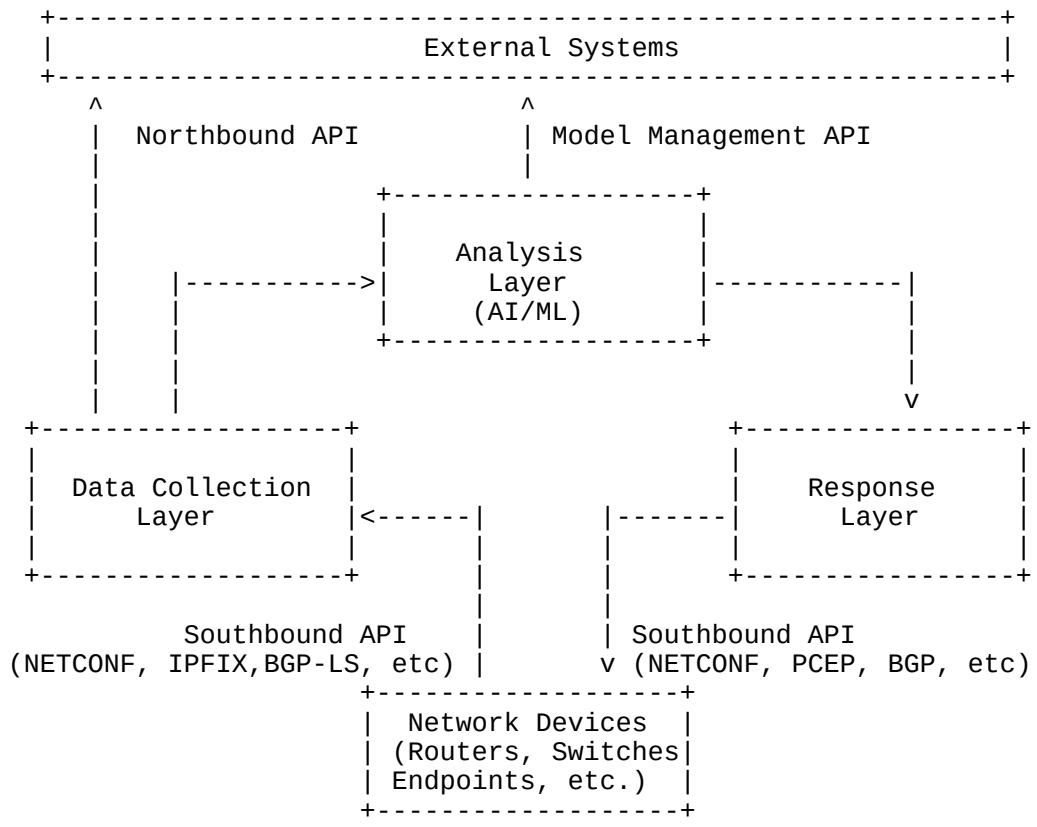


Figure 7: Interfaces of network anomaly detection system

* Protocols

  The following protocols are suggested for communication between the components of the anomaly detection system:

  - NETCONF/RESTCONF: For configuring and managing network devices and retrieving operational data, as defined in [RFC6241] and [RFC8040].

  - gRPC/HTTP2: For high-performance communication between the analysis layer and other components, leveraging HTTP/2 [RFC7540]) for efficient data transfer.

- MQTT: For lightweight, publish-subscribe messaging between distributed components, particularly in IoT environments, as specified in [RFC7252] (CoAP) or MQTT 5.0 (OASIS Standard).

The choice of protocol should consider factors such as latency, bandwidth, and security requirements.

* Data Models

Data models for network anomaly detection should be designed to capture both the structure and semantics of network traffic data. The following models are recommended:

- YANG Data Models: For representing network configuration and state data in a structured format, as defined in [RFC7950] and extended by [RFC8345] for network topologies.

- JSON/XML Schemas: For defining the format of data exchanged between components via APIs, consistent with [RFC8259] (JSON) and [RFC7303] (XML).

- Feature Vectors: For representing the input data to AI models, which may include packet headers, flow statistics, and behavioral patterns. These vectors should align with the IPFIX Information Model [RFC7012] for flow data representation.

These data models should be extensible to accommodate new types of network data and evolving AI techniques.

* Alignment with IETF

The development of AI-based network anomaly detection systems should align with existing IETF standards and working groups, such as:

- NETMOD (Network Modeling): For leveraging YANG data models [RFC7950], [RFC8345] and NETCONF/RESTCONF protocols [RFC8040].

- MILE (Managed Incident Lightweight Exchange, concluded): For standardizing the exchange of security incident information, as outlined in [RFC8329].

- DOTS (DDoS Open Threat Signaling ,concluded): For coordinating responses to distributed denial-of-service attacks, as defined in [RFC8811].

- Awaiting to add more WGs, BGP-LS, PCE, etc.

Collaboration with these groups ensures that the anomaly detection system integrates seamlessly with existing IETF frameworks and contributes to the broader goal of network security and management.

6.4. Network Predictive Maintenance

More to be added.

6.5. Detection of Network Misconfiguration

More to be added.

6.6. Generate Node Configuration

Generate node config with certain customer requirement (e.g., certain QOS, policy, ACL, tunnels, …)

More to be added.

6.7. Cognitive Search On Internal Operator Data

The operation of IP and optical networks comprises a wide range of management, monitoring and optimization tasks, including equipment configuration (switches, routers, OTNs, etc.), implementation of network policies, fault detection, troubleshooting, and capacity planning. The execution of such tasks usually requires access, comprehension and analysis of specific documentation containing information about network topologies, hardware inventory, vendor specifications, and pre-defined procedures.

Given the capacity of LLMs to understand natural language, including technical jargon, and their ability to process large amounts of information in short times, they can be used to build useful tools that support the network operational work, by executing comprehensive cognitive searches through the different documentation available to the operational teams, providing fast and concrete answers to technical enquiries, and making the access to such information a more efficient and interactive process.

To provision an LLM with such knowledge requires either a fine-tuning training job, that retrains an existing LLM, or the implementation of a RAG based architecture, where the information coming from the documentation is stored in a knowledge base and provided as context to the LLM. For this scenario, the RAG based approach has some specific advantages like lower computational cost, faster deployment, no need of retraining when the documentation is updated, and easier scalability.

Therefore, it is often the default approach for this type of solutions. Next section provides an architectural overview of how a RAG based system can be implemented to provide cognitive search for network operations.

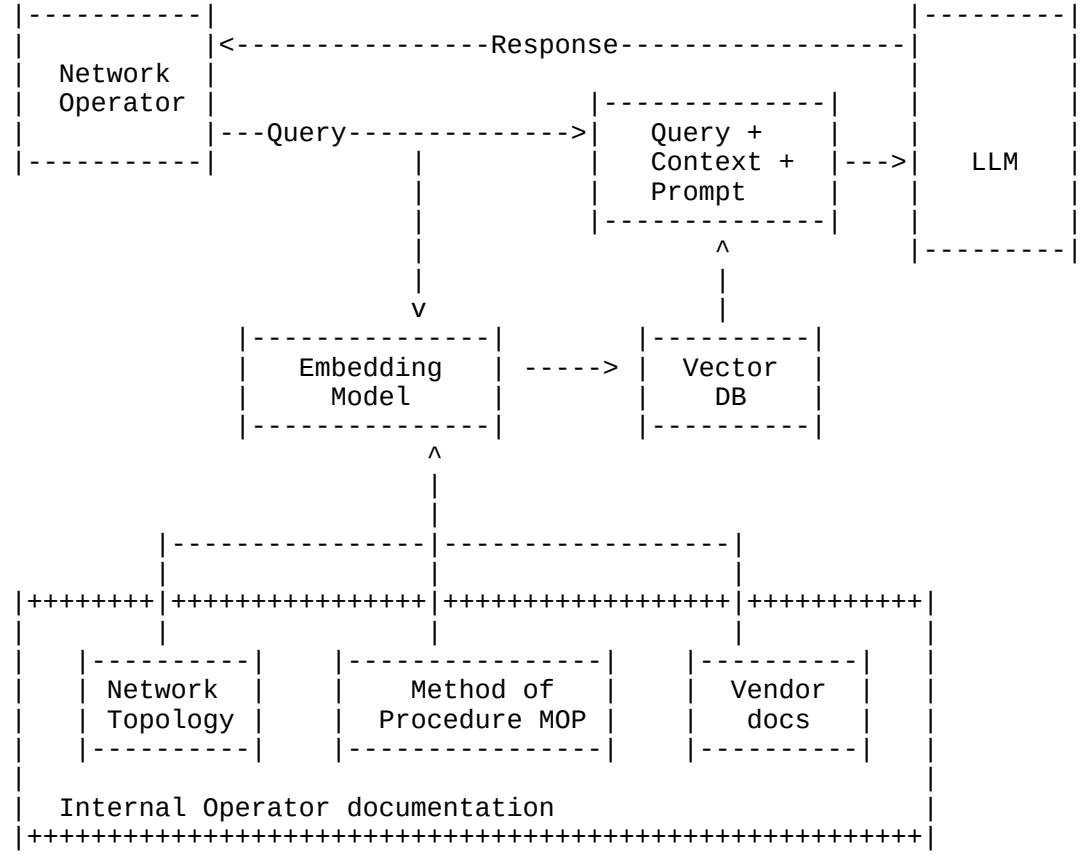
* Architecture

  In a RAG based architecture, a knowledge base is created by using an embedding model capable of splitting and transforming the content of different documents into numerical representations (vectors), and storing them in a data base, also known as Vector Data Base. The general process executed by the system every time a query is made by a user can be summarized in the following steps:

