

KP-A: a Unified Network Knowledge Plane for Catalyzing Agentic Network Intelligence

Yun Tang, *Member, IEEE*, Mengbang Zou, Zeinab Nezami, *Member, IEEE*,
Syed Ali Raza Zaidi, *Senior Member, IEEE*, Weisi Guo, *Senior Member, IEEE*

Abstract—The emergence of large language models (LLMs) and agentic systems is enabling autonomous 6G networks with advanced intelligence, including self-configuration, self-optimization, and self-healing. However, the current implementation of individual intelligence tasks necessitates isolated knowledge retrieval pipelines, resulting in redundant data flows and inconsistent interpretations. There is now a growing need to support greater multi-tenant interoperability in 6G, especially for Open RAN architectures, and we propose *KP-A*: a unified Network Knowledge Plane specifically designed for Agentic network intelligence. By decoupling network knowledge acquisition and management from intelligence logic, KP-A streamlines knowledge development, reduces maintenance complexity for radio engineers, and enhances interoperability for the network intelligence agents. We build and demonstrate KP-A in 2 representative intelligence tasks: (1) live network knowledge Q&A, and (2) edge-AI service orchestration in RAN. All implementation artifacts have been open-sourced to support reproducibility and future standardization efforts.

Index Terms—6G, LLM, Agent, Knowledge Engineering

I. INTRODUCTION

THE sixth generation (6G) of mobile networks, as envisioned in the IMT-2030 framework, aims to revolutionize wireless communication by offering ultra-high data transmission rates, ultra-low latency, greater connection capacity, and ubiquitous connectivity to enable a wide array of novel applications and services, including immersive communications, integrated sensing and communication, and hyper-reliable low-latency communications [1]. To meet the complex and dynamic service provisioning requirements of these use cases, 6G networks must incorporate advanced self-management capabilities such as self-configuration, self-optimization, and self-healing, which enable networks to autonomously adapt to changing conditions with minimal human intervention [2].

The integration of advanced native Artificial Intelligence (AI), particularly large language models (LLMs) or LLM-powered agentic systems, into network operations is critical to realizing these autonomy capabilities. In recent years, researchers have widely adopted LLMs and LLM-powered agents in various components of the telecom network, such as configuration management, network security, network function code generation, network simulation, user intent profiling [3], sensing [4], and AI service orchestration [5].

Yun Tang, Mengbang Zou, and Weisi Guo are with Cranfield University, UK. Emails: {yun.tang, m.zou, weisi.guo}@cranfield.ac.uk. Zeinab Nezami (z.nezami@leeds.ac.uk) and Syed Ali Raza Zaidi (s.a.zaidi@leeds.ac.uk) are with the University of Leeds, UK. This work is supported by EPSRC CHEDAR: Communications Hub For Empowering Distributed Cloud Computing Applications And Research (EP/X040518/1) (EP/Y037421/1). Corresponding author: Weisi Guo.

A. Current Gaps in Native AI Implementation

However, despite the sound benefits, integrating all the above proposed LLM pipelines and agentic frameworks into one RAN architecture from the network service provider's point of view is impractical for several reasons. One of the primary issues is the lack of a unified knowledge plane, and as a result, network intelligence engineers need to implement additional knowledge retrieval and interpretation mechanisms. As illustrated in Fig. 1, such a fragmented practice leads to redundancy, inconsistent interpretations of network data, and increased development and maintenance overhead. For instance, there can be a scenario where ten LLM-powered intelligence pipelines deploy ten xApps independently but subscribe to the same event from the underlying network infrastructure, resulting in severe redundancy, suboptimal network performance, and, more importantly, inconsistent information interpretations.

B. Proposal for a Unified AI Knowledge Layer

To address these challenges, we propose a unified network Knowledge Plane for Agentic intelligence (KP-A) situated between the underlying Network infrastructure plane and the upper Intelligence Plane. This intermediary layer serves as a unified source of truth, ensuring consistent and intuitive information retrieval protocols and interpretation across intelligent applications. By unifying knowledge management, KP-A reduces redundancy, harmonizes interpretations, and streamlines the development of agentic intelligence frameworks.

Our novel contributions in this paper are summarized as follows:

- We design a KP-A architecture, a unified network knowledge plane to facilitate consistent and intuitive knowledge exploration and retrieval by native AI agents in RAN.
- We implement and demonstrate our design through two representative network intelligence tasks: (i) live network knowledge Q&A and (ii) edge-AI service provisioning.
- We open-source all the source code (O-RAN network simulator with KP-A and intelligence agents) to foster further research and standardization in this field.

