CATS Working Group
Internet-Draft
Intended status: Informational
Expires: 1 January 2026

T. Jiang
P. Liu
China Mobile

30 June 2025

CATS Reference Model for AI-Agent Communication Network
draft-jiang-cats-reference-acn-00

Abstract

This draft describes the AI-agents along with the network to provide the communication services among various types of AI-agents, i.e., AI-agent Communication Network or ACN. Thanks to the CATS-like information flow steering in ACN, we propose a CATS reference model that covers the definition of reference points, protocol stacks, service provisioning model, sigaling procedures, message paths, and implementation schemes. This reference model is generalized so as to accommodate both the existing CATS framework and the potential extension for the ACN.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at https://datatracker.ietf.org/drafts/current/.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on 1 January 2026.

Copyright Notice

Table of Contents

1. Introduction

AI agents are software-driven entities with embedded artificial intelligence, including machine learning and natural language processing, to interact multi-modally with applications, end devices and network components. AI agents may exist either in the physical state as embedded HW devices (e.g., robots) or in the virtual state (e.g.,software-implmented applications). With the integration of LLMs, AI agents can understand complex requests, translate them into actionable insights, and orchestrate various services (e.g. communication service, data analyzing service, AI-related services). AI agents play a cruical role in the Telecom domain by enhancing the network efficiency via dynamically optimizing resources, predicting network conditions, making autonomous & intelligent decisions, and facilitating seamless communication among serviced & servicing entities [AI-Agent-6G-ARC][TR.22.870].

1.1. AI-agent Communication Network (ACN)

With the imminent full unfolding of 6G era, the future world is expected to be full of AI agents, bearing versatile morphism and different capabilities. In light of the seeming differentiation between the AI-agent centric and the human-object oriented communication modes, e.g., flow interactions, requirements of capability exposure, and trust control & management models, etc., it is highly imperative to define a new network framework that is

tailored to advance the communication among AI-agents, while simultaneously satiating the requirements of existing network entities.

This new network framework is defined as the AI-agent Communication Network or ACN. ACN targets at architecting a globally interconnected network to satisfy the on-demand communication, interactions & collaborations with secure and controllable information flow paths for AI-agents in distributed deployment mode, regardless of their instantiation formats (i.e., being in physical or virtual state), capability disparities (being in high-, mid- or low-rank), and/or the hosting devices [CMCC-ACN-WP].

Commonly integrated with LLMs, AI-agents demand high computing power and significant energy consumption, which deems the versatility of the specific realization forms of AI agents. For example, an AI-agent can be instantiated as a standalone physical body, or as an intelligent service (in software state) deployed inside the hosting network, or as a cloud-native instance residing in the edge or remote cloud data centers (DCs), or even as hybrid composite entity integrating all the advantages of physical body, hosting network and cloudified deployment.

1.2. ACN Realization Models

The versatile forms of the AI-agent realization may result in three typical architecture and communication models for ACN, namely:

1. Static intra-domain only AI-agent Communication: A network architectural model for the communication among AI-agents residing within a single administrative domain or network. AI-agents in the domain form a network group maintaining the static communication association. AI-agents inside the domain do not communicate with AI-agents outside the domain.

2. Static inter-domain AI-agent Communication: A network architectural model for the communication among AI-agents that reside across multiple administrative domains. AI-agents across these domains form one or more network groups maintaining the static communication associations. Note that in this model, AI-agents in a domain can communicate with AI-agents both inside and outside the domain.

3. Dynamic multi-domain AI-agent Communication: A network architectural model for the communication among AI-agents that may dynamically form a network group to handle a temporarily-generated task. These AI-agents could be in the same or different administrative domains. Once the temporary task is

finished, the dynamically-formed network gorup would be released and the communication session(s) among the involved AI-agents are terminated. This is a dynamic communication mode versus the previous two static modes.

2. Applications of CATS to ACN

As stated in the Section 1.1, an AI-agents may manifest in different instantiation forms, e.g., embedded in a physical body, as an (software) APP service, as a cloud-native instance, or even as a hybrid composite entity. These variations imply AI-agents own different capabilities, functional objectives, resource optimizations, etc. Sometimes, the limitations of AI-agents, either those provisioning a service or those realizing a service, may lead AI-agents in an ACN to pursue the service optimization with the principles that are commensurate with CATS's objectives [CATS-PS-UseCase-Req].

