

ALTO WG
Internet-Draft
Intended status: Informational
Expires: April 28, 2022

LM. Contreras
Telefonica
October 25, 2021

Considering ALTO as IETF Network Exposure Function
draft-contreras-alto-ietf-nef-00

Abstract

This document proposes ALTO as the means for exposure of underlay network capabilities for multiple overlays on top of the network.

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 April 28, 2022.

Copyright Notice

Copyright (c) 2021 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 Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction	2
2. Exposing network capabilities for enhancing service delivery	2
3. ALTO versus network controller conceptualization	3
4. Modes of usage	4
4.1. Existing use cases	4
4.1.1. Network topology and reachability	4
4.1.2. Network performance metrics per path	4
4.1.3. Segmented paths and associated characteristics . . .	5
4.1.4. In-time view of dynamic IP addressing allocation . .	5
4.2. Prospective use cases	5
4.2.1. Determination of optimal compute facility taking into	
account network information	5
4.2.2. Visibility of underlying network information in	
overlay networks	6
4.2.2.1. Cellular case	6
4.2.2.2. Media distribution case	6
5. ALTO as IETF Network Exposure Function	6
6. TODO for next versions of this document	8
7. Security Considerations	8
8. IANA Considerations	8
9. References	8
Acknowledgments	10
Author's Address	10

1. Introduction

Networks are turning on consumable objects by external applications and services. In order to facilitate that, it is necessary to expose the capabilities offered by the networks in such a way that the applications and services can produce informed decisions that assist in the improvement of the service delivery.

Thus it is convenient to define mechanisms for capabilities exposure that could provide required information for IETF networks. ALTO [RFC7285] can play such a role. This memo describes existing and foreseen capabilities that can be exposed by leveraging on ALTO.

2. Exposing network capabilities for enhancing service delivery

More and more, services and applications rely on information retrieved from the network in order to make decisions positively affecting the service delivery, by adapting the applications to the reality observed through the retrieved information. This information is typically offered by specific components in the network with the mission of aggregating, processing and securely exposing such information.

Several initiatives are being developed in order to facilitate such exposure of capabilities and information at different network levels. For example, 3GPP defines the Network Exposure function (NEF) [TS29.522] as a secure, scalable and simplified exposing tool for capabilities (as well as events) supported by the 5G Core (5GC) network. Main capabilities of NEF are the following:

- o Securely expose 3GPP Network Functions (NFs) capabilities to Application Functions (AF).
- o Secure provision of information to 5GC, including authentication and authorization to AF.
- o Rate limit AF access to 5GC functions and information, including charging functions.
- o Translation of internal-external information, e.g. identities.

This is done though a number of APIs defined in [TS29.522]. A specific NEF instance may support only a subset of the APIs specified for capability exposure.

Further examples are present in other network concerns. Thus, in ETSI Multi-Access Edge Computing (MEC) group a number of APIs allow the retrieval of specific network information at the edge (e.g., location API [MEC-013]), or the O-RAN Alliance which is working on exposing information to applications running on top of the non-real time Radio Information Controller (RIC) [O-RAN].

The purpose of this document is to consider ALTO as the means for exposure of underlay network capabilities to multiple overlays on top of the network. In other words, serve as "ground truth" from the network provider perspective to the applications consuming network capabilities in the scope of IETF.

3. ALTO versus network controller conceptualization

A relevant question that could arise is about the difference on purpose between ALTO and a network controller in the network.

Primarily, the final purpose of these components is quite different. In this respect, a network controller (i.e., SDN controller [RFC7149]) can be seen as the element devoted to orchestration, control and management of the network assets, that is, the component in charge of administering network objects. Typically, a network controller leverages on another IETF functional component used for network control, such as the Path Computation Element (PCE) [RFC4655], which is used to compute paths for forwarding purposes

based on network constraints. In contrast to these two elements, ALTO acts as a "one-stop-shop" for retrieving (and correlating) network related information, potentially leveraging on the capabilities of the other two (i.e., SDN controller and/or PCE).

Moreover, ALTO has been included as part of some architectural frameworks, such as ABNO [RFC7491], with the mission of allowing joint network and application-layer optimization precisely by exposing to applications an abstract network topology containing only information relevant to such application. In this manner the application can take an informed decision and request specific control actions in the network.

4. Modes of usage

This section presents different modes of usage of ALTO network exposure capabilities to improve network operations. Some of these usages can be implemented nowadays based on existing specifications, while a set of other use cases is considered as prospective since more specification work is yet needed in IETF.

4.1. Existing use cases

This subsection presents a number of use cases already described that can leverage on ALTO as IETF Network Exposure Function.

4.1.1. Network topology and reachability

The basic ALTO capabilities [RFC7285] provide network maps associated with costs in a manner that for any pair of source and destination can be retrieved information about topology and reachability. This can be considered as the fundamental or baseline information on top of which the other modes of usage are built on.

4.1.2. Network performance metrics per path

Extensions defined in [I-D.ietf-alto-performance-metrics] permit the reporting of standard-based performance metrics associated to the paths generated in the network map. With that view, applications consuming ALTO (i.e., ALTO clients) can determine the performance expectation for the possible paths between an origin and a destination. Thus, not only pure cost but also performance can be considered as an element for decision.

4.1.3. Segmented paths and associated characteristics

Original ALTO concentrates on end-to-end paths. However it may result of interest to get knowledge of specific parts of the end-to-end paths that could produce problems such as e.g. congestion. Then, having means of segmenting the end-to-end paths becomes useful. [I-D.ietf-alto-path-vector] allows for that defining a new abstraction called Abstract Network Element (ANE) to represent components constituting an entire end-to-end network path as a vector of ANEs.

4.1.4. In-time view of dynamic IP addressing allocation

Some architectures allow for dynamic allocation of IP address subnets across the network. An example of that is the Control and User Plane Separation (CUPS) architecture for Broadband Network Gateways (BNGs) [I-D.wadhwa-rtgwg-bng-cups], [TR-459]. In that architecture, the control plane of the BNG has the possibility of dynamically assigning IP address subnets to different elements distributed in the network, acting as user plane functions of the BNG. This dynamic allocation implies that certain IP prefixes could be allocated in different parts of the network along the time. By means of ALTO and its network map is it possible to obtain an up-to-date view of the topological location of each subnet in runtime, facilitating the optimization of some services (e.g. media distribution) in an automated manner.

4.2. Prospective use cases

This subsection presents a number of use cases that could be enabled by ALTO as IETF Network Exposure Function.

4.2.1. Determination of optimal compute facility taking into account network information

ALTO can be used as a component to provide insights on the reachability of suitable compute facilities. An initial case has been documented in [I-D.contreras-alto-service-edge]. The rationale for this case is that ALTO receives information of connected compute capabilities in terms of e.g. CPU, memory and storage. This information can be put together with the network map, in a way that the cost of reaching those capabilities can be easily determined.

Note that if further information apart of cost is included in the map (e.g., performance metrics) then the resulting information provided to applications becomes enriched.

4.2.2. Visibility of underlying network information in overlay networks

Different overlay networks run today leveraging the connectivity provided by the basic underlying transport network. Since specific situations on the transport network can result in relevance for the service being provided by the overlays, it is crucial to facilitate the observation of such situations from the underlay to the overlay.

4.2.2.1. Cellular case

Mobile networks leverage transport networks to connect mobile access nodes with core management and control entities (e.g., for mobility management, policing, etc), running in an overlay mode through tunneling (i.e., the GTP protocol). [I-D.li-alto-cellular-use-cases] presents the benefits of exposing network information for applications running on access devices of a cellular network.

4.2.2.2. Media distribution case

Media delivery systems, as traditional CDNs, deliver content to end-users in an over-the-top fashion. The key aspect for an efficient and optimal delivery of the content is to select the proper delivery point for whatever end-user requesting it is to have a clear view of the network topology (including the associated costs or any other information that could enrich the decision, such as performance metrics). In this respect, the information exposed by ALTO in reference to the requesting end-user can be consumed by CDN control elements for improving the decision on what delivery point to select [RFC7971].