II. RELATED WORKS

A. Knowledge Plane and Knowledge Defined Network

The concept of Knowledge Plane (KP) dates back to 2003 by David Clark [6], which initiated the proliferation of Knowledge-defined Networks (KDN) [7]. The essence of KDN is to construct a global and unified perspective on the data and

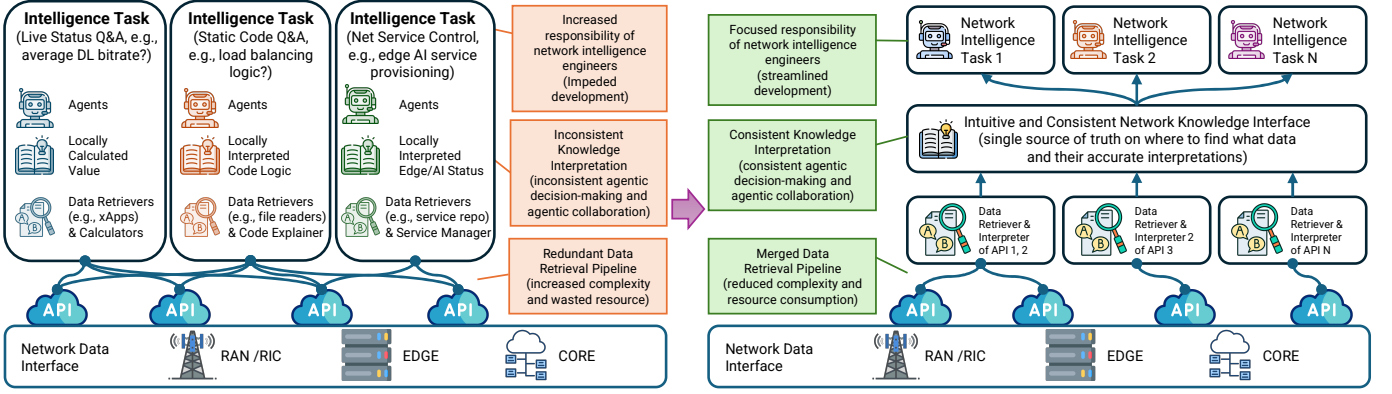


Fig. 1. Comparison of knowledge engineering approaches (left: isolated and inconsistent; right: unified and consistent) for network intelligence tasks (and intelligence engineers) and the envisioned benefits of KP-A.

controls of the underlying physical/software-defined network to enable traditional AI/ML-based network automation. Entering the era of advanced network intelligence powered by LLMs and agents, we adopt the design philosophy of the knowledge plane and propose KP-A, which is specifically designed for agentic network intelligence. These native AI agents are generally divided into several classes of network management:

- Network Data/Event Explainers [8]
- User Intent Translators [9], [10]
- Network Code Generators [11]
- Network Simulation Drivers [12]

For all the application patterns above, having access to external knowledge sources is essential to ensure the trustworthiness and effectiveness of LLMs and agent-based systems, particularly due to the following reasons: (1) new information and knowledge are continuously generated, making static models quickly outdated; (2) LLMs are inherently limited by their training data, and frequent retraining or fine-tuning to incorporate new knowledge is substantially more expensive than in-context learning; and (3) the internal knowledge retrieval process within LLMs, driven by the attention mechanism, is inherently probabilistic; meanwhile external, verifiable knowledge sources can serve as critical ground truth references that guide models towards accurate and reliable responses.

B. Knowledge Engineering for LLMs and Agentic Systems

There are 3 major engineering practices to organize external knowledge based on the nature of the knowledge:

1) Semantic Databases for Static Factual Knowledge:

Semantic databases organize factual knowledge by chunking documents into smaller units, embedding them into semantic vectors (e.g., via OpenAI's *text-embedding-3-small*), and storing them with metadata for efficient retrieval. As shown in the top path of Fig. 3, retrieval combines metadata filtering with semantic matching based on vector similarity, enabling scalable and precise retrieval-augmented generation (RAG) pipelines.

