CATS
Internet-Draft
Intended status: Standards Track
Expires: 29 August 2025

H. Fu
D. Huang
L. Ma
W. Duan
ZTE Corporation

25 February 2025

An SR-TE based Solution For Computing-Aware Traffic Steering
draft-fu-cats-sr-te-based-solution-03

Abstract

Computing-aware traffic steering (CATS) is a traffic engineering approach [I-D.ietf-teas-rfc3272bis] that takes into account the dynamic nature of computing resources and network state to optimize service-specific traffic forwarding towards a given service instance. Various relevant metrics may be used to enforce such computing-aware traffic steering policies.It is critical to meet different types of computing-aware traffic steering requirements without disrupting the existing network architecture. In this context, this document proposes a computing-aware traffic steering solution based on the SR-TE infrastructure of the current traffic engineering technology to reduce device resource consumption and investment and meet the requirements for computing-aware traffic steering of network devices.

Table of Contents

1. Introduction 2
2. Requirements Language 3
3. Terminology 3
4. Requirements and Motivation 4
5. Background and general scenario 5
6. Service Flow 5
6.1. Service Overview 7
6.2. Work Flow Overview 7
7. Control Plane 7
7.1. Considerations 8
7.2. EGW Processing 8
7.3. IGW Processing 10
7.4. Control Plane WorkFlow 11
8. Data Plane 13
9. Security Considerations 14
10. Acknowledgements 14
11. IANA Considerations 14
12. References 14
12.1. Normative References 14
12.2. Informative References 15
Authors' Addresses 15

1. Introduction

Edge computing provides better response time and transmission rate than cloud computing by proximity to the end users. Considering computing resource capacity and locations, peak hours, and economic factors, traffic steering to the nearest node may not meet service requirements. To meet the requirements of users, service providers deploy the same type of service instances at multiple edge sites. This brings about the key problem of steering the service traffic to the most suitable computing instances to meet the (service-specific) requirements of users. When different types of computing services are accessed, the requirement types for CATS are usually different. In general, there are the following three types: 1) Experience, namely, the SLA indicators related to service access QoS are met. 2) Cost, namely, the optimal cost/energy consumption for service access

resources. 3) Resource , namely, the balance of computing resources.

For experience service access, the end-to-end delay is a key factor that affects user experience. This delay includes two parts: Network and computing processing. The CATS would not be able to select a best service instance with regard to only the compute or network factor. As described in [I-D.yao-cats-ps-usecases], multiple edge sites need to be interconnected and coordinated at the network layer to meet service requirements and ensure better user experience.Based on this, the two-level service routing mechanism is employed to reduce the processing load on the control plane and forwarding plane, and a virtual node and a link (including a computing resource status) are created based on a service identifier and a corresponding service instance. The computing and network integration decision-making could thus be reduced to a conventional network-level traffic engineering problem. so as to implement a traffic steering solution of an egress serving gateway for level-1 routing, and a level-2 routing service instance, thereby reducing system complexity and meeting different requirements for traffic steering. For a requirement of a cost or resource type service, a computing resource status is converted into a network factor. Even if the CATS preferentially selects a computing resource, this solution is still applicable by increasing a weight of the factor.

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

* CATS: Computing-Aware Traffic Steering.
* SID: Service Identifier.
* IGW: Ingress GateWay.
* EGW: Egress GateWay.
* TEDB: Traffic Engineering DataBase.
* CADB: Computing-Aware DataBase.
* SRT: Service Routing Table.

* SFT: Service Forwarding Table.

4. Requirements and Motivation

As described in [I-D.yao-cats-ps-usecases], multiple edge sites need to be interconnected and coordinated at the network layer to meet service requirements and ensure better user experience.

Considering the actual deployment and network resource capabilities of edge computing in MANs, we believe that the CATS framework should consider the following requirements and motivations:

1)To meet the requirements of three types of CATS, two problems must be solved at the same time: 1) IGW selects a specific service instance during the user service access process; 2)IGW or the network controller orchestrates the network paths that meet the quality requirements for the selected service instance.To solve any problem above, the quality of computing resources and network quality must be jointly evaluated at the same time. For example, the budgets for server (computing) delay and network delay are almost the same. It makes sense to consider the two types of delay . If the computing domain metric can be converted into the existing network domain metric in a unified manner, the technical solution will be greatly simplified by using the existing traffic engineering technology.