  1. Retrieval: The query made by the user is transformed by the embedding model and used to search and retrieve relevant information from the Vector Data Base.

  2. Augmentation: The information retrieved from the Vector Data Base is used to augment the query made by the user, adding context that might be unknown to the LLM.

  3. Generation: The augmented query is sent to the LLM, which then generates and answer in natural language that is finally delivered to the user.



As previously mentioned, the documents stored in the Vector Database for this specific use case correspond to various types of Network Operation Documentation. Thus, this system serves as a powerful tool, offering quick and efficient access to complex information across different areas of the Network Operations landscape, including network infrastructure, Standard Operating Procedures, security documentation, incident reports, and more.

More to be added.

6.8. Network Operator Assistant

Operator-Assistant as a virtual-expert-network-engineer.

More to be added.

6.9. Gen-AI based Network Operational Insights

More to be added.

6.10. Network Traffic Prediction

Telefonica use-case: Traffic-prediction using AI

More to be added.

6.11. Multi-layer Use-case

Multi-layer aspect of above use-cases, e.g.,

More to be added.

6.12. Multi-layer Network Planning

Several innovations have been developed at the IETF for multi-layer network (MLN) planning. This activity is involves coordinating and optimizing multiple network layers, such as IP, optical, and transport layers, to improve efficiency, resilience, and scalability. The Internet Engineering Task Force (IETF) has developed several technologies and standards to facilitate multi-layer network planning, including protocols for path computation, topology exchange, and resource optimization.

The components and interfaces for MLN planning include:

* Path Computation Element (PCE)

* Generalized Multi-Protocol Label Switching (GMPLS)

* Traffic Engineering Database (TED) and Topology Exchange

* Abstraction and Control of Traffic Engineered Networks (ACTN)

* YANG Models for Network Topologies and Node Inventory

These enabling technologies are discussed in the following sub-sections.

* Architecture

  The Abstraction and Control of Traffic Engineered Networks (ACTN) ACTN [RFC8453] architecture provides a framework for virtualized network resource control and abstraction, enabling efficient multi-layer coordination between packet and optical networks. It

defines key functional components like the Multi-Domain Coordinator (MDSC), which facilitates policy-based control and end-to-end service planning and provisioning.

* Interfaces and APIs

  The Path Computation Element (PCE) is a fundamental component of a Software Defined Networking (SDN) system, responsible for computing optimal traffic paths and dynamically adjusting them based on network conditions or demand. Originally designed for deriving paths for MPLS, and GMPLS, Label Switched Paths (LSPs), PCE delivers these computed routes to the LSP's head end via the Path Computation Element Communication Protocol (PCEP).

  The PCE architecture [RFC4655] enables efficient path computation for traffic-engineered networks by offloading complex calculations to a dedicated entity. Stateful PCE [RFC8051] and [RFC8231] extends the PCE framework by maintaining real-time network state awareness, allowing dynamic path optimization across layers. The Hierarchical PCE (H-PCE) [RFC6805] architecture supports multi-layer and multi-domain path computation by allowing collaboration between multiple PCEs.

* Protocols

  To be added.

* Data Models

  The YANG data modeling language is a cornerstone for MLN planning. It provides a structured way to represent network elements, configurations, and operational states, enabling programmatic control and integration across multiple network layers. Several IETF YANG models provide network topology, traffic engineering, optical transport, and service abstraction.

  A core YANG model for MLN planning is the Network Topology Model [RFC8345], which provides a generic framework for representing network nodes, links, and supporting attributes. This model would facilitate an AI-enabled planning system to define multi-layer relationships, such as the mapping between optical, ethernet, and IP layers, enabling a holistic approach to MLN planning.

* Alignment with IETF

  To be added.

6.13. Causality Discovery

Causality discovery: you want to know to be updated by Vincenzo.

More to be added.

6.14. Network Clean Up

Clean-up procedure in the network

More to be added.

6.15. Multi Agent Interworking

As briefly introduced in chapter 5, effectively deploying multiple AI agents for network management introduces significant interworking challenges that must be addressed for successful and reliable operation. These challenges span several key areas:

1. Communication and Coordination: Multiple agents operating in a shared network environment need to communicate effectively to coordinate their actions. This includes sharing information about network state, learned models, and planned interventions. A lack of standardized protocols and data models can lead to the need to deploy expensive and time consuming adaptation layers. It is also extremely important to determine the appropriate communication frequency and granularity to avoid overloading the communication network between them while keeping a sufficient level of details to avoid suboptimal or even harmful decisions due to incomplete information.

2. Conflict Resolution and Decision Fusion: When multiple agents are responsible for overlapping or interdependent network functions, conflicts in their decisions are inevitable. For example, one agent might decide to reroute traffic to alleviate congestion, while another agent simultaneously decides to scale down resources in the same area. Effective conflict resolution mechanisms are needed to prioritize actions, negotiate solutions, and ensure that the overall impact on the network is positive. This requires defining clear roles and responsibilities for each agent, establishing decision fusion strategies, and potentially incorporating a central arbitration mechanism, like for example in the case of coordination of multiple PCEs. Furthermore, handling conflicting information from different agents, potentially due to noisy or incomplete data, requires robust data validation and aggregation techniques.

3. Consistency and Stability: The dynamic nature of networks requires agents to continuously learn and adapt. However, independent learning by multiple agents can lead to inconsistencies in their learned models and behaviors, potentially causing instability in the network. For example, different agents might learn different optimal routing strategies, leading to oscillations and unpredictable network performance. Mechanisms for sharing learned knowledge, synchronizing models, and ensuring convergence towards a stable and consistent global state are essential. This could involve techniques like federated learning or distributed consensus protocols.

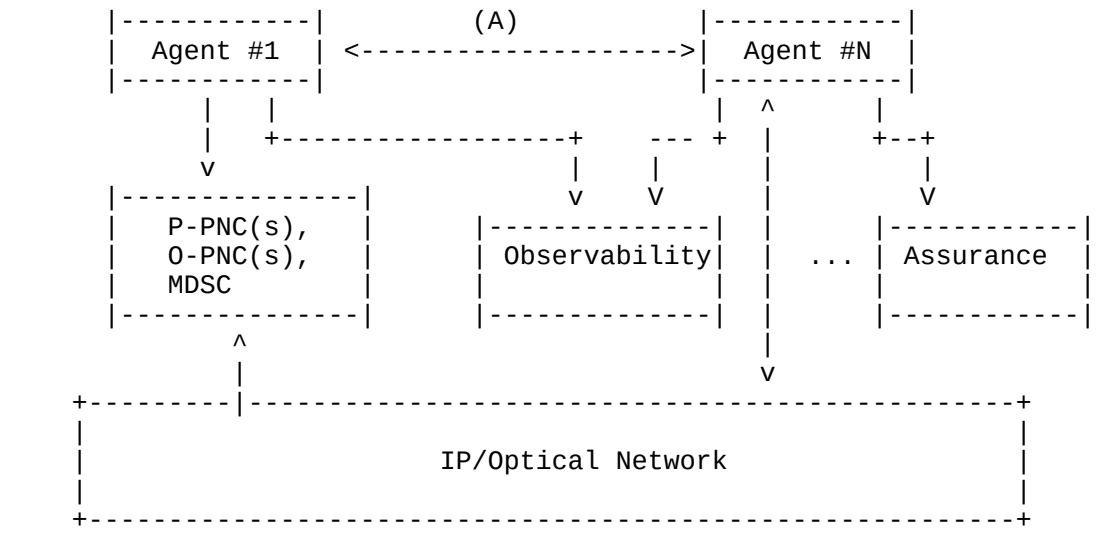
4. Trust and Security: In a multi-agent environment, trust and security become critical concerns. Agents might be vulnerable to malicious attacks or faulty behavior, which can compromise the entire network. Robust authentication and authorization mechanisms are needed to ensure that only legitimate agents can access and control network resources. Establishing trust between agents, potentially through reputation systems or blockchain technologies, can also enhance the overall security and resilience of the network.