2) Knowledge Graphs for Static Structural Knowledge:

Knowledge graphs preserve ontological relationships among entities that would otherwise be lost in naive text chunking (bottom path of Fig. 3). For instance, [13] constructs a dual-level KG from support tickets to improve QA accuracy, while

GRAG [14] retrieves subgraphs via ego-graph-based strategies and integrates both text and graph views for enhanced multi-hop reasoning. In industrial contexts, KG-enhanced RAG frameworks [15] formalize FMEA data into multi-relational graphs, supporting symbolic and numerical reasoning over failure-cause-mitigation chains via efficient graph traversals.

3) Digital Twins for Dynamic Operational Knowledge:

While the above offer robust solutions for encoding and retrieving static knowledge, whether factual or structural, they fall short in scenarios requiring real-time insights from continuously evolving environments. On the other hand, digital twins have been widely adopted in industrial settings as a paradigm for integrating dynamic operational knowledge, where contextual updates, sensor data, and inference mechanisms co-evolve with the physical systems they represent. For instance, Wang et al. [16] introduce the Radio Environment Knowledge Pool (REKP), designed to support 6G digital twin channels by capturing how wireless signals behave in complex, real-world environments. REKP works like a dynamic memory system that learns and stores the relationship between physical surroundings (like buildings or obstacles) and signal behaviors (such as fading or path loss), using data from sensors and simulation tools. This allows the system to predict and adapt to changes in signal quality with greater accuracy and speed. One use case simulates a street-level environment, showing how REKP updates its understanding of signal loss as a receiver moves through areas with different building densities and obstructions.

C. Current Challenges

Despite many successful network intelligence applications with LLMs and agentic systems, from the literature review, we identify several challenges in current knowledge engineering practices.

1) *Redundant Knowledge Pipeline:* Agentic intelligence tasks typically implement their logic for retrieving and interpreting network data individually. This fragmented approach can lead to 1) redundant subscriptions where there can be numerous xApps subscribing to the same network events, leading to unnecessary overhead in both the control and data planes; and 2) hindered development due to increased engineering effort

to develop and maintain parallel, uncoordinated knowledge pipelines across diverse tasks.

2) *Poor Reusability and Interoperability*: Existing knowledge artifacts are closely tied to their target applications and input data formats. This tight coupling thus inhibits reuse and generalization across different network intelligence tasks. For example, extensive prompt engineering and fine-tuning may be required to adapt to different knowledge interfaces designed for existing intelligence tasks. Orchestrating end-to-end workflows that involve multiple LLMs or agents can also become error-prone and operationally brittle as the agents may be forced to communicate in different knowledge “accents” and query knowledge with different protocols. In addition, proprietary knowledge formats or APIs also hinder interoperability across vendors and operators, leading to vendor lock-in.

3) *Inconsistent Interpretations*: The absence of a unified source of truth can lead to diverging decisions or actions triggered by the same underlying network event, due to isolated semantic interpretations or retrieval mechanisms of individual intelligence tasks.

III. KP-A: NETWORK KNOWLEDGE PLANE FOR AGENTS

A. Design Requirements

To ensure the KP-A meets the practical needs of real-world 6G telecom deployments, we outline the following **ten** key design requirements.

1) *Schema*: The KP-A should provide an industry-standard-aligned schema or ontology to represent dynamic network knowledge, such as state, topology, and categorizations. This ensures semantic consistency across networks, vendors, and intelligence pipelines. The schema should be extensible to accommodate evolving 6G concepts.

2) *Freshness*: The KP-A should provision real-time or near-real-time network states through continuous streaming or event-driven mechanisms. Temporal versioning should be supported to enable time-based analysis, historical debugging, or replay of network events.

3) *Coverage*: The KP-A should provision the broadest possible range of network knowledge required by the Intelligence Plane. This includes granular, real-time operational data (e.g., current Channel Quality Indicator (CQI), Cell Individual Offset (CIO), and PRB allocations per UE), high-level orchestration states (e.g., execution logs and policies of xApps and service orchestrators), and even static or procedural artifacts (e.g., source code for scheduling, handover, and load-balancing algorithms). By making such comprehensive and cross-domain knowledge uniformly accessible, the KP-A can support both low-level data retrieval and high-level reasoning, promoting deeper situational awareness and advanced network autonomy.

4) *Semantic Enrichment*: The KP-A should provision lightweight insights wherever possible to bootstrap high-level reasoning from low-level metrics (e.g., congestion prediction, anomaly detection) with rule engines or AI/ML pipelines.

5) *Maintainability*: The KP-A should decouple knowledge acquisition from its consumption, enabling multiple intelligent agents to reuse shared network insights. It should expose modular and intuitive APIs for seamless integration with upstream data sources and downstream intelligence consumers.

