

CATS
Internet-Draft
Intended status: Standards Track
Expires: 13 April 2026

H. Fu
ZTE Corporation
B. Liu
Z. Li
China Mobile
Q. Xiong
ZTE Corporation
10 October 2025

Computing-Aware Traffic Steering (CATS) Operations, Administration, and
Maintenance (OAM) Framework
draft-fu-cats-oam-fw-04

Abstract

This document describes the OAM framework and requirements for Computing-Aware Traffic Steering (CATS). The framework defines the CATS OAM layering model and OAM components. It also describes the requirements to enable the fault and the performance management of end-to-end connections from clients to networks and finally to services instances.

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 13 April 2026.

Copyright Notice

Copyright (c) 2025 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<https://trustee.ietf.org/license-info>) in effect on the date of publication of this document.

Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Revised BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Revised BSD License.

Table of Contents

1. Introduction	2
2. Requirements Language	3
3. Terminology	3
4. Motivation	4
5. CATS OAM Framework	5
5.1. CATS OAM Layering Model	5
5.2. CATS OAM Components	6
5.2.1. SI-OAM Component	6
5.2.2. TC-OAM Component	7
5.2.3. AF-OAM Component	7
6. CATS OAM Requirements	7
6.1. Operation	7
6.2. Administration	8
6.3. Maintenance	8
7. Deployment Considerations	8
8. Security Considerations	10
9. Acknowledgements	11
10. IANA Considerations	11
11. References	11
11.1. Normative References	11
11.2. Informative References	12
Contributors	12
Authors' Addresses	12

1. Introduction

As described in [I-D.ietf-cats-usecases-requirements], edge computing provides lower response time and higher transmission rate than cloud computing by moving computing instances to the network edge. To meet the requirements of users that are highly distributive, service providers deploy the same type of service instances at multiple edge sites, which involves steering traffic from clients to the most appropriate computing instance.

Compute-aware traffic steering (CATS) [I-D.ietf-cats-framework] is a traffic engineering approach as per [I-D.ietf-teas-rfc3272bis] developed to address the aforementioned traffic steering problem. This approach takes into account the dynamic nature of both the computing resources and the network states to optimize the way that traffic is forwarded towards a given service instance. Various

metrics can be taken into account to devise and enforce such service-specific and computing-aware traffic steering policies. To achieve better service assurance, it is necessary to not only rapidly detect whether the QoS provided by the computing networks meets the SLA requirements of clients, but also dynamically trigger the calculation and the adjustment of both the computing and the networking services. There are some OAM technologies developed for networks, but they are only deployed to facilitate the operations and the maintenance of network operators, and cannot provide measurements of an end-to-end connection from a client to a service instance.

To this end, based on the CATS framework as per [I-D.ietf-cats-framework], this document describes the OAM framework and requirements for Computing-Aware Traffic Steering (CATS). The framework defines the CATS OAM layering model and OAM components. It also describes the requirements to enable the fault and the performance management of end-to-end connections from clients to networks and finally to services instances. The deployment considerations are also described as well.

2. Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

3. Terminology

This document makes use of the terms defined in [I-D.ietf-cats-framework].

- * FM: Fault Management.
- * PM: Performance Monitoring.
- * SI-OAM: Service Instance OAM.
- * TC-OAM: Traffic Classifier OAM.
- * AF-OAM: Application Flow OAM.
- * IOAM: In-situ OAM.

4. Motivation

The main objectives of OAM are to detect anomalies before they intensify, reduce the number of traffic flows impacted by these abnormalities, and ensure that network operators fulfill their QoS guarantee commitments to meet the Service Level Agreement (SLA) of clients.

As a traffic engineering method, computing-aware traffic steering (CATS) takes into account the dynamic nature of both the computing resources and the network states to optimize the way that traffic is forwarded toward a given service instance. However, existing OAM technologies cannot be used to collect metrics associated with the computing resources. Therefore, it is necessary to extend the existing OAM technologies to build an end-to-end OAM for CATS. Key objectives include:

