

LF NETWORKING



Architecting Autonomy: The Convergence of Agentic AI and Open Source Networking

Published by LF Networking

Authors

Arpit Joshipura - The Linux Foundation

Sana Tariq - Telus

Ranny Haiby - The Linux Foundation

A Linux Foundation Networking publication

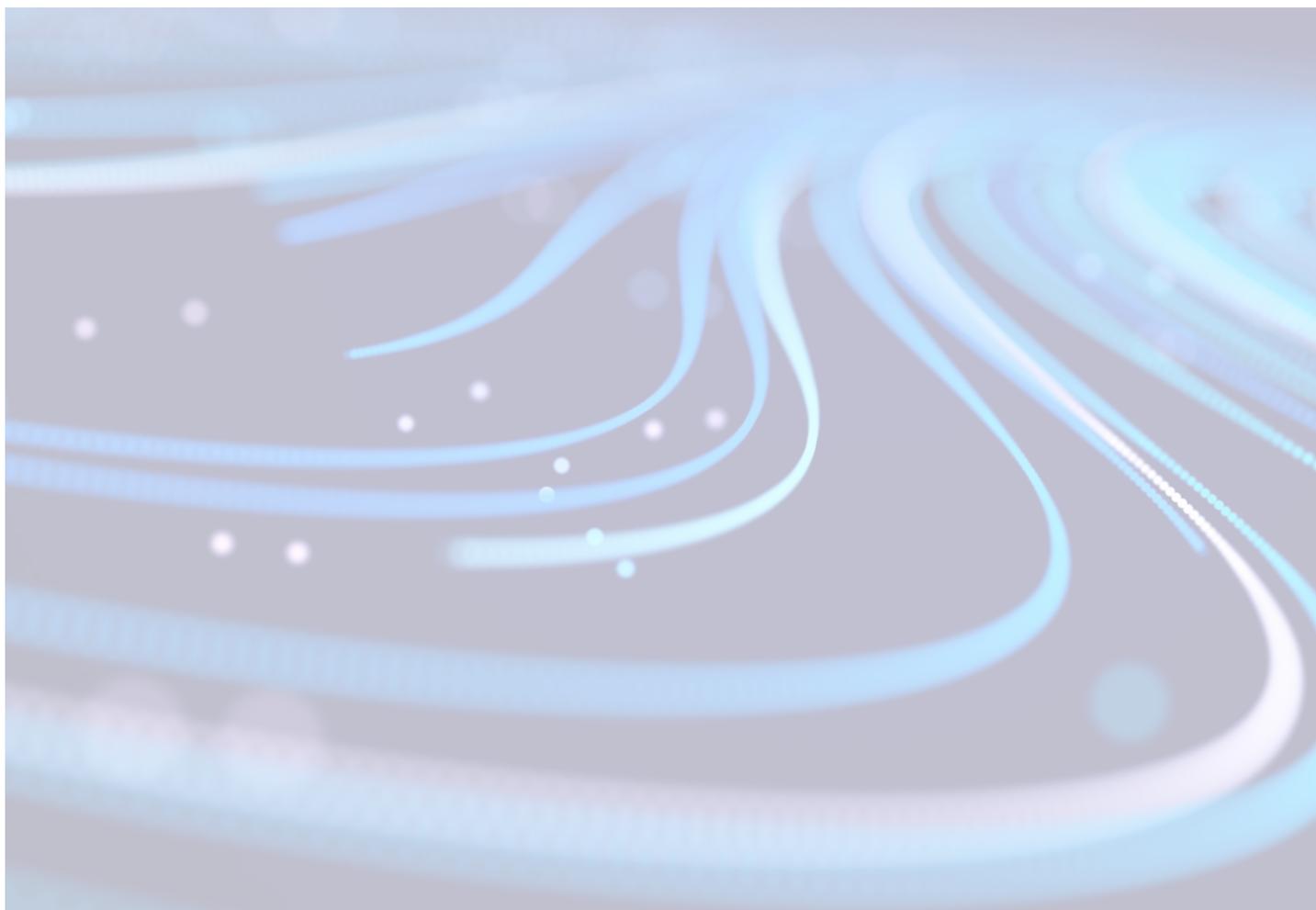
Table of Contents

| | |
|--|-----------|
| Executive Summary | 3 |
| Introduction to the High Level Architecture enabling Agentic AI | 4 |
| Why should Agentic AI be applied to open networking? | 5 |
| Integrating AI Agents and Network APIs | 7 |
| AI Agents leveraging Intent-based Networks | 7 |
| How does Agentic AI in networking work? | 8 |
| Standards for Network Orchestration are Solidifying | 8 |
| Project adoption is boosted through agentic discoverability | 9 |
| Agentic workflows across project boundaries | 9 |
| Architecture Alignment for AI-Centric Networking Frameworks | 9 |
| Practical Use Cases Enabled by AAIF + LFN | 10 |
| The path forward: The Agentic AI Readiness Charter | 11 |
| LF Networking Agentic AI Readiness Principles | 11 |
| Conclusion: The Convergence of Agentic AI and Open Networking | 13 |

Executive Summary

As the industry pivots toward autonomous operations, Agentic AI represents the next frontier in networking. It creates an architectural disruption, transcending the boundaries of legacy frameworks such as the three-tier enterprise application model and the 7-layer OSI standard.. This whitepaper establishes a comprehensive framework for this transition, beginning with a detailed delineation of the Agentic AI technology stack - from underlying LLM foundations to the orchestration layers and specialized agent frameworks. We highlight the indispensable role of open-source ecosystems, which serve as the primary engine for innovation, providing the standardized building blocks necessary to develop, deploy, and scale agentic behaviors across diverse infrastructure environments.