5. Scalability and Management: As the number of agents and the complexity of the network increase, managing the interactions between agents becomes increasingly challenging. Scalable architectures and management frameworks are needed to handle the growing communication overhead, coordination complexity, and resource requirements. One possible option to overcome this problem could be leveraging on a hierarchical agent structure. As previously introduced, in order to allow for scalability, it is also important to foresee advertisement protocols/extensions to let the agents learn about their counterparts and their capabilities.

   Addressing these interworking challenges is essential for realizing the full potential of AI agents in network management. Developing standardized protocols, robust coordination mechanisms, and scalable management frameworks will pave the way for autonomous networks.

* Architecture

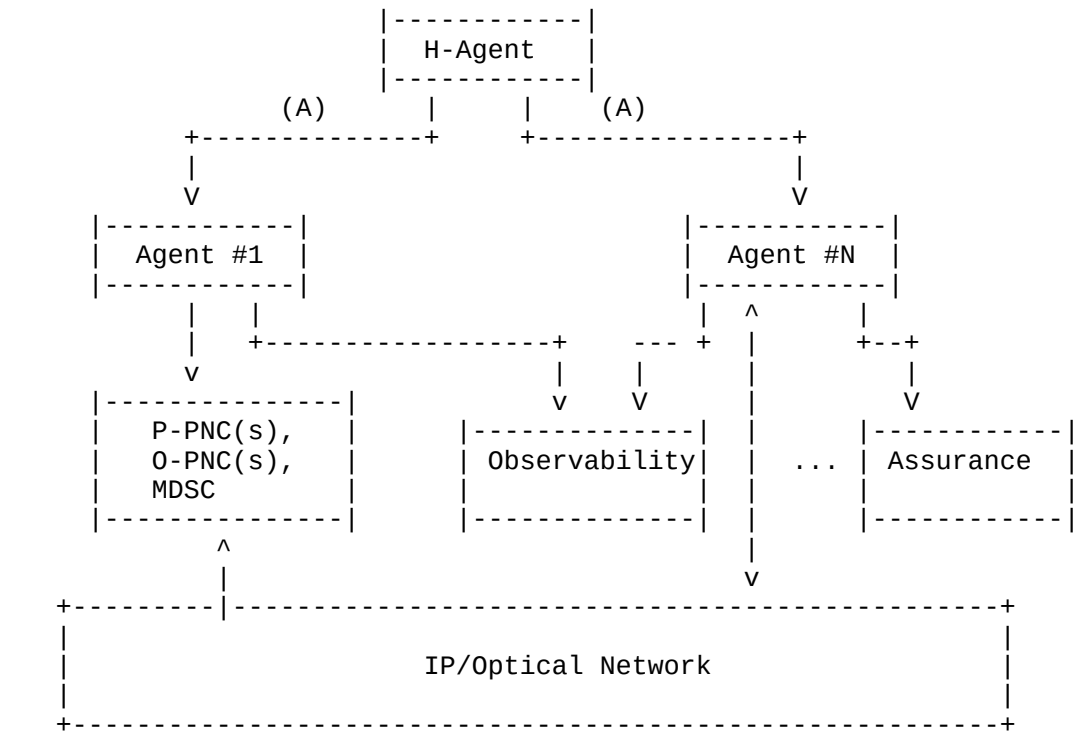
Multi agent architecture can be extremely complex, but figure Figure 8 tries to capture the main interwokring issues of this scenario. An example with an arbitrary number of agents (N) connecting to different components of the management and control stack (SDN controllers, observability function, assurance function, and others) is provided.



Legend:
(A) Inter Agent communication for coordination

Figure 8: Multi-agent architecture

Alternatively, a hierarchical solution can be foreseen, with an agent (H-Agent) specifically designed for coordinating agents, or an agent designated to play the role of H-Agent in addition to its duties, as shown in Figure 9:



Legend:
(A) H-Agent to Agent communication

Figure 9: Multi-agent hierarchical architecture

More to be added.

6.16. Network Traffic Management

Flow placement, traffic engineering/steering along with network resource defragmentation are among important aspects of network operations that can benefit from artificial intelligence.

Network routing protocols automate flow placement for best-effort traffic. Traffic engineering and steering are commonly based on statistical analysis and historical trends of network traffic. They are mostly implemented via configurations and tunnel setups, often employing scripts for automation purposes.

While there are some proactive approach to network resource defragmentation, reactive methods are still quite common. There are short-term approaches and longer-term views on employing AI to address traffic management.

6.16.1. Short term approaches

In the short-term, AI models train on operator’s network traffic patterns and employ a set of APIs to connect to network configuration equipment in order to add, remove, and modify configurations and perform different traffic management related tasks. Model training can be either off-line or on-line.

Initially, AI models perform their inference tasks exclusively based on their training on historical network traffic patterns, and topology changes in a centralized manner. In more advanced approaches, the models not only train on network traffic patterns, and network topology changes, but also learn how to interpret and digest external events. This added capability allows the AI models to be more effective in performing their traffic management tasks.

Generally speaking, IETF/IRTF can work on describing and providing synthetic networks along with synthetic traffic that can be used to train AI models. Furthermore, IETF/IRTF can also define and provide expected reasonable traffic flows.

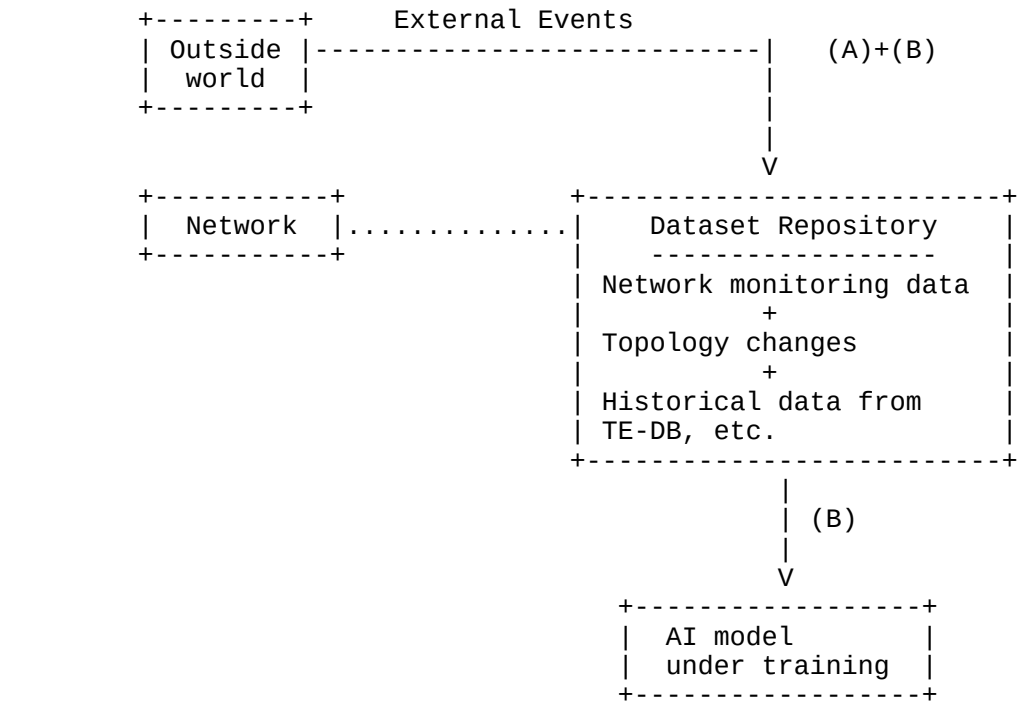
* Offline training

  During off-line training, external events, network monitoring information (available via protocols such as SNMP), historical data from traffic engineering databases, network topology changes, and other traffic-related data from the operator’s network are collected over time. This data is then used later during the training and model performance evaluation process.