Further than that, additional scenarios can benefit from ALTO network information exposure capabilities. For instance, in scenarios of interconnection of CDNs, such as the one described in [I-D.ryan-cdni-capacity-insights-extensions] for advertising capacity associated with the CDN internal to an operator, could leverage on ALTO capabilities for that purpose (with the necessary augmentations).

5. ALTO as IETF Network Exposure Function

From its inception, ALTO was defined as a way of informing applications about network-related aspects for improving the overall service.

The applications under scope can be either internal or external to the operator of the network. The implications can differ in the level of aggregating, abstracting and securely exposing the information, but the purpose keeps being the same.

Figure 1 illustrates the role of ALTO as IETF Network Exposure Function.

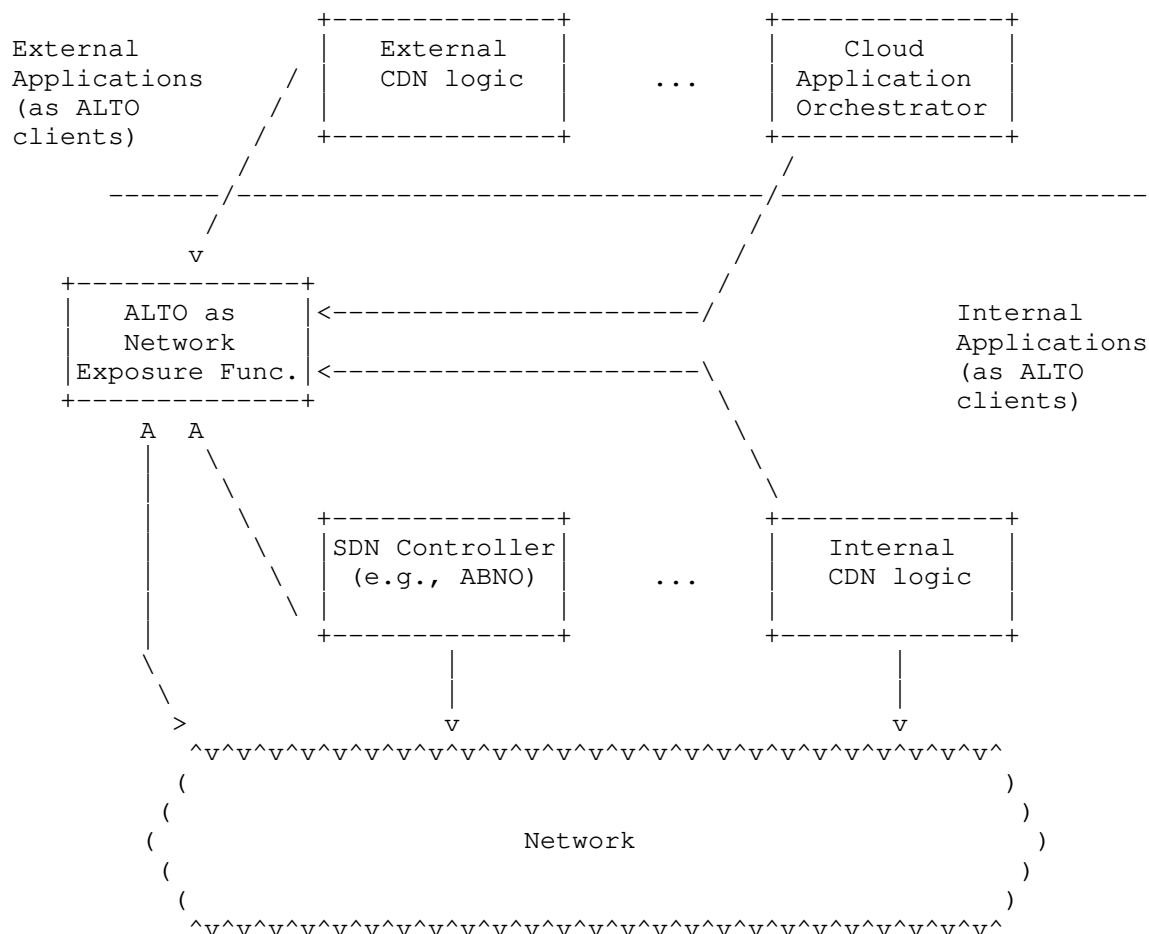


Figure 1: ALTO as IETF Network Exposure Function

Each particular ALTO instance in a certain network could support only a subset of the capabilities discussed in the use cases described before. In this respect, ways of advertising supported capabilities should be defined.

6. TODO for next versions of this document

This version is a work-in-progress. Next versions of the document will address some further aspects such as:

- o Identification of further network capabilities of interest for being exposed by ALTO.
- o Complete security aspects when exposing information to internal and external applications.

7. Security Considerations

ALTO security considerations as reflected in [RFC7285] apply to this document.

(TODO: elaborate on data-related security issues as NEF).

8. IANA Considerations

This draft does not include any IANA considerations

9. References

[I-D.contreras-alto-service-edge]

Contreras, L. M., Lachos, D. A., Rothenberg, C. E., and S. Randriamasy, "Use of ALTO for Determining Service Edge", draft-contreras-alto-service-edge-03 (work in progress), July 2021.

[I-D.ietf-alto-path-vector]

Gao, K., Lee, Y., Randriamasy, S., Yang, Y. R., and J. J. Zhang, "ALTO Extension: Path Vector", draft-ietf-alto-path-vector-18 (work in progress), October 2021.

[I-D.ietf-alto-performance-metrics]

Wu, Q., Yang, Y. R., Lee, Y., Dhody, D., Randriamasy, S., and L. M. C. Murillo, "ALTO Performance Cost Metrics", draft-ietf-alto-performance-metrics-19 (work in progress), October 2021.

[I-D.li-alto-cellular-use-cases]

Gang, L., Randriamasy, S., and C. Xiong, "ALTO Use Cases for Cellular Networks", draft-li-alto-cellular-use-cases-00 (work in progress), July 2021.

- [I-D.ryan-cdni-capacity-insights-extensions]
Ryan, A., Rosenblum, B., and N. B. Sopher, "CDNI Capacity Insights Capability Advertisement Extensions", draft-ryan-cdni-capacity-insights-extensions-01 (work in progress), October 2021.
- [I-D.wadhwa-rtgwg-bng-cups]
Wadhwa, S., Shinde, R., Newton, J., Hoffman, R., Muley, P., and S. Pani, "Architecture for Control and User Plane Separation on BNG", draft-wadhwa-rtgwg-bng-cups-03 (work in progress), March 2019.
- [MEC-013] "GS MEC 013 Location API V2.1.1", ETSI GS MEC 013 V2.1.1 , September 2019.
- [O-RAN] "Non-RT RIC Architecture", O-RAN.WG2.Non-RT-RIC-ARCH-TS-v01.00.02 , July 2021.
- [RFC4655] Farrel, A., Vasseur, J., and J. Ash, "A Path Computation Element (PCE)-Based Architecture", RFC 4655, DOI 10.17487/RFC4655, August 2006, <<https://www.rfc-editor.org/info/rfc4655>>.
- [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/info/rfc6241>>.
- [RFC7149] Boucadair, M. and C. Jacquenet, "Software-Defined Networking: A Perspective from within a Service Provider Environment", RFC 7149, DOI 10.17487/RFC7149, March 2014, <<https://www.rfc-editor.org/info/rfc7149>>.
- [RFC7285] Alimi, R., Ed., Penno, R., Ed., Yang, Y., Ed., Kiesel, S., Previdi, S., Roome, W., Shalunov, S., and R. Woundy, "Application-Layer Traffic Optimization (ALTO) Protocol", RFC 7285, DOI 10.17487/RFC7285, September 2014, <<https://www.rfc-editor.org/info/rfc7285>>.
- [RFC7491] King, D. and A. Farrel, "A PCE-Based Architecture for Application-Based Network Operations", RFC 7491, DOI 10.17487/RFC7491, March 2015, <<https://www.rfc-editor.org/info/rfc7491>>.