REQ 1 It’ s recommended the computing status be converted or mapped into the metric aligning with the existing network metric schemes.

2)Considering the service and resource planning of the existing network, the edge compute nodes need to be deployed in VPN during the notification of computing status. As a result, service-layer routes and Transport-layer path decisions are interdependent. This undoubtedly increases the coupling between the two layers. The transmission services provided by the network are divided into two layers: Service and Transport. Services: services include L2VPN, L3VPN, and VXLANs, which usually use the OVERLAY technology. Transport: Uses Underlay technologies such as IPv6, MPLS to control paths by using traffic engineering technologies. . Therefore, the cats framework should consider the design of an independent service routing layer, abstraction of computing resources and status, and joint TE decision-making involving the public network.

REQ 2 An independent computing service based routing layer should be empolyed by CATS over the underlying public network to enable joint traffic steering of computing and networking.

3)To meet the requirements of CATS, the network needs to be aware of the status changes of computing resources (granularity: minute-level). The status information of a large number of computing instances will bring significant pressure to the control plane and data plane of the network. The CATS framework should consider reducing the pressure on the control plane and data plane, and use two-level routing or even direct egress gateways to preferentially select or aggregate service instances, thereby reducing the scale of computing capability information expansion.

5. Background and general scenario

The edge computing service is being expanded from a single edge site to a networked network and coordinates with multiple edge sites to solve problems such as low costs, service experience, and resource utilization. CATS enables large-scale edge interconnection collaboration, providing optimal service access and load balancing to adapt to service dynamics. The computing capability and network conditions based on the real processing delay could dynamically switch the service requests to proper service nodes, thus improving resource utilization and user experience.

