



Experiential Networked Intelligence (ENI); Use Cases and Requirements for AI Agents Based Core Network

Disclaimer

The present document has been produced and approved by the Experiential Networked Intelligence (ENI) ETSI Industry Specification Group (ISG) and represents the views of those members who participated in this ISG.
It does not necessarily represent the views of the entire ETSI membership.

Reference

DGR/ENI-0055v411_AI_Agents

Keywords

6G, AI-Native, GenAI, use case

ETSI

650 Route des Lucioles
F-06921 Sophia Antipolis Cedex - FRANCE

Tel.: +33 4 92 94 42 00 Fax: +33 4 93 65 47 16

Siret N° 348 623 562 00017 - APE 7112B
Association à but non lucratif enregistrée à la
Sous-Préfecture de Grasse (06) N° w061004871

Important notice

The present document can be downloaded from the
[ETSI Search & Browse Standards](#) application.

The present document may be made available in electronic versions and/or in print. The content of any electronic and/or print versions of the present document shall not be modified without the prior written authorization of ETSI. In case of any existing or perceived difference in contents between such versions and/or in print, the prevailing version of an ETSI deliverable is the one made publicly available in PDF format on [ETSI deliver](#) repository.

Users should be aware that the present document may be revised or have its status changed,
this information is available in the [Milestones listing](#).

If you find errors in the present document, please send your comments to
the relevant service listed under [Committee Support Staff](#).

If you find a security vulnerability in the present document, please report it through our
[Coordinated Vulnerability Disclosure \(CVD\)](#) program.

Notice of disclaimer & limitation of liability

The information provided in the present deliverable is directed solely to professionals who have the appropriate degree of experience to understand and interpret its content in accordance with generally accepted engineering or other professional standard and applicable regulations.

No recommendation as to products and services or vendors is made or should be implied.

No representation or warranty is made that this deliverable is technically accurate or sufficient or conforms to any law and/or governmental rule and/or regulation and further, no representation or warranty is made of merchantability or fitness for any particular purpose or against infringement of intellectual property rights.

In no event shall ETSI be held liable for loss of profits or any other incidental or consequential damages.

Any software contained in this deliverable is provided "AS IS" with no warranties, express or implied, including but not limited to, the warranties of merchantability, fitness for a particular purpose and non-infringement of intellectual property rights and ETSI shall not be held liable in any event for any damages whatsoever (including, without limitation, damages for loss of profits, business interruption, loss of information, or any other pecuniary loss) arising out of or related to the use of or inability to use the software.

Copyright Notification

No part may be reproduced or utilized in any form or by any means, electronic or mechanical, including photocopying and microfilm except as authorized by written permission of ETSI.

The content of the PDF version shall not be modified without the written authorization of ETSI.

The copyright and the foregoing restriction extend to reproduction in all media.

© ETSI 2025.
All rights reserved.

Contents

Intellectual Property Rights	5
Foreword.....	5
Modal verbs terminology.....	5
1 Scope	6
2 References	6
2.1 Normative references	6
2.2 Informative references.....	6
3 Definition of terms, symbols and abbreviations.....	7
3.1 Terms.....	7
3.2 Symbols.....	7
3.3 Abbreviations	7
4 Background	8
4.1 Motivation of AI-Core.....	8
4.1.1 Driving Force.....	8
4.1.2 Key Concept	9
4.1.3 Advantages and Challenges	9
4.2 Business Value of AI-Core.....	10
5 Use Cases and Requirements	10
5.1 Consumer Use Cases	10
5.1.1 Use Case: AI Agents to Enable Smart Life.....	10
5.1.1.1 Description	10
5.1.1.2 Potential Requirements	11
5.1.2 Use Case on Network-Assisted Collaborative Robots.....	12
5.1.2.1 Description	12
5.1.2.2 Potential Requirements	13
5.1.3 Use Case on AI Phone	13
5.1.3.1 Description	13
5.1.3.2 Potential Requirements	14
5.2 Business Use Cases	15
5.2.1 Use Case on AI Agent-based Customized Network for Smart City Traffic Monitoring	15
5.2.1.1 Description	15
5.2.1.2 Potential Requirements	16
5.2.2 Use Case on AI Agents-Based Customized Network for Smart Construction Sites.....	16
5.2.2.1 Description	16
5.2.2.2 Potential Requirements	18
5.2.3 Use Case on AI Agent Ensuring Game Acceleration Experience.....	18
5.2.3.1 Description	18
5.2.3.2 Potential Requirements	19
5.2.4 Use Case on AI Agent-Assisted Collaborative Energy Distribution in Power Enterprises	19
5.2.4.1 Description	19
5.2.4.2 Potential Requirements	21
5.3 Telecom Operator Use Cases	21
5.3.1 Use Case on AI Agent-Based Autonomous Network Management	21
5.3.1.1 Description	21
5.3.1.2 Potential Requirements	23
5.3.2 Use Case on AI Agent-Based Disaster Handling Network Management	23
5.3.2.1 Description	23
5.3.2.2 Potential Requirements	24
5.3.3 Use Case on AI Agent-Based Time-Sensitive Network Management.....	25
5.3.3.1 Description	25
5.3.3.2 Potential Requirements	25
5.3.4 Use Case on AI Agent-Driven Core Network Signalling Optimization	26
5.3.4.1 Description	26
5.3.4.2 Potential Requirements	27

5.3.5	Use Case on AI Agent-Based Core Networks to Enhance User Experience	27
5.3.5.1	Description	27
5.3.5.2	Potential Requirements	28
6	Existing Use Case Summary	29
7	Conclusion and Recommendations	31
Annex A:	Bibliography	33
	History	34

Intellectual Property Rights

Essential patents

IPRs essential or potentially essential to normative deliverables may have been declared to ETSI. The declarations pertaining to these essential IPRs, if any, are publicly available for **ETSI members and non-members**, and can be found in ETSI SR 000 314: *"Intellectual Property Rights (IPRs); Essential, or potentially Essential, IPRs notified to ETSI in respect of ETSI standards"*, which is available from the ETSI Secretariat. Latest updates are available on the [ETSI IPR online database](#).

Pursuant to the ETSI Directives including the ETSI IPR Policy, no investigation regarding the essentiality of IPRs, including IPR searches, has been carried out by ETSI. No guarantee can be given as to the existence of other IPRs not referenced in ETSI SR 000 314 (or the updates on the ETSI Web server) which are, or may be, or may become, essential to the present document.

Trademarks

The present document may include trademarks and/or tradenames which are asserted and/or registered by their owners. ETSI claims no ownership of these except for any which are indicated as being the property of ETSI, and conveys no right to use or reproduce any trademark and/or tradename. Mention of those trademarks in the present document does not constitute an endorsement by ETSI of products, services or organizations associated with those trademarks.

DECT™, **PLUGTESTS™**, **UMTS™** and the ETSI logo are trademarks of ETSI registered for the benefit of its Members. **3GPP™**, **LTE™** and **5G™** logo are trademarks of ETSI registered for the benefit of its Members and of the 3GPP Organizational Partners. **oneM2M™** logo is a trademark of ETSI registered for the benefit of its Members and of the oneM2M Partners. **GSM®** and the GSM logo are trademarks registered and owned by the GSM Association.

Foreword

This Group Report (GR) has been produced by ETSI Industry Specification Group (ISG) Experiential Networked Intelligence (ENI).

Modal verbs terminology

In the present document "**should**", "**should not**", "**may**", "**need not**", "**will**", "**will not**", "**can**" and "**cannot**" are to be interpreted as described in clause 3.2 of the [ETSI Drafting Rules](#) (Verbal forms for the expression of provisions).

"**must**" and "**must not**" are **NOT** allowed in ETSI deliverables except when used in direct citation.

1 Scope

The present document studies potential use cases and new service requirements relevant to AI-Agents based core network (AI-Core). It covers the motivation, key concepts, and business value of AI-Core; identifies potential use cases, including Business to Consumer (B2C), Business to Business (B2B), and telecom operators' internal scenarios, and outlines the corresponding consolidated service requirements for future mobile communication networks.

2 References

2.1 Normative references

Normative references are not applicable in the present document.

2.2 Informative references

References are either specific (identified by date of publication and/or edition number or version number) or non-specific. For specific references, only the cited version applies. For non-specific references, the latest version of the referenced document (including any amendments) applies.

NOTE: While any hyperlinks included in this clause were valid at the time of publication, ETSI cannot guarantee their long-term validity.

The following referenced documents may be useful in implementing an ETSI deliverable or add to the reader's understanding, but are not required for conformance to the present document.

- [i.1] ETSI GR ENI 051 (V4.1.1): "Experiential Networked Intelligence (ENI); Study on AI Agents based Next-generation Network Slicing".
- [i.2] [The global market for humanoid robots could reach \\$38 billion by 2035.](#)
- [i.3] [Intelligent Virtual Assistant Market Size, Share, and Trends 2025 to 2034.](#)
- [i.4] IEEE 802.11™: "IEEE Standard for Information Technology--Telecommunications and Information Exchange between Systems Local and Metropolitan Area Networks--Specific Requirements Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications".
- [i.5] Recommendation ITU-R M.2160-0 (11/2023): "Framework and overall objectives of the future development of IMT for 2030 and beyond".
- [i.6] NGMN Alliance (V1.0): "6G Use Cases and Analysis".
- [i.7] 3GPP TR 22.870 (V0.2.0): "Study on 6G Use Cases and Service Requirements; Stage 1 (Release 20)".
- [i.8] Canalys special report (V1.0): "Now and next for AI-capable Smartphones".
- [i.9] TMForum: "IG1274M AI Agent (v2.0.0)", 2025.
- [i.10] Renze M, Guven E.: "Self-reflection in llm agents: Effects on problem-solving performance", arXiv preprint arXiv:2405.06682, 2024.
- [i.11] Shinn N, Cassano F, Gopinath A, et al.: "Reflexion: Language agents with verbal reinforcement learning", Advances in Neural Information Processing Systems, 2023, 36: 8634-8652.
- [i.12] Zhang W, Tang K, Wu H, et al.: "Agent-pro: Learning to evolve via policy-level reflection and optimization", arXiv preprint arXiv:2402.17574, 2024.
- [i.13] Guo Z, Xu B, Wang X, et al.: "MIRROR: Multi-agent Intra-and Inter-Reflection for Optimized Reasoning in Tool Learning", arXiv preprint arXiv:2505.20670, 2025.

3 Definition of terms, symbols and abbreviations

3.1 Terms

For the purposes of the present document, the following terms apply:

AI-Core: next generation core network which consists of multiple agents

NOTE: See ETSI GR ENI 051 [i.1].

3.2 Symbols

Void.