2.1. AI-agents Services with CATS-like Optimization

AI-agents in an ACN demand normally intensive compute power and accordingly heavy energy consumptions. However, the capabilities (either statically provisioned resources or dynamic runtime loads) among AI-agents, the AI related demands, and the data processing tasks varies dramatically. For example, the compute power of lightweight terminals, such as smartphones, XR glasses, etc., is difficult to handle locally the complex computation tasks with more than billions of parameters. In contrast, if all the complex tasks are delegated to more advanced cloudified instances (in either SW or HW format residing in a remote data center), then it might impair the real-time responsiveness if the remote instances experience the bursty load of multi-users.

Therefore, it is more desirable to consider the seamless collaborations among end devices, networks and (edge or remote) clouds to build a composite AI-agent communication network, in which AI logics are distributed either at the network edge or within the network, either inside or across domains. In that, the compute power at end devices may realize hierarchical AI reasoning with the help of more powerful network entities, which expands the end-side AI services on demand. Further, the network can also provide more advanced AI services, e.g., Ambient IoT [TS.23.369], integrated sensing [TR.23.700-14], etc., to supplement the requirements of (end) AI-agents and lower the compute demands in them. The ultimate goal would be the better balance between achieving the intelligence at AI-agents and the lower energy consumption (of course, potentially more advantages). This certainly conforms to what CATS is promoting.

The Section 1.2 exemplifies three different ACN realization models, i.e., the static intra-domain, the static inter-domain, and the dynamic multi-domain. AI-agents offering varied types of services could reside in any domain in a (multi-domain) ACN. Supposing a complex task is comprised of multiple sub-tasks that need to be handled in a sequence, and every sub-task may be potentially serviced by more than one AI-agent. If these AI-agents are distributed across different domains of an ACN, the service-chaining formed from the (across-domain) AI-agents makes the task processing more challenging. In this scenario, the application of CATS principles to the selection of the optimal AI-agent (among all candidates) for each sub-task may help fulfill the complex task more efficiently.

2.2. CATS-like Metrics Model for ACN

The CATS IETF draft on metrics definition [CATS-Metrics-Definition] specifies two types of metrics, namely the traditional network metrics that focus on the network resources and dynamic runtime information, and the compute metrics that describes the functional capabilities, resource consumption, system performance, etc., for service instances which would normally reside in edge or remote DCs.

* Network metrics: For network entities like routers or switches, they can be bandwidth, capacity, throughput, transmission delay, TX bytes, RX bytes, host bus utilization, etc.

* Compute metrics: For compute nodes, end servers, and/or service instances, they can be CPU, GPU, NPU, memory, storage, system delay.

When the similar metrics model is extended to the AI-agent Communication Network or ACN, there would be a new type of metrics, defined as the 'AI-agent metrics' in this draft, to specify the unique characteristics of AI-agents. These metrics could consist of:

* AI-agent metrics: AI-agent functionalities & capabilities, AI model types, #parameters of models, authentication/authorization policy, etc.

We think the protocol stack of the AI-agent in an ACN should reside above the network & transport layers, which might be considered as in the application layer. Correspondingly, the metrics specifically associated with AI-agents would be exchanged among AI-agents themselves, which makes the peering relationship for the AI-agent message exchanging different from that for the network and the compute metrics.

As shown in the Figure 1, the AI-agents reside on the App-layer, sitting above the network and compute entities. The existing CATS metrics (network & compute) are targeted toward the traffic steering at the network layer, which might be achieved via the network protocol extension. In comparison, the AI-agent metrics are generated for the overlay-exchange among App clients (those served as AI-agents), which are not generally subject to the signaling path over network protocols.