- [RFC7971] Stiemerling, M., Kiesel, S., Scharf, M., Seidel, H., and S. Previdi, "Application-Layer Traffic Optimization (ALTO) Deployment Considerations", RFC 7971, DOI 10.17487/RFC7971, October 2016, <<https://www.rfc-editor.org/info/rfc7971>>.
- [TR-459] "Control and User Plane Separation for a disaggregated BNG", Broadband Forum TR-459 , June 2020.
- [TS29.522] "TS 29.522 Network Exposure Function Northbound APIs V16.9.0.", 3GPP TS 29.522 V16.9.0 , September 2021.

Acknowledgments

...

Author's Address

Luis M. Contreras
Telefonica
Ronda de la Comunicacion, s/n
Sur-3 building, 3rd floor
Madrid 28050
Spain

Email: luismiguel.contrerasmurillo@telefonica.com
URI: <http://lmcontreras.com/>

Application-Layer Traffic Optimization
Internet-Draft
Intended status: Informational
Expires: January 12, 2022

P. Liu
Y. Fu
China Mobile
July 11, 2021

Computing-aware Networking Use case of ALTO
draft-liu-alto-can-usecase-00

Abstract

The ever-emerging new services are imposing more and more highly demanding requirements on the network. In order to meet these requirements, some new technology trends of network emerge as the times require. On the one hand, for the selection of service node and forwarding path, in addition to considering the network topology and link state, more factors are also considered, such as the computing properties of service node; On the other hand, network and application present the trend of mutual perception, including application to perceive the state of network path, or network to perceive the demand of application.

This draft describes a new network scenario and architecture considering computational properties, and assumes that Alto could be used as an important node to realize the deployment of services, and to assist in the selection of service nodes.

Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [RFC2119].

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 January 12, 2022.

Copyright Notice

Copyright (c) 2021 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 Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction	2
2. Usage Scenarios of CAN	4
2.1. AR/VR	4
2.2. Internet of Vehicles	5
3. CAN Framework and ALTO	5
4. Deployment of CAN with ALTO	7
5. Scheduling of CAN with ALTO	8
6. Summary	9
7. Security Considerations	9
8. IANA Considerations	9
9. Acknowledgements	9
10. Normative References	9
Authors' Addresses	10

1. Introduction

For new services with heavy computing tasks, such as AR/VR, video recognition and so on, the computing time and network transmission delay are almost the same order of magnitude. In this kind of scenario, computing attributes become more important than traditional services.

The generation of edge computing is to solve such problems. Edge computing is to deploy service nodes close to the user side, which shortens the distance of network transmission. Moreover, it can deploy specific computing resources, such as CPU/GPU, to meet the needs of different services.

It is predicted by Gartner that by 2025, more than 50% of the computing data needs to be analyzed, processed, and stored at the edge. Since the service providers begins to offer the edge computing infrastructure to provide better response time and transfer rate. There are also some challenges of edge computing itself, which are pointed out in the work of dyncast [draft-liu-dyncast-ps-usecases-01], [draft-li-dyncast-architecture-00] :

- * Geographically Scattered Large Number of Edge Sites. The edge sites are highly distributed and may not have proximate distances to user.
- * Resource Limitation. There are fewer servers of server per node.
- * Heterogeneous Hardware, such as CPU, GPU, Memory, ASICs.
- * Dynamic Load. The available resources may change quickly.
- * Edge-cloud Coordination. Edge does not solve all requests.
- * High Cost. On-site maintenance is expensive.
- * Mission Critical. Users are counting on you for 100% reliability of industry automation.

So how to collaboratively deploy and computing services based on the computing resources in network to meet the diverse computing requirements, and achieve the on-demand allocation and dispatch of service request needs be studied.

Some existing works have begun to consider the computing attributes. For example, coinrg initiated the consideration of computing and storage resources. Dyncast proposed how to introduce the scheme of computing metric at the routing level. Semantic routing[], which also extends the semantics of routing in a broad sense. However, today's routing system and technology has been relatively good, the introduction of more metric in routing still need more theoretical and experimental verification. In the work of ITU, it is more from the perspective of architecture, such as ITU Y.CAN-reqts [Y.CAN-reqts: "functional requirements of Computing-aware Networking"] proposed a new network architecture-computing-aware networking (CAN), CAN schedules service request to the optimal computing site along optimal path to meet service requirements both on network and computing. ITU.Y.CPN-arch [Y.CPN-arch: "Framework and architecture of Computing power Network"] provides the framework and architecture of Computing power Network, specifies its functional entities and defines the functionalities of these functional entities. So the convergence of network and computing brought by edge computing

includes the issue of service deployment and service request scheduling. ALTO has done the work of collect the network information for application, it may help to do some work in the two important issues:

How to deploy service nodes based on computing resources. For this point, [draft-contras-alto-service-edge-02] gives the corresponding idea of using Alto to deploy edge computing nodes. Alto can better interact with the upper application, fully understand the requirements of the application, including computing requirements and collect the information of infrastructure resources.

How to select the most suitable node for service request. Alto can also help this kind of work to a certain extent. Centralized selection of service nodes and paths is relatively easy to implement, such as SDN. However, emerging service requests require high real-time performance, and there may be some efficiency and complexity problems, which have been analyzed in the work of dyncast.

2. Usage Scenarios of CAN

Any multi-point deployment service that requires high computing power or low latency will involve the joint scheduling of network and computing resources.

2.1. AR/VR

Take the AR/VR as an example. The upper bound latency for motion-to-photon (MTP) is less than 20 milliseconds to avoid the motion sickness. It consists of four parts, and the frame rendering computing delay is 5.5 milliseconds, so the network delay demand can be calculated as 5.1milliseconds.

Total MTP delay	50ms
sensor sampling delay	<1.5ms
display refresh delay	5 ms
frame rendering delay computing with GPU	5.5ms
network delay	$20-1.5-5.5-7.9=5.1\text{ms}$

Delay of MTP

So the budgets for computing delay and network delay are almost equivalent. Considering another factor that the computing resources have a big difference in different edges, it is necessary to apply dynamically steer traffic to the 'best' edge.

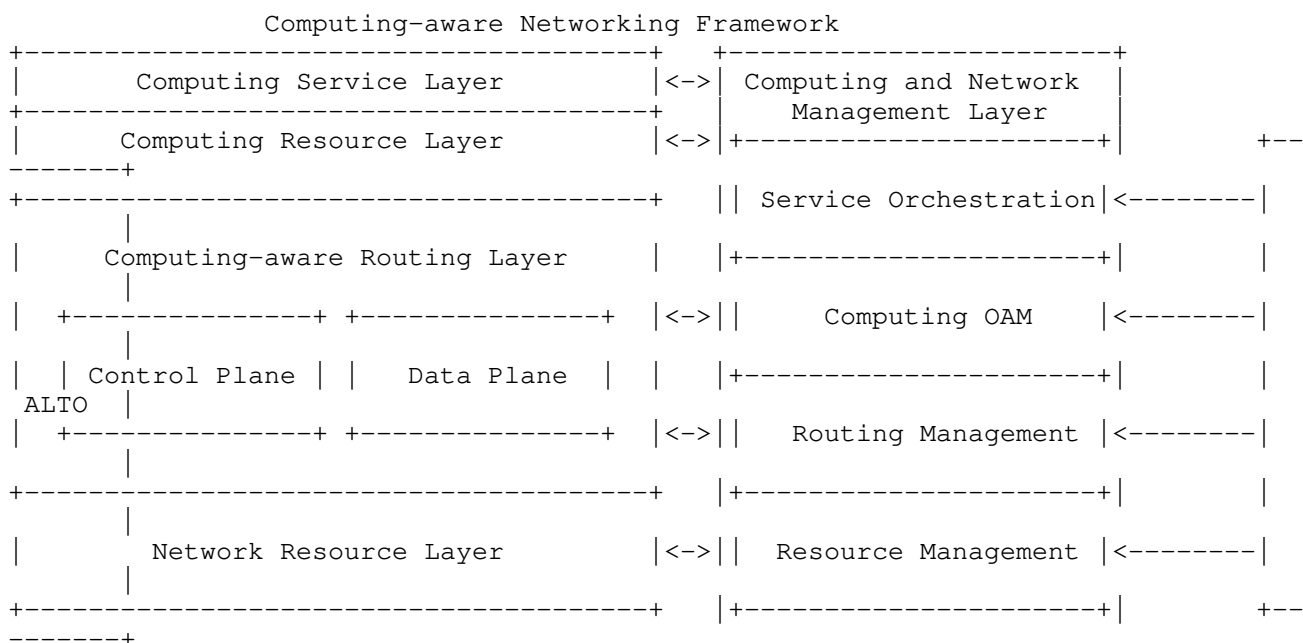
2.2. Internet of Vehicles

Under the scenario of Internet of Vehicles, the services are divided into auxiliary driving and on-board entertainment services. For the auxiliary driving function, for road traffic conditions outside the line of sight due to obstructions, blind areas, etc., the edge computing node obtains comprehensive road traffic information around the location of the vehicle, performs unified data processing, and sends out warning signals for vehicles with potential safety hazards, to assist the safe driving of vehicles.