- * Convergence latency is compressed from the order of tens of seconds to sub-second timescale: In CATS, the status information of the computing instances is collected by the CATS Service Metric Agent (C-SMA) component and processed at the control plane for performance monitoring and failure detection. However, to limit control-plane load, such sensing mechanisms are typically engineered to operate on the order of tens of seconds.. Accordingly, rapid detection of data-plane degradation affecting both service instances and network states is mandatory, so that CATS Path Selector (C-PS) convergence is triggered and its latency compressed from tens of seconds to sub-second scale.
- * Closed-loop network path evaluation : In CATS, the CATS Path Selector (C-PS) calculates and selects the paths towards appropriate egress PEs and computing service instances. In this process, it is necessary to verify whether the calculation and the selection results meet the SLA requirements of clients taking into account both the network states and the computing instance status.
- * Closed-loop service SLAs guarantee for flows : In CATS, subsequent packets of service flows in an established session are forwarded through the CATS Traffic Classifier (C-TC) to the same service instance. However, during such a process, the computing/network performance may degrade. To ensure consistent experience for end users, it is necessary to measure the flow-level performance of service instances and make appropriate adjustments, e.g., change segments of routing paths or enable backup paths, according to the SLA requirements.

- * Fault delimiting and troubleshooting: When user experience deteriorates, it is necessary to rapidly locate the fault on the end-to-end path from the user terminal through the network to the service instance to implement fast end-to-end fault location and troubleshooting.

5. CATS OAM Framework

5.1. CATS OAM Layering Model

The CATS OAM layering model is shown in Fig. 1. In this architecture, both the CATS router and the underlay node are deployed with the existing OAM technologies. These OAM technologies are used to detect anomalies and monitor service performance in the network domain, and can be divided into three categories: link OAM, tunnel OAM, and service OAM.

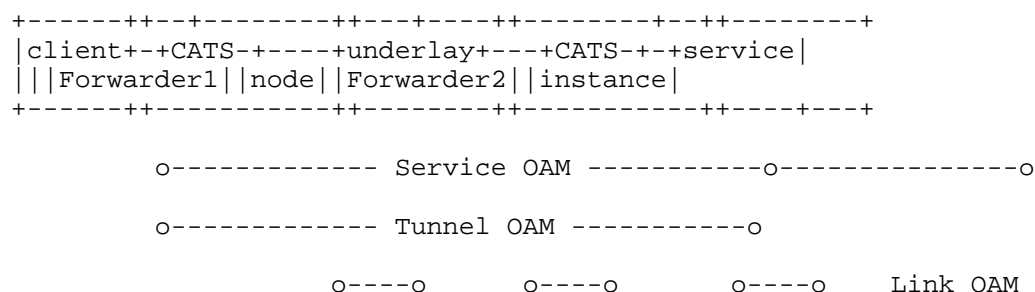


Figure 1: CATS OAM Layering Model

- * In link OAM, anomaly detection and performance monitoring are conducted for a single ethernet link. The link layer is an optional sublayer implemented in the data link layer between the Logical Link Control (LLC) and the MAC sublayer in the Open Systems Interconnection (OSI) model. Common detection tools of link OAM include IEEE-802.3ah.
- * A tunnel bears multiple services so the tunnel OAM must ensure that the performance of a given service is not degraded when the network fails or the number of services in the tunnel increases. As a result, failure detection and performance monitoring are conducted on the LSP layer to implement service protection. Common detection tools of tunnel OAM include ITU-T Y.1711, MPLS-LM-DM, BFD, etc.

- * Service OAM is generally conducted for the L2VPN/L3VPN service layer that is provided by the network to evaluate the service quality and protect services. Common detection tools of service OAM include ITU-T Y.1731, TWAMP, STAMP, etc.

CATS simultaneously steers traffic along network paths and toward compute instances. Within the network domain the three conventional OAM mechanisms remain applicable, yet link-level OAM can at best cover the direct link between compute instances; no effective OAM exists from the ingress/egress gateways to the compute instances themselves. Moreover, the introduction of flow-affinity policies mandates that end-to-end quality assessment of service flows span both network and compute domains.

5.2. CATS OAM Components

The CATS OAM layering model should flexibly support existing OAM detection tools and it consists of the following three components, SI-OAM, TC-OAM and AF-OAM as Figure 2 shown.

