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

Y. Zhao
L. Han
China Mobile
X. Li
H. Zheng
Huawei
D. King
Old Dog Consulting
28 February 2026

Framework and Applicability of Computation-aware Traffic Steering (CATS) in Optical Transport Networks (OTN)
draft-zhao-cats-otn-applicability-00

Abstract

Computation-aware Traffic Steering (CATS) offers a framework for selecting computation service sites based on computation capabilities and load, and considering the network capabilities and state on the paths to the sites.

Optical Transport Networks (OTN) provide guaranteed separation of traffic along with reserved hardware resources offering bandwidth and quality of service promises.

This document describes how OTN may be used to support a CATS system to achieve the stringent performance targets required by demanding service environments.

Table of Contents

1. Introduction

Computing service architectures have evolved toward multi-site environments, where collaborative service sites work together to optimize performance. This decentralized approach addresses critical issues like long response times and ensures a more efficient use of service and network resources, avoiding localized resource under-utilization or exhaustion.

Networking infrastructures that incorporate computing resources have typically employed static service dispatching mechanisms, particularly for the selection of service instances. Within these architectures, service-specific traffic is frequently steered toward the nearest service site based on optical network service availability (such as fixed light-path or pre-established connections). This approach, however, often overlooks the real-time network state (e.g., utilization or congestion) and the dynamic service site state (e.g., GPU availability).

Consistent with the use cases and requirements described in [I-D.ietf-cats-usecases-requirements], various services stand to benefit from traffic steering that integrates knowledge of network capabilities and state with computing resource metrics (such as capabilities and current usage). AI large-model training, some AI inference jobs, and distributed computing workloads impose stringent requirements on network determinism. These tasks rely on high-bandwidth, deterministic latency, and minimal jitter to ensure efficient synchronization between massive GPU clusters. Although the Computing-Aware Traffic Steering (CATS) framework [I-D.ietf-cats-framework] supports making joint compute- and network-aware decisions, the utilization of Optical Transport Network (OTN) features offers a reliable "hard-isolation" infrastructure. This integration is particularly effective for achieving the stringent performance targets required by demanding service environments.

Current enterprise environments frequently distribute AI training and inference workloads across hybrid infrastructures, including on-premises and cloud-based networks. To ensure high availability and responsiveness, the CATS framework enables a specific service to be delivered through one or more service instances deployed across multiple service sites. These service instances are reached by clients via service contact instances. While a single service site, such as an intelligent computing center, can host multiple service

contact instances, its available computing resources (e.g., GPU memory or FLOPS) may be constrained at any given time (usually because they are in use for other services). Since resource availability fluctuates across different service sites, steering traffic via dynamically reconfigurable optical paths provides an effective mechanism to mitigate resource limitations within a specific service site.

The primary objective of traffic steering within the CATS framework is to identify an optimal service contact instance for each request, based on a combination of network and computing metrics. In certain scenarios, such as hierarchical or recursive contexts, this selection process does not necessarily expose the specific service instance that ultimately handles the client’s invocation. Instead, only a service contact instance that acts as a gateway to multiple service instance is identified. Consequently, the metrics associated with a service contact instance may represent aggregate metrics derived from a collection of underlying service instances.

Achieving deterministic performance for packet-based (e.g., IP) traffic steering is challenging because path selection and forwarding are performed on a hop-by-hop basis, which may introduce variability in latency, jitter, and queuing behavior. This limitation may render packet-based CATS insufficient to meet the strict performance requirements of highly performance-sensitive use cases, such as AI training and tele-health.

This document introduces CATS-aware OTN which is intended to complement packet-based CATS by providing deterministic transport capabilities to support highly performance-sensitive use cases. It maps service flows into optimized optical containers (e.g., ODUk or OSU). By incorporating optical-layer characteristics (e.g., deterministic path latency, wavelength continuity constraints, and optical link performance parameters) together with computing-layer metrics, the framework enables the establishment of an end-to-end "hard-isolation" capable of delivering the performance stability required by AI cluster workloads.

The CATS framework serves as an overlay architecture designed to facilitate the selection of optimal service contact instances among multiple candidates. The determination of whether a service instance is deemed ’suitable’ depends on a multi-dimensional evaluation of both networking and computing metrics. This document extends the application of the CATS framework into the OTN domain, specifying how optical path computation and connection establishment can be made compute-aware.

Additionally, this document outlines the operational workflow of the primary CATS procedures (see Section 4) as they are implemented across both the control and data planes within a CATS-aware OTN infrastructure. It is assumed that the CATS functional elements are situated within a single provider network. Consequently, deployment scenarios involving the co-location of these elements at the client site are considered out of scope for this discussion.

2. Terminology

The following terms are defined in [I-D.ietf-cats-framework] and are not redefined here:

* Client
* Computing-Aware Traffic Steering (CATS).
* Metric
* Computing metrics
* Service
* Computing Service
* CATS Service ID (CS-ID)
* Service instance
* Service contact instance
* CATS Service Contact Instance ID (CSCI-ID)
* Service request
* CATS Path Selector (C-PS)
* CATS Service Metric Agent (C-SMA)
* CATS Network Metric Agent (C-NMA)
* CATS forwarder

The following definitions are extended from those provided in [I-D.ietf-cats-framework].