  There is potential to define a set of APIs to collect information or enable a query mechanism to pull the required training data, particularly for external events.

  Selecting the important features from the entire dataset is another crucial aspect of training.

  IETF/IRTF can certainly play a role in both of the above-mentioned cases.



Legend:
--- Potential IETF defined and standardized interface.
(A) Extracting and storing outside world events data.
(B) Important features for training model for traffic management

Figure 10: AI assisted traffic management: Offline training

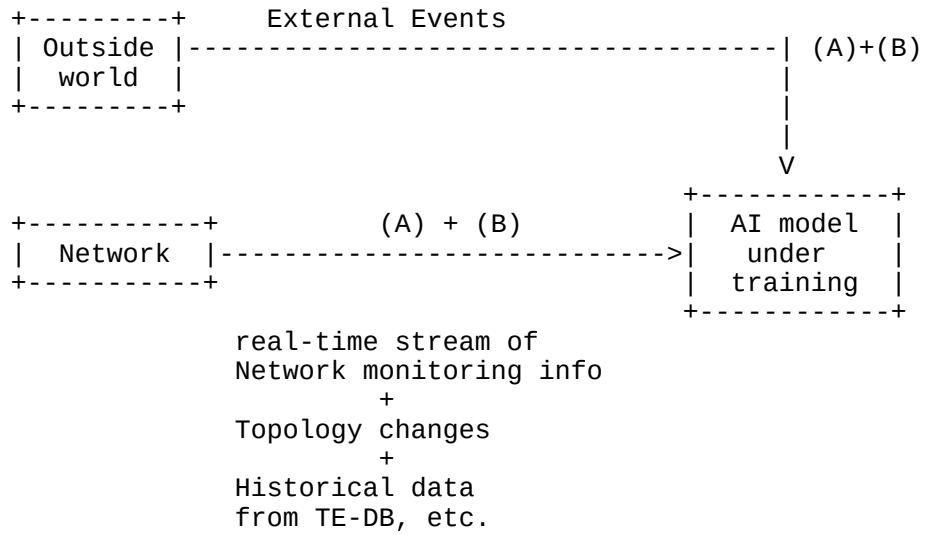
* Online training

Online training takes a more real-time approach. Here model training is based on processing data incrementally as it becomes available. This method is particularly suitable for scenarios such as network traffic management which require real-time learning and adaptation to changes.

A traffic management AI model under online training uses the same input sources as it does in offline training. However, unlike offline training, the data here is not stored in a repository but streamed into the training process. As such, the ground truth for model performance evaluation in online training is derived from observation of actual real time world events and network behavior, rather than stored data.

The training process therefore requires a mechanism to extract important features from the stream of incoming real-time network data and outside world events. These extracted features are then fed to the training process for adjusting model's parameters in a dynamic manner.

IETF/IRTF can work to standardize the mechanisms to identify important feature and implement the above mentioned required real-time data delivery and feature extraction.

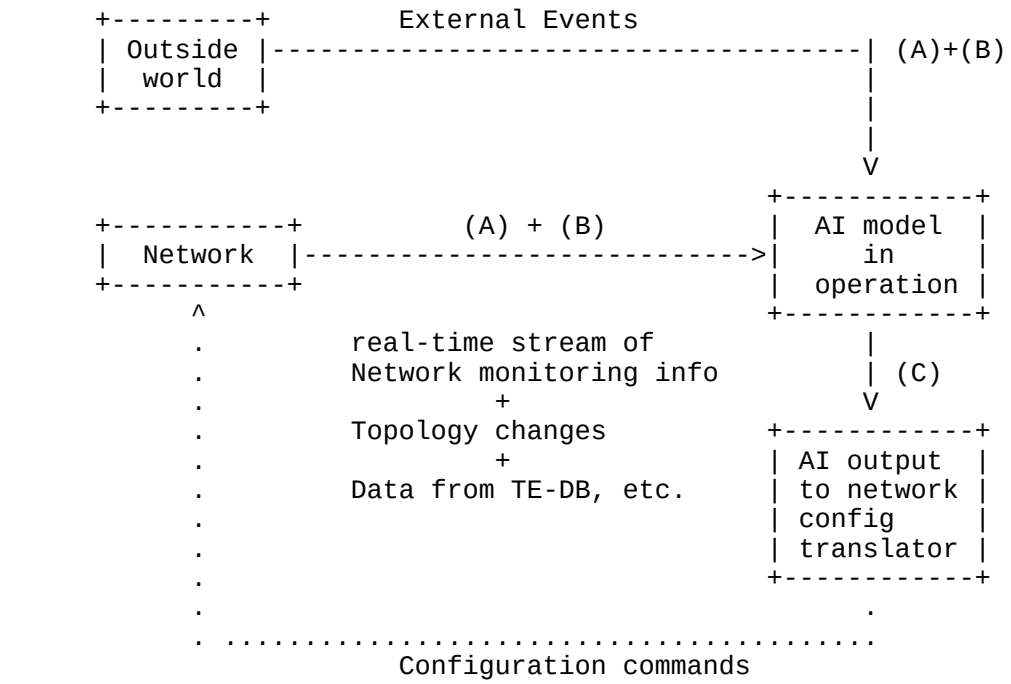


Legend:
--- Potential IETF defined and standardized interface.
(A) Extracting and storing outside world events data.
(B) Important features for training model for traffic management

Figure 11: AI assisted traffic management: On-line training

6.16.2. Inference

Inference phase for traffic management requires an interface to translate AI model's output to a set of network operation tasks and configuration commands. With this information readily available, existing protocols such as NETCONF can be employed to manage the network.



Legend:
--- Potential IETF defined and standardized interface.
(A) Extracting and storing outside world events data.
(B) Important features for training model for traffic management
(C) Standardized output of the AI model delivered for translation

Figure 12: AI assisted traffic management: Inference

6.16.3. Longer term view

Over time, the full integration of AI models and network elements will transform networks from their current state into agent-based or Agentic networks. In a distributed version of Agentic networks, each node is accompanied by an AI agents. Once trained, these agents work together to address flow placement, traffic steering/engineering, and other network related tasks such as traffic management, network resource defragmentation, and even routing.

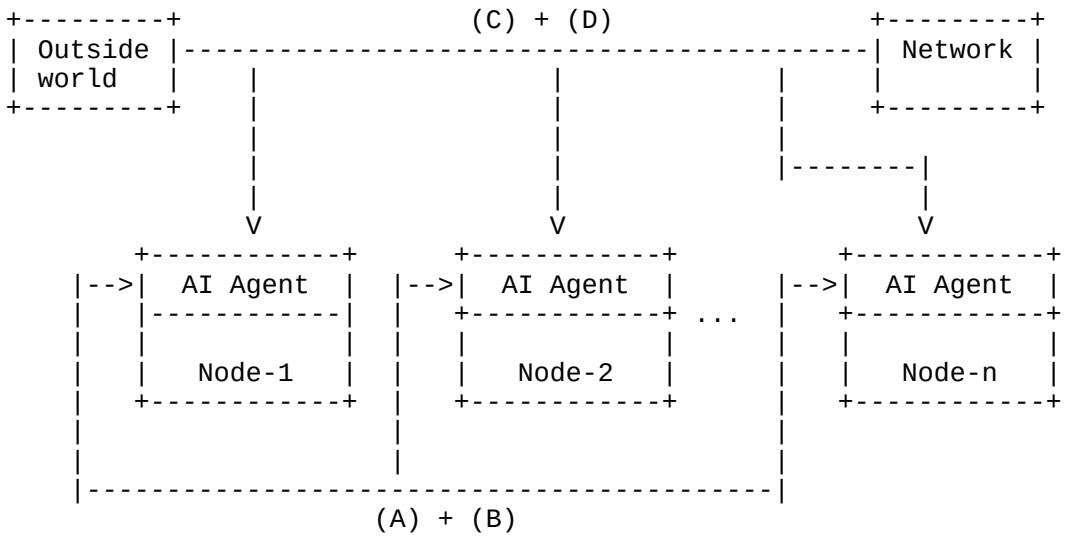
While being different from networks managed by a set of interworking multi agents , the Agentic networks face some of the same challenges outlined in the multi agent interworking section of the document. However, in Agentic networks, distributed training of the agents and proper knowledge sharing between them can enhance their collective training performance and can potentially alleviate some of these difficulties.