6) *Security*: The KP-A should support multi-tenant architectures with fine-grained, role-based access control. This ensures secure and isolated access to knowledge subsets while enforcing privacy and compliance policies such as data masking or anonymization.

7) *Performance*: The KP-A should scale horizontally to accommodate the massive volume and velocity of knowledge generated by dense deployments of heterogeneous 6G elements, such as user equipment, small cells, IoT devices, and edge nodes. Scalability in query handling is essential to ensure that knowledge flows remain timely and actionable, especially for real-time decision-making.

8) *Auditability*: The KP-A should log all knowledge updates and accesses to support traceability, compliance audits, and root-cause analysis. Additionally, it should enable consuming agents to expose which knowledge artifacts informed their decisions, supporting explainability.

9) *Resilience*: The KP-A should operate reliably in the presence of network failures, latency spikes, or cyber threats. Redundancy and failover mechanisms should be incorporated to ensure high availability and fault tolerance in mission-critical deployments.

10) *Observability*: The KP-A should include toolings for developers and operators, such as monitoring dashboards and query testing utilities. Observability mechanisms should report metrics related to data freshness, system health, and query performance.

B. KP-A Design Overview

Fig. 2 illustrates the architectural design of the unified Network Knowledge Plane for Agentic intelligence (KP-A) and its integration within the autonomous 6G network. The design adopts a layered approach, with the KP-A positioned between the lower-level Network Data Ontology Plane and the upper-level Network Intelligence Plane. Its primary roles are to abstract and organize dynamic network information and provision intuitive, consistent, and interpretable textual knowledge, serving as a unified and reliable foundation for developing agentic network intelligence.

At the bottom of the stack lies the *Network Infrastructure Plane*, composed of physical and virtual elements such as user equipment (UE), base stations, edge servers, and core network components. These elements form the operational substrate of the 6G network, continuously generating telemetry and control data.

Immediately above, the *Network Data Ontology Plane* defines structured object-oriented models for these network entities. Each model, such as UE, Cell, Base Station, RIC, or Core Network, includes attributes (e.g., frequency, connection state, resource allocation) and logical methods (e.g., `connect()`, `getLoad()`, `handoverUE()`). This plane collects and organizes the raw data flows from the network infrastructure plane (e.g., via O-RAN or OAM interfaces) with a standardized ontology that enables intuitive, deterministic, and structured knowledge exploration and retrieval for the layers above.

The core of the architecture is the *Network Knowledge Plane*—the KP-A itself. This plane transforms the fragmented