* Flow: A set of packets or signals grouped logically over a defined period. Within the context of CATS-aware OTN, a flow is generally encapsulated into an Optical Data Unit (ODU) or a fine-grain OTN (fgOTN) connection to provide deterministic transport for AI workloads.

* CATS Traffic Classifier (C-TC): A functional entity responsible for identifying which packets or client signals constitute a traffic flow for a particular service request. It operates in coordination with the Ingress CATS-aware OTN edge node to ensure that such traffic is encapsulated into an OTN connection (e.g., ODUk) and follows the path computed by the C-PS. Refer to <<sec-ctc>> for additional details.

This document makes use of the following additional terms:

* CATS-aware OTN edge node: An OTN node deployed at the network edge that is capable of functioning as a CATS-Forwarder. It operates based on forwarding instructions provided by a CATS Path Selector (C-PS), which might be integrated into or external to the CATS-aware OTN edge node.

  A CATS-aware OTN edge node can function in either an Ingress or Egress capacity. Refer to Section 3.4.6 for further details.

  - Ingress CATS-aware OTN edge node: A functional entity that directs service-specific traffic along a path determined by CATS. In a CATS-aware OTN, an ingress CATS-aware OTN edge node connecting to the client site is responsible for mapping of client signals into ODU/fgOTN containers. It serves as the ingress point.

  - Egress CATS-aware OTN edge node: An entity situated at the termination of a CATS-computed path that interfaces with a service site. In a CATS-aware OTN, a Egress CATS-aware OTN edge node connecting to the Service Contact Instance is responsible for de-mapping signals from ODU/fgOTN containers. It serves as the egress point.

3. CATS Framework and Components

3.1. Assumptions

Under the CATS framework, a specific service can be implemented through single or multiple service instances, which may be deployed across one or several service sites. Each service is uniquely identified by a consistent service identifier (see Section 3.2). Furthermore, CATS operates under the premise that these instances are accessible through one or more service contact instances, without requiring further internal details of the instances themselves.

3.2. CATS Identifiers

The CATS architecture utilizes two functional identifiers as defined in [I-D.ietf-cats-framework]: the CATS Service ID (CS-ID) and the CATS Service Contact Instance ID (CSCI-ID).

This document maintains neutrality regarding the internal structure or semantics of the CSCI-ID. Within the context of CATS-aware OTN, a unicast IP address may serve as a CSCI-ID to uniquely identify the location or access point of a service instance.

3.3. Framework Overview

Figure 1 in [I-D.ietf-cats-framework] provides a high-level conceptual overview of the CATS framework, abstracting the internal functional entities within the network.

[I-D.ietf-cats-framework] further categorizes the architecture into three functional planes: the CATS Management Plane, the CATS Control Plane, and the CATS Data Plane. In the context of this document, the CATS Management Plane handles the configuration and maintenance of CATS-aware OTN edge nodes. The CATS Control Plane manages service scheduling by evaluating both computing and network status. In the context of OTN, this augmented Control Plane determines the establishment and cross-connection of optical paths or connections (e.g., ODUk/fgOTN) across the associated CATS-aware OTN edge nodes, relaying these instructions to the CATS Data Plane for execution.

The CATS Data Plane manages compute-aware optical transport, which involves encapsulating service traffic into optical containers (e.g., ODUk), directing them via designated optical paths toward selected service contact instances, and performing signal cross-connections to maintain deterministic performance throughout the transit.

Depending on the specific implementation and deployment scenario, these planes may comprise various functional components; subsequent sections will provide further details. For instance, the control plane may incorporate elements such as C-PS and C-NMA, while the data plane may include CATS-aware OTN edge nodes, C-TC, and other related entities.

3.4. CATS Functional Components

CATS nodes determine the forwarding path for service requests received from clients by evaluating the operational status and capabilities of both service contact instances and the network. Within a CATS-aware OTN environment, this process incorporates the selection and provisioning of deterministic optical paths. The primary functional entities of the CATS framework and their interworking are illustrated in Figure 2 of [I-D.ietf-cats-framework] where CATS-aware OTN edge nodes access the underlying OTN infrastructure.

3.4.1. Service Sites, Service Instances, and Service Contact Instances

Service sites are described in [I-D.ietf-cats-framework]. They represent physical or logical locations hosting the necessary resources (such as GPU clusters for AI training) to provide a specific service.

A compute service is identified by a CATS Service Identifier (CS-ID).

Figure 2 in [I-D.ietf-cats-framework] illustrates two CATS nodes (which in a CATS-aware OTN are CATS-aware OTN edge node 1 and CATS-aware OTN edge node 3) that facilitate access to these service contact instances. These entities function as Egress CATS-aware OTN edge nodes (see Section 3.4.6) implemented as CATS-aware OTN edge nodes.