3.3 Abbreviations

For the purposes of the present document, the following abbreviations apply:

AGV	Automated Guided Vehicle
AI	Artificial Intelligence
AMF	Access and Mobility management Function
API	Application Program Interface
B2B	Business to Business
B2C	Business to Consumer
BSS	Business Support Systems
CAGR	Compound Annual Growth Rate
CN	Core Network
CRM	Customer Relationship Management
CSP	Customer Service Platform
HTN	Hierarchical Task Network
IME	Intent Management Entity
IoT	Internet of Things
IT	Information Technology
IVR	Interactive Voice Response
JSON	JavaScript Object Notation
KG	Knowledge Graph
LiDAR	Light Detection And Ranging
LLM	Large Language Model
MNO	Mobile Network Operator
MW	MegaWatts
NaaP	Network as a Platform
NLP	Natural Language Processing
NMS	Network Management System
NPN	Non-Public Network
NWDAF	NetWork Data Analytics Function
OAM	Operations and Management
OPS	Open Programmability System
O-RAN	Open-Radio Access Network
OSS	Operations Support System
OTT	Over The Top
P2P	Point 2 Point
PCF	Policy Control Function
PDU	Protocol Data Unit
PR	Potentiel Requirement
QCI	Quality of service Class Identifier
QoE	Quality of Experience
QoS	Quality of Service
RAN	Radio Access Network

RAT	Radio Access Technology
RF	Radio Frequency
RSSI	Received Signal Strength Indicator
SLA	Service Level Agreement
SMF	Session Management Function
SNR	Signal to Noise Ratio
TSN	Time Sensitive Network
UE	User Equipment
UPF	User Plane Function
URLLC	Ultra-Reliable Low Latency Communications
USD	United States Dollar
VIP	Very Important Person

4 Background

4.1 Motivation of AI-Core

4.1.1 Driving Force

As shown in [i.5] and [i.6], the 6G network is required to support communication-based usage scenarios, such as immersive communication and massive communication, as well as beyond communication usage scenarios, such as AI and communication, and integrated sensing. The service requirements of various usage scenarios are different and each scenario requires diverse network capabilities.

EXAMPLE: AI and communication scenarios expect a set of AI-related capabilities, including data acquisition and management, preparation and processing, distributed model training and inference.

Integrated sensing and communication require the provision of sensing-related capabilities, including range, velocity, angle estimation, object and presence detection, etc. Thus, the mobile network management system, in conjunction with its orchestrator(s), needs to flexibly orchestrate and provide various combinations of network capabilities to meet the diverse requirements of users.

The service requirements of customers are dynamic, and the environment may change. The network is required to make a prompt adjustment and feedback to adapt to the ever-changing needs of customers so as to ensure good experiences for them. In order to lower the threshold for customers to consume network services that they are interested in, it is beneficial to support customers in initiating service requests through intents. Therefore, the mobile network needs to provide the proper services and resources flexibly to satisfy the various requirements of users based on their intents and adapt to the changes in user needs (expressed through intents), business goals, and environmental conditions in a timely manner to meet their service needs, expressed through (for example) their Service Level Agreement (SLA).

Current network design is a mixture of rule-based and specialized software (e.g. multi-objective optimization systems). In rule-based systems, the actions, inputs and outputs of network entities are pre-defined and structured, which are only used to solve standardized tasks. When the network needs to be extended to new scenarios, the corresponding standards and network functions need to be updated. The typical standardization cycle is long, making the system inflexible and poorly adaptable. However, as described in 3GPP TR 22.870 [i.7], an AI Agent has a set of unique capabilities, e.g. interacting with its environment, acquiring contextual information, reasoning, self-learning, decision-making, and executing tasks (autonomously or in collaboration with other AI Agents) to achieve a specific goal. It would be beneficial to introduce AI Agents into a Core Network (CN). The CN is the logical place for multiple AI agents that manage and control the network. It has a global, end-to-end view of all network services, subscriber data, policies, and resource usage. This makes it the ideal location for non-real-time tasks like translating a subscriber's business intent, orchestrating a complex network slice that spans multiple domains, performing large-scale analytics, and real-time tasks such as dynamic resource scheduling, session modification to ensure that SLAs of customers are met.

In addition, with the development of AI technology, a large number of intelligent devices, such as AI Phones and AI-embodied robots, are emerging. The analysis report of Canalys shows that 16 % of global smartphone shipments were AI Phones in 2024, and this proportion is expected to soar to 54 % by 2028 [i.8]. The devices are gradually evolving from types of equipment that passively respond to commands to intelligent agents with autonomous decision-making and execution capabilities. Due to the limited resources and processing capabilities on devices, the network can provide intelligent assistance services, e.g. assisting safe navigation and optimal path planning, to the intelligent devices through AI Agent in CN.

In conclusion, it is necessary to integrate AI Agents into the CN to provide flexible customized services based on the intents of subscribers and ensure their Quality of Experience (QoE) autonomously, as well as provide intelligent assistance services for AI-capable devices.

4.1.2 Key Concept

As demonstrated in [i.1], AI-Core is the next-generation core network that consists of multiple AI Agents. It presents an horizontal architecture where each agent in a multi-agent system plays a specific role and collaborates with others to accomplish various tasks. The key idea of AI-Core is to utilize multiple AI Agents to handle high-level intents, plan complex task execution, manage and control the network resources, and to flexibly process the data for new services based on the dynamic requirements of various applications. Through the powerful capabilities of AI Agents, various network functions and tools provided by third parties can be flexibly assembled to generate a customized network on-demand to meet the personalized requirements of subscribers. During the operation of these customized networks, AI Agents autonomously perceives the changes in the environment and adjust the constituent functions or tools and resources dynamically to guarantee the QoS and QoE. The networks are automatically recycled when the service completes. In other words, the design, generation, execution, update, and recycling of the customized networks are entirely performed by AI Agents in CN, without human intervention, making CN an autonomous system. Moreover, after receiving feedback from the environment, AI Agents can examine past decisions, so as to make better decisions in the future when facing similar problems through the self-reflection mechanisms such as summarizing behavioural guidelines or policies used for the model or fine-tuning the model through reinforcement learning [i.10], [i.11], [i.12], [i.13]. That is, the performance of AI agents-based core (i.e. AI-Core) improves continuously.

4.1.3 Advantages and Challenges

AI-Core can solve diverse tasks, including standardized and non-standardized (i.e. novel tasks that do not have an associated playbook) tasks. This is thanks to the fact that the basic principle of an AI Agent is that it is knowledge-based (i.e. it generates outputs based on knowledge pre-trained or retrieved from a knowledge base). To address novel tasks, an agentic system leverages its underlying LLM's pre-trained knowledge to generate a multi-step plan. The agent then executes this plan by calling upon a predefined set of software 'tools' (e.g. APIs, diagnostic scripts), which allows for a more rapid response than traditional, lengthy standardization cycles. However, this flexibility introduces a critical trade-off between speed and safety. While accelerating new service deployment, it replaces the deterministic predictability of standardized systems with the stochastic nature of LLM-based agents, creating operational risks for critical infrastructure. Furthermore, an agent's effectiveness is domain-specific; its strength in semantic reasoning and planning does not readily translate to tasks requiring high-precision mathematical optimization, for which traditional numerical algorithms remain superior.

AI-Core aims to simplify service consumption by allowing subscribers to initiate requests through high-level intents expressed in natural language. This approach abstracts away the complexity of network functions and APIs, making services more accessible to non-expert users. The primary technical challenge, however, lies in reliably translating a user's qualitative, and often ambiguous, goal into a set of precise, machine-executable network parameters. For example, an intent like, "*Ensure my video conferences have priority during peak hours*", requires the system to identify the correct application traffic, interpret time-based conditions, and apply specific QoS policies without violating other service level agreements. Bridging this 'semantic gap' between user intent and network configuration is a non-trivial problem, as misinterpretation can lead to incorrect network behaviour or create security vulnerabilities.

4.2 Business Value of AI-Core

In the current ecosystem, the mobile network serves as a bit-pipe to transmit data for Over-The-Top (OTT) service providers, while OTT service providers develop various kinds of applications for end-users. Since AI-Core can utilize advanced agentic AI technologies to customize networks on-the-fly that integrate various network functions and tools provided by third parties as well as associated resources, it is expected to blur the boundary between connectivity and application, providing network operators with similar opportunities as the agentic AI technology offers to the OTT service providers. This can bring new revenue and profit models for network operators.

Several business models can be exploited for AI-Core. The first one is that AI-Core generates tailored services, the monetized approach (or business model) of the operator can be similar to OTT's, including advertisement revenue. The second one is that the operator can integrate multiple OTT services into the network and generate a new service, possibly with a value-added service included, the operator pays the service providers for API calls, while the operator charges the end-users for consuming the integrated service. The third one is that the network operator is the integrator of various third-party services and charges the service providers for expanding their customer base. The fourth one is that AI-Core is a platform that generates highly customized services based on multi-agent collaboration. In such a setting, the mobile network (operator) exposes APIs to the service providers to facilitate service customization, for which the service providers are charged by the operator.

5 Use Cases and Requirements

5.1 Consumer Use Cases

5.1.1 Use Case: AI Agents to Enable Smart Life

5.1.1.1 Description

The growth in the use of AI agents enriches human daily life and supports the intelligent solutions in the industry. For embodied AI agents, there are different types in the market with different capabilities and levels of intelligence, e.g. an intelligent humanoid robot can provide many different types of support across a variety of circumstances, a robot-dog is more lightweight but can provide less service than a humanoid robot, a drone is of low intelligence considering the battery constraint, limiting it to tasks like delivering foods for human beings. For virtual AI entities, the chatbot has entered the area of customer service. According to the latest report from Goldman Sachs [i.2], the global market for humanoid robots could reach \$38 billion by 2035 and the humanoid robot shipment is expected to hit 1 million units by 2035. Precedence Statistics also show that the global intelligent virtual assistant market size was United States Dollar (USD) 16,17 billion in 2023, accounted for USD 20,42 billion in 2024, and is expected to reach around USD 166,97 billion by 2033, expanding at a Compound Annual Growth Rate (CAGR) of 26,3 % from 2024 to 2033 [i.3].

In future production scenarios, such as industrial grounds, smart cities and hospitals, AI Agents will be used to complement and even replace human labour.

EXAMPLE: These AI Agents could be given the order of replacing all instances of part A with part B in the products being manufactured, checking the production line for possible problems or inefficiencies, or reconfiguring the operation of other machines.

As with humans, autonomous AI Agents are not only the recipients of direct commands, but they also have the initiative to inform about the status of other components in the factory or even spontaneously raise alarms. The mobile network can provide connectivity to these AI Agents as well as provide global perception and assisted AI services to AI Agents, rather than limiting them to local perception and intelligence. The reason is that, on the one hand, the accuracy of local sensors and models sometimes are not high enough to ensure safe and efficient operation. On the other hand, the use of local small-scale models for individual AI agents often results in sub-optimal decision-making. In summary, the mobile network can empower AI Agents with network services and resources including sensing, AI, communication, and computing.

Assume that a user owns several types of AI agents, including a smart car, drone, robot-servant and robot-dog, which are produced by different manufacturers and of different capabilities. Further assume that all of these agents have registered with the AI-Core and have access to the mobile network, and each of them is allocated an identity to uniquely identify them. Currently, agents retrieve translated intent from other sources, as this is a complex multi-stage pipeline for all but the simplest of intents. In the future, it is possible to define a set of agents to perform the multi-stage intent translation directly, which will simplify the life of the user.

Considering an example where a user wants to go camping on the weekend, the AI Agents are required to work together to make a good camping plan with the assistance of the network. The service flows of this example are as follows:

- 1) The robot-servant sends the intent of "make a camping plan" to the 6G network on behalf of the user. How the robot-servant gets the user's authorization is out of scope, e.g. it can take verbal commands from the user.
- 2) The AI agents in the mobile network parse the intent and break it down into sub-tasks, which are then assigned to different AI agents based on their capabilities, e.g. the network AI Agent asks local life assistant (a digital AI agent provided by 3rd party) to recommend campsites, instructs the smart car to pick-up family members by designing the optimal route, asks the robot-servant to book food in advance according to the user's taste.
- 3) The network AI agent builds connections for the involved AI Agents since they need to exchange information. For example, the local life assistant needs to send the campsite address to the smart car for designing the routes. Robot-servant selects the restaurant based on the user's taste and sends the restaurant information to the robot-dog while instructing it to pick up the order.
- 4) The AI agents execute the allocated sub-tasks. When they cannot perform the sub-task well, they request for the network services. For example, due to the limitation of local perception, the smart car requests the sensing service of the network when it determines the optimal route from the user's home to the campsite. The robot-servant requests the AI service of the mobile network to help perform inference of motion control.
- 5) The network monitors the AI Agents that perform sub-tasks, when some exception occurs, it finds an alternative AI agent to complete the sub-task.