In these networks, AI agents trained on local traffic patterns and external events will exchange knowledge and network state information through a set of protocols in a distributed manner in order to address network related tasks. Agentic networks will potentially offer highly automated, streamlined, and tunnel-less traffic management that is currently available only for best-effort traffic.

In addition to the potential standardization opportunities outlined in the previous section, IETF/IRTF can alo play a role in defining and standardizing the followings:

* Training
  - Mechanisms for distributed training and knowledge sharing
  - Mechanisms for feeding traffic and overall network state information to agents for training purposes.
  - Mechanisms for feeding external events information to agents during training.
* Inference
  - Mechanisms for distributing agents’ decisions and inference results.
  - Mechanisms for feeding traffic and overall network state information to agents during inference phase.
  - Mechanisms for feeding external events information to agents during inference phase.
* There is also potentially a need to define mechanisms to identify flow requirements to the agents during network operations.

The following figure depicts an example of an Agentic network.



Legend:
--- Potential IETF defined and standardized interfaces
(A) APIs/Interfaces/Protocols for distributing training and knowledge sharing.
(B) APIs/Interfaces/Protocols for distributing agents' decisions and inference results.
(C) APIs/Interfaces/Protocols for feeding regionally observed traffic and network state info. to agents for training and inference.
(D) APIs/Interfaces/Protocols for feeding regionally observed external events info. to agents for training and inference.

Figure 13: Distributed agentic networks

More to be added.

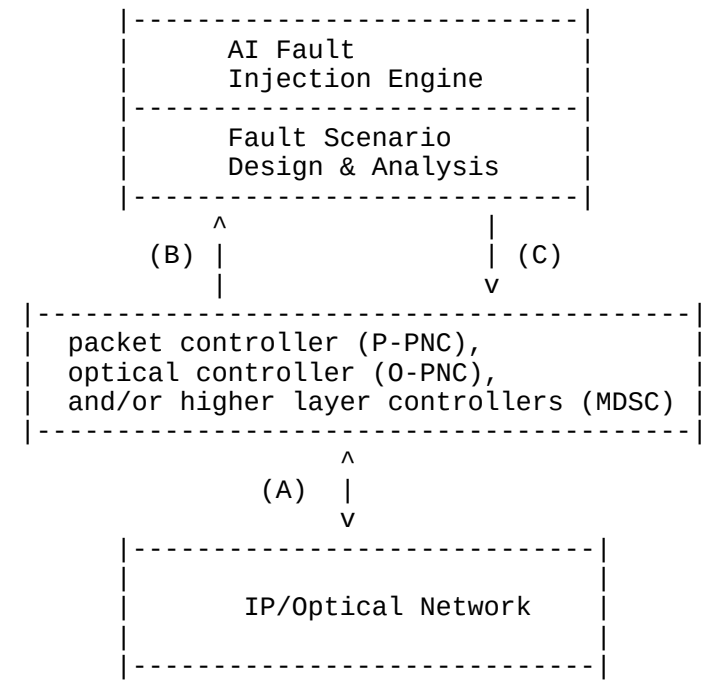
6.17. AI-Driven Resilience Testing

This use case leverages AI to design and execute fault injection scenarios that test the resilience of IP/optical networks under simulated failure conditions. By proactively introducing controlled disruptions-such as packet drops, latency spikes, or optical signal degradation-AI assesses the network's ability to detect, respond, and recover from faults. This approach enhances network robustness by identifying weaknesses and validating automated recovery mechanisms before real failures occur, addressing both single-layer (IP or optical) and multi-layer (IP over optical) scenarios.

The AI system analyzes historical failure data (e.g., fiber cuts, equipment outages), real-time telemetry (e.g., latency, BER), and external factors (e.g., weather events, traffic surges) to model probable failure points. It then injects faults, monitors the network's response, and refines recovery strategies, potentially in a closed-loop manner. For example, an AI model might predict a high-risk optical link based on trending attenuation, simulate a fiber cut, and evaluate whether IP-layer rerouting maintains SLAs. If recovery is suboptimal, it suggests adjustments (e.g., updating TE policies) and retests.

* Architecture

  The architecture integrates an AI Fault Injection Engine with network controllers and elements to simulate faults and assess resilience across IP and optical layers. Figure 14 illustrates this design, showing the AI Fault Injection Engine interfacing with P-PNC, O-PNC network controllers, which manage the IP/optical network. The engine designs fault scenarios, injects them via controller APIs, collects telemetry feedback, and triggers recovery actions as needed. This centralized approach leverages existing IETF control-plane components, ensuring compatibility with multi-layer coordination frameworks like ACTN.



Legend:
(A) Fault injection commands (e.g., disable link, drop packets, degrade signal)
(B) Telemetry feedback (e.g., latency, packet loss, BER)
(C) Recovery actions (e.g., reroute traffic, adjust optical parameters)

Figure 14: Architecture for AI-Driven Resilience Testing

* Interfaces and APIs

  The AI Fault Injection Engine interfaces with network controllers using standard management and telemetry APIs. NETCONF (RFC 6241) or RESTCONF (RFC 8040) enables fault injection by sending commands to disable interfaces, drop packets, or adjust optical parameters (e.g., signal power). Real-time telemetry is collected via gNMI (gRPC Network Management Interface) or OpenConfig streams, providing metrics like latency, packet loss, and Bit Error Rate (BER). External data sources (e.g., weather APIs, threat intelligence feeds) may integrate via REST APIs to enrich fault scenario design.

* Protocols

Several IETF protocols support this use case. PCEP (RFC 5440) extensions could enable dynamic path recomputation during fault scenarios, testing traffic engineering resilience. BGP (RFC 4271) or OSPF (RFC 2328) adjustments validate routing protocol stability under simulated failures. For optical layers, OTN (G.709) or ASON (G.8080) signaling protocols facilitate fault injection (e.g., simulating fiber cuts). Telemetry protocols like IPFIX (RFC 7011) or SNMP (RFC 3411) provide feedback data, though streaming alternatives (e.g., gNMI) are preferred for real-time needs.

* Data Models

  YANG models are central to representing fault injection and resilience data. The base Network Topology Model (RFC 8345) can be extended with new YANG modules to define fault parameters (e.g., failure type, duration, scope) and resilience metrics (e.g., recovery time, SLA compliance). OpenConfig YANG models for interfaces (e.g., openconfig-interfaces) and optical transport (e.g., openconfig-terminal-device) support fault execution and telemetry collection. A new YANG model may be needed to standardize fault injection workflows and outcomes.

* Processes and Procedures

  The process begins with AI training on historical failure data, synthetic scenarios, and real-time network state, using ML techniques like supervised learning for fault prediction and reinforcement learning for recovery optimization. Fault injection tests are scheduled (e.g., off-peak) or triggered on-demand, with operator oversight via an AIOps-Assistant interface (similar to 6.8). Post-test analysis generates reports on resilience gaps, updates network policies (e.g., QoS, routing), and refines the AI model’s training dataset. Closed-loop automation may execute recovery actions autonomously, validated by subsequent tests.

* Alignment with IETF

  This use case aligns with ongoing IETF efforts in multiple working groups. The Network Management Research Group (NMRG) explores AI applications in networking, providing a foundation for fault injection methodologies. The Traffic Engineering Architecture and Signaling (TEAS) working group’s work on resilience and path computation (e.g., RFC 8453 for ACTN) supports multi-layer testing. Extensions to YANG (NETMOD), PCEP (PCE), and telemetry protocols (OPSAWG) could standardize fault injection and resilience assessment, fostering interoperability across vendor implementations.

6.18. Energy Efficiency Optimization

This use case employs AI to optimize energy consumption across IP/optical networks by dynamically adjusting network resources based on traffic demand, equipment performance, and environmental conditions. With routers, switches, and optical amplifiers contributing significantly to power usage, AI-driven energy management reduces operational costs and carbon footprints while maintaining performance and reliability. It addresses both single-layer (IP or optical) and multi-layer (IP over optical) scenarios.

The AI system analyzes real-time telemetry (e.g., power usage, link utilization), historical patterns (e.g., peak/off-peak traffic), and external factors (e.g., electricity costs, cooling needs) to identify energy-saving opportunities. Actions include powering down idle ports, rerouting traffic to consolidate active paths, tuning optical signal parameters, or scheduling high-energy tasks during low-cost periods. For instance, during off-peak hours, the AI might deactivate redundant IP interfaces or reduce optical amplifier gain, ensuring SLAs are met with minimal power draw.