Note: "Egress" refers to the direction of service request placement, specifically identifying the exit point of the CATS infrastructure.

3.4.2. CATS Service Metric Agent (C-SMA)

As described in [I-D.ietf-cats-framework], the CATS Service Metric Agent (C-SMA) is a functional entity that collects data regarding service sites and server resources (such as GPU utilization and memory availability), alongside the operational status of various service instances. Depending on the deployment, a C-SMA can be integrated with or positioned near a service contact instance, or hosted by/adjacent to an Egress CATS-aware OTN edge node (see Section 3.4.6). A given deployment may utilize one or multiple C-SMA

instances.

3.4.3. CATS Network Metric Agent (C-NMA)

The CATS Network Metric Agent (C-NMA) is a functional component described in [I-D.ietf-cats-framework]. It is responsible for acquiring information about the underlay network state. Within the context of CATS-aware OTN, the C-NMA retrieves optical-layer performance indicators, including Optical Signal-to-Noise Ratio (OSNR), wavelength or timeslot availability, and deterministic latency derived from physical fiber distance.

The C-NMA is expected to employ established mechanisms (e.g., [RFC7471], [RFC8570], and [RFC8571]) in addition to specialized optical performance monitoring protocols.

3.4.4. CATS Path Selector (C-PS)

The C-PS receives aggregated data from C-SMAs and C-NMAs to determine the optimal Egress CATS-aware OTN edge nodes for routing specific service requests. In the context of CATS-aware OTN, C-PSes focus on the computation and selection of optical paths (such as ODUk or fgOTN connections) to satisfy the rigorous demands of AI workloads.

A C-PS may employ the Path Computation Element Communication Protocol (PCEP) [RFC5440] or PCEP Link-State (PCEP-LS) [I-D.ietf-pce-pcep-ls] with PCEP-LS optical extensions [I-D.lee-pce-pcep-ls-optical] to advertise metrics and synchronize path selection, adhering to the procedures outlined in [RFC9730].

A C-PS can be embedded within CATS-aware OTN edge nodes or implemented as a standalone entity. Typically, a standalone C-PS functions as a part of a centralized controller, such as a Path Computation Element (PCE) [RFC4655] capable of addressing optical constraints..

3.4.5. CATS Traffic Classifier (C-TC)

As described in [I-D.ietf-cats-framework], the CATS Traffic Classifier (C-TC) is a functional entity responsible for mapping incoming client signals or packets to their respective service requests. Within CATS-aware OTN, the C-TC identifies traffic through physical ports, VLAN tags, or specific Service IDs, ensuring these flows are encapsulated into the appropriate optical containers (e.g., ODUk/fgOTN) as directed by the C-PS.

C-TCs are generally situated within CATS-aware OTN edge nodes (acting as Ingress CATS-aware OTN edge nodes).

3.4.6. CATS-Aware OTN Edge Nodes

Ingress CATS-aware OTN edge nodes are tasked with directing service-specific traffic along a path determined by the CATS framework. In the context of this document, these are CATS-aware OTN edge nodes that encapsulate client signals into deterministic optical pipes. Egress CATS-aware OTN edge nodes function as the exit points for service requests by decapsulating the optical containers back into their original client formats.

Within a CATS-aware OTN infrastructure, these CATS-aware OTN edge nodes execute wavelength or timeslot cross-connections at the physical and link layers. This ensures the zero-jitter and high-bandwidth transmission essential for the synchronization of AI clusters.

3.4.7. Underlay Infrastructure

The "underlay infrastructure" depicted in Figure 2 of [I-D.ietf-cats-framework] represents an OTN and optical network (which may include WDM layers) that does not inherently need to be CATS-aware. The CATS-specific paths determined by a C-PS are distributed to the CATS-aware OTN edge nodes, ensuring that the underlying optical nodes (such as P-nodes) remain unaffected by CATS-level steering.

4. CATS-Aware OTN Workflow

The following subsections outline an operational workflow for CATS-aware OTN. To activate CATS within a specific domain, certain provisioning steps are required, as detailed in Section 5.1. Furthermore, Section 5.3 explores various deployment strategies (including distributed, centralized, and hybrid architectures) to suit different operational environments.

4.1. Service Announcement

A service provider assigns a unique identifier, known as a CS-ID, to each service.

Within CATS-aware OTN, the service announcement procedure facilitates the alignment of service demands with deterministic optical resources. The service provider or the controller links the CS-ID with particular ingress CATS-aware OTN edge nodes to ensure that traffic is accurately identified and encapsulated into the relevant optical containers (e.g., ODUk/fgOTN).

4.2. Metrics Distribution

As outlined in Section 3.4, a C-SMA gathers computing capabilities and performance metrics, linking them to the service-specific CS-ID. The C-SMA is responsible for either aggregating these metrics across multiple service contact instances or maintaining individual records for each, or a combination of both approaches.