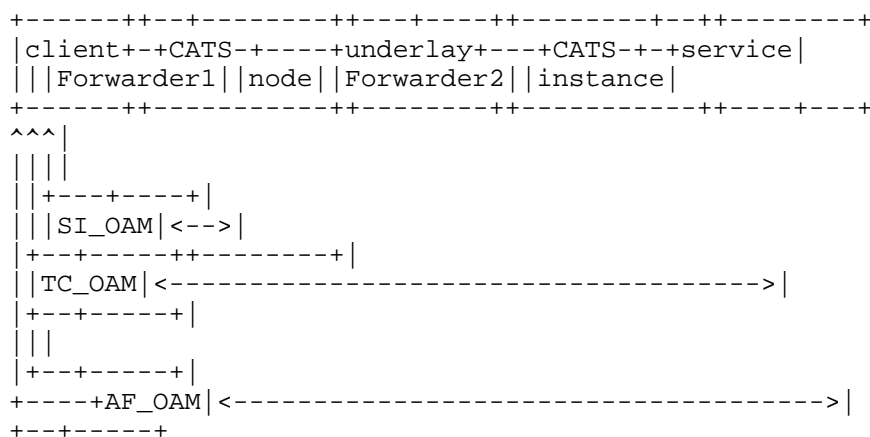


Figure 2: CATS OAM Functional Components

5.2.1. SI-OAM Component

The functions of this component include (but are not limited to) detecting the failures that happen between the CATS-Forwarder 2 and the service instance, and measuring the associated metrics such as latency, packet loss, and bandwidth. The SI-OAM component generally would not dive into the internal structure of the network between the CATS-Forwarder 2 and the service instance and only makes the measurements of the end-to-end connection. These measurements are

generally fed back to the C-SMA component to achieve faster failure detection and performance monitoring than the CATS control plane.

5.2.2. TC-OAM Component

The functions of this component include but are not limited to detecting the failures that happen between the CATS-Forwarder 1 and the service instance of a certain specific ID, and measuring the associated metrics such as delay and packet loss. The testing packets are delivered through the CATS Path Selector (C-PS) to the associated service instance according to the corresponding forwarding table entry of the CATS Traffic Classifier (C-TC) to verify whether the measurements of the connection meet the service level agreement (SLA) requirements. And if it does not, recalculation is triggered.

5.2.3. AF-OAM Component

The functions of this component include but are not limited to measuring the metrics such as delay, packet loss, and bandwidth, of the service flow in CATS. In general, the user experience of an active connection may be affected by a number of factors, such as the processing latency of the service instances may increase or the network performance may degrade due to the increase of the incoming traffic to the service instance. For CATS-Forwarder 1, it is necessary to evaluate whether the SLA requirements of service flows are achieved, and if the SLA requirements are not achieved, conduct appropriate path adjustments to compensate for the deviation as much as possible to ensure the clients have consistent experience. For client terminals, if the experience is degraded, it is necessary to accurately locate where the problem occurs and quickly conduct troubleshooting. It should be noted that related OAM tools can also be developed, so that the entire network stack (L2-L7) can be observed for applications and the entire network stack, instead of merely traditional application-level visibility or network-level visibility, providing a comprehensive solution for operators' efficiency.

6. CATS OAM Requirements

6.1. Operation

- * Sub-second/second-granularity telemetry SHALL be collected for CPU, GPU, memory, accelerator utilization and energy consumption to produce unified compute metrics (e.g., TOPS/W, TFLOPS).

- * These metrics SHALL be fused with network telemetry to generate an integrated “compute-network” telemetry stream encompassing packet loss, latency, throughput and compute load, providing real-time decision inputs to the C-PS.

6.2. Administration

- * Compute-resource provisioning: A node SHALL present a compute-capability template (type, capacity, affinity) at boot; OAM SHALL authenticate the template and synchronize it to the network-wide routing database.
- * Service contract and billing: OAM SHALL generate a billing model from multi-dimensional factors—compute class, usage duration, network distance—and push the model to edge controllers.
- * Unified orchestration: OAM SHALL abstract compute workloads into routable Compute-SIDs and, together with network SIDs, inject them into the SRv6/BGP SR Policy orchestration plane to enable resource scheduling across domains, clouds, and edges.

6.3. Maintenance

- * End-to-end quality assessment: 1) Network segment: Employ BFD, TWAMP and IOAM to detect link/node faults; convergence latency SHALL be 50 ms. 2) Compute segment: Utilize keep-alive plus health probes to monitor container/VM/accelerator liveness; crashes or overload SHALL be detected within seconds.
- * Fault correlation and localization: OAM SHALL correlate “compute unavailable” events with “network-path degradation” events to determine whether the root cause is resource exhaustion or packet loss, eliminating needless path shifts.
- * Intelligent self-healing: 1) Compute-node failure SHALL trigger the CATS Path Selector to re-select a path and move traffic in real time to an alternate node in the same or a remote pool. 2) Network-link failure SHALL invoke TI-LFA/SR-TE protection switching within < 50 ms while simultaneously updating the compute topology to prevent black-holing.

7. Deployment Considerations

To demonstrate the complete CATS OAM procedure, a proper OAM detection tool needs to be selected and deployed on the network and service instance hosts of the CATS OAM architecture. The selection of OAM detection tools is out of the scope of this document.