* Architecture

  The architecture for energy efficiency optimization mirrors the centralized design used for AI-Driven Resilience Testing (see Section 6.17). Figure 12 illustrates this, with the AI Energy Optimization Engine replacing the AI Fault Injection Engine, interfacing with P-PNC and O-PNC to manage the IP/optical network. The engine collects telemetry and external data, computes energy-efficient configurations, and applies them via controller APIs. In this context, (A) represents energy optimization commands (e.g., power down ports, adjust signal gain), (B) denotes telemetry feedback (e.g., power usage, traffic load), and (C) indicates configuration updates (e.g., reroute traffic, schedule operations).

* Interfaces and APIs

  The interfaces and APIs are largely identical to those in Section 6.17, adapted for energy optimization. NETCONF (RFC 6241) or RESTCONF (RFC 8040) delivers commands to adjust power states or reroute traffic, while gNMI or OpenConfig streams provide real-time telemetry (e.g., power consumption, utilization). External inputs, such as electricity pricing or weather data, integrate via REST APIs to inform optimization, consistent with the approach in Section 6.17.

* Protocols

The protocols align closely with those in Section 6.17, tailored for energy goals. PCEP (RFC 5440) supports traffic rerouting to consolidate paths, reducing energy use. BGP (RFC 4271) or OSPF (RFC 2328) adjustments optimize routing efficiency. For optical layers, OTN (G.709) signaling tunes power settings (e.g., lowering laser output). Telemetry protocols like IPFIX (RFC 7011) or SNMP (RFC 3411) provide feedback, with streaming alternatives (e.g., gNMI) preferred for real-time needs, as noted in Section 6.18.

* Data Models

  YANG models for energy optimization build on those in Section 6.17. The Network Topology Model (RFC 8345) can be augmented to define energy-specific parameters (e.g., power states, utilization thresholds) and metrics (e.g., watts consumed, energy cost), extending the fault-related models from Section 6.18. OpenConfig models for interfaces and optical transport support power adjustments and telemetry, with a potential new YANG model to standardize energy efficiency policies.

* Processes and Procedures

  The process begins with AI training on historical energy usage, traffic patterns, and cost data, using reinforcement learning for policy optimization and time-series analysis for demand forecasting. Optimization runs continuously or on a schedule, with operator oversight via an AIOps-Assistant (similar to 6.8). Post-optimization, the AI evaluates energy savings against performance impacts, updating policies and retraining as needed. Closed-loop automation applies adjustments, validated by telemetry, following the workflow principles in Section 6.17.

* Alignment with IETF

  This use case aligns with IETF efforts in sustainability and network management, paralleling Section 6.17 standardization ties. The Operations and Management Area Working Group (OPSAWG) supports telemetry and efficiency metrics, while the Traffic Engineering Architecture and Signaling (TEAS) working group’s path optimization work (e.g., RFC 8453 for ACTN) enables energy-efficient rerouting. Extensions to YANG (NETMOD), PCEP (PCE), and telemetry standards (OPSAWG) could standardize energy optimization, leveraging the same frameworks proposed in Section 6.17.

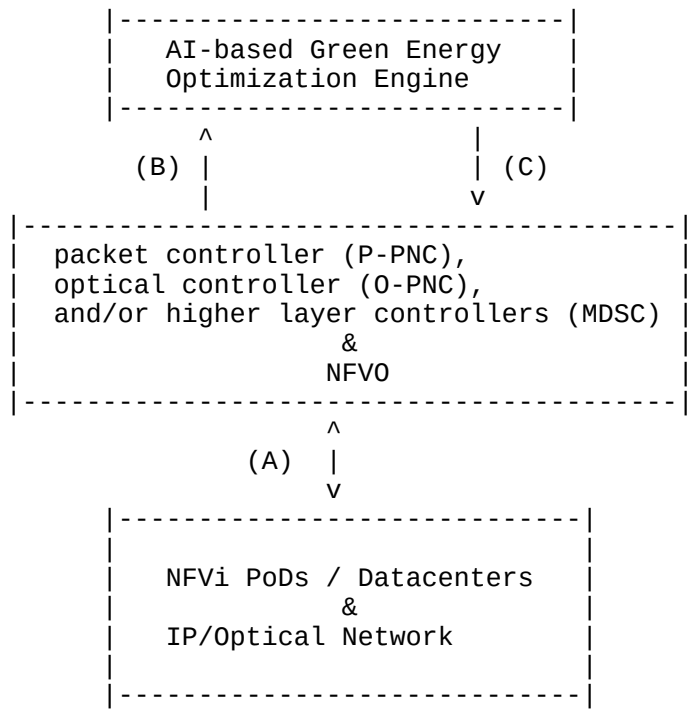
6.19. AI-Driven Green Energy Optimization

This use case leverages AI to optimize network and compute operations by prioritizing resources powered by green energy sources (e.g., solar, wind) over conventional ones, while ensuring performance requirements (e.g., latency, throughput) are met. As networks increasingly rely on distributed compute elements like Virtual Network Functions (VNFs) or edge servers, selecting energy sources for these workloads impacts both sustainability and cost. This applies to single-layer (e.g., IP compute nodes) and multi-layer (e.g., IP over optical with compute) scenarios.

The AI system analyzes real-time telemetry (e.g., latency, traffic load), energy source data (e.g., green vs. conventional availability), and external factors (e.g., renewable energy forecasts) to decide where and how to execute compute tasks. For example, in a mobile core network, a media optimizer VNF processing mobility traffic could be instantiated on a green-energy-powered server in a specific NFVi Point of Delivery (PoD) or datacenter instead of a conventionally powered one, provided latency SLAs are not violated. If green resources are unavailable or insufficient, the AI shifts workloads or adjusts traffic paths dynamically, balancing sustainability with service quality.

* Architecture

  The architecture integrates an AI Green Energy Optimization Engine with both compute orchestration and network control layers to manage workload placement and traffic across IP/optical networks and NFVi PoDs or datacenters. Figure 15 illustrates this design, showing the AI engine interfacing with an NFV Orchestrator (NFVO) to shift compute jobs (e.g., VNFs) between PoDs/datacenters based on green energy availability, and optionally with P-PNC and O-PNC for traffic adjustments. The engine collects telemetry and energy data, computes optimal configurations, and applies them via orchestration and controller APIs.



Legend:
(A) Optimization commands (e.g., instantiate VNF on green PoD, reroute traffic)
(B) Telemetry feedback (e.g., latency, energy source, compute load)
(C) Configuration updates (e.g., shift VNFs, adjust network paths)
NFVO: Network Function Virtualization Orchestrator

Figure 15: Architecture for AI-Driven Green Energy Optimization

* Interfaces and APIs

  Interfaces extend those in Section 6.17 and Section 6.18, with a focus on compute orchestration. The NFVO uses ETSI NFV MANO APIs (e.g., Os-Ma-nfvo reference point) to instantiate or migrate VNFs across NFVi PoDs/ datacenters based on green energy availability. NETCONF (RFC 6241) or RESTCONF (RFC 8040) manages traffic adjustments via P-PNC/O-PNC when needed, while gNMI or OpenConfig streams provide telemetry (e.g., latency, server energy type). REST APIs integrate external data like green energy availability or weather forecasts, consistent with Section 6.17.

* Protocols

Protocols align with Section 6.17 and Section 6.18, with additions for compute management. PCEP (RFC 5440) enables traffic rerouting to align with green-energy-powered nodes, while BGP (RFC 4271) or OSPF (RFC 2328) adjusts routing paths. OTN (G.709) signaling supports optical adjustments if involved. For compute, ETSI NFV protocols (e.g., VE-Vnfm-vnf for VNF management) complement network protocols. Telemetry uses IPFIX (RFC 7011) or SNMP (RFC 3411), with streaming options (e.g., gNMI) preferred, as in Section 6.17.

* Data Models

YANG models build on Section 6.17 and Section 6.18, with compute-specific extensions. The Network Topology Model (RFC 8345) can be augmented to include NFVi PoD/datacenter attributes (e.g., energy source, compute capacity) and metrics (e.g., carbon footprint, latency). OpenConfig models for interfaces and ETSI NFV YANG models (e.g., for VNF descriptors) support configuration and telemetry. A new YANG model may standardize green energy optimization across network and compute domains.