Given that computing metrics often fluctuate rapidly (as discussed in Section 5.3 of [I-D.ietf-cats-usecases-requirements]), the frequency of their distribution is defined by the specific communication protocol employed. Potential update mechanisms include interval-based, threshold-triggered, or policy-driven updates, as well as the use of normalized metrics to ensure stability.

Furthermore, the C-NMA is responsible for collecting optical network-layer capabilities and metrics. This information may be disseminated using PCEP-LS for optical networks [I-D.lee-pce-pcep-ls-optical], which may require extensions to support additional optical parameters such as link latency, OSNR, and wavelength availability. By distributing these optical metrics to C-PSes, the system enables them to evaluate both service and network conditions to identify the optimal Egress CATS-aware OTN edge node for servicing a request. Consistent with computing metrics, optical network data can be distributed through centralized, distributed, or hybrid frameworks, the specifics of which remain deployment-dependent.

Optical network state may also vary over time. To avoid excessive control plane overhead or flooding, a ttol such as PCEP-LS for optical networks [I-D.lee-pce-pcep-ls-optical] can utilize existing mechanisms to manage state change notifications. Similar to C-SMAs, C-NMAs should be configured with specific triggers or intervals to regulate when updates are reported to the C-PSes.

4.3. Service Access Processing

Based on the service and optical network metrics advertised to the C-PS (for example, via PCEP-LS for optical networks [I-D.lee-pce-pcep-ls-optical], a C-PS identifies the optimal paths to the relevant CATS-aware OTN edge nodes (acting as egress points). The C-PS may be integrated into an Ingress CATS-aware OTN edge node (as illustrated in Figure 3 of [I-D.ietf-cats-framework]) or operate as a logically centralized entity, consistent with the centralized or hybrid models discussed in Section 5.3.

In the scenario depicted in Figure 3 of [I-D.ietf-cats-framework], a client initiates a service request through CATS-aware OTN edge node 1, which serves as the Ingress CATS-aware OTN edge node. Such

service requests may involve high-bandwidth data flows (e.g., RDMA or Ethernet) identified by VLAN tags or specific physical ports that convey the CS-ID and associated parameters.

Upon identifying a matching classification entry via the C-TC, the Ingress CATS-aware OTN edge node maps and encapsulates the incoming signals into an Optical Data Unit (ODUk) or fine-grain OTN (fgOTN) container, as defined in [ITU-T-G-709]. This encapsulated traffic is subsequently steered toward the Egress CATS-aware OTN edge node selected by the C-PS, following the optical path established through PCEP.

Once these optical containers arrive at the Egress CATS-aware OTN edge node, the ODUk/fgOTN overhead is stripped away (via decapsulation/demapping per [ITU-T-G-709]), allowing the original client signals to be delivered to the designated service contact instance.

4.4. Service Contact Instance Affinity

Service contact instance affinity requires that all packets or signals constituting a flow for a given service request are consistently routed to the same service contact instance. Additionally, such traffic should follow a uniform path to prevent packet mis-ordering and avoid the introduction of jitter or unpredictable latency. Within a CATS-aware OTN environment, this path consistency is fundamentally maintained through the use of dedicated optical channels which provide a circuit-switched infrastructure that inherently eliminates reordering and guarantees deterministic performance. Any CATS framework implementation for OTN must ensure that both the service instance selection and the subsequent path steering remain stable for the duration of a flow.

Specifically, the traffic must be directed through the same Egress CATS-aware OTN edge node. Maintaining service affinity is a capability that can be provisioned on the C-PS during service deployment (applying to all associated flows) or assigned dynamically when a new service request is initiated (applying to a specific flow).

It should be noted that the definition of a 'flow' may vary across different services. In a CATS-aware OTN infrastructure, a flow can be identified using physical or link-layer attributes as defined in [ITU-T-G-709], such as a designated port or a specific ODUk/fgOTN timeslot. Therefore, any protocol designed to convey affinity information to the C-TC should provide flexible flow identification mechanisms. More broadly, there must be a way to define and recognize the specific set of signals or packets that require affinity.

Crucially, the criteria for flow identification should remain application-independent to prevent the proliferation of service-specific affinity methods. Nonetheless, affinity parameters (such as identification types, methods, and timeout values) may be configurable on a per-service basis, adhering to the mapping and policy frameworks of [ITU-T-G-709].

This document does not specify the particular mechanisms used to define or enforce service contact instance affinity.

5. Operational Considerations

5.1. Provisioning of CATS Components

The deployment of CATS within an OTN can be achieved through an incremental approach. It is not mandatory for all CATS-aware OTN edge nodes (such as Terminal Muxes) to be upgraded simultaneously. Support for CATS awareness may be restricted to specific CATS-aware OTN edge nodes. For example, CATS capabilities could be prioritized on CATS-aware OTN edge nodes that interconnect AI computing data centers (DCI nodes), while the remaining intermediate nodes maintain transparent transport.