Figure 5.1.1.1-1 depicts the described scenario.

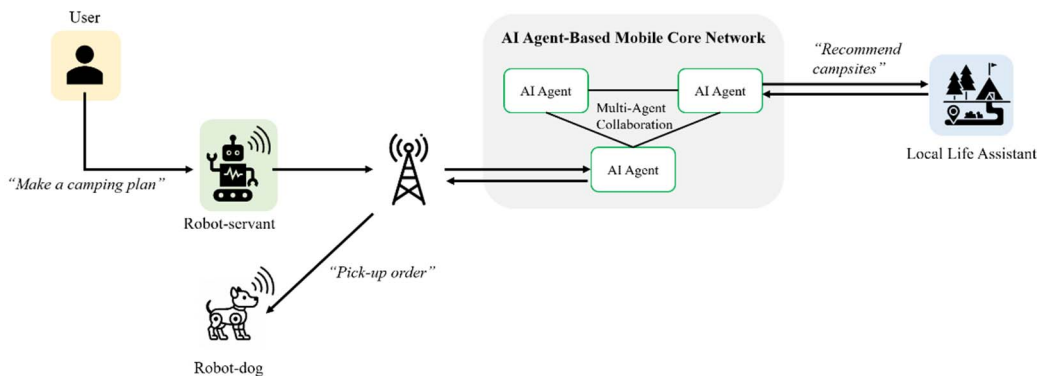


Figure 5.1.1.1-1: Smart life use case key entities and high-level service flows

5.1.1.2 Potential Requirements

To support the above example, five requirements for the mobile network are summarized in the following:

[PR 5.1.1-1] Subject to operator policy and regulatory requirements, the mobile network is used to contact AI-Core, which then provides a mechanism to uniquely identify an AI Agent that acts on behalf of the user.

[PR 5.1.1-2] The mobile network uses a combination of authentication, opaque execution, and policy enforcement to ensure that all AI Agents preserve the privacy of the owner (e.g. user, the 3rd party) when they exchange information with AI Agents.

NOTE 1: Opaque execution is a cornerstone of modern multi-agent communication protocols. It means that when one agent asks another to perform a task, the second agent is able to complete the task without revealing its internal methods, data, or reasoning processes. The collaboration happens through well-defined, standardized interfaces, but the "inner workings" of each agent remain private.

NOTE 2: When sensitive information needs to be exchanged, robust agentic systems have guardrails for sensitive data handling and isolation. This means there are explicit policies that govern how sensitive data is used and shared between agents. Privacy preservation is the set of rules followed during the conversation, ensuring that sensitive personal or business information is not revealed unnecessarily.

[PR 5.1.1-3] The architecture includes a standardized "integration fabric" that mediates all multi-agent communication. This fabric supports multiple interaction patterns, such as request-response for direct commands and publish-subscribe for asynchronous event streaming, enabling decoupled and scalable collaboration.

[PR 5.1.1-4] The system provides a service registry or Knowledge Graph (KG) that allows agents to dynamically discover and consume network-provided services. These services augment agent capabilities by offering specialized data (e.g. high-precision location) or AI models (e.g. congestion prediction) as a service.

[PR 5.1.1-5] A high-level orchestrator agent is responsible for end-to-end task management. It uses formal planning techniques, such as Hierarchical Task Networks (HTNs), to decompose high-level intents into sub-tasks, which are then delegated to specialized agents based on their advertised capabilities.

5.1.2 Use Case on Network-Assisted Collaborative Robots

5.1.2.1 Description

Multiple robots can collaborate to accomplish complex tasks that are beyond the capability of any single robot. Examples include carrying heavy objects, monitoring wide areas, or conducting search and rescue operations in disaster zones. To enhance adaptability and generalization across diverse robotic tasks and various robot designs, large models play a crucial role. Equipped with such, robots can perceive their dynamic environments, make informed decisions through advanced reasoning, and execute actions to interact effectively with the physical world. Agentic AI technology offers great potential to implement such intelligent systems, thereby, revolutionizing the next generation of robotic systems. However, due to the inherent limitations in onboard resources and computational power, individual robot agents often rely on the assistance of AI Agents equipped with more advanced models to assist them in performing complex tasks efficiently.

For instance, multiple robots can cooperatively perceive their surrounding environments by utilizing onboard sensors that capture diverse, multi-modal data such as speech, images, videos, haptic feedback, and RF signals. These sensing processes across different robots and modalities sometimes are performed asynchronously to accommodate varying operational conditions. The collected sensing data are transmitted to a central network, where large AI models analyse and infer the environmental states.

AI models with relatively smaller sizes can be deployed on robots to perform data pre-processing such that the latency of communication and computation can be balanced. To optimize communication latency and computational efficiency, smaller-scale AI models can be deployed directly on the robots for initial data pre-processing. This distributed approach balances the workload between the robot- and the network agents. Additionally, the network is expected to determine which robots and sensing modalities are essential for the cooperative task, minimizing redundant operations and maximizing overall system efficiency. The service flows of the described use case above are listed as follows:

- 1) Multiple robots register with a management and orchestration entity in the AI-Core to participate in environmental sensing tasks, such as gesture recognition.
- 2) The mobile network collects capability information from each robot, including sensor capabilities (i.e. modality, resolution, accuracy), communication capabilities, computational capabilities, and energy consumption profiles.

NOTE 1: This step is a continuation of the registration process. The capability information described is the payload of the registration request from Step 1. This structured data is used to create or update the robot's entity within a suitable entity, such as a Knowledge Graph (KG). The KG serves as the network's "digital twin", storing not just the identity of each robot but a rich profile of its specific attributes, which can be queried. This allows the orchestrator to perform sophisticated queries later, such as, "Find all available robots within Area A with camera resolution > 4K and battery > 50 %".

- 3) One or more robots send service requests specifying their service request (i.e. the intent to collect information about their surroundings, potential requirements and/or preferences on information accuracy, data delivery latency, and energy consumption) to an Intent Management Entity within the AI-Core.

NOTE 2: Here, a robot switches roles from a potential service provider to a service consumer. It submits its high-level intent not to the general "network" but to a specific Intent Management Entity (IME) within the AI-Core. This interaction is handled by an Intent Translation Pipeline. This pipeline uses NLP and LLM-based techniques to parse the intent and its constraints (accuracy, latency) into a formal, machine-readable specification that the orchestration system can act upon.

- 4) Based on these requests, an AI Orchestrator (or Supervisor) Agent within the AI-Core takes the validated intent from Step 3 and uses a formal planning methodology, such as Hierarchical Task Networks, to decompose the abstract goal ("collect information") into a concrete workflow.
- 5) The Orchestrator queries the Knowledge Graph (or other entity that is used) to find the optimal set of robots (from the pool of robots registered in Step 1) whose capabilities match the requirements of the decomposed plan. This is where it would "select appropriate data modalities" by choosing robots with the right sensors.
- 6) The Orchestrator schedules the tasks and delegates specific instructions to the chosen robots.
- 7) Simultaneously, the Orchestrator delegates a sub-task to the relevant Network Management Domain (e.g. the RAN or Core domain manager) to "configure relevant network equipment". This typically involves creating a dedicated, QoS-guaranteed network slice to ensure the sensing data can be transmitted with the required latency and reliability.
- 8) Robots, now acting as worker agents, execute the instructions delegated by the Orchestrator in Step 6. The robots transmit their data to a dedicated Data Analytics Service as specified by the Orchestrator's instructions.
- 9) The raw data sent by the sensing robots is received by a dedicated Data Fusion and Analytics Agent within the AI-Core. This specialized agent is responsible for the post-processing tasks: aggregating data from multiple robotic sources, filtering out noise, and performing higher-level inference (e.g. fusing multiple camera angles to perform gesture recognition). Once this value-added processing is complete, this agent delivers the final, refined information back to the original robot that made the service request in Step 3, thus closing the loop. This entire workflow is monitored by the Orchestrator to ensure the original intent is fulfilled.

NOTE 3: Steps 8 and 9 operate as a pair. Step 8 is the addressable endpoint that the robots transmit their data to. Step 9 is the logical actor within the AI-Core that owns and operates that service. It is the intelligent component that actually receives the data from the service endpoint, fuses data from multiple robots while filtering noise, and generates the final, value-added result.

5.1.2.2 Potential Requirements

The use case implies the following requirements:

[PR 5.1.2-1] The system needs to provide a standardized interface for authorized users to initiate service requests by submitting high-level, declarative intents. This interface abstracts the underlying complexity, allowing users to define their desired outcomes without specifying the implementation details.

[PR 5.1.2-2] Upon receiving a validated intent, the system's AI Orchestration function needs to provide the requested service on-demand. This is achieved by autonomously planning and decomposing the intent into a workflow of sub-tasks, which are then delegated to the appropriate specialized agents for execution.

5.1.3 Use Case on AI Phone

5.1.3.1 Description

Mobile terminals are envisioned to be transformed into AI phones in the near future, supporting a wide range of sophisticated AI applications. Beyond traditional connectivity services, next-generation mobile communication systems will provide new capabilities to AI phones. For example, the phone becomes a dynamic platform that receives AI models provisioned by the network. The brain of an AI agent can be deployed across various platforms, including cloud servers, edge servers, or end devices. In this model, the network operator can push specialized, task-specific AI models or lightweight agents directly to the "AI Phone" as needed. Significantly, the phone's abilities are no longer static and tied to the apps that the user(s) have installed. Rather, its intelligence can be augmented on-the-fly by the network. In addition, instead of a single, massive, all-purpose AI model residing permanently on the device, smaller, more efficient models tailored to the immediate task can be loaded and unloaded, optimizing resource usage.

This capability enables a compelling future vision. Instead of the user having to identify a specific task, select an appropriate application, and launch the application and its (possibly unfamiliar) unique interface, the user can simply express one or more intents in natural language to the phone via text or voice. The phone's agent(s) then interpret the intent and orchestrates all the necessary sub-tasks in the background. For example, instead of having to learn the particulars of each airline's interface, the phone can check the user's calendar, find flights, book a hotel, arrange ground transport, and anything else required, all by invoking the required services or specialized sub-agents. Hence, the user is freed from the cognitive load of managing a constellation of individual apps. The complexity is abstracted away, and the interaction feels like a continuous conversation with a single, capable assistant. The "apps" still exist as underlying services or functions, but the user no longer interacts with them directly.

While a simple chatbot does not require immense intelligence, the vision described in the present document does. However, it is important to realize that this vision does not have to be implemented all at once. To function as a truly personalized AI agent that can seamlessly fulfil complex, multi-modal requests, the system possesses capabilities that are indeed at the forefront of AI research: advanced reasoning and planning, social intelligence (i.e. the ability of the agent to infer the user's unspoken needs, desires, and emotional state from nuanced cues), and multi-modal fusion (a highly complex task that requires sophisticated AI models).

For example, a user might interact with the next generation mobile core network through their AI phone to place an order for a customized product. The core network would interpret multi-modal inputs from the user, such as images, voice, and text, to understand their intent, generate a unique design, and coordinate with the product supplier to fulfil the order. This requires the core network to process diverse data types to understand the user's intent through agentic AI technology and autonomously deliver customized on-demand services to satisfy the user's request. To enable this, seamless and intelligent communication between the AI phone and the core network's AI Agent is essential.