6. Service Flow

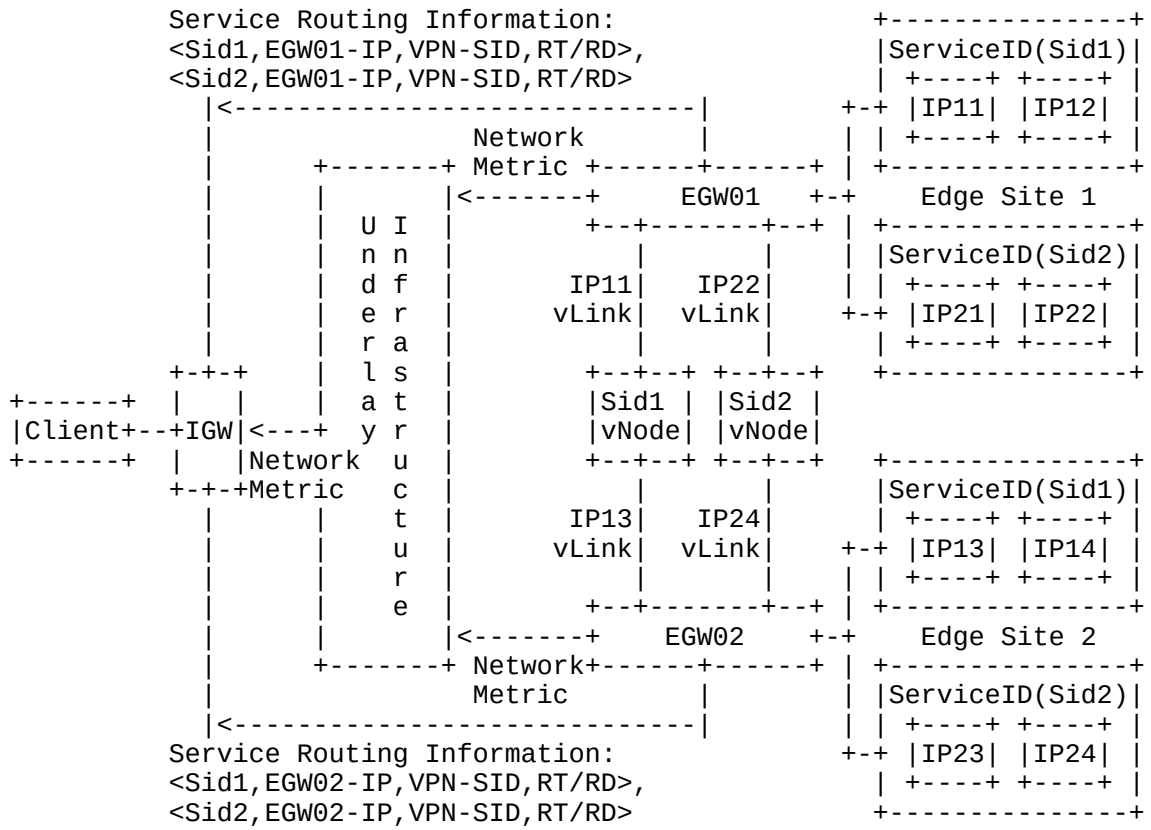

Figure 1: Overall Architecture

Figure 1 indicates the network topology and technical architecture of CATS in terms of service flow. The IGW/EGW node is a functional entity that provides the switching capability in the CATS network, and is interconnected by the transport network (Underlay Infrastructure). The EGW is connected to multiple computing resources and being aware of the status information of the computing resources. The EGW provides the CATS service for customers (the EGW can act as an IGW at the same time). Edge sites often refer to managed edge computing. IGW/EGW node functions are usually provided by physical devices, Such as routers in the access network or MAN.

The "underlay infrastructure" in Figure 1 indicates an IP/MPLS network that is not necessarily CATS-aware. The CATS paths that are computed will be distributed among the overlay IGW/EGW, and will not affect the underlay nodes.

6.1. Service Overview

CATS uses Service Identifier (SID) to represent specific service provided by service nodes on multiple edge sites. The client device always uses SID to initiate service access. The source or destination IP or IP extension header options can be used to carry SID. A CATS request for a single SID could be referred to by different edge locations and compute instances. The client device does not know in advance which edge site to satisfy the request. This service process is a late binding model that selects the appropriate edge site (i.e. EGW egress) and the corresponding service instance and provides the network connectivity channel. As shown in Figure 1, EGW01 is connected to two types of services: Sid1 and Sid2. Computing nodes that provide a Sid1 service include IP11, IP12, or more, and nodes that provide a Sid2 service include IP21, IP22, or more. Details are not described again in EGW02.

6.2. Work Flow Overview

The following is a brief description of the CATS system traffic steering workflow:

(1) The client initiates a computing service request. The packet carries SID in multiple carrying modes . No matter which SID carrying mode is used, the goal is to make the request packet reachable and the IGW perceives the SID.

(2) After receiving the request packet from the client, the IGW identifies the corresponding SID, selects the corresponding EGW, and delivers the specified network path to meet the network quality requirements for service access.

(3) The EGW receives the service request forwarded by the IGW, identifies the corresponding SID, selects a proper service instance, modifies the destination address of the packet to the service instance, lookups the VPN FIB, and forwards the packet to the service instance to implement the service connection.

(4) The service instance responds with a packet. On the EGW, the source IP of the packet is changed to the destination IP corresponding to the service request type. The subsequent procedure is a normal network service procedure.

7. Control Plane

7.1. Considerations

To achieve the goal of computing-aware traffic steering, the general design idea of the control plane of network devices is to enable network link attribute flooding and IGP/BGP extension to implement computing-aware and advertise to upstream nodes to form the traffic engineering database (TEDB) and computing-aware database (CADB). The CADB and TEDB need to be associated across layers. Joint computation (centralized or distributed) is performed in accordance with the service access SLA to obtain the required service instances and network paths.

This bring two issues:

(1) the computing speed requirements are different, both centralized and distributed systems need to be supported. Therefore, a set of SDN architecture similar to the PCEP-based solution would have to be involved repeatedly.