Beyond the CATS steering policies transmitted by a C-PS to an Ingress CATS-aware OTN edge node, several provisioning actions are necessary. These tasks include, but are not limited to:

* Locating Ingress Entities: Supplying C-PS elements with the locators of available Ingress CATS-aware OTN edge nodes (e.g., node identifiers or termination points). These locators may also be dynamically discovered from the network topology via the optical control plane.

* Agent Connectivity: Providing the necessary information to establish communication between C-PS elements, C-NMAs, and C-SMAs.

* Identifier Management: Assigning CS-ID/CSCI-ID identifiers and associating them with particular service contact instances.

* Policy Definition: Configuring C-PS elements with service-specific optimization metrics and policies, emphasizing latency determinism, bandwidth rigidity, and optical-layer availability to meet the requirements of (for example) AI training tasks.

* Traffic Mapping: Configuring the mapping and multiplexing functions of CATS-aware OTN edge nodes, such as allocating AI traffic to designated wavelengths or ODUk/fgOTN timeslots to ensure physical isolation. This also includes credentials for mutual authentication between peer CATS-aware OTN edge nodes.

* Classifier Initialization: Clearing or updating the classification tables within C-TC elements.

* Monitoring and Correlation: Initializing traffic counters and performance monitoring (PM) parameters at CATS-aware OTN edge nodes to facilitate correlation between Ingress and Egress CATS-aware OTN edge nodes. This correlation is essential for identifying performance degradations in the underlying optical transport layer, utilizing the native OAM mechanisms of OTN for end-to-end delay and error-rate monitoring.

Provisioning encompasses both static configuration and dynamic distribution via protocols. These tasks can be implemented through various mechanisms, such as NETCONF [RFC6241], IPFIX [RFC7011], RESTCONF [RFC8040], or YANG-Push [RFC8639]. Detailed discussion of specific CATS extensions for these protocols is beyond the scope of this document.

5.2. Supervision of CATS Components and CATS OAM

Complementary supervision and OAM mechanisms are essential to guide CATS provisioning and evaluate the effectiveness of CATS operations. Key requirements include:

* Capabilities Exposure: Reporting the classification features of C-TC elements (e.g., identifying AI traffic through designated physical ports or VLAN tags for traffic mapping).

* Mapping Capabilities: Identifying the mapping and multiplexing functions supported by CATS-aware OTN edge nodes, adhering to the frameworks established in [ITU-T-G-709].

* State Retrieval: Accessing the active classification and mapping tables from C-TC elements.

* Forwarding Rules: Retrieving current cross-connect and timeslot assignment configurations within CATS-aware OTN edge nodes.

* Policy Auditing: Extracting the active policies currently residing in C-PSes.

* Performance Monitoring: Collecting OTN performance monitoring (PM) data (such as Bit Error Rate (BER), Pre-FEC/Post-FEC status, and wavelength power) from CATS-aware OTN edge nodes to simplify operational correlation between Ingress and Egress CATS-aware OTN edge nodes.

* Hardware-level Verification: Utilizing hardware-based OAM tools (e.g., OTN Overhead and Tandem Connection Monitoring (TCM)) to verify the integrity of various functional entities, including classification, cross-connect, and forwarding behaviors. In contrast to packet-based OAM, OTN OAM leverages dedicated frame overhead, which prevents any impact on service traffic. Refer to Section 5.5.

* Deterministic Measurement: Implementing OAM tools focused on deterministic performance, specifically for high-precision monitoring of latency and jitter.

5.3. Deployment Considerations

This document remains agnostic regarding the specific implementation and deployment of CATS-aware OTN functional entities. In practice, whether a CATS architecture adopts a fully decentralized design or utilizes a combination of centralized (e.g., a centralized C-PS) and distributed components (e.g., C-TCs) is a deployment-specific decision. Within a CATS-aware OTN infrastructure, this typically necessitates coordination between Customer Network Controllers (CNCs) and Physical Network Controllers (PNCs) as outlined in the ACTN framework [RFC8453]. Furthermore, specific use cases [I-D.ietf-cats-usecases-requirements] may influence the chosen deployment strategy.

For instance, in a centralized architecture, a logically centralized path computation entity (such as a PCE or an ACTN MDSC) aggregates both computing metrics from C-SMAs and network performance data. In this scenario, the path computation logic processes service requests to determine the optimal paths to service contact instances. For workloads involving high-bandwidth and long-duration flows (such as AI training), paths and optical channels (e.g., ODUk/fgOTN) may be pre-provisioned to guarantee zero packet loss and immediate service availability. The C-PS then identifies the most suitable path based on current metrics and synchronizes these decisions with the C-TCs.

Depending on the distribution and collection mechanisms for computing metrics, the CATS framework supports three primary deployment models as set out in [I-D.ietf-cats-framework]. In a CATS-aware OTN context, these can be re-stated as follows:

* Distributed model: In this model, the service scheduling function is executed by the CATS-aware OTN edge nodes; consequently, the C-PS is integrated within an Ingress CATS-aware OTN edge node.

