

ALTO WG
Internet-Draft
Intended status: Informational
Expires: 27 April 2023

LM. Contreras
Telefonica
D. Lachos
BENOCS
C. Rothenberg
Unicamp
S. Randriamasy
Nokia Bell Labs
24 October 2022

Use of ALTO for Determining Service Edge
draft-contreras-alto-service-edge-06

Abstract

Service providers are starting to deploy and interconnect computing capabilities across the network for hosting network functions and applications. In distributed computing environments, both computing and topological information are necessary in order to determine the more convenient infrastructure where to deploy such a service or application. This document proposes an initial approach towards the use of ALTO to provide such information and assist the selection of appropriate deployment locations for services and applications.

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 27 April 2023.

Copyright Notice

Copyright (c) 2022 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to [BCP 78](#) and the IETF Trust's Legal Provisions Relating to IETF Documents (<https://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Revised BSD License text as described in Section 4.e of the [Trust Legal Provisions](#) and are provided without warranty as described in the Revised BSD License.

Table of Contents

1.	Introduction	2
2.	Computing needs	3
3.	Usage of ALTO for determining where to deploy a function or application	4
3.1.	Integrating compute information in ALTO	5
3.2.	Association of compute capabilities to network topology	5
3.3.	ALTO architecture for determining serve edge	6
4.	ALTO Design considerations for determining service edge	7
4.1.	Example of entity definition in different domains	7
4.2.	Definition of flavors in ALTO property map	9
5.	IANA Considerations	11
6.	Security Considerations	12
7.	Conclusions	12
8.	References	12
8.1.	Normative References	12
8.2.	Informative References	12
	Authors' Addresses	13

[1.](#) Introduction

The advent of virtualization is enabling the operators with a dynamic instantiation of network functions and applications by using different techniques on top of commoditized computation infrastructures, permitting a flexible and on-demand deployment of services, aligned with the actual needs observed as demanded by the customers.

Operators are starting to deploy distributed computing environments in different parts of the network with the objective of addressing the different service needs in terms of latency, bandwidth, processing capabilities, etc. This is translated in the emergence of a number of data centers of different sizes (e.g., large, medium, small) characterized by distinct dimension of CPUs, memory and storage capabilities, as well as bandwidth capacity for forwarding the traffic generated in and out the corresponding data center.

The probable future situation, with the generalization and proliferation of the edge computing approach, will increase the potential footprint where a function or application can be deployed. These different dimensioning rules result in a different unitary cost per CPU, memory, and storage in each computing environment because of the scale.

All the available distributed computing capabilities can complicate the decision of what infrastructure use for instantiating a given function or application. Such a decision influences not only the resources that are consumed in a given computing environment, but also the network capacity of the path that connects such environment with the rest of the network from traffic source to destination.

It is then essential for a network operator to have mechanisms assisting on the decision by considering a number of constraints related to the function or application to be deployed understanding how a given decision on the computing environment for the service edge affects to the transport network substrate. This would allow to integrate network capabilities in the function placement decision and further optimize performance of the deployed application.

This document proposes the usage of ALTO [[RFC7285](#)] for assisting with such a decision.

2. Computing needs

A given network function or application typically shows certain requirements in terms of processing capabilities (i.e., CPU), as well as volatile memory (i.e., RAM) and storage capacity.

Cloud computing providers, such as Amazon Web Services or Microsoft Azure, typically structure their offerings of computing capabilities by bundling CPU, RAM and storage units as quotas, instances or flavors that can be consumed in an ephemeral or temporal fashion, during the actual lifetime of the required function or application.