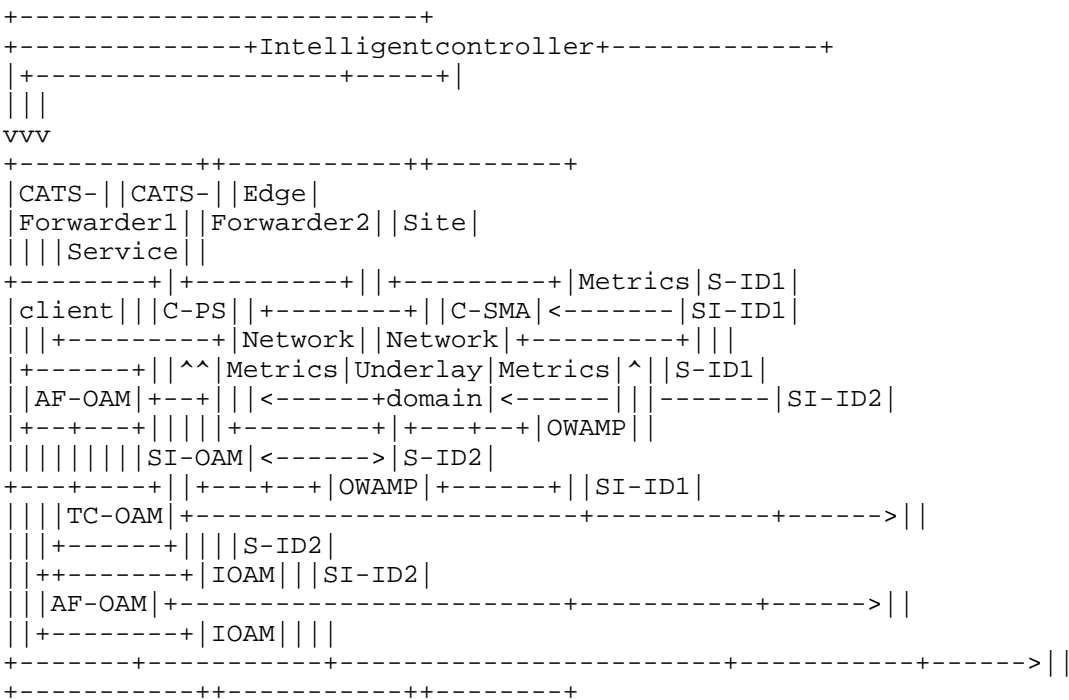


Figure 3: An Example Of CATS OAM Deployment

As illustrated in Fig. 3, the OWAMP and the IOAM tools are selected as examples to describe how the CATS OAM component works with these detection tools to fulfill the four objectives :

- * Convergence latency is compressed from the order of tens of seconds to sub-second timescale: The SI-OAM component is deployed on the CATS-Forwarder 2 and the OWAMP tool is used to measure the delay and packet loss from the CATS-Forwarder 2 to the associated service instance. The source and the destination IP of the detection packets are the CATS-Forwarder 2 interface IP and the service instance IP, respectively. According to the returned packets, the status and the metrics of both the service instance and the network that connects the service instance with the clients are obtained. The SI-OAM component feeds back the measurement results to the C-SMA component, which further spreads the computing resource information in the CATS network to accelerate CATS Path Selector(C-PS) convergence to avoid black holes.

- * Closed-loop network SLA guarantee: The TC-OAM component is deployed on the CATS-Forwarder 1 and the OWAMP tool is used to measure the delay and packet loss from the CATS-Forwarder 1 to the associated service instance. To ensure OWAMP packets are delivered according to the table item of TC, the source and the destination IP addresses of the detection packets are set to the IP address of the interface of CATS-Forwarder 1 and the IP address corresponding to the service ID, respectively. OWAMP packets usually pass through the tunnel to the egress network and are forwarded to the service instance. According to the returned OWAMP packets, the TC-OAM obtains the measurement results and feeds back the results to the C-PS component. If the measurement results deviate from the expected SLAs, recalculation is triggered to fulfill the closed-loop network SLA guarantee for the service ID.
- * Closed-loop SLA guarantee for service flow: for service flows that have been initiated, the flow affinity function is executed to guarantee that subsequent packets reach the same service instance as the first packet. To conduct measuring and performance monitoring for the entire end-to-end flows, the flow-based detection tool such as IOAM is selected and the AF-OAM component is deployed on the CATS-Forwarder 1. Note that the PostCard or the PassPort modes are generally used in the flow-based detection and a centralized collector is required to obtain the measurement results and feed the results back to the C-PS. The network path can be adjusted according to the difference between the OAM measurement results and the SLA requirements to ensure a consistent user experience.
- * Service fault delimiting and troubleshooting: For fast delimitation and troubleshooting under user experience degradation, the AF-OAM component can be deployed on a user terminal when a flow detection tool such as IOAM is performed. The IOAM can use the postcard mode and can directly report the location where packet loss or longer delay occurs according to the measurement results obtained by a centralized collector. This is a typical scenario of IOAM, and details are not described herein.