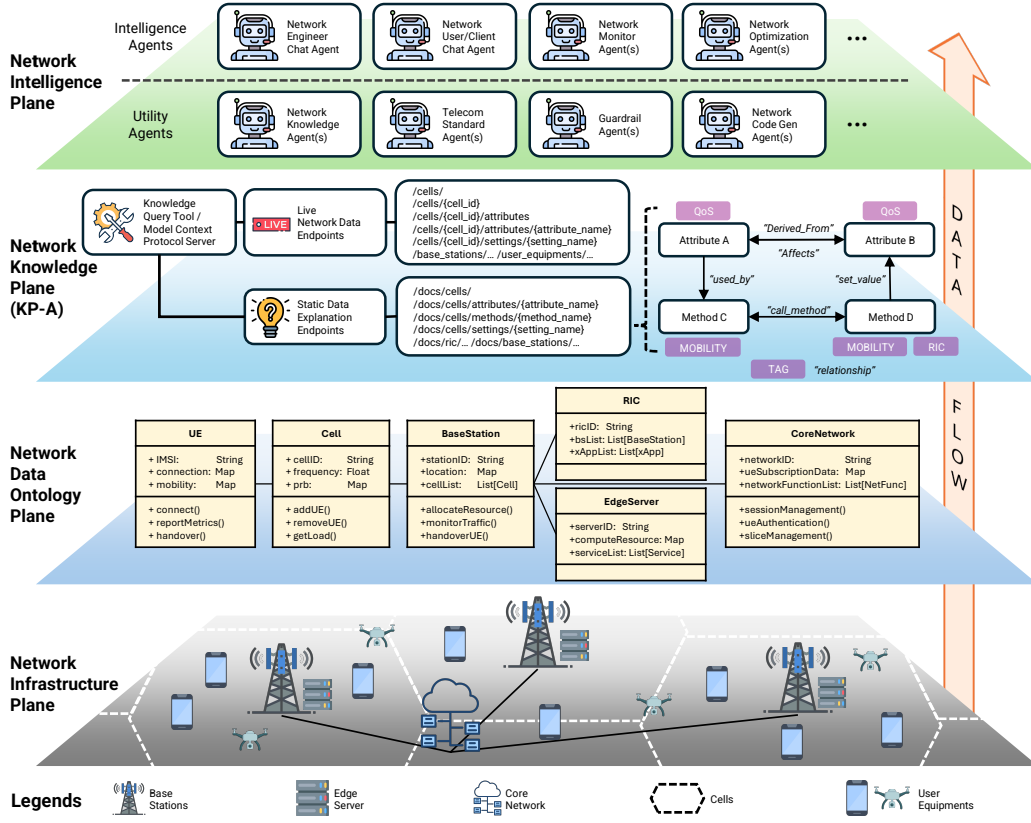


Fig. 2. Overview of KP-A and its relative logical position in the autonomous 6G network. Note that in Software-Defined Networks, there can be multiple knowledge provisioning servers across the network software cluster, and the intelligence agents can run wherever applicable (detailed in Section III).

data from the Data Ontology Plane into REST-style queryable knowledge constructs. It exposes two primary types of interfaces: *live network data endpoints* and *static data explanation endpoints*. The live endpoints provide real-time access to evolving network states, while the static endpoints document how the available attributes, methods, and configurations shall be interpreted by the agents. Beyond this, the KP-A introduces semantic structures, such as attribute and method relationships, to describe how various elements affect one another, facilitating deeper semantic understanding and iterative knowledge exploration. For example, a QoS attribute might be *derived_from* another metric or *used_by* a particular network method. These relationships form a lightweight knowledge graph that enables reasoning and context-aware querying by the intelligence agents.

Above the KP-A, the *Network Intelligence Plane* hosts a diverse set of LLM-powered agents responsible for question-answering and decision-making (optimization, diagnostics, and network planning). These agents can be broadly categorized into *intelligence agents*, which are tasked with high-level functions such as network engineering, client interaction, and optimization, and *utility agents*, which focus on specific tasks like monitoring, standard enforcement, or guardrail enforcement. These agents access the KP-A through a dedicated knowledge query tool or model context protocol, allowing them to retrieve, interpret, and reason over dynamic network knowledge without needing to parse telemetry feeds or hardcode the configuration logic individually.

IV. PROTOTYPE DEMONSTRATION

To demonstrate, we implemented and open-sourced a prototype knowledge plane based on our in-house lightweight RAN simulator [17].

A. Network Engineer Chat Agent

Fig. 4 demonstrates how intelligence plane agents, such as the Network Engineer Chat Agent, seamlessly retrieve both static and dynamic knowledge from KP-A to answer complex queries. In this example, an engineer requests a detailed explanation of the handover mechanism in the simulated RAN environment—a query requiring source code-level insights across multiple simulated entities, including the UE, cell, base station, and RIC. Upon activation, the Network Engineer Chat Agent hands off the query to the Network Knowledge Agent, which is equipped with knowledge query tools to interact with KP-A. Through two rounds of knowledge exploration, the agent collects sufficient source code references and explanations to formulate a comprehensive response. The knowledge endpoints in KP-A are designed to be self-contained and interlinked. For example, the `/docs/ue/methods/execute_handover` endpoint returns the actual code snippets (not pseudo-code), explanations on how the code shall be interpreted, and references to related attributes and methods. Queries for dynamic operational data, such as the number of UEs connected to the network, are resolved similarly through real-time knowledge endpoints.

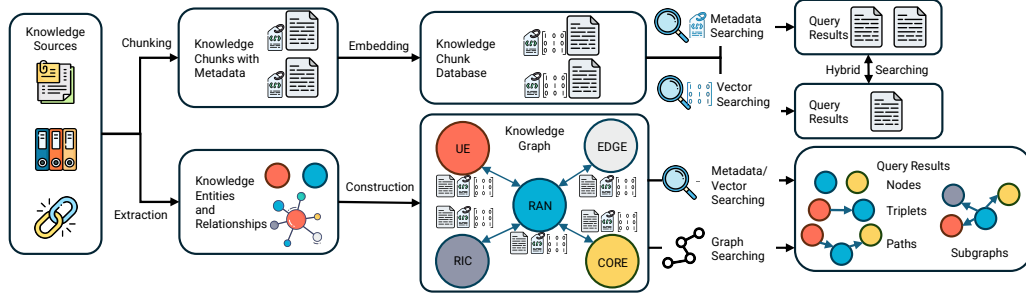


Fig. 3. Common knowledge engineering approaches for RAG: (top) knowledge chunking and semantic matching, and (bottom) knowledge graph construction and graph search.