* Processes and Procedures

The process starts with AI training on historical traffic, latency, and energy source data, using reinforcement learning to optimize green energy use and predictive models for renewable availability. Optimization runs continuously, shifting VNFs to green PoDs/datacenters or adjusting traffic when viable, with operator oversight via an AIOps- Assistant (similar to 6.8). Post-optimization, the AI assesses sustainability gains against performance, updating policies and retraining. Closed-loop automation adjusts configurations, validated by telemetry, following Section 6.18 workflow.

* Alignment with IETF

This use case aligns with IETF sustainability and compute-network integration efforts. The Operations and Management Area Working Group (OPSAWG) supports telemetry for energy metrics, while the Computing in the Network Research Group (COINRG) and Network Function Virtualization Research Group (NFVRG) address compute placement, applicable to VNFs. The Traffic Engineering Architecture and Signaling (TEAS) working groups efforts (e.g., RFC 8453 for ACTN) enable green-energy-aware routing. Extensions to YANG (NETMOD), PCEP (PCE), and telemetry standards (OPSAWG) could standardize this, leveraging Section 6.17 and Section 6.18 frameworks.

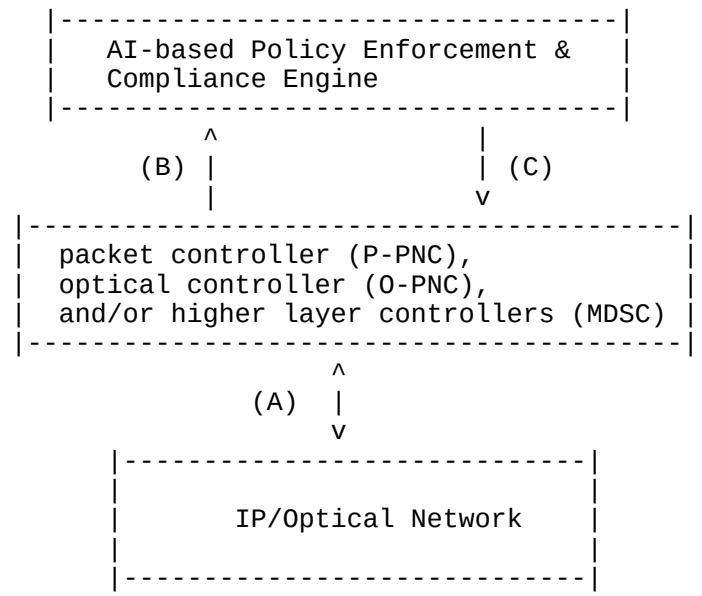
6.20. AI-Driven Policy Enforcement and Compliance Auditing

This use case leverages AI to automate the enforcement of network policies and auditing of compliance with regulatory standards and internal guidelines. By continuously monitoring network configurations, traffic, and security postures, AI ensures that policies are consistently applied and compliance requirements are met. This use case addresses both single-layer (e.g., IP) and multi-layer (e.g., IP over optical) scenarios, as well as cross-domain environments.

The AI system analyzes real-time telemetry (e.g., configuration changes, traffic flows, security logs), historical data, and external inputs (e.g., regulatory updates, threat intelligence) to enforce policies and audit compliance. For example, AI can detect unauthorized changes to firewall rules, enforce encryption standards for sensitive data, or ensure that network configurations align with GDPR requirements. If violations are detected, AI can automatically remediate issues or alert operators for manual intervention.

* Architecture

  The architecture integrates an AI Policy Enforcement and Compliance Engine with network controllers, security systems, and orchestration platforms. Figure 16 illustrates this design, showing the AI engine interfacing with P-PNC, O-PNC and Security Information and Event Management (SIEM) systems. The engine collects telemetry, enforces policies, and audits compliance, applying corrective actions via controller APIs.



Legend:
(A) Policy enforcement commands (e.g., block traffic, adjust QoS)
(B) Telemetry feedback (e.g., configuration changes, security logs)
(C) Compliance reports and alerts

Figure 16: Architecture for AI-Driven Policy Enforcement and Compliance Auditing

* Interfaces and APIs

  The AI engine interfaces with network controllers and security systems using standard management and telemetry APIs. NETCONF (RFC 6241) or RESTCONF (RFC 8040) delivers policy enforcement commands, while gNMI or OpenConfig streams provide real-time telemetry (e.g., configuration changes, traffic flows). SIEM systems integrate via REST APIs to provide security logs and threat intelligence.

* Protocols

  The protocols align with existing IETF standards. PCEP (RFC 5440) supports traffic engineering adjustments to enforce QoS policies, while BGP (RFC 4271) or OSPF (RFC 2328) ensures routing compliance. For security, protocols like IPsec (RFC 4301) and TLS (RFC 8446) enforce encryption standards. Telemetry protocols like IPFIX (RFC 7011) or SNMP (RFC 3411) provide feedback, with streaming alternatives (e.g., gNMI) preferred for real-time needs.

* Data Models

  YANG models are central to representing policies and compliance data. The Network Topology Model (RFC 8345) can be extended to define policy parameters (e.g., access control, encryption) and compliance metrics (e.g., audit logs, violation counts). OpenConfig YANG models for interfaces and security support configuration and telemetry. A new YANG model may standardize policy enforcement and compliance workflows.

* Processes and Procedures

  The process begins with AI training on historical configuration data, security logs, and regulatory requirements. Policy enforcement runs continuously, with operator oversight via an AIOps-Assistant (similar to 6.8). Post-audit, the AI generates compliance reports, updates policies, and retrains as needed. Closed-loop automation applies corrective actions, validated by telemetry.

* Alignment with IETF

  This use case aligns with IETF efforts in network management, security, and policy enforcement. The Operations and Management Area Working Group (OPSAWG) supports telemetry for compliance metrics, while the Security Area Working Group (SEC) addresses policy enforcement. Extensions to YANG (NETMOD), PCEP (PCE), and telemetry standards (OPSAWG) could standardize this use case, leveraging frameworks proposed in Section 6.17 and Section 6.18.

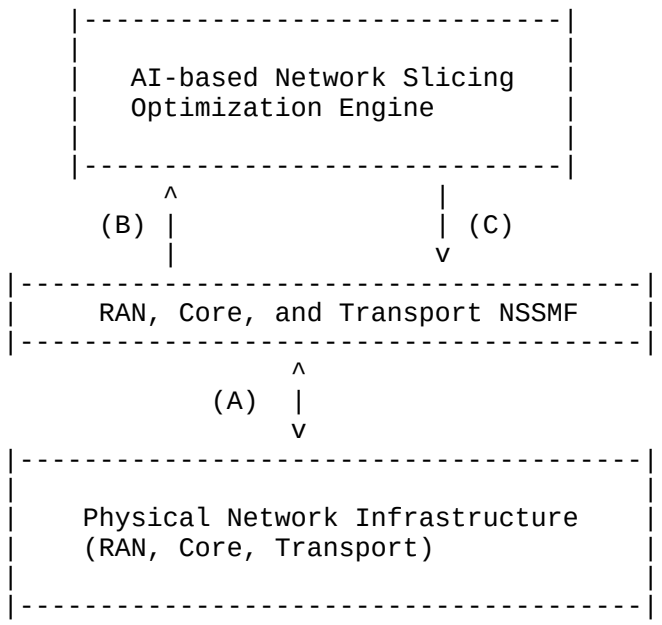
6.21. AI-Driven Network Slicing Optimization

This use case leverages AI to optimize the creation, management, and performance of network slices in 5G and beyond. By dynamically allocating resources, predicting SLA violations, and coordinating across multiple domains, AI ensures that each slice meets its performance requirements while efficiently utilizing the underlying physical infrastructure. This use case addresses both single-domain (e.g., RAN) and multi-domain (e.g., RAN, transport, core) scenarios.

The AI system analyzes real-time telemetry (e.g., traffic patterns, resource utilization), historical data, and external inputs (e.g., service requirements, network topology) to optimize network slices. For example, AI can allocate additional bandwidth to a slice experiencing high traffic or reroute traffic to prevent congestion. If SLA violations are predicted, AI can take proactive measures (e.g., scaling resources, adjusting configurations) to ensure compliance.

* Architecture

The architecture integrates an AI Network Slicing Optimization Engine with the NSMF (Network Slice Management Function) and NSSMFs (Network Slice Subnet Management Functions) for RAN, Core, and Transport domains. Figure 17 illustrates this design, showing the AI engine interfacing with the NSMF, which coordinates with the NSSMFs to manage and optimize network slices. The engine collects telemetry, predicts SLA violations, and provides optimization recommendations to the NSMF, which implements changes via the NSSMFs.