* Centralized model: Centralized control entities (e.g., CNC or MDSC) collect all computing metrics and compute forwarding paths for service requests via PCEP and synchronize with the Ingress CATS-aware OTN edge nodes. Here, the C-PS resides within the centralized controller.

* Hybrid model: This model integrates elements of both distributed and centralized architectures.

In the hybrid approach, some computing metrics are shared among network devices while others are gathered by a centralized controller. For example, static optical parameters (such as fiber distance, Shared Risk Link Groups (SRLG), or maximum port capacity) may be distributed among network devices due to their stability. Conversely, highly dynamic information (such as GPU resource utilization, wavelength availability, or optical power fluctuations) is centralized to prevent excessive flooding within the distributed control plane. Service scheduling may be performed by a centralized controller, Ingress CATS-aware OTN edge nodes (co-located with a C-PS), or both, based on local policies. When path computation is distributed, centralized entities must communicate collected path information to the Ingress CATS-aware OTN edge nodes (co-located with a C-PS) to ensure the full metric set is considered for scheduling.

5.4. Implementation Considerations on Using CATS Metrics

[I-D.ietf-cats-framework] observes the scaling concerns when distributing computing-related metrics.

Within CATS-aware OTN infrastructure, normalization of metrics is important for managing heterogeneous hardware accelerators, such as GPUs, NPUs, or FPGAs. These normalized computing scores can then be correlated with OTN-specific network resources (including available ODUk/fgOTN timeslots or bandwidth) to create a composite metric for path selection. For further discussion on metrics and their distribution, refer to [I-D.ietf-cats-metric-definition].

The placement of normalization and aggregation functions depends on the available processing capacity of the CATS components. One strategy is to host these functions away from C-PSes, particularly when C-PSes are integrated into CATS-aware OTN edge nodes. Consequently, these functions may be situated at service contact instances, C-SMAs, or specialized computing gateways interfaced with the OTN ingress.

In scenarios where C-SMAs are co-located with CATS-aware OTN edge nodes that have limited processing power, implementing normalization within the C-SMA may generate excessive overhead and degrade the efficiency of metric distribution (for example via PCEP-LS optical extensions [I-D.lee-pce-pcep-ls-optical]. Therefore, this document recommends performing normalization at the service contact instances. Aggregation functions, however, may reside in either the C-SMA or the service contact instances.

To maintain consistency in CATS path selection, all participating CATS components must utilize identical normalization and aggregation functions. Furthermore, in environments involving multiple vendors or where service contact instances and C-SMAs are provided by different parties, a standardized set of common functions is necessary to ensure fair selection across all instances. To this end, these functions must be standardized, potentially leveraging YANG models compatible with ACTN PNC/CNC architectures. CATS implementations must provide a configuration parameter to manage and activate these specific functions in contexts where multiple versions are supported.

5.5. Verifying Correct Operations

A CATS implementation must maintain logs of error events (such as light-path switching failures, wavelength conflicts, or computing resource downtime) to facilitate enhanced network management and operations. Mechanisms to evaluate reachability and perform CATS path tracing should be provided.

Within a CATS-aware OTN infrastructure, reachability assessment utilizes hardware-level monitoring of end-to-end optical or electrical trails. The operational status of a CATS path can be verified in real-time using ODUk/fgOTN maintenance signals (e.g., Alarm Indication Signal (AIS) or Locked (LCK) signals) as specified in [ITU-T-G-709], thereby removing the requirement for active probe packets.

Additionally, path tracing is supported by the Trail Trace Identifier (TTI) within the OTN frame overhead. This enables the physical verification that traffic is traversing the exact sequence of nodes

as determined by the C-PS. Such verification data should be synchronized with the PNC or CNC to maintain alignment between the control plane steering policies and the actual state of the data plane.

5.6. Impact on Network Operations

The collection and distribution of computing metrics within the CATS framework necessitate a management function to coordinate interactions between network and computing resources. This role can be fulfilled by an orchestrator, such as a Customer Network Controller (CNC) or a Multi-Domain Service Coordinator (MDSC) within the ACTN framework [RFC8453], which interfaces with both the C-SMA and C-NMA. Utilizing existing optical control hierarchies in this manner minimizes the requirement for entirely new functional entities.

While introducing this coordination function may increase network management complexity (particularly if it is exclusively dedicated to CATS) this is balanced by the superior determinism offered by the OTN layer for workloads. In contrast to connectionless IP networks, CATS-aware OTN is connection-oriented. Once computing-aware paths are provisioned (for example, through PCEP-LS mechanisms for optical networks [I-D.lee.pce-pcep-ls-optical]) operational efforts transition from addressing routing oscillations to maintaining stable, high-bandwidth "hard-isolations." This approach greatly streamlines the supervision of long-duration traffic flows, such as for AI training traffic.

Additionally, the CNC can act as a Northbound interface for external computing platforms, such as AI job schedulers, to enable coordinated resource allocation. This allows the CATS-aware OTN infrastructure to adapt reliably to the specific latency and topological demands of, for example, distributed AI clusters.