Apparently, there are obviously differences between services requirements of auxiliary driving services and entertainment services, which needs to be processed by different edge nodes

3. CAN Framework and ALTO

In order to realize ubiquitous computing and service awareness, interconnection and collaborative scheduling, the computing-aware networking architecture can be divided into computing service layer, computing resource layer, computing routing layer, network resource layer, and computing and network management layer. Among them, the computing routing layer contains the control plane and data plane which is shown in Figure 2. Based on the ubiquitous computing resources of the network, the computing resource layer abstracts and models based on a unified measurement and modeling template, and announces computing information to the computing routing layer. And then the computing routing layer comprehensively considers user needs and network resource status and computing resource status, to schedule service requests to appropriate computing nodes. In addition, the computing management layer completes the control and management of computing resources.



CAN Framework and ALTO

* Computing service layer: computing service layer is computing service provider, which deploys, operates and manages many kinds of computing services and applications. In addition, it can realize the functions of service decomposition and service scheduling through the API gateway.

* Computing resource layer: it is based on the existing computing infrastructure, and includes a combination of computing resources from single-core CPU to multi-core CPU, CPU+GPU+FPGA, etc. And it could supply computing modeling function, computing API function, computing resource identification and other functions to meet the diverse computing needs of different applications based on physical computing resources.

* Computing-aware routing layer: It contains control plane and data plane, performs computing-aware routing and generates service scheduling policy in network layer. Based on the discovery of abstracted computing and network resources, computing routing layer generates new routing tables which include the information of computing in network, considers the state of network and computing comprehensively, and thus generates routing policy for different service requests. Network resource layer: It utilizes the existing network infrastructure, which includes access network, metropolitan area network and backbone network, to provide ubiquitous network connection.

* Computing and network management layer: It adds management functions towards computing resources and computing services based on the traditional network management function. Therefore, the computing and network management layer performs service orchestration, resource management, routing measurement and computing OAM. In addition, the computing and network management layer could

be used to perform the pre-configuration function and management function, which interacts with each functional layer.

* Network resource layer: using the existing network infrastructure to provide network connection, network infrastructure includes access network, metropolitan area network and backbone network.

Based on the five functional modules defined above, computing-aware networking can realize the awareness, control and scheduling of computing and network resources, and further perform dynamic and on-demand service scheduling. The function of computing and network management layer may be realized by Alto or by opening interface with Alto server.

4. Deployment of CAN with ALTO

With the development of edge computing, there is multiple services duplication deployed in different edge nodes. To improve the effectiveness of service deployment, the problem of how to choose optimal edge node to deploy services needs to be solved. More stable static information should be considered in service deployment, such as:

* Network topology: the overall consideration of network access, connectivity, path protection or redundancy

* The topology of computing resources: including the location and overall distribution of computing resources in network, and the relative position towards network topology.

* Types of computing resources of edge nodes: such as CPU / GPU, etc

* Location: the number of users brought, the differentiation of service types requested by users, etc

* Location of edge nodes: for edge nodes located in popular area, which with large amount of users and service requests, the service duplication can be deployed more than other areas.

* Capacity of multiple edge nodes: not only a single node, but also the total number of requests that can be processed by the resource pool composed of multiple nodes

* Service category: different types of services require different computing resources. It's necessary to consider the category of computing resources required by the services to deploy services. For example, whether the business is multi-user interaction, such as video conferencing, games, or just resource acquisition, such as

short video viewing Alto can help to obtain one or more of the above information, so as to provide suggestions or formulate principles and strategies for service deployment.

For the collection of those information, second level or minute level frequency is enough, while serious real-time processing isn't necessary. For example, periodically collecting the total consumption of computing resources, or the total number of sessions accessed, to notify where to depoly more VMS or containers. Unlike the scheduling of request, service deployment should still follow the principle of proximity. The more local access, the more resources should be deployed. If the resources are insufficient, the operator can be informed to increase the hardware resources. Alto can be used to transmit information.

5. Scheduling of CAN with ALTO

Compared to the deployment, scheduling needs to consider more dynamic information to select and adjust the optimal (rather than the shortest) path in real timesuch as:

- * Mobility: CAN schedules service request to the optimal service node among several service nodes near to users. So when user mobiles, the nearby service nodes changes and new scheduling are needed to chooses new path and new service node.

- * Real time delay of network: network delay is always in the process of dynamic change, and more and more services propose strict time requirements, which is one of the most important factors affecting user experience.

- * Real time status of computing resources: computing resources change frequently and the status of computing resources heavily affect the completion time of service, which is also one of the most important factors affecting user experience.

- * Comprehensive status of network status and computing status: the update frequency of computing status and network status is different, it is necessary to generate a comprehensive value to reflect the status of them.

- * Various service requirements: different services propose different service requirements on computing and network, including bandwidth, latency, computing resources etc, and the latency includes both transmission latency in network and processing latency in service node, for transmission-intensive services, the transmission latency accounts a lot , so the network latency of services are more important.

Availability of network or computing resources: such as temporary unavailability caused by network equipment or service node failure.

Alto can still help collect real-time network and service node information：

* Providing the best choice of network and service nodes. Based on the collected network information, computing information, and service requirements on network and computing, Of course, there are still some real-time and complexity problems.

* Providing data analysis and policy distribution, real-time node selection still depends on distributed routing, such as dyncast.

6. Summary

The converge of network and computing, as well as the interaction with applications has become one of the current technical development directions. This draft analyzes the development status of the current computing aware network and the functional modules in its architecture that can interact with Alto. As a protocol to connect networks and applications, Alto may play a more important role.

7. Security Considerations

TBD.

8. IANA Considerations

TBD.

9. Acknowledgements

Thanks to Yizhou Li, Qin Wu, Tianji Jiang for helpful suggestion. Thanks go to Dirk Trossen, Luigi Iannone, and Carsten Bormann for their inspiring Dyncast work.

10. Normative References

[I-D.li-dyncast-architecture]

Li, Y., Iannone, L., Trossen, D., and P. Liu, "Dynamic-Anycast Architecture", draft-li-dyncast-architecture-00 (work in progress), February 2021.

[I-D.liu-dyncast-ps-usecases]

Liu, P., Willis, P., and D. Trossen, "Dynamic-Anycast (Dyncast) Use Cases & Problem Statement", draft-liu-dyncast-ps-usecases-01 (work in progress), February 2021.

[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>>.

Authors' Addresses

Peng Liu
China Mobile
Beijing 100053
China

Email: liupengyjy@chinamobile.com

Yuxia Fu
China Mobile
Beijing 100053
China

Email: fuyuxia@chinamobile.com

ALTO Working Group
Internet-Draft
Intended status: Standards Track
Expires: April 27, 2022

Y. Yang
Yale University
October 24, 2021

ALTO Transport using HTTP/2
draft-yang-alto-http2-transport-01

Abstract

The ALTO base protocol [RFC7285] uses HTTP/1.x as the transport protocol and hence ALTO transport includes the limitations of HTTP/1.x. ALTO/SSE [RFC8895] addresses some of the limitations, but is still based on HTTP/1.x. This document introduces ALTO/H2, which provides the transport of ALTO information resources on top of HTTP/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.