This same approach is being taken nowadays for characterizing bundles of resources on the so-called Network Function Virtualization Infrastructure (NFVI) Points of Presence (PoPs) being deployed by the telco operators. For instance, the Common Network Function Virtualisation Infrastructure Telecom Taskforce (CNTT) [[CNTT](#)], [[GSMA](#)], jointly hosted by GSMA and the Linux Foundation, intended to harmonize the definition of above-mentioned computing capability instances or flavors for abstracting capabilities of the underlying NFVI facilitating a more efficient utilization of the infrastructure and simplifying the integration and certification of functions, where certification means the assessment of the expected behavior for a

given function according to the level of resources determined by a given flavor. Evolution of this initiative is Anuket [[Anuket](#)], working on different architectures for well known tools such as OpenStack and Kubernetes.

Taking CNTT as an example, the flavors or instances can be characterized according to:

- * Type of instance (T): the types of instances are characterized as B (Basic), or N (Network Intensive). The latter can come with extensions for network acceleration for offloading network intensive operations to hardware.
- * Interface Option (I): it refers to the associated bandwidth of the network interface.
- * Compute flavor (F): it refers to a given predefined combination of resources in terms of virtual CPU, RAM, disk, and bandwidth for the management interface.
- * Optional storage extension (S): allows to request additional storage capacity.
- * Optional hardware acceleration characteristics (A): to request specific acceleration capabilities for improving the performance of the infrastructure.

The naming convention of an instance is thus encoded as T.I.F.S.A.

3. Usage of ALTO for determining where to deploy a function or application

ALTO can assist the deployment of a service or application on a specific flavor or instance of the computing substrate by taking into consideration network cost metrics.

A generic and primary approach is to take into account metrics related to the computing environment, such as availability of resources, unitary cost of those resources, etc.

Nevertheless, the function or application to be deployed on top of a given flavor is interconnected outside the computing environment where it is deployed, also requiring to guarantee transport network requirements to ensure the application performance, such as bandwidth, latency, etc.

The objective then is to leverage on ALTO to provide information about the more convenient execution environments to deploy virtualized network functions or applications, allowing the operator to get a coordinated service edge and transport network recommendation.

3.1. Integrating compute information in ALTO

CNTT proposes the existence of a catalogue of compute infrastructure profiles collecting the computing capability instances available to be consumed. Such kind of catalogue could be communicated to ALTO or even incorporated to it.

ALTO server queries are required to support T.I.F.S.A encoding in order to retrieve proper maps from ALTO. Additionally, filtered queries for particular characteristics of a flavor could also be supported.

3.2. Association of compute capabilities to network topology

It is required to associate the location of the available instances with topological information to allow ALTO construct the overall map. The expectation is to manage the network and cloud capabilities by the same entity, handling both network and compute abstractions jointly, producing an integrated map. While this can be straightforward when an ISP own both the network and the cloud infrastructure, it could require multiple administrative domains to interwork for composing the integrated map. Details on potential scenarios will be provided in future versions of the document.

At this stage four potential solutions could be considered:

- * To leverage on (and possibly extend) [[I-D.ietf-teas-sf-aware-topo-model](#)] for disseminating topology information together with notion of function location (that would require to be adapted to the existence of available compute capabilities). A recent effort in this direction can be found in [[I-D.llc-teas-dc-aware-topo-model](#)].
- * To extend BGP-LS [[RFC7752](#)], which is already considered as mechanism for feeding topology information in ALTO, in order to also advertise computing capabilities as well.
- * To combine information from the infrastructure profiles catalogue with topological information by leveraging on the IP prefixes allocated to the gateway providing connectivity to the NFVI PoP.

- * To integrate with Cloud Infrastructure Managers that could expose cloud infrastructure capabilities as in [\[CNTT\]](#), [\[GSMA\]](#).

The viability of these options will be explored in future versions of this document.

3.3. ALTO architecture for determining service edge

The following logical architecture defines the usage of ALTO for determining service edges.