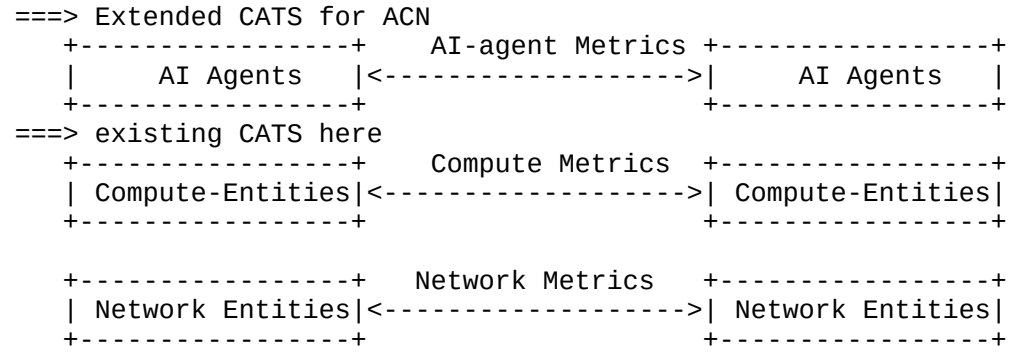

Figure 1: AI-agent Protocol Stack

In the following section, we will define a CATS-based holistic model operating on general reference points that could be leveraged for the signaling exchanges of all the three types of metrics, i.e., network, compute and AI-agent metrics.

3. CATS Reference Model for ACN

The Section 2.1 provides use cases to explain why the CATS scheme may be applicable to optimize the AI-agent services in an ACN. The Section 2.2 describes two existing CATS metrics, i.e., network and compute metrics, as well as defines a new metric type, i.e., the AI-agent metrics. The same section also explains the protocol stack and the interactions among AI-agents themselves and between the CATS compute- & network- entities. The uniqueness of AI-agents along with their instantiation states and the corresponding deployment models of ACNs make the associated metrics exchange model different from what is possibly adopted by the existing CATS metrics. Thanks to the variations among the three metrics, we propose a holistic CATS reference model to accommodate the metrics extension of AI-agents in an ACN.

The Figure 2 demonstrates the integrated CATS reference architecture that accommodates the existing C-PS, C-NMA, C-SMA as well as the (new) AI-agents in ACN. The AI-agent#0 is a physical-form AI-agent (e.g., embedded in a server) and the AI-agent#1 is a virtual instance deployed in the cloud service site#1. The service instances #2, #3a and #3b are normal CATS instances deployed in the site#2 and site#3, respectively. The CATS entity C-SMA-2 is in the service site#2 and the C-SMA-3 in the site#3, handling the capture, processing and distribution of compute metrics at service sites. The provider network in the figure contains two CATS network metric agents, i.e., C-NMA-2 and C-NMA-3, for the handling of network metrics. Both C-SMAs and C-NMAs talk to the CATS entity C-PS for metrics distribution. Here C-NMAs, C-SMAs and C-PS are defined in [CATS.Framework].

When the CATS framework is extended to accommodate the AI-agent metrics in an ACN, there will be either a new type of CATS agent to be introduced, e.g., named as CATS AI-agent Metric Agent or C-AMA that can talk with C-PS, or the AI-agents directly engaging & communicating with C-PS.

C-NMA: CATS Network Metric Agent
C-SMA: CATS Service Metric Agent
C-AMA: CATS AI-agent Metric Agnet (new)