The service flows describing the interaction between the AI phone and the core network AI agents are as follows:

- 1) The user launches the AI phone client and submits a product customization request using multi-modal inputs, including images and audio.

NOTE: Here, the AI phone is the initiating agent. In order for this to work, the Intent Management Entity needs to expose a set of standardized, secure interfaces over the Integration Fabric that is accessible to authenticated user agents on mobile devices.

- 2) The phone's AI agent performs initial validation, data compression, and possibly restructures the raw inputs before sending them to the Intent Management Entity within the AI-Core, via the system's secure Integration Fabric.
- 3) The Orchestrator Agent within the AI-Core receives the intent and validates it against the network's KG to check for feasibility, resource availability, and potential policy conflicts. It uses a formal planning technique like Hierarchical Task Networks (HTNs) to decompose the complex intent ("create and deliver a custom product") into a workflow of sub-tasks. The Orchestrator then delegates these sub-tasks to the appropriate specialized agents (e.g. a Multi-Modal Fusion Agent, a Product Design Agent, a Supplier Logistics Agent (a third-party tool), and a Network Slice Manager Agent (for handling the QoS requirements mentioned in the next step).

The AI Orchestrator, as part of its overall plan, issues a derived intent to the relevant Network Management Domain (e.g. the 5G Core domain manager). This intent would be something like, "Establish a network slice for user X with latency < 50 ms and bandwidth > 100 Mbps for the duration of this session". The domain manager is then responsible for configuring the low-level network resources to meet this QoS requirement.

5.1.3.2 Potential Requirements

[PR 5.1.3-1] The Intent Management Entity of the AI-Core needs to expose a standardized, secure interface over the Integration Fabric that is accessible to authenticated user agents on mobile.

[PR 5.1.3-2] All interactions between the AI Phone agent and the AI-Core, transmitted over the mobile network, are required to be encrypted and authenticated to prevent eavesdropping or man-in-the-middle attacks. This is a foundational security posture for any trusted system.

[PR 5.1.3-3] The AI-Core needs to maintain a discoverable registry of available services and their capabilities, allowing the AI Phone agent to query which network-provided functions (e.g. data fusion, design generation, QoS management) it can invoke.

5.2 Business Use Cases

5.2.1 Use Case on AI Agent-based Customized Network for Smart City Traffic Monitoring

5.2.1.1 Description

The new use cases and services supported by the next-generation mobile network are outlined in [i.5] and [i.6]. These emerging services require the mobile networks to provide various capabilities, including AI-driven and sensing-related capabilities, as well as multi-dimensional resources such as computing power, communication, and data storage. Given these new services and requirements, next-generation mobile networks are required to be able to instantiate isolated, logical networks on-demand, each with guaranteed Quality of Service (QoS) parameters (e.g. bandwidth, latency) tailored to a specific application.

Traditional network management processes that are based on expert knowledge and historical experience are no longer sufficient for providing adequately customized networks for new services that require the flexible assembly of multiple functionalities and multi-dimensional resources. AI Agents with their strong capabilities in intent understanding, tool usage, planning, decision-making, task execution and self-evolution, offer a promising approach to customizing networks for new services.

For instance, in a smart city scenario, the mobile network is required to provide a customized network, i.e. a logic network that assembles various network capabilities and resources, to the smart city operator for monitoring passenger and vehicle flow. During peak tourism periods, smart city operators need real-time information on passenger and vehicle flow in scenic spots to implement intelligent traffic dispatch, alleviate congestion, and improve citizens' travel convenience. In this context, the mobile network needs to offer a customized network to monitor these flows during peak hours. Outside the peak commute hours, the monitoring service is automatically terminated by the operators, and the dedicated network for monitoring service is subsequently deleted. Thus, the creation and reclamation of this customized network for services occur on demand. The service flows of this use case are illustrated in Figure 5.2.1.1-1 and are given as follows:

- 1) The smart city operator submits a high-level, multi-modal intent to a specific service endpoint provided by the network's AI-Core. This request is ingested by an Intent Translation Pipeline, which uses LLMs to parse the natural language and temporal constraints (e.g. "scenic spot A", "from 8:00 to 17:00") into a structured, machine-readable format.
- 2) The AI Orchestrator receives the structured intent. It then:
 - a) Plans the Workflow using Hierarchical Task Networks by decomposing the intent into a sequence of sub-tasks (e.g. "provision slice", "configure analytics", "deploy monitoring agents", "deliver results").
 - b) It queries the network's Knowledge Graph to identify available resources, such as cameras in "scenic spot A" and available edge computing nodes for data processing.
 - c) It then issues specific sub-intents to the relevant domain managers. For example, it instructs the 5G Core domain manager to instantiate a dedicated network slice with the required QoS, and it instructs an edge orchestration manager to deploy the necessary analytics functions (e.g. object detection) on selected compute nodes. This process of connecting functions is known as service function chaining.
- 3) Closed-loop assurance is realized by the AI Orchestrator (in the AI-Core) continuously monitoring telemetry data from the active network slice and the analytics functions. If it detects a deviation from the intent (e.g. latency increases, video quality drops), it automatically initiates a self-optimization routine, such as allocating more bandwidth or migrating a processing task to a different edge node, to maintain the agreed-upon service level.
- 4) The specialized Data Analytics Agent, having processed the raw data from the sensors, delivers the structured output (e.g. a JSON object with vehicle and passenger counts) to the smart city operator's designated endpoint at the specified one-minute interval.
- 5) At 17:00, or upon receiving a termination command, the AI Orchestrator executes the final stage of its plan. It issues delete commands to the relevant domain managers to tear down the network slice, terminate the analytics functions, and release all reserved compute and network resources, ensuring efficient resource utilization.

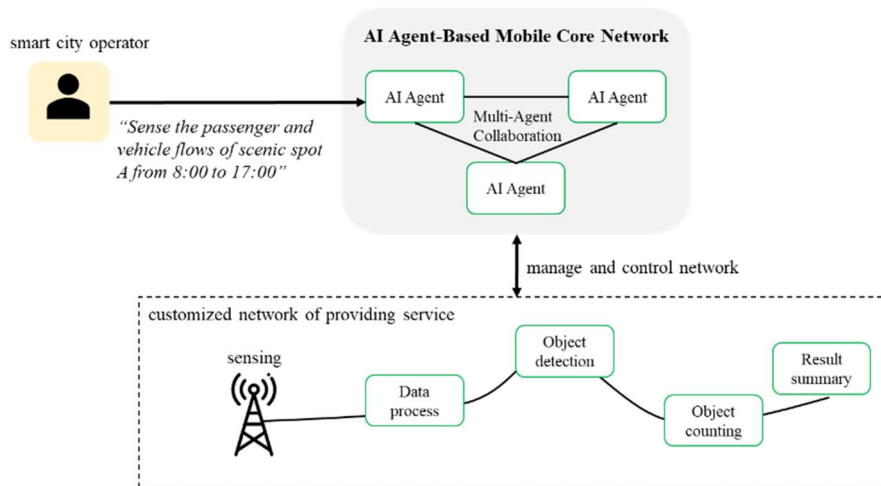


Figure 5.2.1.1-1: Customized network generated by AI Agent-based core network

5.2.1.2 Potential Requirements

For this use case, the following requirements are proposed for the mobile network.

[PR 5.2.1-1] The system's AI Orchestrator is required to be capable of dynamically composing and deploying logical networks (i.e. network slices) on-demand. This process integrates network capabilities (e.g. QoS guarantees) with application-layer functions (e.g. AI-based video analytics) by decomposing a user's intent and chaining the required services from both network and third-party domains.

NOTE: The generated networks are not pre-designed and are generated on-the-fly by the multi-agent tailored to the requests of users. The logic networks do not only include the network capabilities/functions but also include the application functions according to the request of users.

[PR 5.2.1-2] The AI Orchestrator is required to autonomously manage the entire lifecycle of the requested service. This includes using Hierarchical Task Network-based planning to design the service chain, delegating deployment tasks to specialized agents, monitoring execution, and ensuring the complete reclamation of all resources upon service termination.

[PR 5.2.1-3] The system is required to implement a closed-loop assurance mechanism. The AI Orchestrator needs to monitor cross-domain telemetry in real-time to verify that the operational state of the logical network continuously complies with the performance and QoS constraints defined in the original intent.

[PR 5.2.1-4] The AI Orchestrator is required to be capable of self-optimization. When the assurance loop detects a performance degradation or potential SLA violation, the orchestrator needs to autonomously trigger corrective actions, such as re-allocating resources or adjusting network configurations, to restore compliance with the user's intent.

5.2.2 Use Case on AI Agents-Based Customized Network for Smart Construction Sites

5.2.2.1 Description

Traditional network management will no longer be sufficient for highly flexible and demanding use cases such as smart construction sites, depicted in Figure 5.2.2.1-1.

At a construction site, multiple shareholder companies share the same space and need to cooperate, requiring high performance for data connectivity. This communication will be handled with nomadic networks, which are mobile and can be deployed at any location. Current construction sites only connect workers with information about the building project or interconnect worker timekeeping systems. In future construction sites, more vehicles and tools, especially connected and automated vehicles like AGVs, trucks, and machinery such as cranes, will be connected to the mobile network:

- 1) The manager submits a high-level intent (e.g. "Provide basic data connectivity for 50 workers at Site B, accessing project plans on server X"). This request is ingested by the Intent Translation Pipeline, which parses it into a structured format for the AI Orchestrator. The Orchestrator then plans and deploys a simple network slice with standard QoS parameters.
- 2) The construction site manager orders a network configuration for transmitting LiDAR data and communication for AGV routing, shortly before the AGV's arrival at the construction site. This is a Modification Intent, which is more complex than the initial request. The manager is updating the service requirements. The Orchestrator receives this new intent (e.g. "Add a high-bandwidth, low-latency service for LiDAR data and a high-reliability, low-latency service for AGV control to the existing network at Site B"). This demonstrates the system's crucial ability to manage the entire service lifecycle, not just initial deployment.
- 3) After receiving the request, the AI Orchestrator takes the new, modified intent and updates its plan for providing communication services to smart construction site operators. This consists of:
 - a) Re-planning using Hierarchical Task Networks to determine the necessary changes to the existing service, such as creating new, dedicated network slices for the LiDAR and AGV traffic or modifying the QoS of the existing slice.
 - b) It queries the KG for available network resources and deploys the necessary application functions (e.g. data processing for LiDAR, routing algorithms for AGVs) onto edge compute nodes.
 - c) It issues updated configuration commands to the relevant Network Domain Managers to implement the changes.
- 4) During the runtime of the generated generative network, the AI Orchestrator performs a closed-loop service assurance function by continuously monitors the performance of both the basic connectivity and the new high-performance slices. If an AGV's connection latency spikes, the orchestrator's assurance loop detects the SLA deviation and can autonomously trigger an optimization, such as rerouting traffic or allocating more radio resources to maintain the intent. This is a "dynamically generated network".
- 5) A Monitoring Agent provides periodic status reports (e.g. network uptime, AGV connectivity status, data throughput) to the construction site manager's dashboard, confirming that the service is operating as intended.
- 6) After the termination of the service, the AI Orchestrator executes the teardown plan, instructing all domain managers to release the network slices and all associated compute and storage resources.



Figure 5.2.2.1-1: Example representative figure for smart construction sites

5.2.2.2 Potential Requirements

For this use case, the following requirements are proposed for the mobile network.