6. IANA Considerations

This document does not make any requests of IANA.

7. Security Considerations

Computing resource information is highly dynamic, fluctuating rapidly as service instances are initialized or terminated. If this information is disseminated via a distribution protocol (such as PCEP-LS for optical networks [I-D.lee.pce-pcep-ls-optical], an excessive volume of updates can undermine network stability. An attacker might exploit this vulnerability by rapidly creating and deleting service instances to trigger instability. Consequently,

CATS solutions must implement safeguards against such behavior, including aggregation techniques, dampening mechanisms, and threshold-triggered updates. Within CATS-aware OTN, where path setup is resource-intensive, the architecture should incorporate a "computing fluctuation window." This ensures that optical layer reconfigurations are only initiated by significant or sustained shifts in compute metrics.

The data distributed by C-SMAs and C-NMAs is often sensitive, as it may reveal network intelligence, the precise topology of GPU clusters, and the specific locations of compute resources within service sites. Attackers could leverage this data to pinpoint vulnerabilities in a provider’s infrastructure. Furthermore, unauthorized modification of this information could disrupt service delivery or redirect traffic to malicious service instances. CATS-aware OTN provides a distinct security advantage by supporting Layer 1 physical layer encryption (e.g., OTN-SEC). This provides high-throughput security for data flows without the header overhead or latency increases typical of higher-layer encryption like IPsec, which is vital for the performance of latency-sensitive AI training.

CATS implementations must provide robust authentication and integrity protection between C-SMAs/C-NMAs and C-PSes, as well as between C-PSes and Ingress CATS-aware OTN edge nodes. In an ACTN-based environment, stringent mutual authentication is required between the PNC/CNC and the CATS-aware OTN edge nodes to prevent unauthorized changes to optical cross-connects or timeslot allocations. Additionally, C-SMAs must have mechanisms to authenticate the services for which they provide data to the C-PS selection logic.

This document is restricted to a single service provider scenario. The centralized architecture of the OTN control plane within a single domain facilitates a closed management loop, effectively minimizing the external attack surface. Therefore, security issues specific to multi-provider deployments are considered out of scope.

8. Privacy Considerations

CATS solutions are required to prevent on-path nodes within the underlay infrastructure from performing client fingerprinting or tracking (e.g., identifying which client is accessing a particular service). Generally, the CATS framework must ensure that personal data is not disclosed to external parties, exceeding the information already present in the original packets transmitted by the client.

Within a CATS-aware OTN infrastructure, privacy is naturally bolstered by the use of Layer 1 rigid "Hard-isolations." Because intermediate elements (such as Optical Amplifiers) function at the

physical layer, they are unable to inspect the payload of the encapsulated traffic. This transparency at the physical layer ensures that on-path nodes cannot perform traffic analysis or track application behavior through packet header inspection.

In certain scenarios, a CATS solution might require knowledge of specific applications, clients, or user identities. Such sensitive data must be protected via encryption. To mitigate the risk of information leakage among CATS components, path information computed by the C-PS and specific mapping instructions (such as ODUk/fgOTN timeslot assignments) should be encrypted during distribution. For instance, communication between the PNC/CNC and CATS-aware OTN edge nodes should be protected using secure southbound protocols like NETCONF over TLS or RESTCONF. The choice of encryption-whether at the network, transport, or application layer-is implementation-dependent and remains outside the scope of this document.

This document is restricted to a single service provider environment. Consequently, privacy issues related to multi-provider deployments are not addressed here.

For further details on privacy, refer to [RFC6462] and [RFC6973].

9. Acknowledgements

The authors wish to acknowledge Adrian Farrel for helpful discussions.

10. References

10.1. Normative References

[I-D.ietf-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-ietf-cats-framework-20, 26 February 2026, <https://datatracker.ietf.org/doc/html/draft-ietf-cats-framework-20>.

[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-14, 2 February 2026, <https://datatracker.ietf.org/doc/html/draft-ietf-cats-usecases-requirements-14>.

[ITU-T-G-709]
ITU-T, "G.709 - Interfaces for the optical transport network", June 2020, <https://www.itu.int/rec/T-REC-G.709/>.

[RFC6462] Cooper, A., "Report from the Internet Privacy Workshop", RFC 6462, DOI 10.17487/RFC6462, January 2012, <https://www.rfc-editor.org/rfc/rfc6462>.

[RFC6973] Cooper, A., Tschofenig, H., Aboba, B., Peterson, J., Morris, J., Hansen, M., and R. Smith, "Privacy Considerations for Internet Protocols", RFC 6973, DOI 10.17487/RFC6973, July 2013, <https://www.rfc-editor.org/rfc/rfc6973>.

[RFC8453] Ceccarelli, D., Ed. and Y. Lee, Ed., "Framework for Abstraction and Control of TE Networks (ACTN)", RFC 8453, DOI 10.17487/RFC8453, August 2018, <https://www.rfc-editor.org/rfc/rfc8453>.