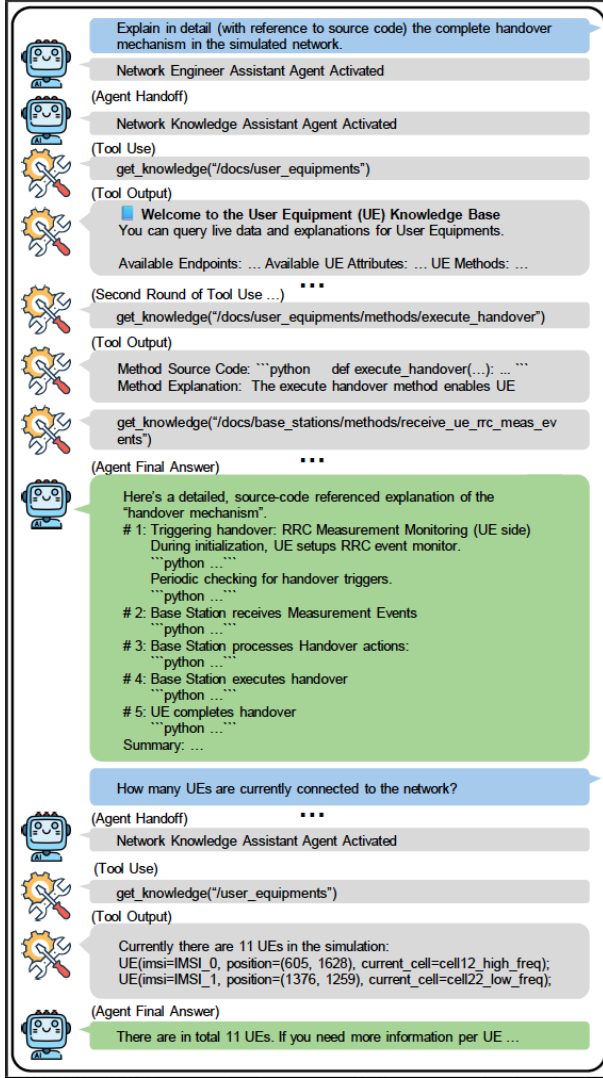


Fig. 4. Agent querying the KP-A for static and dynamic network knowledge.

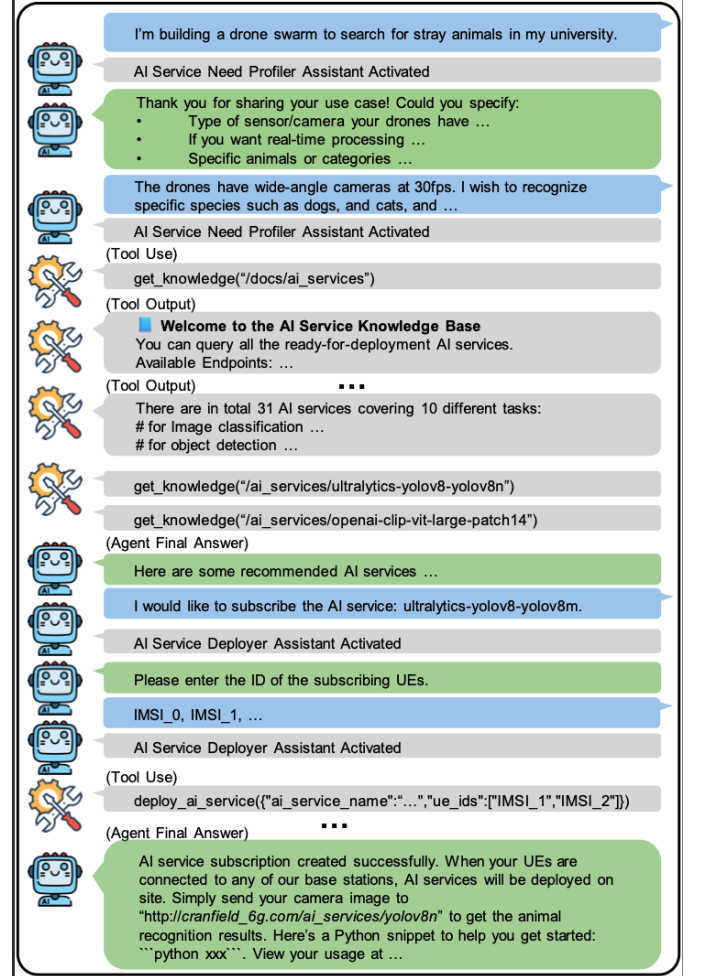


Fig. 5. Agent querying KP-A for edge AI service provisioning.

