DMSC Working Group
Internet-Draft
Intended status: Standards Track
Expires: 19 July 2026

X. Li
A. Wang
China Telecom

15 January 2026

Dynamic Multi-agents Secured Collaboration Infrastructure architecture
draft-li-dmsc-inf-architecture-02

Abstract

This document presents an architectural framework for dynamic multi-agent collaboration from an infrastructure perspective. It outlines the network requirements introduced by large-scale agent collaboration, and proposes a systematic approach to enabling Dynamic Multi-agent Secured Collaboration (DMSC) through infrastructure capabilities. The architecture focuses on how network control and forwarding functions can actively participate in agent collaboration.

Table of Contents

1. Introduction

Intelligent agents have evolved rapidly in recent years, driven by advances in artificial intelligence models, computing platforms, and network connectivity. Early forms of agents were typically embedded within isolated systems and designed to perform narrowly defined tasks under predefined conditions. Their interactions with external entities were limited and often mediated by tightly coupled application logic [IoA].

With the increasing availability of large-scale AI models, edge computing resources, and programmable network infrastructures, agents are becoming more autonomous, adaptive, and capable of operating across distributed environments. Modern agents can perceive changes in their environment, make decisions based on local or shared information, and interact with other agents and tools in order to achieve complex objectives. These interactions are no longer confined to static configurations or single administrative domains, but increasingly span devices, networks, and application platforms.

As agents continue to proliferate, they are forming large-scale collaborative systems in which multiple agents dynamically discover each other, exchange information, and coordinate actions. Such systems exhibit highly dynamic behavior, including frequent changes in agent population, roles, and interaction patterns. The resulting agent ecosystems resemble an open, interconnected environment rather than a collection of isolated applications.

The evolution toward large-scale, dynamic agent ecosystems introduces new challenges for the underlying network infrastructure. While agents are capable of sophisticated reasoning and decision-making, their ability to collaborate effectively depends on the availability of common, scalable, and interoperable networking support.

This document focuses on the architectural aspects of enabling dynamic multi-agent collaboration from a network and infrastructure perspective. It examines how network control and forwarding functions can be extended to recognize agents as first-class entities and provide generic support for agent identification, discovery, semantic-aware communication, and coordination. The architecture is intended to support a wide range of agent types, including on-device agents, network-resident agents, and third-party agents, without imposing assumptions about their internal implementation.

The scope of this document is limited to architectural concepts and functional building blocks. It does not define specific protocols, data models, or security mechanisms, nor does it prescribe particular deployment scenarios or application workflows. Instead, it provides a foundational framework upon which more detailed specifications, including protocol designs and security architectures, can be developed in subsequent documents.

2. Conventions used in this document

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC2119] .

3. Terminology

The following terms are defined in this draft:

* DMSC: Dynamic Multi-agent Secured Collaboration. The framework and infrastructure enabling secure and efficient collaboration among dynamic agents.

* Agent: An autonomous software entity capable of perception, planning, decision-making, and execution.

* SemR: Semantic Routing. The process of routing an Agent request based on the meaning or intent of the request, rather than solely on a pre-defined address or identifier.

4. Network Requirements

The proliferation of intelligent agents fundamentally reshapes interaction patterns and control dynamics in future networks. Agent interactions are typically short-lived, context-dependent, and driven by task semantics rather than static endpoints. Moreover, agents may dynamically join or leave collaborative groups, migrate across administrative domains, or change roles over time. These characteristics introduce new requirements for network infrastructures, including agent-level identity management, capability-aware communication, scalable registration and discovery, cross-domain collaboration support, and adaptive routing, as also reflected in [draft-yu-ai-agent-use-cases-in-6g].[usecase]

Collectively, these requirements indicate that future networks must go beyond passive connectivity and actively support dynamic multi-agent collaboration. The core idea of Dynamic Multi-agent Secured Collaboration (DMSC) is to elevate key collaboration-related functions into the network infrastructure. Instead of embedding all coordination logic within applications or agent frameworks, DMSC leverages infrastructure-level capabilities exposed through control-plane and forwarding-plane functions. This approach enables the network to recognize agents as first-class entities, maintain high-level collaboration context, and make informed decisions on discovery, routing, and coordination support in a scalable and interoperable manner.