The core of our analysis explores the symbiotic relationship between AI and connectivity through two lenses: “AI for Networks,” where agents optimize performance and self-heal infrastructure, and “Networks for AI,” where fabric design is tailored to meet the extreme demands of distributed training and inference. Through real-world use cases, we demonstrate how established open-source networking projects are evolving to integrate agentic capabilities, offering a technical roadmap for this transformation. To guide future development, the paper proposes ten architectural principles for building agentic-ready network applications and concludes with a call to action for community participation, ensuring that the future of intelligent networking remains open, interoperable, and resilient.



Introduction to the High Level Architecture enabling Agentic AI

The creation of the Agentic AI Foundation (AAIF) under the Linux Foundation creates a timely opportunity to prepare organizations using LF Networking (LFN) projects for both “AI for Networks” and “Networks for AI,” establishing a much-needed synergy to evolve the LFN ecosystem in response to critical industry demands.

Agentic AI Foundation provides LF Networking with a shared, open framework for building autonomous, goal-driven AI agents that can reason, act, and collaborate across complex network environments. By standardizing agent capabilities such as intent handling, policy enforcement, observability, and secure tool access, the foundation accelerates innovation across LF Networking projects while avoiding fragmentation. This enables the LF Networking ecosystem to embed AI-driven automation, closed-loop control, and cross-domain intelligence consistently across open networking stacks—from design and deployment to operations and optimization.

The transition toward agent-centric AI represents a fundamental architectural disruption, transcending the boundaries of legacy frameworks such as the three-tier enterprise application model and the 7-layer OSI standard. To maintain competitive relevance, enterprise and network architects must pivot toward machine-native interfaces, recognizing autonomous agents as first-class citizens within the digital ecosystem. As new paradigms for agentic orchestration emerge at an unprecedented velocity, open-source ecosystems have become the primary engine for industry-wide synchronization. The following diagram delineates the Agentic AI technology stack and the open-source initiatives catalyzing its development.