This demonstration exemplifies how KP-A empowers agents with direct, structured, and consistent access to comprehensive network knowledge, thereby potentially facilitating explainability, root cause analysis, or rapid in-situ troubleshooting without requiring manual inspection of fragmented logs or raw data from the underlying data ontology plane and infrastructure plane.

B. Edge AI Service Provisioning Agent

Fig. 5 illustrates the workflow of the Edge AI Service Provisioning Agent leveraging KP-A to facilitate AI service

deployment for user equipment (UEs) at the network edge. In this example, a use case developer (network customer) initiates a conversation with the AI Service Need Profiler Assistant, outlining their objective of deploying a drone swarm equipped with wide-angle cameras to detect stray animals such as dogs and cats. The profiler agent collects deployment requirements, including sensor specifications, real-time processing needs, and target categories, before invoking the AI Service Knowledge Base to query the catalog of ready-to-deploy AI services. Upon retrieving the available services, it identifies and recommends suitable models such as YOLOv8 for real-time object detection. Following the developer's service selection, the AI Service Deployer Assistant is activated to create AI service subscriptions for the specified UEs. The final response confirms successful subscription creation and provides operational details, including the AI inference endpoint URL and a Python code snippet to support immediate integration.

This demonstration exemplifies how KP-A extends beyond static knowledge retrieval to enable on-demand service provisioning and potentially general network control tasks.

C. More Evaluation Examples

More evaluation examples can be found in our project repository [17], containing 1) single attribute/method Q&A conversations that require only one knowledge query for a single attribute or method of a network entity class (e.g., *what's the current cell connected by the UE of ID: IMSI_1?*); 2) multi-method logic Q&A conversations that require multiple queries on different methods of a single network entity class (e.g., *explain what happens when an UE powers up with code-level reference*); 3) cross-entity logic Q&A conversations that require multiple queries on different methods of multiple network entity classes (e.g., *explain the complete handover mechanism implementation in the simulated network with code-level reference*); and 4) knowledge endpoint reusability validation conversations which involves multiple rounds of Q&A and the agent are expected to infer knowledge endpoints based on previous rounds.

V. CONCLUSION & FUTURE WORK

In this paper, we introduced KP-A, a unified Network Knowledge Plane designed to overcome key challenges in developing agentic intelligence for 6G networks, including fragmented knowledge pipelines, redundant data flows, inconsistent interpretations, and hindered interoperability. By decoupling knowledge acquisition and management from intelligence logic, KP-A provides a reusable, consistent, and intuitive foundation for building diverse network intelligence applications. We demonstrated its practical value through two representative use cases: network engineer chat for source-code-referenced explanations and edge AI service provisioning for dynamic service orchestration. Future work will focus on systematic evaluation, extending the schema coverage to encompass emerging 6G technologies, enabling cross-domain knowledge ingestion and orchestration, adding guardrails that safeguard across the native AI RAN ecosystem [18], and potentially standardizing KP-A interfaces for widespread adoption in real-world telecom deployments. We believe KP-A will catalyze the next generation

of trustworthy, explainable, and autonomous 6G networks by serving as a foundation for large-scale AI and agentic systems.