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 <http://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 April 27, 2022.

Copyright Notice

Copyright (c) 2021 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 (<http://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 Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction	2
2. ALTO/H2 Design Requirements	3
3. ALTO/H2 Information Resource Directory (IRD)	4
4. Security Considerations	7
5. ALTO based on HTTP/3 Considerations	7
6. IANA Considerations	7
7. Acknowledgments	7
8. References	7
8.1. Normative References	7
8.2. Informative References	8
Author's Address	8

1. Introduction

Application-Layer Traffic Optimization (ALTO) provides a means for network applications to obtain network status information. The ALTO base protocol [RFC7285] is based on the sequential request and response model of HTTP/1.1 [RFC7230]; hence, in the base protocol, an ALTO client only can issue a sequence of requests on network information resources, and the ALTO server sends the information resources one-by-one, in the order of the request sequence.

To address the use cases where an ALTO client may need to efficiently monitor changes to a set of network information resources and the protocol is still based on the HTTP/1.1 model, the ALTO Working Group introduces ALTO/SSE (ALTO Incremental Update based on Server-Sent-Event) [RFC8895], so that an ALTO client can manage (i.e., add and remove) a set of requests maintained at an ALTO server, and the server can continuously, concurrently, and incrementally push updates whenever a monitored network information resource changes. Figure 1 shows the architecture and message flow of ALTO/SSE, which can be considered as a more general transport protocol than the ALTO base transport protocol. Although ALTO/SSE allows the concurrent transport of multiple ALTO information resources, it has complexities and limitations: (1) it requires that the server provide a separate control URI, leading to complexity in management; (2) the data

connection still suffers HTTP/1.x problems such as head-of-line (HOL) blocking.

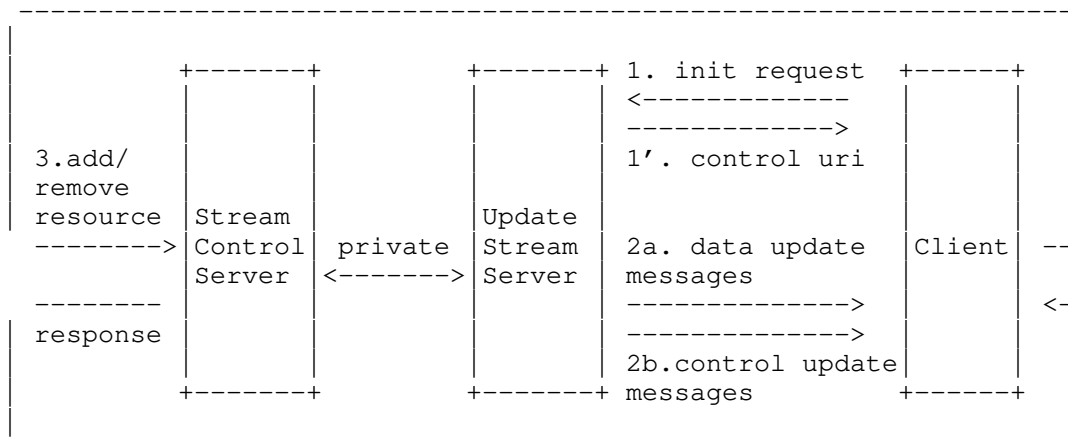


Figure 1: ALTO SSE Architecture and Message Flow.

This document specifies ALTO/H2, based on HTTP/2 [RFC7540]. The newer transport can provide multiple benefits:

- o ALTO based on HTTP/2 natively supports multiple concurrent requests pending at the server, realizing a main design goal of ALTO SSE. At the same time, instead of using multiple TCP connections (a control connection and a data connection) as ALTO/SSE does, ALTO based on HTTP/2 can use a single TCP connection, achieving a design with only one connection.
- o ALTO based on HTTP/2 can take advantage of the benefit provided by HTTP/2 to avoid head-of-line blocking in the data connection, reducing latency.
- o ALTO based on HTTP/2 can take advantage of other benefits of HTTP/2, including potentially higher encoding efficiency and stronger enforcement of security (i.e., using https).

2. ALTO/H2 Design Requirements

To realize the functions of ALTO/SSE, there are the following requirements:

- o R0: Client can request any resource using the connection, just as using ALTO base protocol using HTTP/1.x.
- o R1: The client can signal the addition (start) of incremental updates to a resource.
- o R2: The client can signal the deletion (stop) of incremental updates to a resource.
- o R3: The server can signal to the client the start or stop of incremental updates to a resource.
- o R4: The server can choose the type of each incremental update encoding, as long as the type is indicated to be acceptable by the client.

3. ALTO/H2 Information Resource Directory (IRD)

Extending the IRD example in Section 8.1 of [RFC8895], Figure 2 is the IRD of an ALTO server supporting ALTO base protocol, ALTO/SSE, and ALTO/H2.

In particular,

```
"my-network-map": {
  "uri": "https://alto.example.com/networkmap",
  "media-type": "application/alto-networkmap+json",
},
"my-routingcost-map": {
  "uri": "https://alto.example.com/costmap/routingcost",
  "media-type": "application/alto-costmap+json",
  "uses": ["my-networkmap"],
  "capabilities": {
    "cost-type-names": ["num-routingcost"]
  }
},
"my-hopcount-map": {
  "uri": "https://alto.example.com/costmap/hopcount",
  "media-type": "application/alto-costmap+json",
  "uses": ["my-networkmap"],
  "capabilities": {
    "cost-type-names": ["num-hopcount"]
  }
},
"my-filtered-cost-map": {
  "uri": "https://alto.example.com/costmap/filtered/constraints",
  "media-type": "application/alto-costmap+json",
  "accepts": "application/alto-costmapfilter+json",
}
```



```
    "uses": ["my-networkmap"],
    "capabilities": {
      "cost-type-names": ["num-routingcost", "num-hopcount"],
      "cost-constraints": true
    }
  },
  "my-simple-filtered-cost-map": {
    "uri": "https://alto.example.com/costmap/filtered/simple",
    "media-type": "application/alto-costmap+json",
    "accepts": "application/alto-costmapfilter+json",
    "uses": ["my-networkmap"],
    "capabilities": {
      "cost-type-names": ["num-routingcost", "num-hopcount"],
      "cost-constraints": false
    }
  },
  "my-props": {
    "uri": "https://alto.example.com/properties",
    "media-type": "application/alto-endpointprops+json",
    "accepts": "application/alto-endpointpropparams+json",
    "capabilities": {
      "prop-types": ["priv:ietf-bandwidth"]
    }
  },
  "my-pv": {
    "uri": "https://alto.example.com/endpointcost/pv",
    "media-type": "multipart/related;
      type=application/alto-endpointcost+json",
    "accepts": "application/alto-endpointcostparams+json",
    "capabilities": {
      "cost-type-names": [ "path-vector" ],
      "ane-properties": [ "maxresbw", "persistent-entities" ]
    }
  },
  "update-my-costs": {
    "uri": "https://alto.example.com/updates/costs",
    "media-type": "text/event-stream",
    "accepts": "application/alto-updatestreamparams+json",
    "uses": [
      "my-network-map",
      "my-routingcost-map",
      "my-hopcount-map",
      "my-simple-filtered-cost-map"
    ],
    "capabilities": {
      "incremental-change-media-types": {
        "my-network-map": "application/json-patch+json",
        "my-routingcost-map": "application/merge-patch+json",

```



```
        "my-hopcount-map": "application/merge-patch+json"
      },
      "support-stream-control": true
    }
  },
  "update-my-costs-h2": {
    "uri": "https://alto.example.com/updates-h2/costs",
    "media-type": "application/alto-h2",
    "accepts": "application/alto-updatestreamparams+json",
    "uses": [
      "my-network-map",
      "my-routingcost-map",
      "my-hopcount-map",
      "my-simple-filtered-cost-map"
    ],
    "capabilities": {
      "incremental-change-media-types": {
        "my-network-map": "application/json-patch+json",
        "my-routingcost-map": "application/merge-patch+json",
        "my-hopcount-map": "application/merge-patch+json"
      },
      "support-stream-control": true
    }
  },
  "update-my-props": {
    "uri": "https://alto.example.com/updates/properties",
    "media-type": "text/event-stream",
    "uses": [ "my-props" ],
    "accepts": "application/alto-updatestreamparams+json",
    "capabilities": {
      "incremental-change-media-types": {
        "my-props": "application/merge-patch+json"
      },
      "support-stream-control": true
    }
  },
  "update-my-pv": {
    "uri": "https://alto.example.com/updates/pv",
    "media-type": "text/event-stream",
    "uses": [ "my-pv" ],
    "accepts": "application/alto-updatestreamparams+json",
    "capabilities": {
      "incremental-change-media-types": {
        "my-pv": "application/merge-patch+json"
      },
      "support-stream-control": true
    }
  }
}
```


}

Note that it is straightforward for an ALTO sever to run HTTP/2 and support concurrent retrieval of multiple resources such as "my-network-map" and "my-routingcost-map" using multiple HTTP/2 streams with the need to introducing ALTO/H2.