(2) The dimensions of the computing domain parameters (health score, average processing delay, economic cost, and resource occupation) and the networking domain parameters (bandwidth, delay, jitter, and packet loss) would be hard to be unified. The computation consumption time increases with the increase of constraint conditions, and CPU resources consumed by a large number cannot be deployed on a large scale.

In addition, computing instances that provide the same service type can be flexibly deployed to the same EGW and/or different IGWs. If the status of an EGW computing resource is continuously updated to the upstream IGWs, the update of mass computing status information would overwhelm the control plane of network devices and even cause system breakdown.

7.2. EGW Processing

			Computing Metric			
VRF-ID	Service Identifier	Service Instance	Processing delay	Processing bandwidth capability	Bandwidth occupancy	Processing costs
100	Sid1	IP11	* 1ms	10G	9G	100
100	Sid1	IP12	2ms	10G	5G	200
100	Sid2	IP21	10ms	40G	8G	30
100	Sid2	IP22	20ms	40G	* 5G	30

Figure 2: The intention of the Local Service Routing Table

The EGW perceives the status of the computing instance from the Edge Manager, the corresponding status includes four attributes (we call computing metric):

(1)Processing delay: the time when a service instance processes a single service.

(2)Processing bandwidth: Physical bandwidth capability of the service instance or network port bandwidth for computing resources.

(3)Occupied bandwidth: The service instance occupies the processing bandwidth or the bandwidth of the computing resource network interface.

(4)Processing cost: Cost of service instance resources. In most cases, physical costs are related to energy consumption.

For details, see Figure 2. EGW maintains the corresponding service instance entry in the SRT in accordance with the VPNs deployed on the computing resources,VRF-ID, and SID, and the latency, bandwidth, and cost elements of the service instance VRF-ID and computing resources. The EGW performs local processing in accordance with the SLA corresponding to SID (if the service SLA focus on latency, the EGW preferably selects the local service instance in accordance with the latency of the instance), generates the local SRTs.

When a preferred service instance exists in a specific SID in the local SRT, the EGW advertises a VRF route update message to the IGW. Once the preferred service instances becomes zero due to resource deterioration, the EGW advertises a VRF route revocation message to the IGW. The bearer protocol is implemented through the MP-BGP protocol suite. The carried elements include the message type, SID, EGW-IP, VPN-SID, and RT/RD. The EGW advertises a service route to the IGW instead of the specific service instance information. In this way, a service routing layer independent of VPN IP routes is formed, reducing the pressure on the control plane.

The EGW installs, in the IGP, a virtual node and a virtual link that are corresponding to SID based on an entry that is preferred by each SID and that is based on a local SRT. The virtual node is connected to the EGW by using a virtual link (refer to Figure 1). A delay, bandwidth, and a COST that are of a preferred service instance are used as link attributes of the virtual link, and flood and spread network metric values are performed in an IGP area, which greatly reduces a scale of spreading control-plane information.

7.3. IGW Processing

```
+-----------------------------------+
|TE policy py-Sid1-EGW01            |
|  Color color1 end-point EGW01-IP  |
|  SGLIST:{P1-SID,..., EGW01-SID}   |
|                                   |
|TE policy py-Sid2-EGW01            |
|  Color color2 end-point EGW01     |
|  SGLIST:{P1-SID,..., EGW01-SID}   |
+-----------------------------------+
+------+----------+--------+-------+--------------+
|VRF-ID|Service   | EGW IP | COLOR |    VPN-SID   |
|      |Identifier|        |       |              |
+------+----------+--------+-------+--------------+
|      |          |EGW01-IP|color 1|vidx-EGW01-SID|
|100   |Sid1      +--------+-------+--------------+
|      |          |EGW02-IP|color 1|vidx-EGW02-SID|
+------+----------+--------+-------+--------------+
|      |          |EGW01-IP|color 2|vidx-EGW01-SID|
|100   |Sid2      +--------+-------+--------------+
|      |          |EGW02-IP|color 2|vidx-EGW02-SID|
+------+----------+--------+-------+--------------+
```

Figure 3: TE Calculation Result and Global Service Routing Table

As shown in Figure 3, traffic steering from accessing the computing service SID on the IGW to preferred node is converted into a conventional network traffic engineering process: That is, path computation is performed between virtual nodes corresponding to SID connected to the IGW and the EGW according to an SR-POLICY-1 constraint corresponding to the service SID, and a corresponding SR-POLICY-1 path (color, endpoint: SID) is generated, where a penultimate SEGMENT ID (NODE) in the segment list indicates an EGW preferred for service access in a current condition, Convert SR-POLICY-1 into the required SR-POLICY-2 path (color, endpoint: EGW-IP).