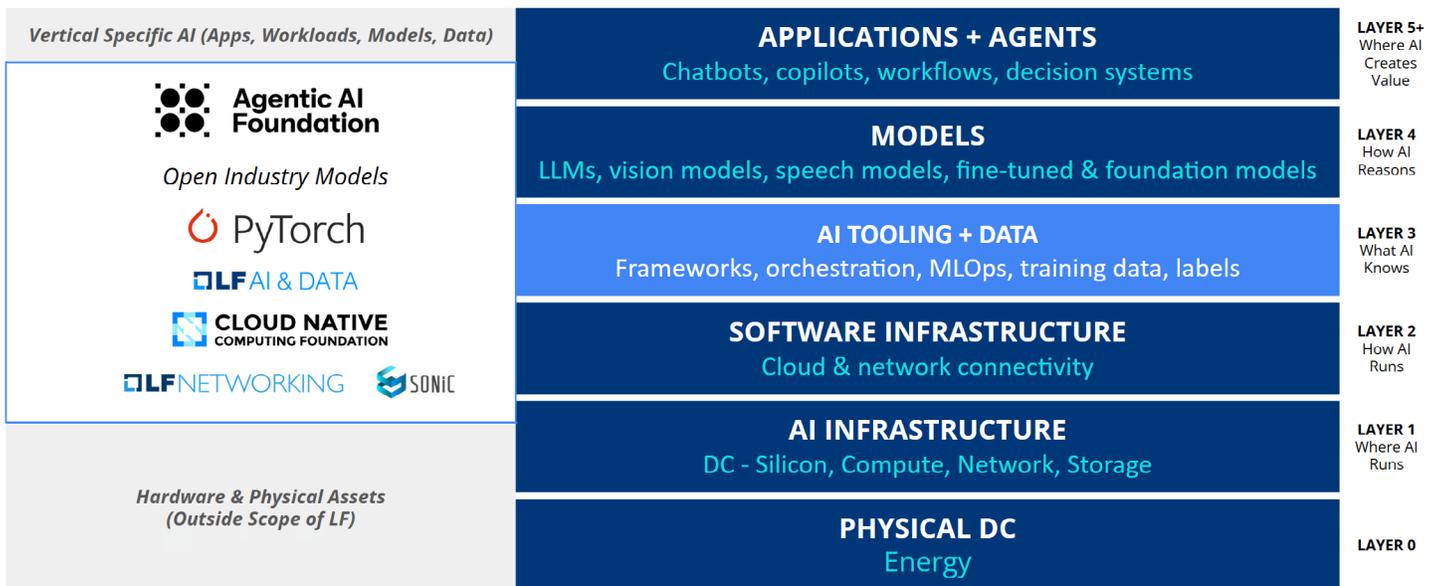


Fig 1. The Layers of modern agentic AI

Why should Agentic AI be applied to open networking?

Networks are rapidly shifting toward cloud-native, intent-driven, and AI-assisted operations. The Linux Foundation's recent launch of the Agentic AI Foundation (AAIF) signals a major inflection point for how networks and AI systems will evolve together. For years, telecom and cloud networking communities have been building toward more autonomous, intent-driven operations, but progress has often been slowed by fragmentation: different APIs, incompatible automation frameworks, proprietary agent models, and inconsistent governance across AI technologies.

AAIF changes that landscape by providing a unifying, vendor-neutral home for the core standards that will define how AI agents interact with tools, data, and infrastructure. This gives the open source networking ecosystem (spanning LFN, CAMARA, LF AI & Data, and emerging AI-native platforms like Essedum) a shared architectural foundation to build upon.

With MCP, AGENTS.md, and related initiatives stabilized under the Linux Foundation, networking projects can begin to align around a common integration model. Orchestrators such as Nephio and ONAP gain a predictable way to expose intent, configuration, and lifecycle management operations to AI agents. CAMARA APIs become first-class "tools" that agents can safely invoke to deliver customer-facing capabilities. And AI-centric frameworks like Essedum can build their control loops and inference models directly on top of standardized agent interfaces, rather than bespoke connectors.

The result is a more cohesive and extensible ecosystem, one where AI systems and networks don't just coexist but collaborate. Operators benefit from interoperability instead of lock-in, vendors innovate on stable ground rather than shifting interfaces, and developers gain a consistent, intuitive way to build applications that leverage both AI intelligence and carrier-grade network capabilities.

AI agents are rapidly becoming the operational backbone of organizations. Orchestrating these agents through MCP, and standardizing on a shared agentic framework across LF Networking projects, is therefore a must-have for enabling the next chapter of AI-native networking.

AI for Networks

- AI agents optimize, operate, and automate networks
- Focus: intent translation, assurance, remediation, lifecycle automation

Networks for AI

- Networks adapt to AI workloads (training, inference, edge AI)
- Focus: GPU fabrics, edge discovery, data gravity, SLA-aware connectivity

Agentic AI is the missing control plane that allows these two transformations to reinforce each other.