5. DMSC Infrastructure Architecture

5.1. DMSC Infrastructure Architecture

Figure 1 illustrates the overall architecture for dynamic multi-agent collaboration from an infrastructure-centric perspective. The architecture positions the network infrastructure as an active participant in agent collaboration, while preserving the autonomy and task-level reasoning of individual agents. In this architecture, the network does not execute agent logic or interpret task semantics. Instead, it provides generic support functions that enable agents to collaborate more efficiently and reliably. Agents remain autonomous, while the network supplies shared infrastructure capabilities.

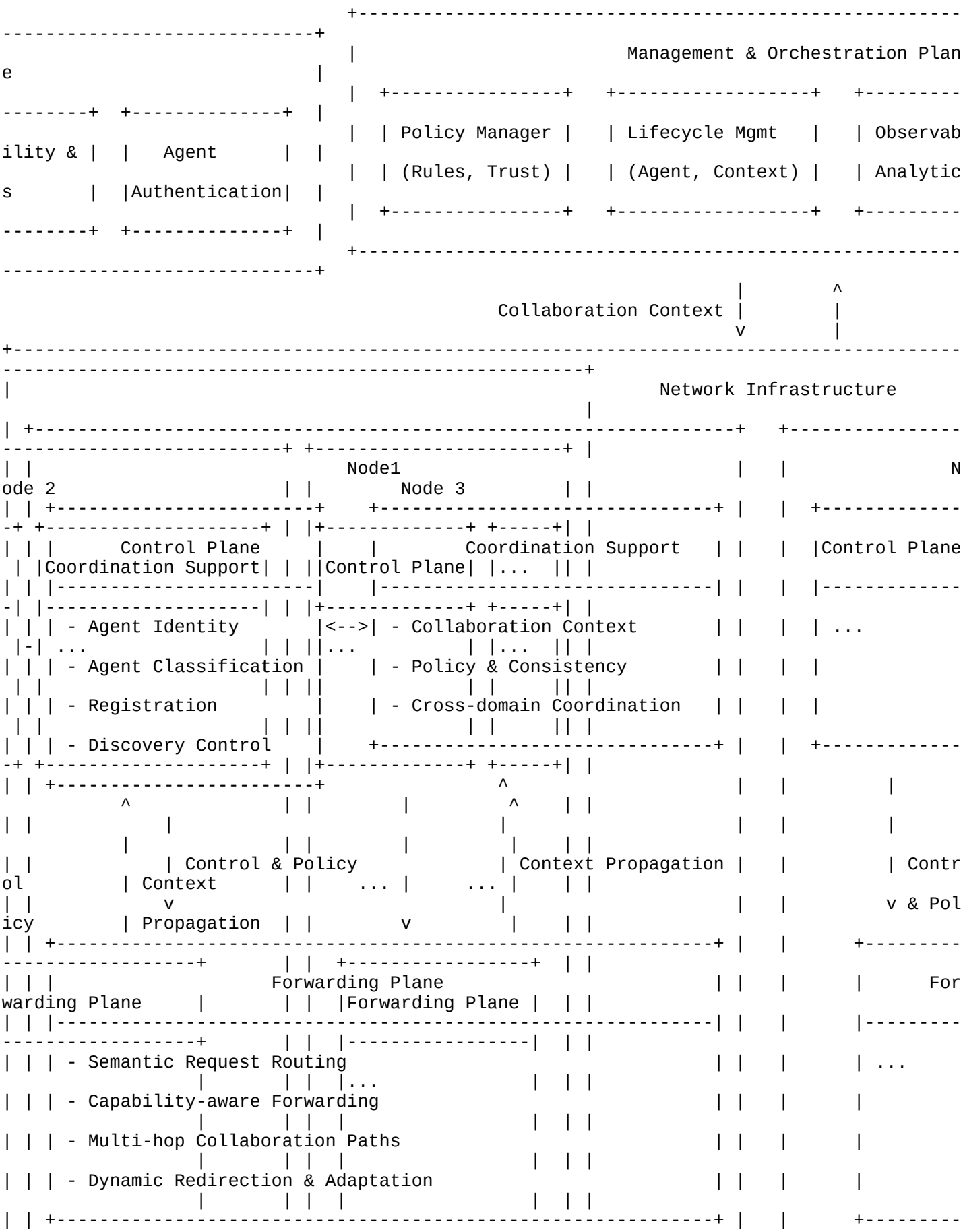
From an infrastructure perspective, the architecture is organized into three logical layers:

* Management Plane: governs policies, trust, lifecycle and authentication aspects.