[PR 5.2.2-1] The system's AI Orchestrator is required to be capable of dynamically composing and deploying logical networks that co-provision network services (e.g. connectivity with specific QoS) and application-layer functions (e.g. data processing) as a single, integrated on-demand service.

[PR 5.2.2-2] The system is required to support the autonomous, end-to-end lifecycle management of the composed service. This includes the initial design and deployment based on intent, runtime monitoring and modification, and the complete recycling of all network and compute resources upon service termination, all managed by the AI Orchestrator.

[PR 5.2.2-3] The system is required to provide closed-loop assurance by continuously monitoring the performance of the deployed network and application functions. The AI Orchestrator is required to autonomously detect and remediate any deviations from the user's intent to ensure correct execution and SLA compliance.

[PR 5.2.2-4] In the case of non-public networks (NPN), where the network service is not provided by a public Mobile Network Operator, the mobile network needs to be usable in a nomadic node manner. This results in the fact that the cellular infrastructure is highly mobile and can be used at multiple various construction sites.

5.2.3 Use Case on AI Agent Ensuring Game Acceleration Experience

5.2.3.1 Description

Acceleration services in mobile networks are designed to enhance the speed, responsiveness, and overall quality of user experiences by optimizing how data are transmitted and managed across the network. These services are critical as mobile data traffic continues to grow rapidly, driven by applications like video streaming and gaming.

Currently, acceleration services offer various packages tailored to specific apps or users. However, the future landscape will involve users interacting with multiple non-network AI agents - such as game agents provided by third parties - that manage phone operations and screen projection. This necessitates a standardized, secure way for agents to communicate, discover each other's capabilities, and exchange intent information, which is the role of an Integration Fabric and a Service Registry.

The existing acceleration service framework is inadequate to support this intelligence-driven ecosystem or to provide truly personalized acceleration services. Currently, the mobile network operators deploy proprietary nodes to directly connect to game servers and uses dynamic routing technology to provide acceleration services for video games based on their latency or bandwidth requirement. However, the customization of the acceleration service according to the intent of users or applications is not supported. Moreover, the network adjusts the acceleration service in a reactive way with the changing user requirements, which in turn affects the user experience due to the delayed reaction time.

Network AI agents can adjust network resources from the core network to the base station in an end-to-end manner based on the received intent to provide a personalized service experience. This intent typically originates directly from the user, from an external AI agent (e.g. a game agent), or from a game company's server that considers the terminal conditions of all game participants. In addition to passively responding to these intents, the network AI agents can proactively predict fluctuations in user experience by analysing network status and historical data, enabling it to adjust policies in advance to maintain service quality.

Moreover, future charging models can leverage the capabilities of network AI agents. For example, subscription fees for AI agents that understand individual users' intents or other agents' intents could be higher. Pricing could also factor in the computing resources consumed by these agents. This approach enables MNOs to offer more perceptible, personalized acceleration experiences that users are willing to pay a premium for, thereby, enhancing revenue growth by delivering personalized and flexible service experiences to mobile subscribers.

The service flows of network AI agent providing game acceleration service are as follows:

- 1) The consumer subscribes to the game acceleration solution empowered by AI agents. The user plans to play the subscribed game while commuting from City A to City B. A proactive system enables the user's "Personal AI Agent" to share this high-level context (e.g. "User is starting a commute and typically plays Game X") with the network's AI-Core. This allows the network to anticipate the need for the acceleration service.

- 2) While traveling by taxi to the rail station, the consumer opens the game app. The game agent sends a structured intent (e.g. "Requesting low-latency, high-reliability service for Game X") to the network's Intent Management Entity (IME).
- 3) The AI Orchestrator receives the intent. It queries the KG and network monitoring services to gather context (user location, speed, current network load).
- 4) Based on the intent and context, the Orchestrator generates and deploys the initial network policies, likely by provisioning a dedicated network slice with specific QoS parameters.
- 5) At the railway station, where many users require a guaranteed game experience, their AI agents all connect to the AI-Core. The AI Orchestrator receives multiple, potentially competing, intents. It is required to use a conflict resolution or resource optimization function to balance the needs of all users against the available network resources (e.g. base station capacity). This is a highly advanced capability that moves from single-user service fulfilment to multi-user resource management.
- 6) When traveling, the user sometimes passes through areas with poor signal coverage (e.g. tunnels). To minimize the degradation in the quality of service, the AI agent can proactively predict the service degradation using historical data and notify the consumer in advance, which possibly affects the user's decision to save or end the game. This is predictive, closed-loop assurance, and consists of three phases:
 - a) An Analytics Agent within the AI-Core continuously analyses the user's trajectory and compares it against a KG containing historical network performance data for that geographic route.
 - b) The agent's predictive model forecasts an imminent drop in signal quality as the user approaches a tunnel.
 - c) This prediction triggers a proactive notification, sent via the Integration Fabric, to the user's game agent, which can then alert the user. This demonstrates the system's ability to prevent a poor user experience rather than just reacting to it.

5.2.3.2 Potential Requirements

[PR 5.2.3-1] The system is required to include an Intent Translation Pipeline that leverages Natural Language Processing to parse high-level, unstructured user intents into a formal, machine-readable specification for the orchestration layer.

[PR 5.2.3-2] The architecture is required to use an AI Orchestrator to manage the end-to-end service lifecycle and an Integration Fabric to facilitate secure, standardized communication between internal network agents and external third-party agents.

[PR 5.2.3-3] The system is required to support closed-loop assurance and optimization, where the AI Orchestrator continuously monitors service performance against the user's intent and autonomously triggers corrective actions. This requires a Service Registry where agents can discover each other and their capabilities.

[PR 5.2.3-4] The system is required to expose its network and AI capabilities (e.g. QoS management, predictive analytics) as discoverable, well-defined services on the Integration Fabric, allowing authorized external agents to invoke them via standardized APIs.

5.2.4 Use Case on AI Agent-Assisted Collaborative Energy Distribution in Power Enterprises

5.2.4.1 Description

In today's heterogeneous and distributed energy systems, which comprise solar, wind, hydro, and fossil fuel power, energy companies face the challenge of real-time coordination across multiple power stations. These challenges arise from heterogeneity in site capabilities (computing, communication), dynamic fluctuations in energy supply and demand, and the complexity of maintaining synchronization across collaborative operations. By embedding AI Agents within the mobile network architecture, energy enterprises can significantly enhance the intelligence, efficiency, and responsiveness of their operations.

NOTE 1: This describes a complex cyber-physical system where AI acts as the intelligent controller for a distributed power grid, using the mobile network as its nervous system.

AI Agents act as intelligent controller between power infrastructure and the mobile network:

- AI Agents receive user-intent inputs and manage historical data to support adaptive energy distribution strategies. By learning from past consumption patterns and real-time operational status, they can dynamically optimize energy allocation across multiple sites.
- AI Agents analyse multi-dimensional data, such as grid load and environmental conditions (solar, wind, hydro), to dynamically adjust the operation and output of power generation equipment.
- AI Agents analyse real-time energy consumption and distribution loss metrics to make intelligent decisions on whether to prioritize local consumption or dispatch energy across regions, enabling dynamic, large-scale optimization of power distribution.
- AI Agents continuously monitor real-time data streams to detect abnormal energy consumption patterns or equipment anomalies. Based on predefined operational logic or learned behaviours, they can autonomously trigger protective actions within the energy control system.

The integration of AI Agents with mobile networks brings key technical benefits:

- Ultra-Reliable Low-Latency Communication (URLLC) enabled by network slicing and optimized traffic routing, ensures timely delivery of control commands. Critical control commands are prioritized through network slicing, while edge computing enables rapid local decisions. Together, these capabilities form a closed-loop "sense-decide-act" system that enhances renewable energy utilization and maintains grid stability.
- AI Agents leverage lightweight models at the edge to process large volumes of sensor data close to the source, supporting fast and localized decision-making. Meanwhile, core network-based AI Agents leverage large models to perform global analysis and coordination. Through collaboration among multiple models across the edge and core network, this distributed architecture enhances system responsiveness and strengthens overall grid stability.

NOTE 2: This is a two-tiered intelligence model: "lightweight models at the edge" for rapid, local data processing and decision-making, and "core network-based AI Agents" with "large models to perform global analysis and coordination". The edge agents handle real-time control and data filtering, while a central, more powerful agent performs strategic, system-wide optimization.

- AI Agents configure the mobile network to dynamically establish communication links between different power generation sites as well as between the sites and central control centres. This ensures low-latency, reliable, and secure data exchange, enabling real-time coordination and intelligent energy dispatch across distributed facilities.

NOTE 3: The mobile network is transformed from a data transport pipe to a programmable resource that is actively configured by the AI system.

- AI Agents can collaborate with core network functions (e.g. AMF, SMF, PCF as an example for 5G core) to dynamically adjust communication policies, ensuring that energy control traffic is prioritized and resilient to disruptions.

NOTE 4: In conclusion, the system is initiated by a high-level goal, such as "maximize green energy use while maintaining power stability". This declarative intent is the starting point for a complex chain of analysis, planning, and execution.

The service flows of AI Agent-assisted collaborative energy distribution are as follows:

- 1) The Service Registration process begins with Agents representing each energy site connecting to the AI-Core's Service Registry and publishing their capabilities (power output, storage levels, etc.) to the network's KG. The KG now has a real-time digital twin of all available energy assets, which is essential for the planning phase.
- 2) The operator submits the high-level business goal to the Intent Management Entity (IME) within the AI-Core. This intent is declarative; it specifies what to achieve, not how to achieve it. An example is "maximize green energy use while maintaining power stability".

- 3) The central AI Orchestrator performs the following sub-tasks:
 - a) The Orchestrator queries the KG and invokes specialized Analytics Agents to forecast energy generation and load based on historical and real-time data.
 - b) Using the forecast and the intent's constraints, the Orchestrator uses a planning engine (like Hierarchical Task Networks) to create an optimal energy distribution plan. It queries the KG to select the best combination of energy sites to fulfil the plan.
 - c) The Orchestrator issues derived intents to the 5G Core's Network Domain Manager to configure the necessary network slices with URLLC QoS for the control commands.
- 4) The Orchestrator dispatches specific, executable instructions (e.g. "discharge battery at 50 MW", "curtail solar output by 10 %") to the lightweight agents at the selected energy sites and edge nodes. The closed loop assurance process consists of the following steps:
 - a) The local agents at each site execute their assigned tasks.
 - b) The AI Orchestrator continuously ingests telemetry data from the sites and the network. It compares the real-world outcomes against the predicted model. If there's a deviation (e.g. an unexpected drop in solar output), it automatically re-runs its planning and optimization loop to generate and dispatch an updated strategy, ensuring the high-level intent is always being met.

5.2.4.2 Potential Requirements

[PR 5.2.4-1] The AI-Core is required to provide a Service Registry that allows authorized agents representing energy sites to register and publish their operational capabilities to the network's KG in a standardized format.

[PR 5.2.4-2] The system is required to expose a secure Intent Management Interface that allows authorized operators to initiate energy optimization services by submitting high-level, declarative intents.

[PR 5.2.4-3] The architecture is required to support a hierarchical agent model. A central AI Orchestrator is required to perform global, system-wide optimization and planning, while delegating real-time data processing and task execution to lightweight agents deployed at the network edge and on-site.

[PR 5.2.4-4] The AI Orchestrator is required to be capable of translating high-level intent into an executable workflow. This includes analysing contextual data from the KG, generating an optimal coordination strategy, delegating tasks to subordinate agents, and issuing derived intents to network domain managers to configure the required QoS.