Here is an overview of some of the Agentic AI functionality provided by LF Networking projects:

| | Project | Current Focus | AI for Networks (Enabled by AAIF) | Networks for AI (Enabled by AAIF) |
|---|----------------|---|---|---|
| LFN Projects | Nephio | Kubernetes-native intent-based network orchestration | MCP-exposed intent, lifecycle, and policy interfaces enabling agent-driven closed-loop automation and remediation | AI workload-aware cluster and network orchestration optimized for distributed training and inference |
| | ONAP | Service orchestration and policy management | Agents reason over service models and policies to automate assurance, healing, and lifecycle operations | SLA-driven service chaining and orchestration for AI-native services |
| | Essedum | Networking Agentic AI Framework | Agent-assisted capacity planning, fault isolation, and optimization across domains | AI fabric optimization, high-bandwidth low-latency paths for distributed AI workloads |
| Other LF Open Source Networking Projects | Sylva | Telco cloud reference architecture and lifecycle | AI-assisted infrastructure provisioning, compliance validation, and lifecycle optimization | Accelerator-aware (GPU/AI) cloud profiles and optimized underlay networking |
| | CAMARA | Standardized exposure of telco network capabilities | Network capabilities exposed as agent-callable tools via MCP for intelligent service automation | Dynamic edge discovery, QoS, slicing, and latency-aware connectivity for AI applications |
| | SONiC | Distributed networking fabric for data centers, edge and enterprise | Self-healing fabrics, predictive maintenance and anomaly detection | Scalable GPU Fabrics using RoCEv2 and Near-Zero Tail Latency using Adaptive Routing and Packet Spraying |

Integrating AI Agents and Network APIs

LFN's sister project, CAMARA, already standardizes Open Gateway APIs that expose telco capabilities (QoS, location, slicing, edge discovery, etc.). With MCP, AI agents can now understand, call, and orchestrate these APIs programmatically which leads to unlocking of intelligent service exposure, automated provisioning, and cross-network coordination.

CAMARA's [white paper](#), "In Concert: Bridging AI Systems and Network Infrastructure through MCP" explores this relationship more deeply and investigates:

How MCP can serve as the "translation layer" between AI agents and CAMARA's API ecosystem. Architectural models for safe agentic access to network capabilities.

Concrete use cases such as dynamic QoS, automated SLA management, and multi-cloud edge application provisioning.

How operators can achieve secure, auditable workflows even as AI takes on active operational roles.

Most importantly, this alignment accelerates the industry's progress toward AI-native, self-managing networks. By harmonizing standards across foundations and domains, the Linux Foundation is enabling the telecom ecosystem to move faster, integrate more safely, and explore sophisticated automation use cases that were previously too complex or siloed to achieve.

AI Agents leveraging Intent-based Networks

Intent-based, declarative automation has long been a foundational principle across LF Networking projects such as Nephio, ONAP, and the broader GitOps-driven ecosystem, enabling operators to describe what the network should achieve rather than how it should be configured. Agentic AI naturally extends this model by introducing reasoning, goal decomposition, and adaptive decision-making on top of existing intent frameworks. Through MCP, AI agents can interpret high-level business or application intents, map them to CAMARA-exposed network capabilities, and invoke Nephio or ONAP workflows to realize those intents in a controlled and auditable manner. This convergence enables a new class of intelligent network APIs that are not only monetizable, but dynamically optimized and continuously assured. By aligning agentic reasoning with declarative intent systems and GitOps-based execution, LFN projects provide a deterministic, vendor-neutral path for operators to scale automation, safely expose network capabilities, and evolve toward truly AI-native, self-managing networks.

How does Agentic AI in networking work?

The creation of the AAIF solidifies the position of MCP as the de-facto standard for communication between AI agents and external tools. LF Networking hosts several projects that control and orchestrate the network and thus are natural candidates for exposing services to agents through MCP servers.

The AAIF Roadmap for MCP focuses on making it more robust and scalable, which brings it closer to deployment in carrier-grade environments. The roadmap focus area around scalability, server discovery and authentication address many of the challenges that were prohibiting developers from leveraging MCP in production environments.

Standards for Network Orchestration are solidifying (Nephio, ONAP, Sylva-aligned stacks)

With MCP now governed under the LF, orchestration initiatives across the LFN ecosystem gain a dependable, vendor-neutral standard for agent interaction. This directly:

- Boosts confidence for Nephio, ONAP, and related cloud-native orchestration systems to adopt MCP as a de-facto integration standard.
- Reduces fragmentation across operator environments.
- Enables consistent tooling, schema definitions, and automation workflows.