10.2. Informative References

[I-D.ietf-cats-metric-definition]
Yao, K., Li, C., Contreras, L. M., Ros-Giralt, J., and G. Zeng, "CATS Metrics Definition", Work in Progress, Internet-Draft, draft-ietf-cats-metric-definition-05, 2 February 2026, <https://datatracker.ietf.org/doc/html/draft-ietf-cats-metric-definition-05>.

[I-D.ietf-pce-pcep-ls]
Dhody, D., Peng, S., Lee, Y., Ceccarelli, D., Wang, A., and G. S. Mishra, "PCEP extensions for Distribution of Link-State and TE Information", Work in Progress, Internet-Draft, draft-ietf-pce-pcep-ls-04, 14 October 2025, <https://datatracker.ietf.org/doc/html/draft-ietf-pce-pcep-ls-04>.

[I-D.lee-pce-pcep-ls-optical]
Zhao, Y., Ceccarelli, D., LiXiao, Yoon, B. Y., and A. Farrel, "PCEP Extensions for Distribution of Link-State and TE Information for Optical Networks", Work in Progress, Internet-Draft, draft-lee-pce-pcep-ls-optical-17, 7 February 2026, <https://datatracker.ietf.org/doc/html/draft-lee-pce-pcep-ls-optical-17>.

[I-D.lee.pce-pcep-ls-optical]
"*** BROKEN REFERENCE ***".

[RFC4655] Farrel, A., Vasseur, J.-P., and J. Ash, "A Path Computation Element (PCE)-Based Architecture", RFC 4655, DOI 10.17487/RFC4655, August 2006, <https://www.rfc-editor.org/rfc/rfc4655>.

[RFC5440] Vasseur, JP., Ed. and JL. Le Roux, Ed., "Path Computation Element (PCE) Communication Protocol (PCEP)", RFC 5440, DOI 10.17487/RFC5440, March 2009, <https://www.rfc-editor.org/rfc/rfc5440>.

[RFC6241] Enns, R., Ed., Bjorklund, M., Ed., Schoenwaelder, J., Ed., and A. Bierman, Ed., "Network Configuration Protocol (NETCONF)", RFC 6241, DOI 10.17487/RFC6241, June 2011, <https://www.rfc-editor.org/rfc/rfc6241>.

[RFC7011] Claise, B., Ed., Trammell, B., Ed., and P. Aitken, "Specification of the IP Flow Information Export (IPFIX) Protocol for the Exchange of Flow Information", STD 77, RFC 7011, DOI 10.17487/RFC7011, September 2013, <https://www.rfc-editor.org/rfc/rfc7011>.

[RFC7471] Giacalone, S., Ward, D., Drake, J., Atlas, A., and S. Previdi, "OSPF Traffic Engineering (TE) Metric Extensions", RFC 7471, DOI 10.17487/RFC7471, March 2015, <https://www.rfc-editor.org/rfc/rfc7471>.

[RFC8040] Bierman, A., Bjorklund, M., and K. Watsen, "RESTCONF Protocol", RFC 8040, DOI 10.17487/RFC8040, January 2017, <https://www.rfc-editor.org/rfc/rfc8040>.

[RFC8570] Ginsberg, L., Ed., Previdi, S., Ed., Giacalone, S., Ward, D., Drake, J., and Q. Wu, "IS-IS Traffic Engineering (TE) Metric Extensions", RFC 8570, DOI 10.17487/RFC8570, March 2019, <https://www.rfc-editor.org/rfc/rfc8570>.

[RFC8571] Ginsberg, L., Ed., Previdi, S., Wu, Q., Tantsura, J., and C. Filsfils, "BGP - Link State (BGP-LS) Advertisement of IGP Traffic Engineering Performance Metric Extensions", RFC 8571, DOI 10.17487/RFC8571, March 2019, <https://www.rfc-editor.org/rfc/rfc8571>.

[RFC8639] Voit, E., Clemm, A., Gonzalez Prieto, A., Nilsen-Nygaard, E., and A. Tripathy, "Subscription to YANG Notifications", RFC 8639, DOI 10.17487/RFC8639, September 2019, <https://www.rfc-editor.org/rfc/rfc8639>.

[RFC9730] Zheng, H., Lin, Y., Zhao, Y., Xu, Y., and D. Beller, "Interworking of GMPLS Control and Centralized Controller Systems", RFC 9730, DOI 10.17487/RFC9730, March 2025, <https://www.rfc-editor.org/rfc/rfc9730>.

Contributors

Minxue Wang
China Mobile
Email: wangminxue@chinamobile.com

Authors' Addresses

Yang Zhao
China Mobile
China
Email: zhaoyangyjy@chinamobile.com

LiuYan Han
China Mobile
China
Email: hanliuyan@chinamobile.com

Xiao Li
Huawei
China
Email: lixiao33@huawei.com

Haomian Zheng
Huawei
China
Email: zhenghaomian@huawei.com

Daniel King
Old Dog Consulting
United Kingdom
Email: daniel@olddog.co.uk