[PR 5.2.4-5] The AI Orchestrator is required to implement a closed-loop assurance function, continuously monitoring execution outcomes and using this feedback to dynamically learn and adapt its energy distribution strategies to persistently meet the user's intent.

5.3 Telecom Operator Use Cases

5.3.1 Use Case on AI Agent-Based Autonomous Network Management

5.3.1.1 Description

As future mobile systems are expected to support an increasing array of new technologies and services, mobile network operators are actively seeking opportunities to reduce their operational expenses while meeting the evolving customization needs of these applications. The requirements of these applications are dynamic, necessitating solutions that can adapt rapidly to changing demands.

Predefined standardized processes are generally fixed and do not perform well in situations when the application requirements are changing frequently. This is particularly important for those applications that demand multiple new capabilities, such as AI, sensing, data collection and processing capability. For example, a vehicle related application requires the sensing and data process capability of network to percept the road condition and process the perception data. Thus, the mobile system is required to orchestrate these new capabilities and multi-dimensional resources (e.g. computing, communication and data resources) to meet the requirements of various applications in an efficient way. That means that the mobile network is required to autonomously generate personalized solutions according to customized requirements of applications.

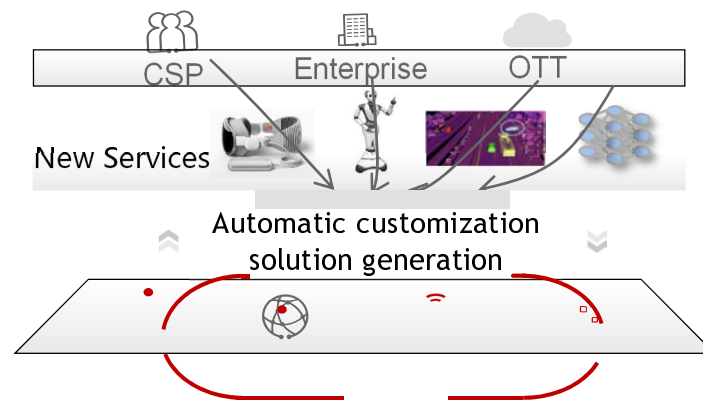


Figure 5.3.1.1-1: Agent-Based Autonomous Network Management

Take autonomous network fault analysis and recovery as an example. When a user finds that the network speed is very slow when surfing the internet, the user complains to the network through natural language. The system's first and most critical task is to understand this natural language input and translate it into a formal, machine-readable goal that can trigger the entire automated workflow. The service flows are as follows:

- 1) The user's natural language complaint is the high-level intent. It is submitted to the NMS, which serves as the user-facing portal to the AI-Core. The request is immediately ingested by an Intent Translation Pipeline. This pipeline uses NLP and LLMs to parse the unstructured complaint, identifying the core intent ("diagnose and resolve performance issue") and extracting key entities like the user's identity, location, and the nature of the problem ("slow speed").
- 2) Before a plan can be formulated, the system requires context. The "AI Agent for OPS" is best understood as the central AI Orchestrator. Its first action is to query the network's Knowledge Graph (KG) - the "knowledge database" - to gather all relevant information about the user and the affected network segment. This includes topology, active services, current traffic loads, and recent alarms in that area.
- 3) The AI Orchestrator manages the task decomposition:
 - a) Using a formal planning technique like Hierarchical Task Networks, the Orchestrator decomposes the high-level goal into a logical workflow of sub-tasks.
 - b) The Orchestrator delegates these sub-tasks to specialized worker agents via the Integration Fabric:
 - i) Subtask 1 (Exception Detection): This is assigned to a Monitoring Agent, which invokes data collection tools to pull real-time performance metrics (latency, packet loss, throughput) from the relevant network elements.
 - ii) Subtask 2 (Root Cause Analysis): The abnormal metrics are passed to a Diagnostics Agent. This specialized agent uses its analytical models to correlate the data and pinpoint the root cause (e.g. a congested backhaul link, a misconfigured router, or interference in the radio access network).
 - iii) Subtask 3 (Fault Rectification): The root cause analysis is passed to a Remediation Agent. This agent generates a solution (e.g. "reroute traffic via backup path") and, critically, can first test this solution in a Simulation Environment or "digital twin" to verify its effect before applying it to the live network. This validation step is essential for ensuring that automated actions do not inadvertently cause new problems.
 - c) Step 4: Reporting and Closing the Loop. The AI Orchestrator, having successfully managed the workflow, synthesizes the final report (e.g. "Root cause was a congested link. Traffic has been rerouted. Verify your connection speed.") and delivers it back to the user via the NMS interface, thus closing the loop on the original complaint.

5.3.1.2 Potential Requirements

The following requirements are deduced by this use case:

[PR 5.3.1-1] The system is required to expose a single, unified Intent Management Interface as the northbound entry point for all management tasks. This interface needs to be supported by an Intent Translation Pipeline capable of parsing natural language requests to abstract away the complexity of the underlying infrastructure.

[PR 5.3.1-2] The system's AI Orchestrator is required to be capable of fulfilling user objectives by autonomously translating a validated intent into an executable workflow, decomposing it into sub-tasks, and delegating those tasks to the appropriate specialized agents.

[PR 5.3.1-3] The architecture is required to support hierarchical multi-agent collaboration for closed-loop operations. A central AI Orchestrator needs to manage the overall workflow, delegating specific functions like exception detection, root cause analysis, and remediation to distinct, specialized worker agents that collaborate to achieve the system's goal.

5.3.2 Use Case on AI Agent-Based Disaster Handling Network Management

5.3.2.1 Description

The development of sustainable networks for disaster handling is critical for ensuring the resilience of communication infrastructure and the efficient delivery of essential services in the event of a disaster. The challenge in network disaster handling is not only to provide robustness and resilience to communication infrastructure in disaster-prone areas but also to provide sustainable solutions that can be rapidly deployed and require minimal maintenance, reducing the overall cost and environmental impact of network operations. As approaches to resiliency often imply, incorporating a certain amount of extra service capacity and allocating additional network resources, sustainability of these approaches is crucial to create incentives to adoption, and it is an important aspect of overall network sustainability.

Figure 5.3.2.1-1 shows a representative system architecture for network disaster handling.

The sustainable networks for network disaster handling focus on solutions for the use of an integrated, resource shared edge cloud continuum network in the case of disasters or unintended events such as earthquakes, flooding, major power shutdown, or system failures. The aim is to ensure the efficient operation of communication systems for disaster response operations, coordination of rescue and relief operations, and always-on network services such as emergency, medical, and rescue services. This use case considers also the process of restoring the network and its services to their pre-disaster state or to a new sustainable state. It is critical to ensure that the network can seamlessly transition from the emergency state to the normal operating state with minimal downtime and without compromising the quality of service.

The common network solution is controlled by a centralized location based on predefined rules. However, in this case, the AI Agents need to be trained to work in distributed mode with partial knowledge of the network and without a centralized location.

NOTE: While an AI Orchestrator in the AI-Core might manage the network during normal operations, individual agents need to be capable of autonomous, peer-to-peer collaboration to restore services when that central authority is unreachable. This decentralized structure provides robustness, as the failure of some agents does not cripple the entire system.

The set of AI Agents have to cooperate to enable a communication network based on the current physical network conditions. This use case is split to two main stages: before a disaster event and during one. In the first stage, the AI Agents monitor different types of data, including network telemetry collected by AI Monitoring Agents, external environmental feeds ingested via APIs by special Environmental Agents, physical infrastructure sensors collected by special Physical Sensing Agents, and other contextual data, as part of its normal operations. The AI-Core uses a Knowledge Graph for all relevant data, acting as the "digital twin" of the operational environment. A specialized Analytics Agent continuously analyses the aggregated data in the KG and performs cross-domain correlation. It identifies patterns that indicate a potential disaster, such as: "The National Weather Service API is reporting a hurricane with a projected path directly over our primary fibre optic route in Florida".

By correlating external environmental data with internal network and infrastructure data, the system can move from being reactive to being predictive. It can anticipate the impact of a disaster before it happens and begin taking preparatory actions, such as rerouting traffic away from the projected storm path or spinning up resources in a safe location.

In the second stage, the AI Agents try to re-establish connection between themselves and reconfigure the network the bring back communication and network services. The corresponding service flows are:

- 1) During normal operation: The AI Orchestrator directs Monitoring Agents to collect telemetry. This data is fed into a specialized Analytics Agent that runs predictive models to identify potential threats (e.g. tracking a storm's path) and forecast their likely impact on specific network infrastructure. This analysis continuously updates the network's Knowledge Graph (KG), which can serve as a distributed source of truth for all agents.
- 2) During disaster operation:
 - a) One or more AI Agents in each isolated network segment use their local sensors and cached knowledge of the network's last known good state to identify which connections have failed. This is a reactive, autonomous behaviour based on local perception.
 - b) Each surviving AI Agent needs to collaborate with other surviving AI Agents in its local segment. Using robust Peer-to-Peer (P2P) communication protocols, they negotiate a strategy to re-establish connectivity. This could involve activating backup links (e.g. satellite) or dynamically creating a mesh network between surviving nodes. This requires AI Agents to have the authority to reconfigure local network elements.
 - c) To restore service in a completely isolated segment, the collaborating local agents could decide to instantiate a lightweight, containerized version of essential core network functions (e.g. a local AMF or UPF) on an available edge compute node. This creates a self-contained, functional "network-in-a-box" until connectivity to the main core network can be restored.

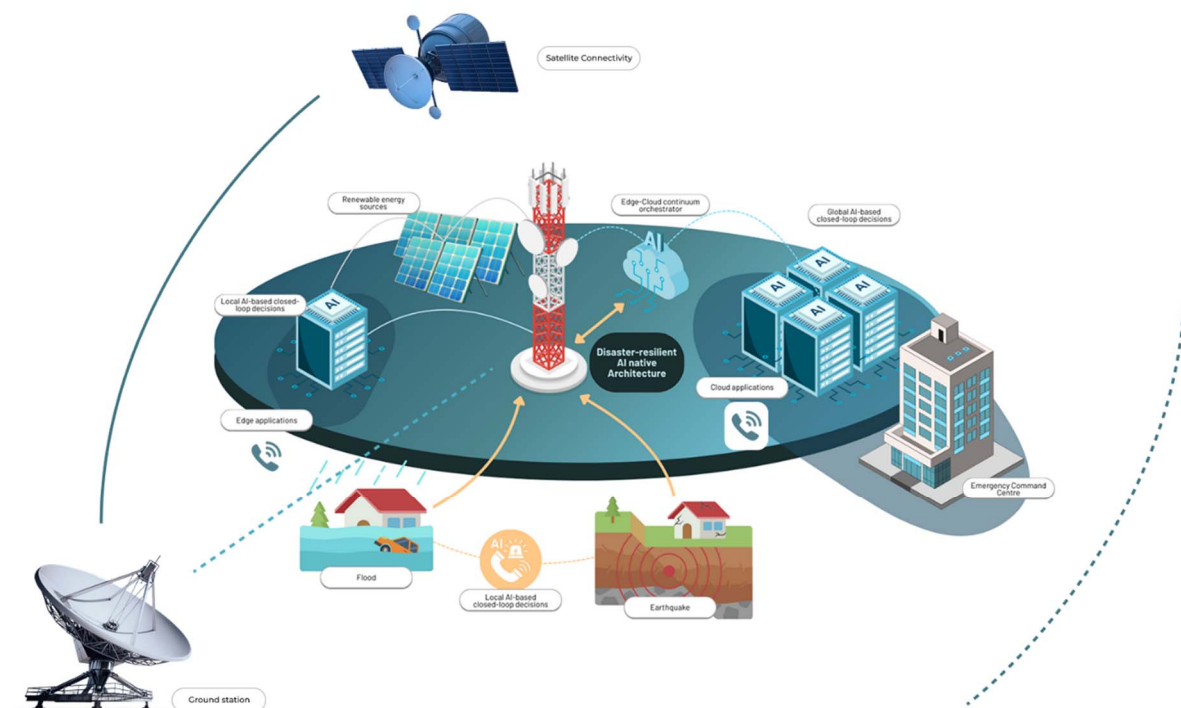


Figure 5.3.2.1-1: Representative figure for network disaster handling

5.3.2.2 Potential Requirements

[PR 5.3.2-1] The system's Integration Fabric is required to support both hierarchical communication patterns for centralized "peacetime" orchestration and robust, decentralized Peer-to-Peer (P2P) protocols to enable agent collaboration when the network is fragmented and central control is lost.