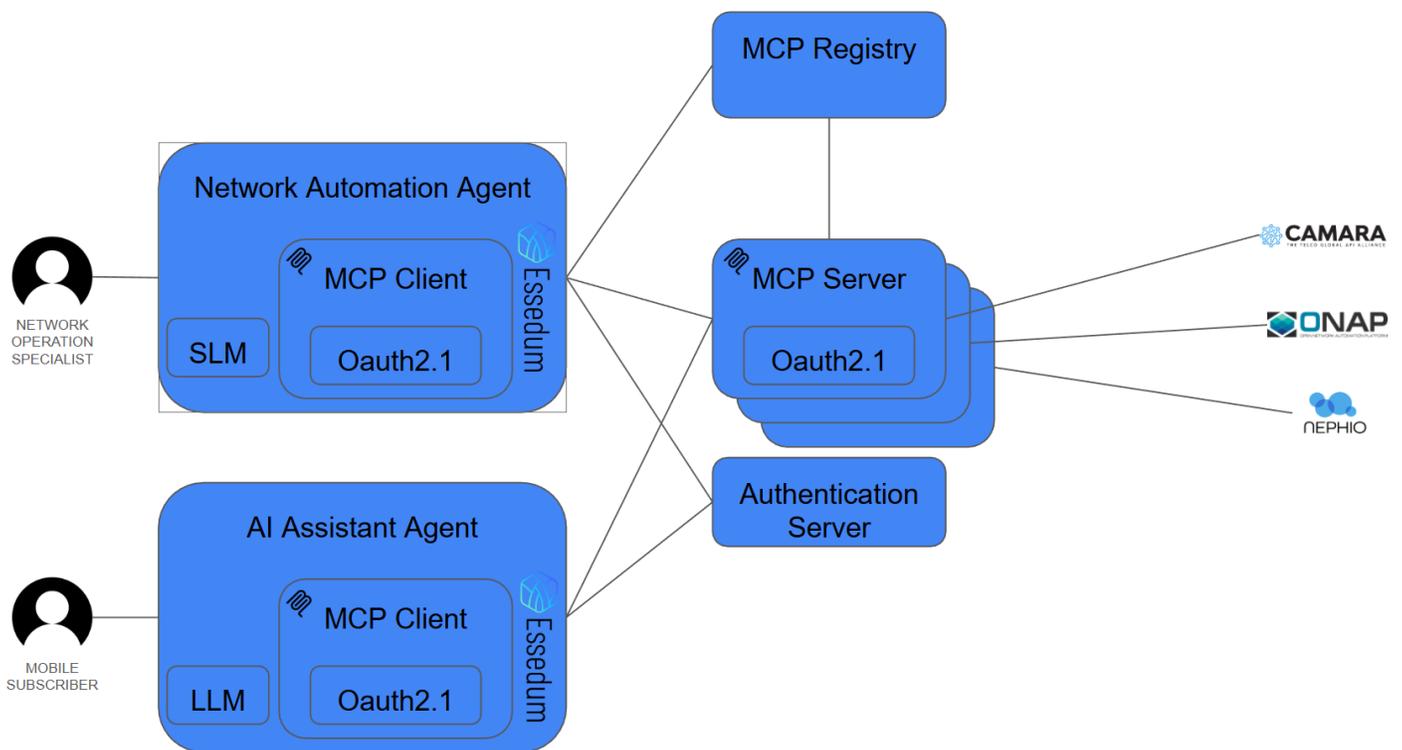


Fig 2. Standard based orchestration architecture

Project adoption is boosted through agentic discoverability

One of the areas where LF Networking projects can benefit from the new direction MCP is taking under the AAIF is increased discoverability. The challenge of getting more end users to adopt the open source technology is decades old. With MCP server registries and discoverability mechanisms coming from the AAIF, LFN projects can now expose a new type of interface to be consumed by AI agents. Since agents do not read project documentation like humans, more machine-consumable exposure is required. By creating MCP servers for the projects and making them discoverable on public registries, projects can ensure their visibility and attractiveness to AI agents, especially AI code generation agents.

Agentic workflows across project boundaries

LFN projects and other networking projects across the LF can now expose MCP-native interfaces, allowing AI agents to:

- Provision network resources end-to-end while responding to dynamic load requirements, resulting in efficient resource utilization and providing a better experience to end users.
- Execute troubleshooting or remediation tasks autonomously (within operator-defined guardrails) reducing operational costs.
- Drive closed-loop automation workflows across diverse network domains (RAN, core, transport, edge).

This shifts AI in networking from offline analytics to actionable automation that reacts in real time and proactively adapts the network for providing optimal experience.