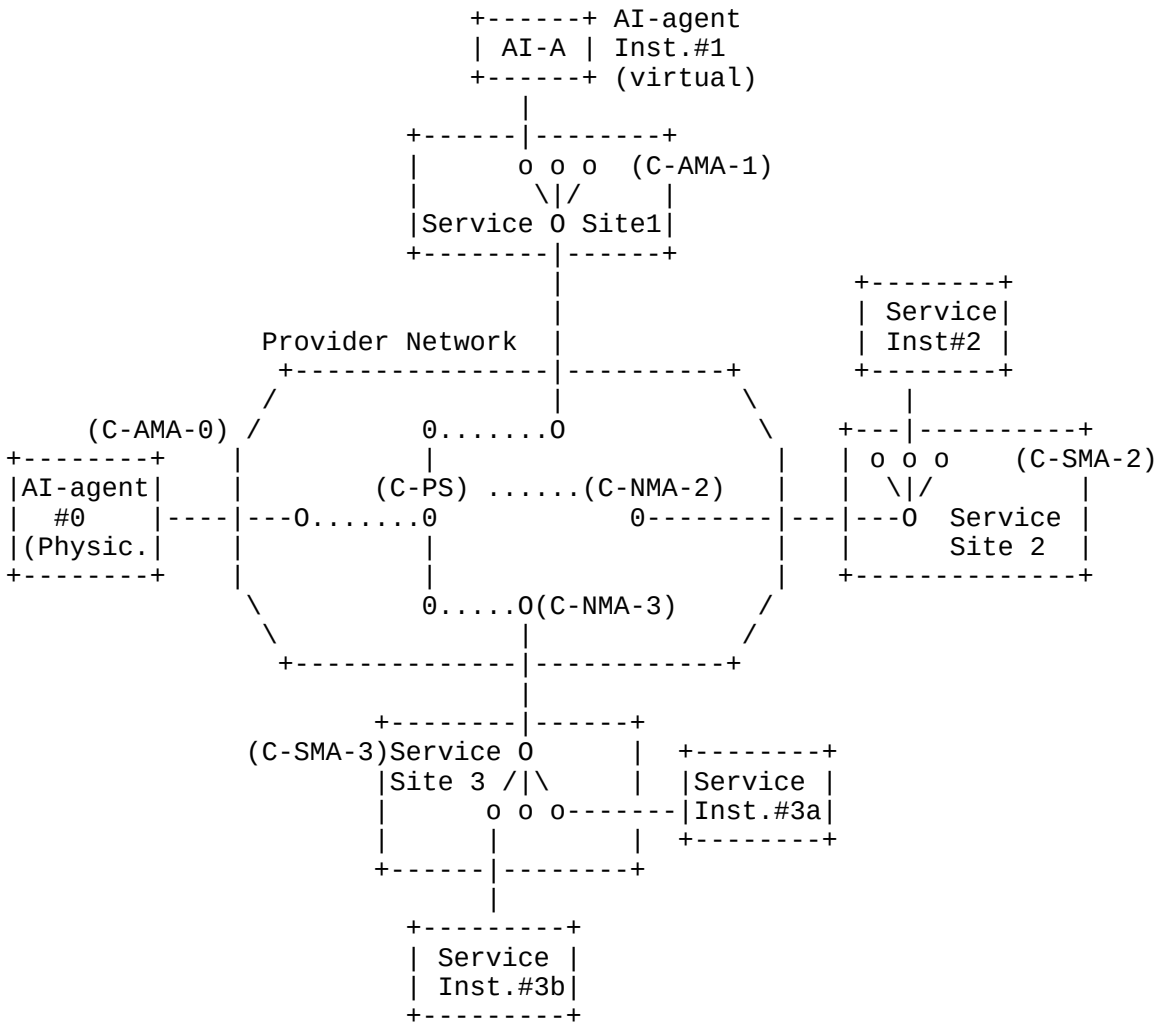

Figure 2: CATS Reference Architecture for Integrated ACN

3.1. CATS Reference Model and Reference Points

We propose to define a unified & generalized CATS reference model with reference points for the signaling process between CATS entities. CATS entities consist of C-PS, C-NMA, C-SMA and, as introduced in the draft, C-AMA. The reference points shall be standardised to support the functionalities and the interactions over the reference interfaces between CATS entities.

The CATS reference model with reference points shall cover the following aspects:

* Service model: Proposed to be a producer-consumer model, with each CATS entity being either a producer or a consumer or both.

* Reference interfaces and points (between entities): definition, identity, name, parameters, etc.

* Singaling messages: definitions, parameters, types of exchanged messages (network-, compute-, and AI-agent metrics), etc.

* Singaling & management procedures: may include:

  - CATS entity registration, authentication and authorization.

  - Peer discovery and selection: the discovery scheme(s) can be from either the draft [IETF-Cisco-AIagent-draft] and the 3GPP NRF-like scheme [TS.23.501] that are applicable within the same domain, or the IETF MSDP-like scheme [RFC3618] applicable across domains.

  - Peering session establishment, message-exchange, peering-state sync-up, peering-session update and modification, and peering release, etc.

* Overlay-based implementation and protocol stacks. Please reference to the Section 2.2 for protocol stack discussion.