[PR 5.3.2-2] The system is required to include a specialized Analytics Agent capable of performing predictive analysis on network telemetry and external data (e.g. weather patterns) to forecast the impact of potential disasters and proactively trigger preparatory actions.

[PR 5.3.2-3] AI Agents deployed at the network edge need to be empowered with the autonomous capability to reconfigure local network elements and instantiate essential network functions. This enables them to restore connectivity and services within an isolated segment without real-time instruction from a central orchestrator.

5.3.3 Use Case on AI Agent-Based Time-Sensitive Network Management

5.3.3.1 Description

Communication forms a bedrock for industrial automation in the realm of Industry 4.0. Communication networks for Industry 4.0 applications need to support deterministic, low latency and high reliability communication. Currently many industrial applications utilize wired Time Sensitive Networking (TSN), however, to improve flexibility and support mobility, wireless TSN extensions are needed. This use case focuses on time sensitive applications in a Non-Public Network (NPN) deployment that supports multiple Radio Access Technologies (RATs), including IEEE 802.11 [i.4] in addition to cellular network. Time-sensitive networking can be utilized in different parts of the NPN (i.e. in the Radio Access Network (RAN), in the transport network, to inter-connect different open RAN (O-RAN) components, or to interconnect different core network components. When using inside RAN, the time sensitive features can be enabled in multiple RATs (multi-RAT). This use case considers the utilization of multi-RAT for enabling end-to-end time-sensitive networking for industry 4.0 applications such as factory automation, control-to-control communication, process control, logistics (automated guided vehicles), operator assistance, collaborative robots, and remote control of robots.

This use cases covers the establishment of a unified method to manage and control multi-RAT within access network enabling seamless and end-to-end network optimization. A unified multi-RAT data plane with time-sensitive networking capabilities will be utilized to improve reliability by performing replication in different RATs and channel bands, to support seamless and deterministic handover and to improve end-to-end cross-domain latency. On top of the unified network, an unified AI-agent based management and orchestration plane will be responsible to manage and orchestrate/coordinate TSN features across different network domains (multi-RAT RAN, transport, core). The management of TSN features will be done on different time scales based on the time criticality of network control loops. The unified AI-agent based network management and orchestration plane will support real-time control loops for TSN features on the RAN and transport, near real-time control loops for TSN features inside the O-RAN components and core and non-real-time control loops for other network policies. The service flows of this use case are as follows:

- 1) Real time monitoring on different granularity scale (on per-hop, per-node and per-flow scale) information are shared with the unified AI Agent-based management and orchestration planes. Such information includes low-level information (e.g. Received Signal Strength Indicator (RSSI), Signal to Noise Ratio (SNR), interference levels, current resource utilization) from multi-RAT RAN as well as other monitoring information from transport and core.
- 2) The network operators will share the information about applications requirements, this information will identify applications that will run and/or are admitted to be started in the network.
- 3) The AI-agent based management, and orchestration plane will split the task in other sub-task to train AI-agents for different sub-tasks based on monitoring and application requirements information.
- 4) Different AI-agents will perform different tasks, e.g. one AI-agent will determine the TSN schedules in multi-RAT RAN links, another will determine TSN schedules in transport, another can determine the coordination mechanism between TSN features in different network domain etc.
- 5) Once the AI-agents' outputs are combined, the decision is communicated from unified AI-agent based management and orchestration plane towards the network nodes that need to be (re)-configured.

5.3.3.2 Potential Requirements

[PR 5.3.3-1] The architecture is required to support Monitoring Agents capable of collecting granular, real-time telemetry from multi-RAT RAN nodes and other network domains, streaming this data to a central Knowledge Graph via the Integration Fabric.

[PR 5.3.3-2] All agents and network nodes involved in the TSN control plane need to be synchronized to a unified, high-precision network clock. This is a foundational requirement for calculating and enforcing the deterministic schedules essential for time-sensitive applications.

[PR 5.3.3-3] Specialized agents, under the direction of the AI Orchestrator, is required to have the capability to configure low-level control parameters on network nodes to manage TSN schedules, coordinate resources, and enforce QoS policies.

[PR 5.3.3-4] The system's Integration Fabric needs to provide secure and reliable communication channels that enable specialized agents across different domains (e.g. RAN, transport) to share state information and collaboratively plan end-to-end service configurations.

[PR 5.3.3-5] The AI Orchestrator needs to include an Intent Reconciliation Function capable of detecting and resolving conflicts between multiple TSN service requests that compete for the same limited network resources, ensuring deterministic service for all accepted intents.

5.3.4 Use Case on AI Agent-Driven Core Network Signalling Optimization

5.3.4.1 Description

The Core Network handles a wide variety of control plane processes to support registration, session establishment, mobility management, policy enforcement, and user authentication. As the number of connected devices grows and network services become more diverse, the volume, complexity, and frequency of control plane interactions increase dramatically.

This surge poses challenges in:

- Control plane overload during busy hours or traffic bursts.
- Inefficient signalling flows in dynamic or unpredictable network conditions.
- Redundant or low-value signalling messages that waste processing power and delay user service setup.

To address these challenges, AI Agents embedded within the Core Network that are co-located with the AMF, SMF, PCF, and NWDAF are leveraged to analyse real-time and historical control plane data and dynamically optimize behaviours across the network.

These AI Agents are trained to:

- Identify high-cost or redundant signalling paths.
- Detect patterns of signalling storms or inefficient UE behaviours.
- Predict signalling traffic loads and proactively offload signalling to secondary AMF instances or local UPF branches.
- Recommend changes to signalling timers, retry policies, registration areas, or PDU session procedures.
- Trigger intent-based signalling policies (e.g. "optimize for minimal registration latency" or "reduce signalling load during congestion").
- In some cases, the AI Agent can initiate signalling flow compression, reduce message exchanges for known UE behaviour profiles, or even adapt the network's state machine behaviour for specialized use cases (e.g. stationary IoT UEs).

The service flows of AI Agent-driven core network signalling optimization are as follows:

- 1) The AI Agent continuously monitors appropriate metrics from relevant Network Functions (e.g. the AMF, SMF, and PCF). Examples include registration request frequency, PDU session setups, mobility-triggered signalling, and retransmissions.
- 2) Based on policies and historical learning, the AI Agent detects excessive signalling in a specific region (e.g. frequent re-registration by stationary IoT devices).
- 3) It evaluates the situation and:
 - Recommends suppressing redundant signalling or modifying UE policy rules via PCF.

- Instructs AMF to aggregate signalling procedures (e.g. combined registration and session management).
- Informs NWDAF and OAM to adjust mobility area planning.

4) The signalling load is reduced, setup latency improves, and energy consumption is minimized.

The AI Agent continues learning from the effects of these optimizations for future scenarios.

5.3.4.2 Potential Requirements

[PR 5.3.4-1] Core Network Functions (e.g. AMF, SMF, PCF) are required to expose standardized, real-time telemetry interfaces to allow authorized Analytics Agents (such as the NWDAF) to monitor control-plane signalling metrics.

[PR 5.3.4-2] Core Network Functions need to provide secure, programmable APIs that allow an authorized Orchestrator Agent to dynamically adjust signalling policies and parameters (e.g. timers, registration areas) in real time.

[PR 5.3.4-3] The system's Intent Management Interface is required to accept high-level operational intents (e.g. "minimize signalling load for IoT devices"). The AI Orchestrator is responsible for directing the translation of these intents into specific, actionable optimization strategies.

[PR 5.3.4-4] The system is required to support a workflow where an Analytics Agent infers signalling inefficiencies from network data, and an Orchestrator Agent autonomously enacts the corresponding optimizations by issuing configuration and policy commands to the relevant Core Network Functions.

[PR 5.3.4-5] The system is required to implement a closed-loop assurance mechanism that continuously measures the impact of signalling optimizations and uses the performance outcomes as feedback to refine the underlying AI models for future decisions.

5.3.5 Use Case on AI Agent-Based Core Networks to Enhance User Experience

5.3.5.1 Description

With the rapid development of mobile Internet, users' expectations for network quality and experience have grown significantly. The mobile core network is required to ensure basic communication functions as well as cater to users' diverse and personalized demands. However, traditional core networks face significant challenges in adapting to complex and dynamic user behaviour, network conditions, and heterogeneous service requirements, often falling short of delivering an optimal user experience. Some typical examples of such are listed as follows:

- During network congestion, high-value users cannot receive timely handling and responses when their services experience abnormalities, thereby affecting the user experience.
- Insufficient analysis of user equipment performance and network adaptation leads to inefficient resource utilization and compromised user experience.

To address these challenges, AI Agents are envisioned to become deeply integrated sub-components of the mobile core network. These can perceive customer attributes, network status, and service experience in real time, which typically involves multi-agent collaboration cross technical domains (e.g. CN, OSS/BSS, IT domains). By leveraging advanced analysis and decision-making capabilities, they enable significant enhancements to the user experience throughout network usage.

These AI Agents are trained to:

- Accurately identify customer attributes and distinguish the needs of different user groups based on multi-dimensional data such as user identity, consumption level, and usage habits.
- Monitor and understand various network-specific data such as network load, link quality, node status.
- Evaluate service experience by analysing indicators such as service response time, throughput, and packet loss rate from various monitoring systems, and locate key factors affecting user experience.
- Adjust resource allocation strategies dynamically based on comprehensive analysis of customer attributes, network status, and service experience, giving priority to high-priority traffic and critical services.

- Predict user behaviour and business demand trends, schedule resources and adjust strategies in advance to achieve proactive service assurance.
- Learn from historical allocation results and services feedback to continuously optimize the core network resource scheduling model and adapt to new scenarios.