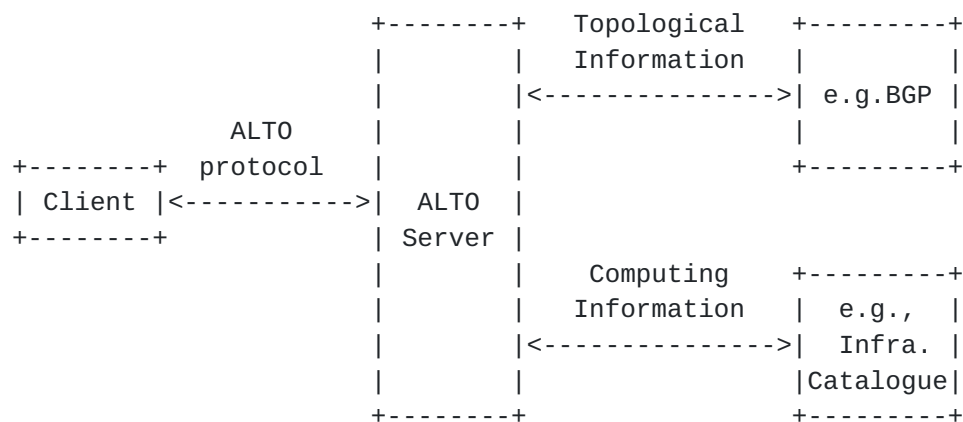


Figure 1: Service Edge Information Exchange

In order to select the optimal edge server from both the network perspective (e.g., the one showing better path cost) and cloud capabilities (e.g. in terms of processing capabilities, available RAM, storage capacity, etc), it is needed to see an edge server as both an IP entity (as in [\[RFC7285\]](#)) and an ANE entity (as in [\[RFC9275\]](#)).

Currently there is no way (neither in [\[RFC9275\]](#) nor [\[RFC9240\]](#)) to see the same edge server as an entity in both domains. The design of ALTO however is flexible enough to allow extensions to indicate that an entity can be defined in several domains. These different domains and their related properties can be specified in extended ALTO property maps, as proposed in the next sections.

4. ALTO Design considerations for determining service edge

This section is in progress and gathers the ALTO features that are needed to support the exposure of both networking capabilities and compute capabilities in ALTO Maps.

In particular, ALTO Entity Property Maps defined in [[RFC9240](#)] can be extended. [[RFC9240](#)] generalizes the concept of endpoint properties to domains of other entities through property maps. Entities can be defined in a wider set of entity domains such as IPv4, IPv6, PID, AS, ISO3166-1 country codes or ANE. [RFC 9240](#) in addition specifies how properties can be defined on entities of each of these domains.

4.1. Example of entity definition in different domains

First, as many applications do not necessarily need both compute and networking information, it is fine to keep separate entity domains with their related relevant properties. However, some applications need information on both topics and a scalable and flexible solution consists in defining an ALTO property type, that:

- * indicates that an entity can be defined in several domains,
- * specifies, for an entity, the other domains where this entity is defined.

The following is an example where property "entity domain mapping" lists the other domains in which an entity is defined. This property type should be usable in all entity domains types.

One possible way, for instance, can be to introduce entity properties that list other entity domains where an edge server is identified. This property type should be usable in all entity domains types.

Suppose an edge server is identified:

- * in the IPV4 domain, with an IP address, e.g. ipv4:1.2.3.4
- * in the ANE domain, with an identifier, e.g. ane:DC10-HOST1

To get information on this edge server as an entity in the "ipv4" entity domain, an ALTO client queries the properties "pid" and "entity-domain-mappings" and gets a response as follows.


```
POST /propmap/lookup/dc-ip HTTP/1.1
Host: alto.example.com
Accept: application/alto-propmap+json,application/alto-error+json
Content-Length: TBC
Content-Type: application/alto-propmapparams+json
{
  "entities" : ["ipv4:1.2.3.4"],
  "properties" : [ "pid", "entity-domain-mappings"]
}
```

```
HTTP/1.1 200 OK
Content-Length: TBC
Content-Type: application/alto-propmap+json
{
  "meta" : {},
  "property-map": {
    "ipv4:1.2.3.4" :
      { "pid" : DC10,
        "entity-domain-mappings" : ["ane"]}
  }
}
```