* Control Plane: Manages agent identity, discovery, policies, and collaboration context.

* Forwarding Plane: Supports semantic-aware routing and data forwarding for agent interactions.

* Coordination Support Functions: Provide higher-level abstractions that bridge agent collaboration and network operation.


Management & Orchestration Plane
Policy Manager (Rules, Trust)
Lifecycle Mgmt (Agent, Context)
Observability & Analytics
Agent Authentication
Collaboration Context
Network Infrastructure
Node1
Node 2
Node 3
Node N
Control Plane
Coordination Support
Coordination Support Control Plane
Control Plane
- Agent Identity
- Agent Classification
- Registration
- Discovery Control
- Collaboration Context
- Policy & Consistency
- Cross-domain Coordination
Control & Policy
Context Propagation
Control & Policy & Context Propagation
Forwarding Plane
Forwarding Plane
- Semantic Request Routing
- Capability-aware Forwarding
- Multi-hop Collaboration Paths
- Dynamic Redirection & Adaptation

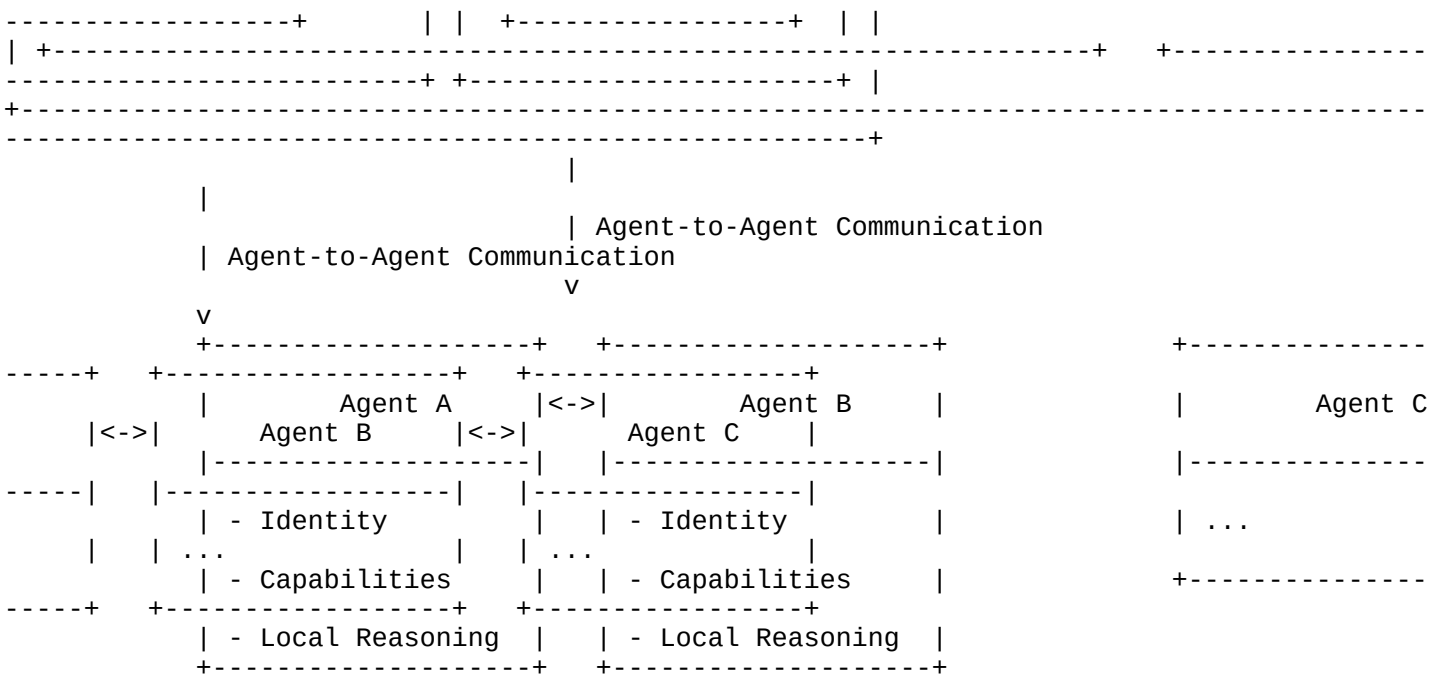


Figure 1 The infrastructure architecture of dynamic multi-agent collaboration

At the top of the architecture, agents engage in collaborative activities driven by task intents, shared goals, and contextual information. Agents are responsible for local reasoning, decision-making, and execution of task-specific logic. The network does not interpret agent semantics or execute agent logic; instead, it provides common infrastructure capabilities that support efficient and scalable collaboration among agents. Above the network infrastructure, a Management and Orchestration Plane provides non-real-time management functions, including policy management, agent and context lifecycle management, observability and analytics, and agent authentication support. This plane supplies policy, trust, and state-related inputs to the network infrastructure.

The network infrastructure itself is composed of multiple network nodes, each implementing a common set of logical functions. Within each node, the Control Plane provides agent-aware control functions, including agent identity management, classification, registration, and discovery control. These functions enable the network to recognize agents as first-class entities and maintain a consistent view of agent-related information across the infrastructure. By decoupling agent identity from physical location, the control plane supports dynamic agent lifecycle events such as mobility, instantiation, and termination.

Complementing the control plane, Coordination Support Functions maintain and propagate collaboration context at an abstract level. This includes information related to collaboration state, policy constraints, and cross-domain consistency. Coordination support functions do not encode task semantics but provide a common substrate for maintaining coherence among dynamic collaboration activities, particularly when agents operate across administrative or network domains.

The Forwarding Plane extends traditional packet forwarding by incorporating semantic-aware decision-making. Instead of relying solely on static addresses, forwarding decisions may consider agent capabilities, collaboration context, and network conditions. This enables semantic request routing, multi-hop collaboration paths, and dynamic redirection when agent availability or network conditions change. Such capabilities are essential for supporting adaptive and resilient agent collaboration at scale.