In such a setting, the service flows of AI agent-based core networks to enhance user experience are as follows:

- 1) The system performs continuous, multi-source data ingestion into a central Knowledge Graph (KG):
 - a) Monitoring Agents within the network domain stream real-time telemetry (traffic, link status, service indicators) into the KG.
 - b) A specialized BSS/IT Integration Agent securely queries the operator's CRM and BSS databases and populates the KG with relevant, anonymized customer attributes (e.g. user_ID: 123 has tier: "platinum" and active_SLA: "business_video").
- 2) A specialized Analytics Agent (akin to a 3GPP NWDAF) continuously queries the now-rich KG. It executes complex queries that correlate data across domains, such as: "Find all users with tier: "platinum" whose active video service flows are currently experiencing packet loss > 5 %". This allows the agent to identify not just network problems, but network problems that are impacting high-value customers and violating their SLAs.
- 3) According to the analysis results, the Analytics Agent detects an SLA violation for a VIP user and raises an alert. This alert triggers the central AI Orchestrator, which plans and delegates a remediation strategy:
 - a) VIP Video Conference: The Orchestrator issues a command to the Policy Control Function (PCF) in the 5G Core, instructing it to dynamically increase the QoS Class Identifier (QCI) for the VIP user's specific data flow, ensuring it receives priority treatment.
 - b) Gaming Congestion: The Orchestrator, recognizing a pattern of many users accessing the same gaming service, instructs the SMF to steer the UPF selection for these users towards an edge UPF with a more direct, low-latency path to the game servers.
 - c) Device Performance: The Orchestrator, having ingested the user's device capabilities into the KG, can instruct the PCF to apply a policy that signals an application server (e.g. a video platform) to serve a lower-bitrate stream to that specific device, ensuring a stable experience.
- 4) The Monitoring Agents continue to stream telemetry, allowing the Analytics Agent to verify that the remediation actions had the desired positive impact on the user's QoE. This feedback is used to continuously refine the AI models, improving the effectiveness of future interventions.

5.3.5.2 Potential Requirements

[PR 5.3.5-1] The system is required to provide authorized Analytics Agents with secure access to a comprehensive, cross-domain dataset. This includes real-time telemetry from network functions as well as relevant customer and service data from the operator's OSS/BSS and IT domains, ideally unified within a query able KG.

[PR 5.3.5-2] The AI Orchestrator is required to be empowered to enact decisions by issuing dynamic configuration and policy adjustments to Core Network Functions (e.g. PCF, SMF). It also needs to be able to manage the lifecycle of the underlying virtualized resources, such as scaling containerized network function instances, to meet service demands.

[PR 5.3.5-3] The system is required to implement a closed-loop assurance mechanism. The AI Orchestrator and Analytics Agent need to continuously verify that optimization actions have successfully improved the user experience and use these performance outcomes as feedback to refine their decision-making models.

6 Existing Use Case Summary

The major roles AI agents are expected to play in future networks fall into three main architectural patterns:

- 1) **AI-Driven Service Orchestration and Assurance.** This is the most common pattern, where a network-based multi-agent system is responsible for the entire lifecycle of a network service. This includes translating a user's high-level intent into a concrete service configuration, orchestrating its deployment across multiple network domains, and providing continuous, closed-loop assurance to guarantee that the service meets the user's requirements.
- 2) **Network-Hosted Agents and Digital Twins.** In this pattern, the network acts as a platform for hosting and managing agents that represent or act on behalf of users, devices, or enterprises. These agents can perform tasks autonomously, even when the end device is offline, effectively acting as a "digital twin" or proxy in the network.
- 3) **Network as a Platform (NaaP) for Agent Services.** This pattern reverses the roles. Instead of agents managing the network, the network exposes its unique capabilities - such as high-precision sensing, data analytics, and distributed edge computing - as services that can be consumed by agents running on User Equipment (UE) or by third-party applications. This "empowers" the agents by offloading computationally intensive tasks and providing them with valuable, context-rich data.

For 3GPP SA1, numerous use cases involving AI Agents have been proposed and discussed. These use cases can be broadly categorized into three groups. The first category involves integrating multiple AI Agents within the mobile network to handle tasks such as intent translation, service orchestration, task execution, and Quality of Service (QoS) assurance. The second category focuses on AI Agents performing tasks on behalf of or representing User Equipment (UE), including devices, individuals, and vehicles. The third category focuses on the network providing computational offloading for AI Agent applications running on UEs. A summary of typical use cases and their associated requirements is presented in Table 6-1.

Table 6-1: Use cases of AI Agent discussed in 3GPP SA1

Category	Title	Source as contributed to 3GPP	Summary	Main Requirements
Network AI Agents understand intent, orchestrate network services and perform tasks collaboratively	Use case on network-based intelligent assistance for autonomous driving	Huawei, HiSilicon, China Telecom, Toyota, China Mobile, China Unicom, KPN, UIC	Network provides intelligent assistance service to UEs through AI Agent	- The 6G system is able to provide suitable means for subscribers to request the services via Intent. - The 6G system is able to support AI Agents (i.e. in core network) to assure in runtime the QoS of the provided service or to provide desired network performance. - The 6G system is able to support multi-agent collaboration to perform specific tasks. - The 6G system is able to enable registration and discovery of network AI agents and their capabilities to fulfil network tasks requested via natural language. - The 6G system is able to support security identification and trusted network access for 3rd party AI agents.
	Use case on providing on-demand scalable customized services with quality assurance	Huawei, HiSilicon, China Mobile, KPN, China Telecom	Multi-agent provides customized service based on intent	
	Use case on big events assurance	ZTE, China Telecom	Multiple AI Agents are used to provide the desired network performance	
	Use case on AI-agents communication	China Mobile, China Mobile (Hangzhou) Inf., ETRI, OPPO, KPN, LG Uplus	Task-oriented communication network is established for multiple agents of UE	
	Use Case on 6G AI Agent Collaboration with Third-Party AI using LLM	Rakuten Mobile, NIST, Nokia	Network AI Agent communicates with 3 rd party agent based on texts	
	AI Agents for 6G System	NEC Corporation	Network supports multi-agent collaboration	
	Network AI agent collaboration	Nokia	Network supports multi-agent collaboration	
	Use case on AI agent-based AI service	China Telecom, Huawei, ZTE	AI Agent is used in network to understand intent and allocate tasks to network entities	
AI Agents perform tasks representing UEs	Use case on Collaborative AI Agents	KPN, China Mobile	AI Agents make decisions and execute tasks for UEs, e.g. when they are in sleep	- The 6G system is able to support hosting of large amounts of AI agent applications managed and controlled by the 6G core network and/or multiple AI Agent applications on a UE. - The 6G system is able to support secure interoperability between AI Agents and between AI Agents and applications to achieve a collaborative task. - The 6G system is able to support multi-agent interoperability through the Personal AI Agent.
	Use case on Personal AI Agent communication	LG Uplus	Personal AI Agent services in 6G networks, enabling seamless multi-agent collaboration, secure identity management, and efficient AI Agent communication for personalized and adaptive service delivery	

Category	Title	Source as contributed to 3GPP	Summary	Main Requirements
Network provides assistances or computing resources to AI agent application on UE	Use Case on 6G system assisted AI agent service	OPPO	6G system assists AI agent device for awareness and decision	- The 6G system is able to provide a suitable means for an AI agent application on UE to invoke some 3GPP services. - The 6G system is able to support a mechanism for providing Compute Services to the UE or 3rd parties and ensure security and privacy.
	Use case on personal AI agent	MediaTek Inc., Nvidia, Toyota	Personal AI agents offload the computationally intensive tasks to the network	

TM Forum discussed a set of use cases on AI agents for network management in IT related use cases, including AI agent on ongoing testing, AI agent on customer service and marketing, and home broadband self-service AI agent [i.9]. The main aspects of these are summarized in Table 6-2.

Table 6-2: Use cases on AI Agents discussed in TM Forum

Title	Key Contents
AI agent on ongoing testing	- AI agent interacts with the customer directly through text and voice, and provides effective approach to improve customer satisfaction and service reliability. - AI agent continuously monitors the service quality of customers and notify them if they decrease below acceptable levels. - AI agent hands over to a human agent or schedules a home visit when the customer issues persist.
AI agent on customer service and marketing	- AI agent assists product managers in writing business specification documents by innovating the human-computer interaction mode using AI technology. - AI agent obtains customer needs, understands customer intentions and recommends appropriate telecom services to customers by interacting with customers, which invokes the core business capabilities of the business support system to complete the whole business process.
Home broadband self-service AI agent	AI agents replace traditional Interactive Voice Response (IVR) systems or chatbots, achieve user natural language understanding, API orchestration, and provide end-to-end execution processes.

7 Conclusion and Recommendations

The present document outlines three categories of use cases for adopting AI Agents within the mobile core network. Each use case is accompanied by a set of key potential requirements and their implications for the next-generation mobile network and/or AI agents, ensuring the respective use case can be effectively supported. It is recommended that the consolidated requirements for each use case, which are summarized in Table 7-1, be considered when designing next-generation cellular systems empowered by agentic AI technology.

As illustrated by the selected use cases, the integration of AI agents in the mobile core will redefine telecommunications and is expected to significantly enhance the experience of mobile network subscribers, vertical industries, and telecom operators. This will be achieved by enabling new services beyond basic connectivity, delivering added value to businesses through improved operations and customer satisfaction, and increasing the level of autonomy in network management.

Table 7-1: Consolidated potential requirements for mobile network

Consolidated Potential Requirement Number	Consolidated Potential Requirements	Original Potential Requirement Number
CPR-1	The mobile network is able to receive and parse intents from subscribers.	PR 5.1.2-1 PR 5.2.3-1 PR 5.2.4-2 PR 5.3.1-1 PR 5.3.4-3
CPR-2	The mobile network is able to support secure communication, collaboration, information sharing and closed-loop optimization of multi-agents. See note.	PR 5.1.1-2 PR 5.1.1-3 PR 5.1.3-1 PR 5.1.3-2 PR 5.2.3-2 PR 5.2.3-3 PR 5.2.4-5 PR 5.3.1-3 PR 5.3.2-1 PR 5.3.3-4 PR 5.3.4-5 PR 5.3.5-3
CPR-3	The mobile network provides a service registry that allows agents to discover and consume network-provided services.	PR 5.1.1-4 PR 5.1.3-3 PR 5.2.4-1
CPR-4	The network is required to be capable of translating a high-level intent into an executable workflow.	PR 5.2.4-4 PR 5.3.1-2
CPR-5	The mobile network is able to generate networks to provide services on demand based on AI agents.	PR 5.2.1-1 PR 5.2.2-1
CPR-6	The mobile network is able to support multi-agent to design, deploy, execute and recycle the generated networks.	PR 5.2.1-2 PR 5.2.2-2
CPR-7	The mobile network is able to support multi-agent to ensure correct execution, runtime QoS and self-optimization of network.	PR 5.2.1-3 PR 5.2.1-4 PR 5.2.2-3 PR 5.3.2-2 PR 5.3.3-5 PR 5.3.5-1 PR 5.3.5-2
CPR-8	The mobile network is able to support AI agents to orchestrate and control tasks or network services.	PR 5.1.1-5 PR 5.2.3-2
CPR-9	The mobile network is able to expose the capabilities of network AI agents to external AI agents.	PR 5.2.3-4
NOTE: The multi-agents include network agents and UE agents.		

The study on use cases and requirements for agent-based core networks reveals that various organizations are defining novel use cases for agent-based next-generation mobile networks. Therefore, it is recommended to establish appropriate liaisons with these organizations to ensure alignment and synergy between our use cases and theirs. Furthermore, it is recommended to select a subset of these use cases for in-depth study within practical scenarios. This will help demonstrate the feasibility of generative AI applications in those use cases and assess whether the identified requirements sufficiently address real-world needs. Such practical validation will also provide valuable insights for refining the use case frameworks and identifying potential gaps or new requirements.

Annex A: Bibliography

Association for Advancing Automation (A3): "[Understanding the Importance of Robot Interoperability on the Facility Floor](#)".

Liu B, Li X, Zhang J, et al.: "Advances and Challenges in Foundation Agents: From Brain-Inspired Intelligence to Evolutionary, Collaborative, and Safe Systems". arXiv preprint arXiv: 2504. 01990, 2025.

History

Version	Date	Status
V4.1.1	October 2025	Publication