The resource "update-my-costs-h2" provides an ALTO/H2 based connection, and this is indicated by the media-type "application/alto-h2". For an ALTO/H2 connection, the client can send in a sequence of control requests using media type application/alto-updatestreamparams+json. The server creates HTTP/2 streams and pushes updates to the client.

4. Security Considerations

The properties defined in this document present no security considerations beyond those in Section 15 of the base ALTO specification [RFC7285].

5. ALTO based on HTTP/3 Considerations

One consideration of the ALTO/H2 design is the pending deployment of HTTP3.

6. IANA Considerations

IANA will need to register the alto-h2 media type under ALTO registry as defined in [RFC7285].

7. Acknowledgments

The authors of this document would also like to thank many for the reviews and comments.

8. References

8.1. 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>>.
- [RFC7230] Fielding, R., Ed. and J. Reschke, Ed., "Hypertext Transfer Protocol (HTTP/1.1): Message Syntax and Routing", RFC 7230, DOI 10.17487/RFC7230, June 2014, <<https://www.rfc-editor.org/info/rfc7230>>.

- [RFC7285] Alimi, R., Ed., Penno, R., Ed., Yang, Y., Ed., Kiesel, S., Previdi, S., Roome, W., Shalunov, S., and R. Woundy, "Application-Layer Traffic Optimization (ALTO) Protocol", RFC 7285, DOI 10.17487/RFC7285, September 2014, <<https://www.rfc-editor.org/info/rfc7285>>.
- [RFC7540] Belshe, M., Peon, R., and M. Thomson, Ed., "Hypertext Transfer Protocol Version 2 (HTTP/2)", RFC 7540, DOI 10.17487/RFC7540, May 2015, <<https://www.rfc-editor.org/info/rfc7540>>.
- [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>>.
- [RFC8895] Roome, W. and Y. Yang, "Application-Layer Traffic Optimization (ALTO) Incremental Updates Using Server-Sent Events (SSE)", RFC 8895, DOI 10.17487/RFC8895, November 2020, <<https://www.rfc-editor.org/info/rfc8895>>.

8.2. Informative References

- [RFC7971] Stiemerling, M., Kiesel, S., Scharf, M., Seidel, H., and S. Previdi, "Application-Layer Traffic Optimization (ALTO) Deployment Considerations", RFC 7971, DOI 10.17487/RFC7971, October 2016, <<https://www.rfc-editor.org/info/rfc7971>>.

Author's Address

Y. Richard Yang
Yale University
51 Prospect St
New Haven, CT 06520
USA

Email: yry@cs.yale.edu

ALTO WG
Internet-Draft
Intended status: Informational
Expires: 13 January 2022

J. Zhang
Tongji University
D. Dhody
Huawei Technologies
K. Gao
Sichuan University
12 July 2021

A Yang Data Model for Operations, Administration, and Maintenance of
ALTO Protocol
draft-zhang-alto-oam-yang-00

Abstract

This document defines a YANG data model for the operations and management of Application-Layer Traffic Optimization (ALTO) Protocol. The operator can use the data model to create and update ALTO information resources, manage the access control, configure server-to-server communication and server discovery, and collect statistical data.

Discussion Venues

This note is to be removed before publishing as an RFC.

Discussion of this document takes place on the ALTO Working Group mailing list (alto@ietf.org), which is archived at <https://mailarchive.ietf.org/arch/browse/alto/>.

Source for this draft and an issue tracker can be found at <https://github.com/openalto/draft-alto-oam-yang>.

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 January 2022.

Copyright Notice

Copyright (c) 2021 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 Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction	3
2. Requirements Language	3
3. Terminology	3
3.1. Tree Diagrams	3
3.2. Prefixes in Data Node Names	3
4. Objectives	4
5. Design of ALTO OAM Data Model	4
5.1. Overview of ALTO OAM Data Model	4
5.2. Meta Information of ALTO Server	6
5.3. Intent-based Interfaces for ALTO Information Resources Management	7
5.3.1. Information Resource Creation Algorithm Example	9
5.4. Data Sources	10
5.4.1. Internal Data Source	11
5.4.2. External Data Source	11
5.5. Model for ALTO Server-to-server Communication	11
5.6. Model for ALTO Statistics	11
6. Security Considerations	11
7. References	11
7.1. Normative References	12
7.2. Informative References	13
Appendix A. The Full ALTO OAM Data Model	14
Authors' Addresses	14

1. Introduction

This document defines a YANG data model for the operations and management of Application-Layer Traffic Optimization (ALTO) Protocol. The basic propose of this YANG data model is discussed in Section 16 of [RFC7285]. The operator can use the data model to create and update ALTO information resources, manage the access control, configure server-to-server communication and server discovery, and collect statistical data.

The basic structure of this YANG data model is guided by Section 16 of [RFC7285] and [RFC7971]. Although the scope of the YANG data model in this document mainly focuses on the support of the base ALTO protocol [RFC7285] and the existing ALTO standard extensions (including [RFC8189], [RFC8895] and [RFC8896]), the design will also be extensible for future standard extensions (e.g., [I-D.ietf-alto-path-vector], [I-D.ietf-alto-unified-props-new], [I-D.ietf-alto-cost-calendar], and [I-D.ietf-alto-performance-metrics]).

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. When the words appear in lower case, they are to be interpreted with their natural language meanings.

3. Terminology

3.1. Tree Diagrams

A simplified graphical representation of the data model is used in this document. The meaning of the symbols in these diagrams is defined in [RFC8340].

3.2. Prefixes in Data Node Names

In this document, names of data nodes and other data model objects are often used without a prefix, as long as it is clear from the context in which YANG module each name is defined. Otherwise, names are prefixed using the standard prefix associated with the corresponding YANG module, as shown in Table 1.

Prefix	YANG module	Reference
yang	ietf-yang-types	[RFC6991]
inet	ietf-inet-types	[RFC6991]

Table 1: Prefixes and corresponding
YANG modules

4. Objectives

This section describes the design objectives for the YANG data model:

- * The data model should provide intent-based interfaces for administrators to create, update and remove ALTO information resources.
 - The data model should be extensible for new ALTO information resources.
 - The data model should allow developers to augment new APIs for ALTO information resource generation.
- * The data model should support access control at the information resource level.
- * The data model should collect statistics information of the requests to each ALTO information resource.

NOTE: The data model supporting configuration for the ALTO client and the communication between the administrated ALTO server and other ALTO servers will be considered in a future version of the document.

5. Design of ALTO OAM Data Model

5.1. Overview of ALTO OAM Data Model

The ALTO YANG module defined in this document has all the common building blocks for ALTO OAM.

NOTE: So far, the ALTO YANG module only focuses on the ALTO server related configuration. The ALTO client related configuration will be added in a future version of the document.