After receiving the routing information advertised by each EGW, the IGW generates global SRTs to multiple VPNs, that is, the VRF-ID and SID are used as the KEY, and different EGW-IP are used as multiple next hops. The SR-POLICY-2 is matched based on each COLOR and EGW-IP in SRTs to obtain the preferred global SRTs and generate the global SFT, which is delivered to the forwarding plane for traffic steering requests.

7.4. Control Plane WorkFlow

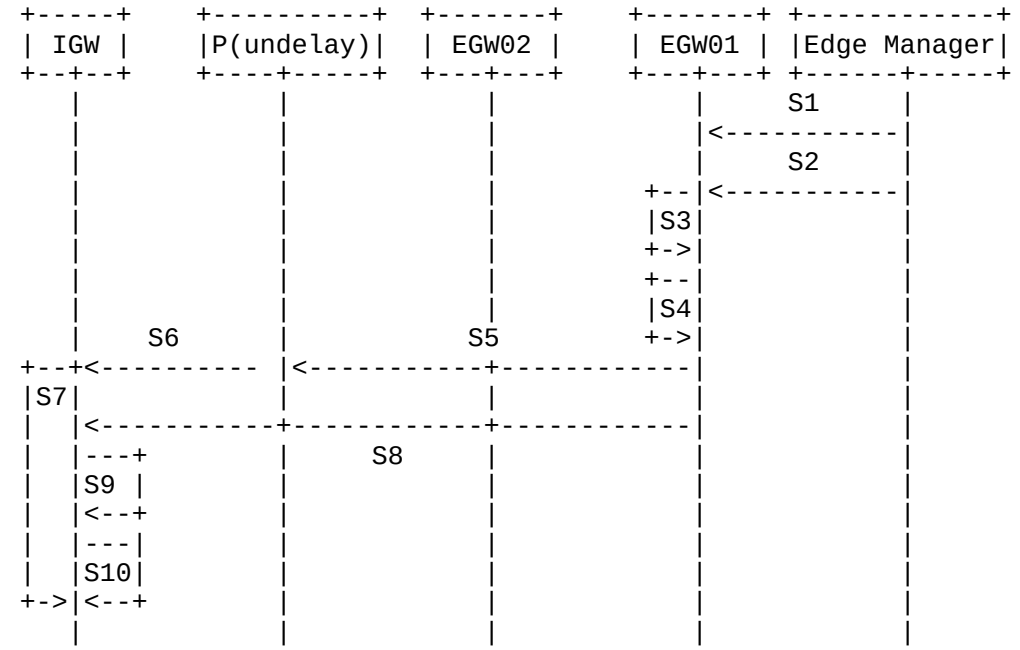

Figure 4: Control Plane Workflow

Figure 4 shows the complete control plane procedure.The related steps are described as follows:

S1: Edge Manager sends a registration/update/deregistration message to the EGW01, including SID and the list of the corresponding instance IP,such as [Sid1, IP11, IP12], [Sid2, IP21, IP22].

S2: Edge Manager periodically sends computing resource status information to the EGW01, including SID, the corresponding instance and computing METRIC information, such as [Sid1, IP11 METRIC, IP12 METRIC], and [Sid2, IP21 METRIC, and IP22 METRIC].

S3: EGW01 generates a local SRT in accordance with the obtained computing resource status and the deployed VPNs. The entries include [VRF-ID, Sid1, IP11, METRIC], [VRF-ID, Sid2, IP21, METRIC>.