Agent-to-Agent communication flows traverse the forwarding plane, while control and context information is exchanged through interactions with control-plane and coordination functions. The separation of concerns among agents, control functions, and forwarding functions ensures that agent autonomy is preserved, while the network provides reusable and interoperable support for collaboration.

Overall, this architecture establishes a clear division of responsibilities: agents focus on intelligent behavior and task execution, while the network infrastructure supplies agent-aware control, semantic-aware forwarding, and coordination support. This division enables dynamic multi-agent collaboration to scale across heterogeneous environments and evolve independently of specific agent implementations.

6. Infrastructure Functions Enabling Active Network Participation

6.1. Agent Identification and Classification

In large-scale dynamic multi-agent environments, agents cannot be effectively supported using traditional host- or service-based identifiers alone. Agents may be instantiated dynamically, migrate across network locations, or operate concurrently on the same physical node. As a result, the network requires a mechanism to identify agents as logical entities that are decoupled from network topology.

The proposed architecture introduces network-visible agent identifiers that represent agents independently of their physical location or hosting environment. These identifiers enable the network to consistently recognize agents across control and forwarding functions, even as underlying network bindings change. Beyond basic identification, the architecture supports agent classification based on capabilities, roles, and contextual attributes. Classification information may describe, for example, whether an agent operates on a device, within the network, or as a third-party service, as well as the functional roles it can assume in collaborative processes. Such information is not intended to expose internal agent logic, but to provide sufficient abstraction for network-level decision-making.

6.2. Infrastructure-Level Agent Discovery

Agent discovery is a fundamental prerequisite for collaboration, yet traditional discovery mechanisms are typically designed for relatively static services or tightly scoped environments. In contrast, multi-agent collaboration requires discovery mechanisms that can operate across heterogeneous platforms, adapt to dynamic agent populations, and respect administrative boundaries.

In DMSC architecture, agent discovery is provided as an infrastructure-level function, rather than being entirely implemented within agent frameworks. The network supports discovery queries based on agent identifiers, advertised capabilities, policy constraints, and dynamic state information. This allows agents to locate suitable collaborators without requiring global knowledge or centralized coordination. Discovery mechanisms may differ between intra-domain and inter-domain contexts. Within a domain, discovery may leverage localized registries or control-plane functions for efficiency. Across domains, discovery must account for policy, trust, and information exposure constraints, potentially relying on aggregated or abstracted representations of agent capabilities.