The container "alto-server" in the ALTO yang module contains all the configured and operational parameters of the administrated ALTO server instance.

```

module: ietf-alto
  +--rw alto-server
    +--rw hostname inet:host
    +--rw cost-type* [cost-type-name]
      |   +--rw cost-type-name string
      |   +--rw cost-mode      cost-mode
      |   +--rw cost-metric    cost-metric
    --rw meta* [meta-key]
      |   +--rw meta-key      string
      |   +--rw meta-value    string
    +--rw resource* [resource-id]
      |   +--rw resource-id    resource-id
      |   +--rw resource-type  identityref
      |   +--rw description?   string
      |   +--rw accepted-group* [user-group]
      |   +--rw dependency*     resource-id
      |   ...
    +--rw (resource-params)
      +--:(ird)
      |   +--rw alto-ird-params
      |   |   +--rw delegation          inet:uri
      +--:(networkmap)
      |   +--rw alto-networkmap-params
      |   |   +--rw is-default?          boolean
      |   |   +--rw filtered?            boolean
      |   |   +--rw (algorithm)
      +--:(costmap)
      |   +--rw alto-costmap-params
      |   |   +--rw filtered?            boolean
      |   |   +--rw cost-type-names*     string
      |   |   +--rw cost-constraints?    boolean
      |   |   +--rw max-cost-types?      uint32 {multi-cost}?
      |   |   +--rw testable-cost-type-names*
      |   |   |   string {multi-cost}?
      |   |   +--rw calendar?            boolean {cost-calendar}?
      |   |   +--rw (algorithm)
      +--:(endpointcost)
      |   +--rw alto-endpointcost-params
      |   |   +--rw cost-type* [cost-mode,cost-metric]
      |   |   |   +--rw cost-mode      cost-mode
      |   |   |   +--rw cost-metric    cost-metric
      |   |   +--rw cost-constraints?  boolean
      |   |   +--rw max-cost-types?    uint32 {multi-cost}?
      |   |   +--rw testable-cost-type-names*

```



```

|         |         string {multi-cost}?
|         |         +---rw calendar?         boolean {cost-calendar}?
|         |         +---rw (algorithm)
|         +---:(endpointprop)
|         |         +---rw alto-endpointprop-params
|         |         +---rw (algorithm)
|         +---:(propmap) {propmap}?
|         |         +---rw alto-propmap-params
|         |         +---rw (algorithm)
|         +---:(cdni) {cdni}?
|         |         +---rw alto-cdni-params
|         |         +---rw (algorithm)
|         +---:(update) {incr-update}?
|         |         +---rw alto-update-params
|         |         +---rw (algorithm)
+---rw data-source* [source-id]
|   +---rw source-id    string
|   +---rw source-type  identityref
|   +---rw (update-policy)
|   |   +---:(reactive)
|   |   |   +---rw reactive          boolean
|   |   +---:(proactive)
|   |   |   +---rw poll-interval      uint32
+---rw (source-params)
|   +---:(internal)
|   |   +---rw internal-source-params
|   |   |   +---rw source-path        yang:xpath1.0
|   +---:(external)
|   |   +---rw external-source-params
|   |   |   +---rw source-uri          inet:uri
|   |   |   +---rw query-data?         string

```

5.2. Meta Information of ALTO Server

The ALTO server instance contains the following basic configurations for the server setup.

The hostname is the name that is used to access the ALTO server. It will be also used in the URI of each information resource provided by the ALTO server.

The cost type list is the registry for the cost types that can be used in the ALTO server.

The "meta" list contains the customized meta data of the ALTO server. It will be populated into the meta field of the default Information Resource Directory (IRD).


```
module: ietf-alto
  +--rw alto-server
    +--rw hostname inet:host
    +--rw cost-type* [cost-type-name]
      +--rw cost-type-name string
      +--rw cost-mode cost-mode
      +--rw cost-metric cost-metric
    --rw meta* [meta-key]
      +--rw meta-key string
      +--rw meta-value string
    ...
```

5.3. Intent-based Interfaces for ALTO Information Resources Management

The ALTO server instance contains a list of "resource" entries. Each "resource" entry contains the configurations of an ALTO information resource (See Section 8.1 of [RFC7285]). The operator of the ALTO server can use this model to create, update, and remove the ALTO information resource.

Each "resoruce" entry is considered as an intent to create or update an ALTO information resource. Adding a new "resource" entry will submit an ALTO information resource creation intent to the intent system to create a new ALTO information resource. Updating an existing "resource" entry will update the corresponding ALTO information resource creation intent. Removing an existing "resource" entry will remove the corresponding ALTO information resource creation intent and also the created ALTO information resource.

The parameter of the intent interface defined by a "resource" entry MUST include a unique "resource-id" and a "resource-type".

It can also include an "accepted-group" node containing a list of "user-group"s that can access this ALTO information resource.

For some "resource-type", the parameter of the intent interface MUST also include the a "dependency" node containing the "resource-id" of the dependent ALTO information resources (See Section 9.1.5 of [RFC7285]).

For each type of ALTO information resource, the creation intent MAY also need type-specific parameters. These type-specific parameters include two categories:

1. One categories of the type-specific parameters are common for the same type of ALTO information resource. They declare the Capabilities of the ALTO information resource (See Section 9.1.3 of [RFC7285]).
2. The other categories of the type-specific parameters are algorithm-specific. The developer of the ALTO server can implement their own creation algorithms and augment the "algorithm" node to declare algorithm-specific input parameters.

Except for the "ird" resource, all the other types of "resource" entries have augmented "algorithm" node. The augmented "algorithm" node can reference data sources subscribed by the "data-source" entries (See Section 5.4).

The developer cannot customize the creation algorithm of the "ird" resource. The default "ird" resource will be created automatically based on all the added "resource" entries. The delegated "ird" resource will be created as a static ALTO information resource (See Section 9.2.4 of [RFC7285]).

```

module: ietf-alto
  +--rw alto-server
    ...
    +--rw resource* [resource-id]
      +--rw resource-id      resource-id
      +--rw resource-type    identityref
      +--rw description?     string
      +--rw accepted-group*  [user-group]
      +--rw dependency*      resource-id
      | ...
      +--rw (resource-params)
        +--:(ird)
          +--rw alto-ird-params
            +--rw delegation          inet:uri
        +--:(networkmap)
          +--rw alto-networkmap-params
            +--rw is-default?         boolean
            +--rw filtered?           boolean
            +--rw (algorithm)
        +--:(costmap)
          +--rw alto-costmap-params
            +--rw filtered?           boolean
            +--rw cost-type-names*    string
            +--rw cost-constraints?   boolean
            +--rw max-cost-types?     uint32 {multi-cost}?
            +--rw testable-cost-type-names*
              string {multi-cost}?

```



```

|         +---rw calendar?          boolean {cost-calendar}?
|         +---rw (algorithm)
+---:(endpointcost)
|   +---rw alto-endpointcost-params
|   +---rw cost-type* [cost-mode,cost-metric]
|   |   +---rw cost-mode          cost-mode
|   |   +---rw cost-metric        cost-metric
|   +---rw cost-constraints?    boolean
|   +---rw max-cost-types?      uint32 {multi-cost}?
|   +---rw testable-cost-type-names*
|   |   string {multi-cost}?
|   +---rw calendar?          boolean {cost-calendar}?
|   +---rw (algorithm)
+---:(endpointprop)
|   +---rw alto-endpointprop-params
|   +---rw (algorithm)
+---:(propmap) {propmap}?
|   +---rw alto-propmap-params
|   +---rw (algorithm)
+---:(cdni) {cdni}?
|   +---rw alto-cdni-params
|   +---rw (algorithm)
+---:(update) {incr-update}?
|   +---rw alto-update-params
|   +---rw (algorithm)
...

```

5.3.1. Information Resource Creation Algorithm Example

The following example shows how the developer can augment a creation algorithm for the network map resource.

```

augment /alto:alto-server/alto:resource/alto:resource-params
  /alto:networkmap/alto:alto-networkmap-params
  /alto:algorithm:
+---rw l3-unicast-cluster-algorithm
|   +---rw l3-unicast-topo
|   |   -> /alto:alto-server/data-source/source-id
|   +---rw depth?      uint32

```

This example defines a creation algorithm called "l3-unicast-cluster-algorithm" for the network map resource. It takes two algorithm-specific parameters:

l3-unicast-topo This parameter refers to the source id of a data source node subscribed in the "data-source" list (See Section 5.4). The corresponding data source is assumed to be an internal data source (See Section 5.4.1) for an IETF layer 3

unicast topology defined in [RFC8346]. The algorithm uses the topology data from this data source to compute the ALTO network map resource.

depth This optional parameter sets the depth of the clustering algorithm. For example, if the depth sets to 1, the algorithm will generate PID for every 13-node in the topology.