* Communication channels & implementation schemes, possibly being

  - Authenticated APIs: For example: REST API methods (get, post, put, delete, etc.).

  - Message brokers: a SW/HW intermediary, facilitating communication and data exchange between different CATS entities and AI-agents.

3.2. Examples of CATS Reference Points

The Figure 3 applies the CATS reference model to exemplify the reference points and reference interfaces between different CATS entities and AI-agents. For example, the reference point "RP_ps_nma" is between the C-PS and the C-NMA, and "RP_aia_ps" is between the AI-agent and the C-PS. Note that the (c1) and (c2) are reference interfaces within the scope of their associated reference point.

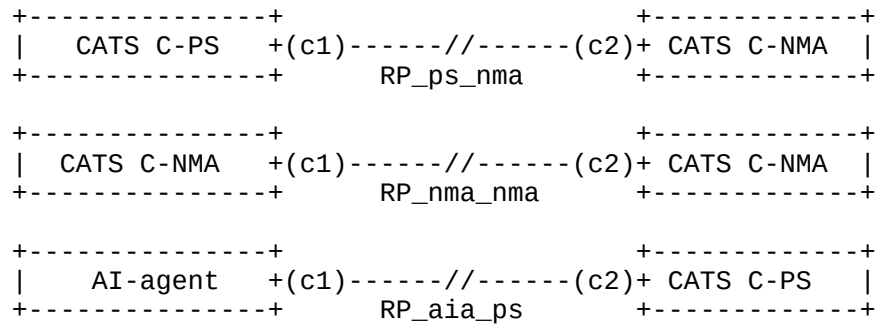

Figure 3: Reference points & interfaces btwn C-PS and C-NMA

4. Security Considerations

There is no security concern.

5. IANA Considerations

There is no IANA requirement.

6. References

6.1. Normative References

[CATS-Metrics-Definition]
Yao, K., et al., "CATS Metrics Definition", draft-ietf-cats-metric-definition, March 2025.

[CATS-PS-UseCase-Req]
Yao, K., et al., "Computing-Aware Traffic Steering (CATS) Problem Statement, Use Cases, and Requirements", draft-ietf-cats-usecases-requirements, June 2025.

[CATS.Framework]
Li, C., et al., "A Framework for Computing-Aware Traffic Steering (CATS)", draft-ietf-cats-framework, June 2025.

[IETF-Cisco-AIagent-draft]
Rosenberg, J., et al., "Framework, Use Cases and Requirements for AI Agent Protocols", draft-rosenberg-ai-protocols, May 2025.

[RFC3618] Fenner, B., Ed. and D. Meyer, Ed., "Multicast Source Discovery Protocol (MSDP)", RFC 3618, DOI 10.17487/RFC3618, October 2003, <https://www.rfc-editor.org/info/rfc3618>.

[TR.22.870]
"3GPP TR 22.870 v0.3.0: Study on 6G Use Cases and Service Requirements; Stage 1, Rel-20", 3GPP TR 22.870, May 2025.

[TR.23.700-14]
"3GPP TR 23.700-14 v0.2.0: Study on Stage 2 for Integrated Sensing and Communication", 3GPP TR 23.700-14, June 2025.

[TS.23.369]
"3GPP TS 23.369: Architecture support for Ambient power-enabled Internet of Things; Stage 2", 3GPP TS 23.369, June 2025.

[TS.23.501]
"3GPP TS 23.501 (V19.0.0): System Architecture for 5G System; Stage 2", 3GPP TS 23.501, June 2024.

[TS.23.502]
"3GPP TS 23.502 (V19.0.0): Procedures for the 5G System; Stage 2", 3GPP TS 23.501, June 2024.

6.2. Informative References

[AI-Agent-6G-ARC]
"Enabling Mobile AI Agent in 6G Era: Architecture and Key Technologies", https://dl.acm.org/doi/abs/10.1109/MNET.2024.3422309, September 2024.

[CMCC-ACN-WP]
"AI-agent Communication Network White Paper", CMCC ACN White Paper, July 2024.

Authors' Addresses

Tianji Jiang
China Mobile
Email: tianjijiang@yahoo.com

Peng Liu
China Mobile
Email: liupengyjy@chinamobile.com