For different detection targets, flexible choices of detection protocols and mechanisms can be made, which will not be elaborated upon here.

8. Security Considerations

To be discussed in future versions of this document.

9. Acknowledgements

To be added upon contributions, comments and suggestions.

10. IANA Considerations

TBD.

11. References

11.1. Normative References

- [I-D.ldbcs-cats-framework]
Li, C., Du, Z., Boucadair, M., Contreras, L. M., and J. Drake, "A Framework for Computing-Aware Traffic Steering (CATS)", Work in Progress, Internet-Draft, draft-ldbcs-cats-framework-06, 8 February 2024, <<https://datatracker.ietf.org/doc/html/draft-ldbcs-cats-framework-06>>.
- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <<https://www.rfc-editor.org/rfc/rfc2119>>.
- [RFC4656] Shalunov, S., Teitelbaum, B., Karp, A., Boote, J., and M. Zekauskas, "A One-way Active Measurement Protocol (OWAMP)", RFC 4656, DOI 10.17487/RFC4656, September 2006, <<https://www.rfc-editor.org/rfc/rfc4656>>.
- [RFC7276] Mizrahi, T., Sprecher, N., Bellagamba, E., and Y. Weingarten, "An Overview of Operations, Administration, and Maintenance (OAM) Tools", RFC 7276, DOI 10.17487/RFC7276, June 2014, <<https://www.rfc-editor.org/rfc/rfc7276>>.
- [RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <<https://www.rfc-editor.org/rfc/rfc8174>>.
- [RFC8402] Filsfils, C., Ed., Previdi, S., Ed., Ginsberg, L., Decraene, B., Litkowski, S., and R. Shakir, "Segment Routing Architecture", RFC 8402, DOI 10.17487/RFC8402, July 2018, <<https://www.rfc-editor.org/rfc/rfc8402>>.

- [RFC8754] Filsfils, C., Ed., Dukes, D., Ed., Previdi, S., Leddy, J., Matsushima, S., and D. Voyer, "IPv6 Segment Routing Header (SRH)", RFC 8754, DOI 10.17487/RFC8754, March 2020, <<https://www.rfc-editor.org/rfc/rfc8754>>.
- [RFC9378] Brockners, F., Ed., Bhandari, S., Ed., Bernier, D., and T. Mizrahi, Ed., "In Situ Operations, Administration, and Maintenance (IOAM) Deployment", RFC 9378, DOI 10.17487/RFC9378, April 2023, <<https://www.rfc-editor.org/rfc/rfc9378>>.

11.2. Informative References

- [I-D.ietf-cats-usecases-requirements]
Yao, K., Contreras, L. M., Shi, H., Zhang, S., and Q. An, "Computing-Aware Traffic Steering (CATS) Problem Statement, Use Cases, and Requirements", Work in Progress, Internet-Draft, draft-ietf-cats-usecases-requirements-07, 10 June 2025, <<https://datatracker.ietf.org/doc/html/draft-ietf-cats-usecases-requirements-07>>.
- [I-D.ietf-teas-rfc3272bis]
Farrel, A., "Overview and Principles of Internet Traffic Engineering", Work in Progress, Internet-Draft, draft-ietf-teas-rfc3272bis-27, 12 August 2023, <<https://datatracker.ietf.org/doc/html/draft-ietf-teas-rfc3272bis-27>>.

Contributors

Daniel Huang
ZTE Corporation
Email: huang.guangping@zte.com.cn

Cheng Huang
ZTE Corporation
Email: huang.cheng13@zte.com.cn

Wei Duan
ZTE Corporation
Email: duan.weil@zte.com.cn

Authors' Addresses

Huakai Fu
ZTE Corporation
Email: fu.huakai@zte.com.cn

Bo Liu
China Mobile
Email: liubo@chinamobile.com

Zhenqiang Li
China Mobile
Email: lizhenqiang@chinamobile.com

Quan Xiong
ZTE Corporation
Email: xiong.quan@zte.com.cn