The creation algorithm can be reactively called once the referenced data source updates. Therefore, the ALTO network map resource can be updated dynamically. The update of the reference data source depends on the used "update-policy" (See Section 5.4).

5.4. Data Sources

The ALTO server instance contains a list of "data-source" entries to subscribe the data sources from which ALTO information resources are derived (See Section 16.2.4 of [RFC7285]).

A "data-source" entry MUST include:

- * a unique "source-id" for resource creation algorithms to reference,
- * the "source-type" attribute to declare the type of the data source,
- * the "update-policy" to specify how to get the data update from the data source,
- * the "source-params" to specify where and how to query the data.

The update policy can be either reactive or proactive. For the reactive update, the ALTO server gets the update as soon as the data source changes. For the proactive update, the ALTO server has to proactively fetch the data source periodically.

To use the reactive update, the "reactive" attribute MUST be set true. To use the proactive update, the "poll-interval" attribute MUST be greater than zero. The value of "poll-interval" specifies the interval of fetching the data in milliseconds. If "reactive" is false or "poll-interval" is zero, the ALTO server will not update the data source.

The target of the data source can be either internal or external.


```

module: ietf-alto
  +--rw alto-server
    ...
    +--rw data-source* [source-id]
      +--rw source-id    string
      +--rw source-type  identityref
      +--rw (update-policy)
        | +--:(reactive)
        | | +--rw reactive          boolean
        | +--:(proactive)
        | | +--rw poll-interval    uint32
      +--rw (source-params)
        +--:(internal)
        | +--rw internal-source-params
        | | +--rw source-path      yang:xpath1.0
        +--:(external)
        | +--rw external-source-params
        | | +--rw source-uri       inet:uri
        | | +--rw query-data?     string

```

5.4.1. Internal Data Source

The "internal-source-params" is used to subscribe the internal data source which is located in the same YANG model-driven data store supplying the current ALTO OAM data model. The "source-path" is used to specify the XPath of the data source node.

5.4.2. External Data Source

The "external-source-params" is used to subscribe the external data source which is located in other database systems, e.g., an SNMP server, a prometheus monitor, or a SQL database. The "source-uri" is used to establish the connection with the external data source. The "query-data" is used to specify the potential query expression.

5.5. Model for ALTO Server-to-server Communication

TBD.

5.6. Model for ALTO Statistics

TBD.

6. Security Considerations

TBD.

7. References

7.1. 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>>.
- [RFC6991] Schoenwaelder, J., Ed., "Common YANG Data Types", RFC 6991, DOI 10.17487/RFC6991, July 2013, <<https://www.rfc-editor.org/info/rfc6991>>.
- [RFC7285] Alimi, R., Ed., Penno, R., Ed., Yang, Y., Ed., Kiesel, S., Previdi, S., Roome, W., Shalunov, S., and R. Woundy, "Application-Layer Traffic Optimization (ALTO) Protocol", RFC 7285, DOI 10.17487/RFC7285, September 2014, <<https://www.rfc-editor.org/info/rfc7285>>.
- [RFC7286] Kiesel, S., Stiernerling, M., Schwan, N., Scharf, M., and H. Song, "Application-Layer Traffic Optimization (ALTO) Server Discovery", RFC 7286, DOI 10.17487/RFC7286, November 2014, <<https://www.rfc-editor.org/info/rfc7286>>.
- [RFC7971] Stiernerling, M., Kiesel, S., Scharf, M., Seidel, H., and S. Previdi, "Application-Layer Traffic Optimization (ALTO) Deployment Considerations", RFC 7971, DOI 10.17487/RFC7971, October 2016, <<https://www.rfc-editor.org/info/rfc7971>>.
- [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>>.
- [RFC8189] Randriamasy, S., Roome, W., and N. Schwan, "Multi-Cost Application-Layer Traffic Optimization (ALTO)", RFC 8189, DOI 10.17487/RFC8189, October 2017, <<https://www.rfc-editor.org/info/rfc8189>>.
- [RFC8340] Bjorklund, M. and L. Berger, Ed., "YANG Tree Diagrams", BCP 215, RFC 8340, DOI 10.17487/RFC8340, March 2018, <<https://www.rfc-editor.org/info/rfc8340>>.
- [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/info/rfc8571>>.

- [RFC8686] Kiesel, S. and M. Stiernerling, "Application-Layer Traffic Optimization (ALTO) Cross-Domain Server Discovery", RFC 8686, DOI 10.17487/RFC8686, February 2020, <<https://www.rfc-editor.org/info/rfc8686>>.
- [RFC8895] Roome, W. and Y. Yang, "Application-Layer Traffic Optimization (ALTO) Incremental Updates Using Server-Sent Events (SSE)", RFC 8895, DOI 10.17487/RFC8895, November 2020, <<https://www.rfc-editor.org/info/rfc8895>>.
- [RFC8896] Randriamasy, S., Yang, R., Wu, Q., Deng, L., and N. Schwan, "Application-Layer Traffic Optimization (ALTO) Cost Calendar", RFC 8896, DOI 10.17487/RFC8896, November 2020, <<https://www.rfc-editor.org/info/rfc8896>>.

7.2. Informative References

- [I-D.ietf-alto-cost-calendar]
Randriamasy, S., Yang, R., Wu, Q., Deng, L., and N. Schwan, "Application-Layer Traffic Optimization (ALTO) Cost Calendar", Work in Progress, Internet-Draft, draft-ietf-alto-cost-calendar-21, 17 March 2020, <<https://www.ietf.org/archive/id/draft-ietf-alto-cost-calendar-21.txt>>.
- [I-D.ietf-alto-path-vector]
Gao, K., Lee, Y., Randriamasy, S., Yang, Y. R., and J. J. Zhang, "ALTO Extension: Path Vector", Work in Progress, Internet-Draft, draft-ietf-alto-path-vector-14, 22 February 2021, <<https://www.ietf.org/archive/id/draft-ietf-alto-path-vector-14.txt>>.
- [I-D.ietf-alto-performance-metrics]
Wu, Q., Yang, Y. R., Lee, Y., Dhody, D., Randriamasy, S., and L. M. Contreras, "ALTO Performance Cost Metrics", Work in Progress, Internet-Draft, draft-ietf-alto-performance-metrics-16, 11 July 2021, <<https://www.ietf.org/archive/id/draft-ietf-alto-performance-metrics-16.txt>>.
- [I-D.ietf-alto-unified-props-new]
Roome, W., Randriamasy, S., Yang, Y. R., Zhang, J. J., and K. Gao, "ALTO Extension: Entity Property Maps", Work in Progress, Internet-Draft, draft-ietf-alto-unified-props-new-17, 16 April 2021, <<https://www.ietf.org/archive/id/draft-ietf-alto-unified-props-new-17.txt>>.

[RFC8346] Clemm, A., Medved, J., Varga, R., Liu, X.,
Ananthakrishnan, H., and N. Bahadur, "A YANG Data Model
for Layer 3 Topologies", RFC 8346, DOI 10.17487/RFC8346,
March 2018, <<https://www.rfc-editor.org/info/rfc8346>>.

Appendix A. The Full ALTO OAM Data Model

TODO: The complete tree diagram and YANG modules will be updated soon.

Authors' Addresses

Jingxuan Jensen Zhang
Tongji University
4800 Cao'An Hwy
Shanghai
201804
China

Email: jingxuan.n.zhang@gmail.com

Dhruv Dhody
Huawei Technologies
Divyashree Techno Park, Whitefield
Bangalore 560066
Karnataka
India

Email: dhruv.ietf@gmail.com

Kai Gao
Sichuan University
No.24 South Section 1, Yihuan Road
Chengdu
610000
China

Email: kaigao@scu.edu.cn