Architecture Alignment for AI-Centric Networking Frameworks (e.g., Essedum)

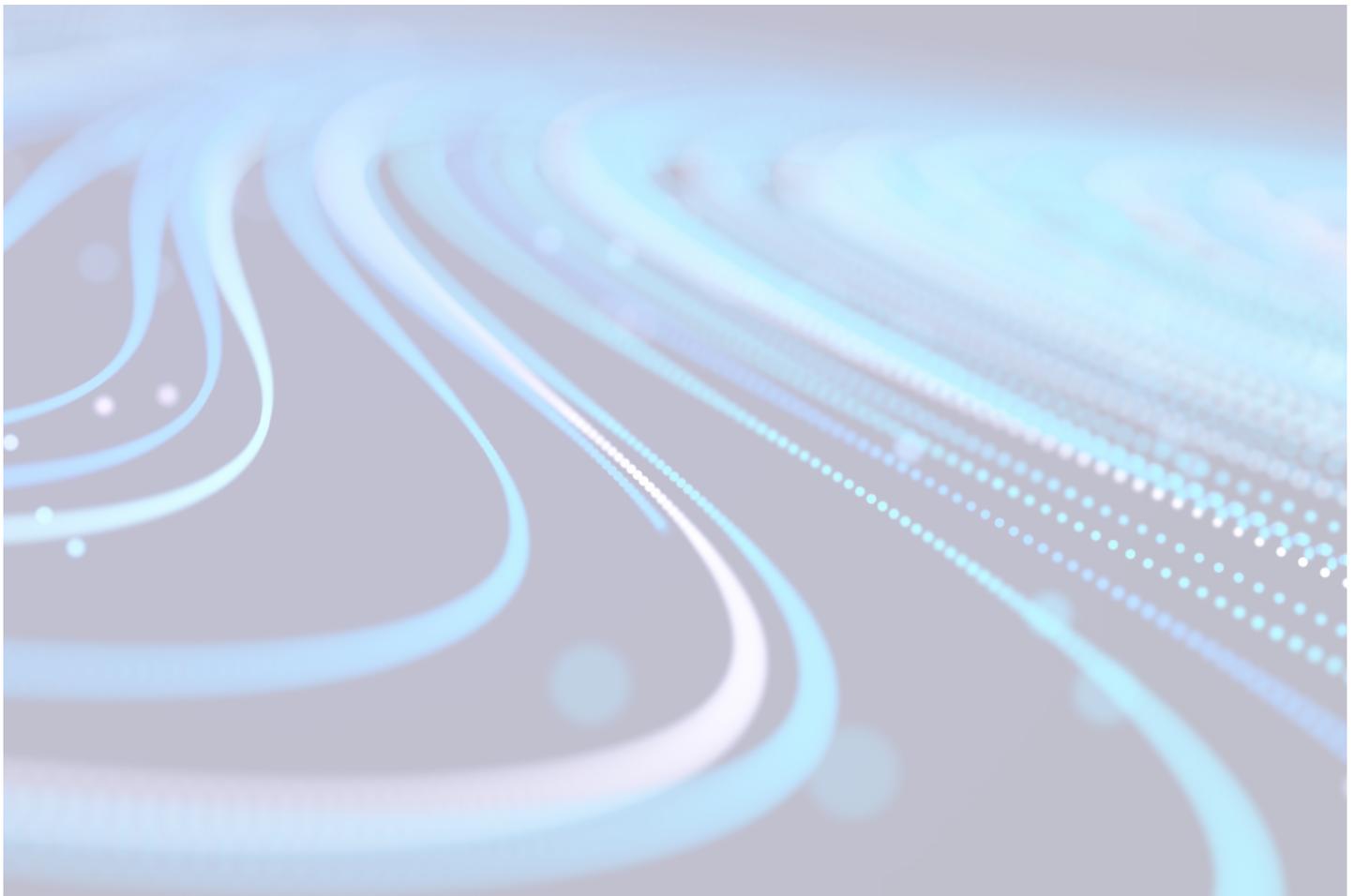
Projects like Essedum, which are explicitly architected for AI-native networking, can now align their models, APIs, and orchestration logic with broader AAIF standards. Benefits include:

- A unified architectural language between AI inference layers and network control layers.
- Easier integration with other AAIF-backed projects and tools.
- Clear interoperability pathways for future agent-framework evolution.

Practical Use Cases Enabled by AAIF + LFN

AAIF gives the industry the connective tissue needed to transform AI's potential into operational reality - and LFN communities are now well positioned to lead this evolution.