REFERENCES

- [1] "Framework and overall objectives of the future development of imt for 2030 and beyond," November 2023, ITU-R Recommendation M.2160-0.
- [2] "Zero-touch network and service management (zsm); enablers for artificial intelligence-based network and service automation," December 2022, published: 21 December 2022.
- [3] Y. Tang and W. Guo, "Automatic retrieval-augmented generation of 6g network specifications for use cases," *IEEE Communications Magazine*, vol. 63, no. 4, pp. 95–102, 2025.
- [4] M. Zakir Khan, Y. Ge, M. Mollel, J. Mccann, Q. H. Abbasi, and M. Imran, "Rfsensingpt: A multi-modal rag-enhanced framework for integrated sensing and communications intelligence in 6g networks," *IEEE Transactions on Cognitive Communications and Networking*, vol. 12, pp. 298–311, 2026.
- [5] Y. Tang, M. Zou, U. C. Srinivasan, O. Umealor, D. Kevogo, B. J. Scott, and W. Guo, "Building ai service repositories for on-demand service orchestration in 6g ai-ran," *IEEE Globecom*, 2025.
- [6] D. D. Clark, C. Partridge, J. C. Ramming, and J. T. Wroclawski, "A knowledge plane for the internet," in *Proceedings of the 2003 conference on Applications, technologies, architectures, and protocols for computer communications*, 2003, pp. 3–10.
- [7] A. Mestres, A. Rodriguez-Natal, J. Carner, P. Barlet-Ros, E. Alarcón, M. Solé, V. Muntés-Mulero, D. Meyer, S. Barkai, M. J. Hibbett *et al.*, "Knowledge-defined networking," *ACM SIGCOMM Computer Communication Review*, vol. 47, no. 3, pp. 2–10, 2017.
- [8] S. Long, F. Tang, Y. Li, T. Tan, Z. Jin, M. Zhao, and N. Kato, "6g comprehensive intelligence: network operations and optimization based on large language models," *IEEE Network*, 2024.
- [9] O. G. Lira, O. M. Caicedo, and N. L. da Fonseca, "Large language models for zero touch network configuration management," *IEEE Communications Magazine*, 2024.
- [10] J. Tong, W. Guo, J. Shao, Q. Wu, Z. Li, Z. Lin, and J. Zhang, "Wirelessagent: Large language model agents for intelligent wireless networks," *arXiv preprint arXiv:2505.01074*, 2025.
- [11] F. Jiang, Y. Peng, L. Dong, K. Wang, K. Yang, C. Pan, D. Niyato, and O. A. Dobre, "Large language model enhanced multi-agent systems for 6g communications," *IEEE Wireless Communications*, 2024.
- [12] F. Rezazadeh, A. A. Gargari, S. Lagen, H. Song, D. Niyato, and L. Liu, "Toward generative 6g simulation: An experimental multi-agent llm and ns-3 integration," *arXiv preprint arXiv:2503.13402*, 2025.
- [13] Z. Xu, M. J. Cruz, M. Guevara, T. Wang, M. Deshpande, X. Wang, and Z. Li, "Retrieval-augmented generation with knowledge graphs for customer service question answering," in *ACM Conference on Research and Development in Information Retrieval*, 2024, pp. 2905–2909.
- [14] Y. Hu, Z. Lei, Z. Zhang, B. Pan, C. Ling, and L. Zhao, "Grag: Graph retrieval-augmented generation," *ACL NAACL*, 2025.
- [15] L. Bahr, C. Wehner, J. Wewerka, J. Bittencourt, U. Schmid, and R. Daub, "Knowledge graph enhanced retrieval-augmented generation for failure mode and effects analysis," *Journal of Industrial Information Integration*, vol. 45, p. 100807, 2025.
- [16] J. Wang, J. Zhang, Y. Zhang, Y. Sun, G. Nie, L. Shi, P. Zhang, and G. Liu, "Radio environment knowledge pool for 6g digital twin channel," *IEEE Communications Magazine*, 2025.
- [17] Y. Tang, "ntutangyun/ai-ran-sim," <https://github.com/ntutangyun/ai-ran-sim>, 2025, accessed: 2025-07-03.
- [18] Y. Tang, M. Zou, W. Guo, and S. A. R. Zaidi, "Guardrailing llm and agentic decisions for 6g ai-ran," in *IEEE Consumer Communications & Networking Conference (CCNC)*, 2026.



DSPACE

<https://dspace.org/>

KP-A: a unified network knowledge plane for catalyzing agentic network intelligence

Tang, Yun; Zou, Mengbang; Nezami, Zeinab; Zaidi, Syed A. R.; Guo, Weisi

2026-05-24

Institute of Electrical and Electronics Engineers (IEEE)

<https://doi.org/10.1109/iccworkshops63917.2026.11586700>

Tang Y, Zou M, Nezami Z, et al., (2026) KP-A: a unified network knowledge plane for catalyzing agentic network intelligence. In: Proceedings of the 2026 IEEE International Conference on Communications Workshops (ICC Workshops), 24-28 May 2026, Glasgow, UK