6.3. Semantic Request Routing

Traditional routing mechanisms forward packets based on destination addresses without awareness of application intent or collaboration context. However, in dynamic multi-agent collaboration, interactions are often driven by what is requested rather than where a specific endpoint is located. The DMSC architecture introduces semantic request routing, where requests can be expressed in terms of agent capabilities, roles, or collaboration context. The network forwarding plane may use such semantic information, together with network conditions and policy constraints, as input to routing and forwarding decisions.

Semantic routing enables several advanced behaviors. Requests may be dynamically directed to different agents capable of fulfilling a given role, rather than a fixed endpoint. Multi-hop collaboration paths can be constructed, where intermediate agents contribute partial results. When agent availability or network conditions change, requests can be redirected without requiring agents to reinitiate discovery. Importantly, semantic routing does not require the network to interpret task semantics or agent logic. The network operates on abstracted descriptors and policies, enabling adaptive and resilient collaboration while preserving agent autonomy.

6.4. Secure Collaboration Context Propagation

Effective collaboration among dynamic agents requires shared context, such as session state, coordination constraints, and policy information. When collaboration spans multiple domains or network segments, maintaining consistent context becomes increasingly challenging. The DMSC architecture supports collaboration context propagation at the infrastructure level. Context information associated with a collaboration can be attached to control-plane interactions and, where appropriate, influence forwarding-plane behavior.

This enables the network to maintain coherence across dynamic collaboration activities without requiring agents to explicitly manage all contextual information. Security-related attributes, such as authorization scope or policy constraints, may be bound to collaboration context to ensure that interactions remain consistent with domain-specific requirements. In cross-domain scenarios, context propagation mechanisms support controlled translation or abstraction to maintain interoperability while respecting local policies.

6.5. Operational Visibility

As multi-agent systems scale, the lack of visibility into collaboration-level behavior becomes a significant operational challenge. Traditional network observability focuses on flows or endpoints, offering limited insight into agent interactions and coordination dynamics. The DMSC architecture introduces operational visibility at the collaboration level. Observable entities include agent interactions, coordination relationships, and their association with network resources and conditions. This visibility is not intended to expose agent internals, but to provide sufficient information for monitoring, troubleshooting, and optimization.

Operational visibility enables feedback-driven adaptation. Information collected by the infrastructure can inform control-plane decisions, such as adjusting discovery policies or routing preferences, and forwarding-plane behavior, such as load-aware redirection. Over time, this feedback loop supports continuous optimization of collaboration efficiency and network resource utilization. At the same time, the architecture recognizes that increased visibility introduces potential risks, which are addressed at the architectural level through controlled exposure and policy mechanisms.

7. Conclusion

This document presents an infrastructure-centric architecture for dynamic multi-agent collaboration. By introducing agent-aware abstractions into network control and forwarding functions, the architecture enables scalable discovery, semantic-aware communication, and coordination support without constraining agent autonomy or interpreting agent semantics. The proposed framework defines clear architectural boundaries between agent intelligence and network responsibility, and provides a common foundation for subsequent protocol, security, and deployment-specific specifications that support the evolution of the Internet of Agents.

8. Security Considerations

This architecture introduces several security considerations, including risks related to agent identity spoofing, capability misrepresentation, semantic routing manipulation, cross-domain trust inconsistencies, and information leakage through enhanced observability. Detailed security mechanisms are outside the scope of this document.

9. IANA Considerations

TBD

10. Acknowledgement

TBD

11. Normative References

[IoA] L, J., "Internet of Agents Definition, Architecture and Applications. https://aip.openatom.tech/explore/journalism/detail/501037383572131840", October 2025.

[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/info/rfc2119>.

[usecase] Y, M., "draft-yu-ai-agent-use-cases-in-6g. https://datatracker.ietf.org/doc/html/draft-yu-dmsc-ai-agent-use-cases-in-6g", July 2025.

Authors' Addresses

Xueting Li
China Telecom
Beiqijia Town, Changping District
Beijing
Beijing, 102209
China
Email: lixt2@foxmail.com

Aijun Wang
China Telecom
Beiqijia Town, Changping District
Beijing
Beijing, 102209
China
Email: wangaj3@chinatelecom.cn