Legend:
(A) Optimization commands (e.g., adjust slice resources, reroute traffic)
(B) Telemetry feedback (e.g., slice performance, resource utilization)
(C) SLA compliance reports and alerts
NSSMF: Network Slice Subnet Management Function (RAN, Core, Transport)

Figure 17: Corrected Architecture for AI-Driven Network Slicing Optimization

* Interfaces and APIs

The AI engine interfaces with the NSMF using standard management and telemetry APIs. The NSMF communicates with the NSSMFs using 3GPP- defined interfaces (e.g., Nsmf_PDUSession_Create, Nsmf_EventExposure_Subscribe). The AI engine collects telemetry via gNMI or OpenConfig streams and provides optimization recommendations to the NSMF via REST APIs.

* Protocols

The protocols align with 3GPP and IETF standards. The NSMF and NSSMFs use 3GPP-defined protocols (e.g., HTTP/2 for service-based interfaces). For transport, protocols like PCEP (RFC 5440) and BGP (RFC 4271) support traffic engineering adjustments. Telemetry protocols like IPFIX (RFC 7011) or SNMP (RFC 3411) provide feedback, with streaming alternatives (e.g., gNMI) preferred for real-time needs.

* Data Models

YANG models are central to representing slice configurations and performance data. The Network Topology Model (RFC 8345) can be extended to define slice parameters (e.g., latency, bandwidth) and metrics (e.g., resource utilization, SLA compliance). 3GPP YANG models for RAN, Core, and Transport support configuration and telemetry. A new YANG model may standardize network slicing optimization workflows.

* Processes and Procedures

The process begins with AI training on historical traffic data, slice configurations, and SLA requirements. Optimization runs continuously, with operator oversight via an AIOps-Assistant (similar to 6.8). Post-optimization, the AI evaluates slice performance, updates configurations, and retrains as needed. Closed-loop automation applies corrective actions, validated by telemetry.

* Alignment with IETF and 3GPP

This use case aligns with 3GPP efforts in network slicing and IETF efforts in network management and traffic engineering. The Operations and Management Area Working Group (OPSAWG) supports telemetry for slice performance metrics, while the Traffic Engineering Architecture and Signaling (TEAS) working group’s work on path optimization (e.g., RFC 8453 for ACTN) enables slice-aware routing. Extensions to YANG (NETMOD), PCEP (PCE), and telemetry standards (OPSAWG) could standardize this use case, leveraging frameworks proposed in Section 6.17 and Section 6.18.

6.22. Other Use Cases

To be discussed and agreed.

* Architecture

To be added.

* Interfaces and APIs

To be added.

* Protocols

To be added.

* Data Models

To be added.

* Alignment with IETF

To be added.

7. Security Considerations

To be discussed in future versions of this document.

8. 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>.

[RFC4655] Farrel, A., Vasseur, J.-P., and J. Ash, "A Path Computation Element (PCE)-Based Architecture", RFC 4655, DOI 10.17487/RFC4655, August 2006, <https://www.rfc-editor.org/rfc/rfc4655>.

[RFC5557] Lee, Y., Le Roux, JL., King, D., and E. Oki, "Path Computation Element Communication Protocol (PCEP) Requirements and Protocol Extensions in Support of Global Concurrent Optimization", RFC 5557, DOI 10.17487/RFC5557, July 2009, <https://www.rfc-editor.org/rfc/rfc5557>.

[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>.

[RFC6805] King, D., Ed. and A. Farrel, Ed., "The Application of the Path Computation Element Architecture to the Determination of a Sequence of Domains in MPLS and GMPLS", RFC 6805, DOI 10.17487/RFC6805, November 2012, <https://www.rfc-editor.org/rfc/rfc6805>.

[RFC7012] Claise, B., Ed. and B. Trammell, Ed., "Information Model for IP Flow Information Export (IPFIX)", RFC 7012, DOI 10.17487/RFC7012, September 2013, <https://www.rfc-editor.org/rfc/rfc7012>.

[RFC7252] Shelby, Z., Hartke, K., and C. Bormann, "The Constrained Application Protocol (CoAP)", RFC 7252, DOI 10.17487/RFC7252, June 2014, <https://www.rfc-editor.org/rfc/rfc7252>.

[RFC7303] Thompson, H. and C. Lilley, "XML Media Types", RFC 7303, DOI 10.17487/RFC7303, July 2014, <https://www.rfc-editor.org/rfc/rfc7303>.

[RFC7540] Belshe, M., Peon, R., and M. Thomson, Ed., "Hypertext Transfer Protocol Version 2 (HTTP/2)", RFC 7540, DOI 10.17487/RFC7540, May 2015, <https://www.rfc-editor.org/rfc/rfc7540>.

[RFC7950] Bjorklund, M., Ed., "The YANG 1.1 Data Modeling Language", RFC 7950, DOI 10.17487/RFC7950, August 2016, <https://www.rfc-editor.org/rfc/rfc7950>.

[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>.

[RFC8051] Zhang, X., Ed. and I. Minei, Ed., "Applicability of a Stateful Path Computation Element (PCE)", RFC 8051, DOI 10.17487/RFC8051, January 2017, <https://www.rfc-editor.org/rfc/rfc8051>.

[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>.

[RFC8231] Crabbe, E., Minei, I., Medved, J., and R. Varga, "Path Computation Element Communication Protocol (PCEP) Extensions for Stateful PCE", RFC 8231, DOI 10.17487/RFC8231, September 2017, <https://www.rfc-editor.org/rfc/rfc8231>.

[RFC8259] Bray, T., Ed., "The JavaScript Object Notation (JSON) Data Interchange Format", STD 90, RFC 8259, DOI 10.17487/RFC8259, December 2017, <https://www.rfc-editor.org/rfc/rfc8259>.

[RFC8329] Lopez, D., Lopez, E., Dunbar, L., Strassner, J., and R. Kumar, "Framework for Interface to Network Security Functions", RFC 8329, DOI 10.17487/RFC8329, February 2018, <https://www.rfc-editor.org/rfc/rfc8329>.

[RFC8345] Clemm, A., Medved, J., Varga, R., Bahadur, N., Ananthakrishnan, H., and X. Liu, "A YANG Data Model for Network Topologies", RFC 8345, DOI 10.17487/RFC8345, March 2018, <https://www.rfc-editor.org/rfc/rfc8345>.

[RFC8446] Rescorla, E., "The Transport Layer Security (TLS) Protocol Version 1.3", RFC 8446, DOI 10.17487/RFC8446, August 2018, <https://www.rfc-editor.org/rfc/rfc8446>.

[RFC8453] Ceccarelli, D., Ed. and Y. Lee, Ed., "Framework for Abstraction and Control of TE Networks (ACTN)", RFC 8453, DOI 10.17487/RFC8453, August 2018, <https://www.rfc-editor.org/rfc/rfc8453>.

[RFC8811] Mortensen, A., Ed., Reddy.K, T., Ed., Andreasen, F., Teague, N., and R. Compton, "DDoS Open Threat Signaling (DOTS) Architecture", RFC 8811, DOI 10.17487/RFC8811, August 2020, <https://www.rfc-editor.org/rfc/rfc8811>.

Appendix A. IANA Considerations

This document has no IANA actions.

Acknowledgments

This work has benefited from several discussions at the IETF and the AI4NETWORK Side Meetings.

Contributors

Daniele Ceccarelli
Cisco

Email: dceccare@cisco.com

Arashmid Aakhavain
Huawei
Email: arashmid.akhavain@huawei.com

Oscar Gonzlez de Dios
Telefonica
Email: oscar.gonzalezdedios@telefonica.com

Ignacio Dominguez Martinez-Casanueva
Telefonica
Email: ignacio.dominguezmartinez@telefonica.com

Vincenzo Riccobene
Huawei
Email: vincenzo.riccobene@huawei-partners.com

Nathalie Romo-moreno
Telekom
Email: nathalie.romo-moreno@telekom.de

Ali Tizghadam
Telus
Email: ali.tizghadam@telus.com

Authors' Addresses

Reza Rokui
Ciena
Email: rrokui@ciena.com

Cheng Li
Huawei
Email: c.l@huawei.com

Daniel King
Lancaster University
Email: d.king@lancaster.ac.uk