- **Closed-loop assurance:** agents detect anomalies and trigger orchestrators. Closed-loop automation must expand into continuous learning loops, allowing agentic systems to improve decisions over time while preserving deterministic execution boundaries.
- **Intent-based networking:** natural language intents mapped to network policies via MCP-connected orchestrators. Intent must become a first-class, machine-reasoned construct, enabling AI agents to interpret objectives, constraints, and trade-offs rather than static configuration artifacts.
- **Multi-domain service provisioning:** agents use CAMARA APIs + Nephio/ONAP workflows.
- **Operational troubleshooting:** agents initiate diagnostics using standardized MCP tool interfaces.
- **Developer experiences:** access to CAMARA APIs is made available to AI agents through a well defined MCP "contract". This allows continuous evolution of the agents to utilize new API capabilities.
- **Network observability** must evolve from passive telemetry collection into semantic, cross-domain intelligence that can be consumed directly by AI agents.
- **Network monetization and exposure APIs** must shift from static request/response models to dynamic, intent-driven services optimized in real time by agentic systems.



The path forward: The Agentic AI Readiness Charter

The establishment of the Agentic AI Foundation (AAIF) under the Linux Foundation, combined with the accelerating adoption of agentic AI systems, creates a clear imperative for the LF Networking (LFN) ecosystem to evolve in support of MCP-based agent orchestration. The following is a set of guiding principles for LFN project leads to adapt their projects to expose and consume capabilities in an agentic AI environment.

LF Networking Agentic AI Readiness Principles

1. Agent-Native Interfaces by Design

Agent-Native operational interfaces

LF Networking projects should expose key capabilities through agent-consumable interfaces enabling safe and predictable interaction through mechanisms such as MCP

Agent-Native design interfaces

Projects should support discovery and development by coding agents through standardized project artifacts (e.g., agents.md, llms.txt), enabling safe contribution and evolution of projects by AI systems alongside human developers.

2. Intent as the Primary Abstraction

Projects should prioritize declarative, intent-based interfaces that describe desired outcomes, constraints, and policies, allowing AI agents to reason over objectives rather than manipulate imperative workflows.

3. Deterministic Execution with Governed Autonomy

While AI agents may reason and plan autonomously, execution within LFN projects must remain deterministic, auditable, and bounded by operator-defined policies and guardrails.

4. Separation of Reasoning and Execution Planes

AI-driven reasoning and decision-making should remain logically separated from control and data plane execution, preserving reliability, safety, and operational clarity.

5. First-Class Observability for Agents

Projects should expose meaningful, structured observability data, beyond raw metrics, so AI agents can understand system state, outcomes, and cause-and-effect relationships.

6. Secure, Auditable Agent Access

All agent interactions must be authenticated, authorized, logged, and traceable, ensuring compliance with carrier-grade security, regulatory, and governance requirements.

7. Composable and Discoverable Capabilities

Project capabilities should be discoverable through standardized schemas and registries, enabling AI agents to dynamically identify, evaluate, and compose network functions.

8. Multi-Domain and Cross-Project Coordination

Projects should assume participation in broader, multi-domain workflows, enabling AI agents to coordinate actions across RAN, core, transport, cloud, and edge components.

9. Backward Compatibility and Incremental Adoption

Agentic AI enablement should be additive, allowing existing users and workflows to continue operating while enabling incremental adoption of MCP and agent-driven capabilities.

10. Open Governance and Vendor Neutrality

All agentic integrations should align with Linux Foundation principles of openness, interoperability, and neutral governance to avoid fragmentation and vendor lock-in.