S4: The EGW01 preferentially generates the local SRT in accordance with the SLA of SID. Preferred entries generate virtual nodes and links,such as [vNode Sid1, vLink Sid1], and [vNode Sid2, and vLink Sid2].

S5-S6: Flood the information of virtual nodes and links between EGW and P node, and between P node and IGW..

S7: SR-TE Path Calculation Between the IGW and the virtual node vNode-sid1/2 in accordance with the SLAs corresponding to SID.

S8: EGW01 advertises VPNs, such as [Sid1, EGW01-IP, vidx-EGW01-SID, RT/RD], and [Sid2, EGW01-IP, vidx-EGW01-SID, RT/RD].

S9: IGW receives the service route advertised by EGW01/02, and generates the global SRT entries with multiple egress next hops, such as {VRF-ID, Sid1, [EGW01-IP, vidx-EGW01-SID], [EGW02-IP, vidx-EGW02-SID]}.

S10: Combined with S7 and S9 contents, Selects the specified EGW next hop based on the SR-POLICY and Global SRT.

Because the service and computing instance status have been converted into network virtual nodes and links, although the distributed head node computing is used as an example here, it is still applicable to the centralized PCE computing architecture.

This solution unifies the traffic to the end-to-end access delay, cost, bandwidth, jitter, and packet loss in accordance with the SLA target. Based on different objectives: 1) Experience, the system focuses on delay, jitter, and packet loss; 2) Costs: Pay attention to costs and energy consumption, that is, costs; 3) Resource: Check the resource usage/status. If the remaining cloud resources are converted into available bandwidth, check the available bandwidth. In actual service deployment, one of the five measurement indicators is selected as the preferred indicator in accordance with different objectives, and other indicators are selected as constraint conditions.

8. Data Plane

CATS traffic steering belongs to the late binding model, and the forwarding plane has the ability to assign user flows to the "best" service instance and network path. When new traffic arrives, the IGWs select the most appropriate EGW egress in accordance with the network status and computing resources, and ensure flow affinity (the data packets of the same flow are sent to the same service instance).

To be added.

9. Security Considerations

(1) There are many computing instances and the resource information changes rapidly with time, Information is carried in routing protocols, and network changes may occur frequently. Section 5.2 provides a solution to avoid sending too many updates.

(2) As the two-level Service routing model is used, the EGW does not need to advertise the details of service instances or aggregate routes to IGW. Client can only access service instances by carrying SID. In the future, the authorization management of SID will be added, greatly improving system access security.

10. Acknowledgements

To be added upon contributions, comments and suggestions.

11. IANA Considerations

There are no IANA considerations in this document.

12. References

12.1. Normative References

[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/info/rfc2119>.

[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/info/rfc5440>.

[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/info/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/info/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/info/rfc8754>.

12.2. Informative References

[I-D.huang-service-aware-network-framework]
Huang, D., Tan, B., and D. Yang, "Service Aware Network Framework", Work in Progress, Internet-Draft, draft-huang-service-aware-network-framework-01, 22 November 2022, <https://datatracker.ietf.org/doc/html/draft-huang-service-aware-network-framework-01>.

[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>.

[I-D.li-dyncast-architecture]
Li, Y., Iannone, L., Trossen, D., Liu, P., and C. Li, "Dynamic-Anycast Architecture", Work in Progress, Internet-Draft, draft-li-dyncast-architecture-08, 16 January 2023, <https://datatracker.ietf.org/doc/html/draft-li-dyncast-architecture-08>.

[I-D.yao-cats-ps-usecases]
Yao, K., Trossen, D., Boucadair, M., Contreras, L. M., Shi, H., Li, Y., and S. Zhang, "Computing-Aware Traffic Steering (CATS) Problem Statement, Use Cases, and Requirements", Work in Progress, Internet-Draft, draft-yao-cats-ps-usecases-03, 30 June 2023, <https://datatracker.ietf.org/doc/html/draft-yao-cats-ps-usecases-03>.

Authors' Addresses

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

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

Liwei Ma
ZTE Corporation
Email: ma.liwei1@zte.com.cn

Wei Duan
ZTE Corporation
Email: duan.wei1@zte.com.cn