To get information on this edge server as an entity in the "ane" entity domain, an ALTO client queries the properties "entity-domain-mappings" and "network-address" and gets a response as follows.


```
POST /propmap/lookup/dc-ane HTTP/1.1
Host: alto.example.com
Accept: application/alto-propmap+json,application/alto-error+json
Content-Length: TBC
Content-Type: application/alto-propmapparams+json
{
  "entities" : ["ane:DC10-HOST1"],
  "properties" : [ "entity-domain-mappings", "network-address"]
}

HTTP/1.1 200 OK
Content-Length: TBC
Content-Type: application/alto-propmap+json
{
  "meta" : {},

  "property-map": {
    "ane:DC10-HOST1"
      {"entity domain mappings" : ["ipv4"]",
       "network-address" : ipv4:1.2.3.4}
    }
  }
```

Thus, if the ALTO Client sees the edge server as an entity with a network address, it knows that it can see the server as an ANE on which it can query relevant properties.

Further elaboration will be provided in next versions of this document.

4.2. Definition of flavors in ALTO property map

The ALTO Entity Property Maps [\[RFC9240\]](#) generalize the concept of endpoint properties to domains of other entities through property maps. The term "flavor" or "instance" refers to an abstracted set of computing resources, with well specified properties on their capabilities. Capabilities are typically CPU, RAM, Storage. So a flavor can be seen as an ANE, with properties declined in terms for example of TIFSA. A flavor or instance is a group of 1 or more elements that can be reached via one or more network addresses. So an instance can be also be seen as a PID that groups 1 or more IP addresses. In a context such as the one defined in CNTT, an ALTO property map could be used to expose T.I.F.S.A information of potential candidate flavors, i.e., potential NFVI PoPs where an application or service can be deployed.

Figure 2 below shows an illustrative example of an ALTO property map with property values grouped by flavor name.

flavor -name	type (T)	inter face (I)	f-c pu (F)	f-ra m (F)	f-di sk (F)	f-b w (F)	S	A
small- 1	basic	1 Gbps	1	512 MB	1 GB	1 G bps		
small- 2	network- intensive	9 Gbps	1	512 MB	1 GB	1 G bps		
medium- 1	network- intensive	25 Gbps	2	4 GB	40 GB	1 G bps		
large- 1	compute- intensive	50 Gbps	4	8 GB	80 GB	1 G bps		
large- 2	compute- intensive	100 Gbps	8	16 GB	160 GB	1 G bps		

Figure 2: ALTO Property Map

The following example uses the filtered property map resource to request properties "type", "cpu", "ram", and "disk" on 5 ANE flavors named "small-1", "small-2", "medium-1", "large-1", "large-2" defined in the example before.


```
POST /propmap/lookup/ane-flavor-name HTTP/1.1
Host: alto.example.com
Accept: application/alto-propmap+json,application/alto-error+json
Content-Length: 155
Content-Type: application/alto-propmapparams+json
{
  "entities" : ["small-1",
                "small-2",
                "medium-1",
                "large-1"],
                "large-2"],
  "properties" : [ "type", "cpu", "ram", "disk"]
}

HTTP/1.1 200 OK
Content-Length: 295
Content-Type: application/alto-propmap+json
{
  "meta" : {
  },
  "property-map": {
    "small-1":
      {"type" : "basic", "cpu" : 1,
        "ram" : "512MB", "disk" : 1GB},
    "small-2":
      {"type" : "network-intensive", "cpu" : 1,
        "ram" : "512MB", "disk" : 1GB},
    "medium-1":
      {"type" : "compute-intensive", "cpu" : 2,
        "ram" : "4GB", "disk" : 40GB},
    "large-1":
      {"type" : "compute-intensive", "cpu" : 4,
        "ram" : "8GB", "disk" : 80GB},
    "large-2":
      {"type" : "compute-intensive", "cpu" : 8,
        "ram" : "16GB", "disk" : 160GB},
  }
}
```

Figure 3: Filtered Property Map query example

5. IANA Considerations

This document includes no request to IANA.