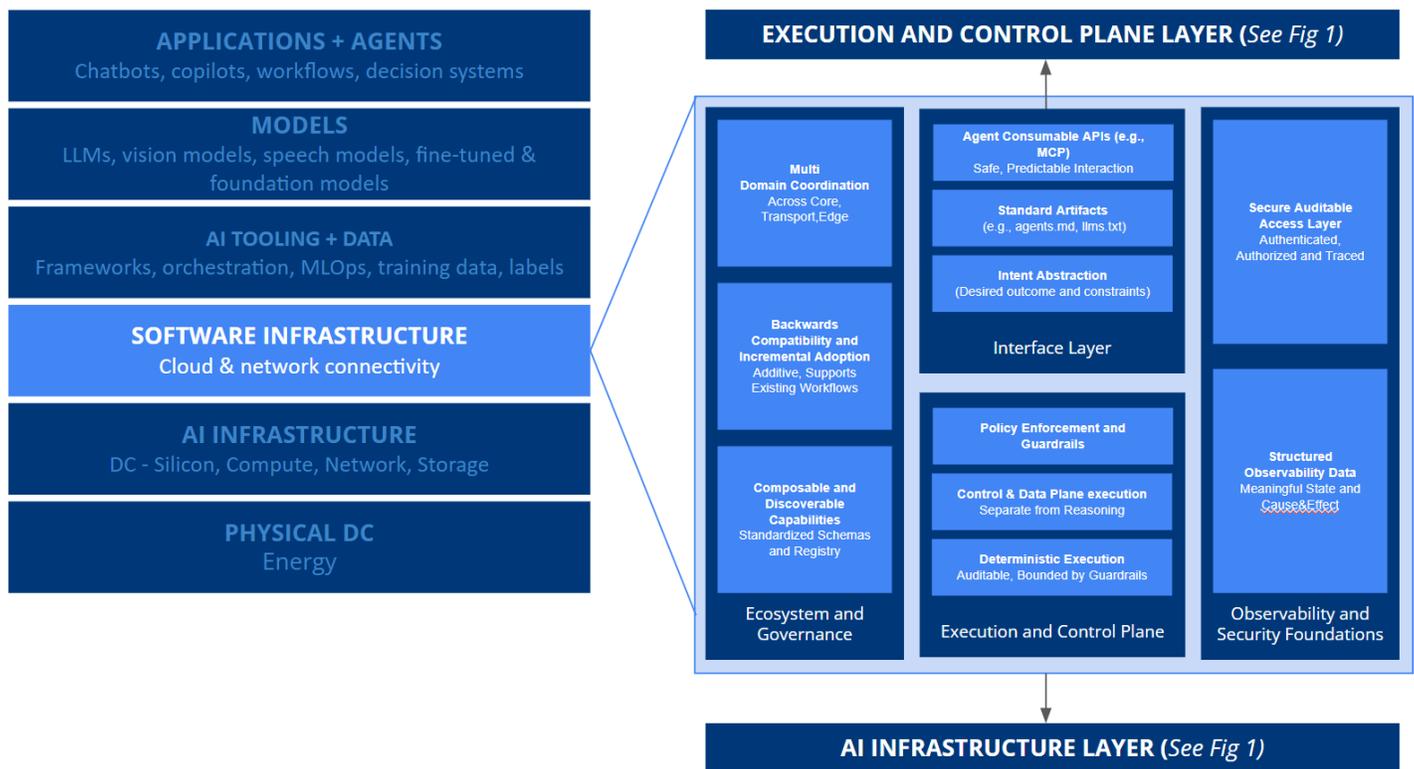


Fig 3. Agentic AI networks architectural principles. This diagram represents the expanded view of the "Software Infrastructure (Cloud & Network Connectivity) layer from Fig 1 on page 4.

Conclusion: The Convergence of Agentic AI and Open Networking

The Agentic AI Foundation (AAIF) establishes a unified, vendor-neutral framework for integrating autonomous, goal-driven agents into the LF Networking ecosystem, bridging the gap between high-level reasoning and carrier-grade infrastructure. By leveraging the Model Context Protocol (MCP) as a standardized “translation layer,” the initiative transforms complex network capabilities, managed by projects like Nephio, ONAP, and CAMARA, into discoverable, agent-callable tools. This creates a powerful synergy: “AI for Networks” (using agents to automate remediation and lifecycle management) and “Networks for AI” (optimizing infrastructure for massive AI workloads). Ultimately, AAIF moves the industry beyond static automation toward a deterministic, AI-native control plane where agents reason over declarative intents to deliver self-managing, interoperable networks. By adopting the Agentic AI readiness principles, networking open source projects secure their position in the forefront of network innovation.

Looking ahead, AAIF and LF Networking lay the groundwork for an entirely new wave of open-source innovation with agent-native control planes, intent intelligence, semantic observability, and AI-aware networks. This enables operators, vendors, and developers to co-create capabilities that will shape how autonomous networks are built, governed, and scaled in the years ahead. The velocity of innovation in agentic AI leaves no room for silos. Staying ahead requires more than just internal agility. It demands collective intelligence. Open-source collaboration provides the essential engine for this journey, turning fragmented efforts into robust, industry-wide methodologies. By contributing to these networking and AI projects today, you aren't just adopting standards - you are authoring them. Let's build the foundation together.