6. Security Considerations

TBD.

7. Conclusions

Telco networks will increasingly contain a number of interconnected data centers, of different size and characteristics, allowing flexibility in the dynamic deployment of functions and applications for advance services. The overall objective of this document is to begin a discussion in the ALTO WG regarding the suitability of the ALTO protocol for determining where to deploy a function or application in distributed computing environments. The result of such discussions will be reflected in future versions of this draft.

8. References

8.1. Normative References

- [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>>.
- [RFC7752] Gredler, H., Ed., Medved, J., Previdi, S., Farrel, A., and S. Ray, "North-Bound Distribution of Link-State and Traffic Engineering (TE) Information Using BGP", [RFC 7752](#), DOI 10.17487/RFC7752, March 2016, <<https://www.rfc-editor.org/info/rfc7752>>.
- [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>>.
- [RFC9240] Roome, W., Randriamasy, S., Yang, Y., Zhang, J., and K. Gao, "An Extension for Application-Layer Traffic Optimization (ALTO): Entity Property Maps", [RFC 9240](#), DOI 10.17487/RFC9240, July 2022, <<https://www.rfc-editor.org/info/rfc9240>>.
- [RFC9275] Gao, K., Lee, Y., Randriamasy, S., Yang, Y., and J. Zhang, "An Extension for Application-Layer Traffic Optimization (ALTO): Path Vector", [RFC 9275](#), DOI 10.17487/RFC9275, September 2022, <<https://www.rfc-editor.org/info/rfc9275>>.

8.2. Informative References

- [Anuket] "Anuket Project", <<https://wiki.anuket.io/>>.
- [CNTT] "Cloud Infrastructure Telco Taskforce Reference Model", <<https://github.com/cntt-n/CNTT/wiki>>.
- [GSMA] "Cloud Infrastructure Reference Model, Version 2.0", October 2021, <<https://www.gsma.com/newsroom/wp-content/uploads//NG.126-v2.0-1.pdf>>.
- [I-D.ietf-teas-sf-aware-topo-model]
Bryskin, I., Liu, X., Lee, Y., Guichard, J., Contreras, L., Ceccarelli, D., Tantsura, J., and D. Shytyi, "SF Aware TE Topology YANG Model", Work in Progress, Internet-Draft, [draft-ietf-teas-sf-aware-topo-model-09](#), 27 February 2022, <<http://www.ietf.org/internet-drafts/draft-ietf-teas-sf-aware-topo-model-09.txt>>.
- [I-D.llc-teas-dc-aware-topo-model]
Lee, Y., Liu, X., and L. Contreras, "DC aware TE topology model", Work in Progress, Internet-Draft, [draft-llc-teas-dc-aware-topo-model-01](#), 12 July 2021, <<https://www.ietf.org/archive/id/draft-llc-teas-dc-aware-topo-model-01.txt>>.

Authors' Addresses

Luis M. Contreras
Telefonica
Ronda de la Comunicacion, s/n
28050 Madrid
Spain
Email: luismiguel.contrerasmurillo@telefonica.com
URI: <http://lmcontreras.com/>

Danny Alex Lachos Perez
BENOCS GmbH
Reuchlinstraße 10
10553 Berlin
Germany
Email: dlachos@benocs.com

Christian Esteve Rothenberg
University of Campinas
Av. Albert Einstein 400
Campinas-Sao Paulo
13083-970
Brazil
Email: chesteve@dca.fee.unicamp.br
URI: <https://intrig.dca.fee.unicamp.br/christian/>

Sabine Randriamasy
Nokia Bell Labs
Route de Villejust
91460 Nozay
France
Email: sabine.randriamasy@nokia-bell-labs.com

