

ALTO  
Internet-Draft  
Intended status: Informational  
Expires: September 3, 2012

M. Stiernerling, Ed.  
NEC Europe Ltd.  
S. Kiesel, Ed.  
University of Stuttgart  
S. Previdi  
Cisco Systems, Inc.  
March 2, 2012

**ALTO Deployment Considerations**  
**draft-ietf-alto-deployments-04**

Abstract

Many Internet applications are used to access resources, such as pieces of information or server processes, which are available in several equivalent replicas on different hosts. This includes, but is not limited to, peer-to-peer file sharing applications. The goal of Application-Layer Traffic Optimization (ALTO) is to provide guidance to these applications, which have to select one or several hosts from a set of candidates, that are able to provide a desired resource. The protocol is under specification in the ALTO working group. This memo discusses deployment related issues of ALTO for peer-to-peer and CDNs, some preliminary security considerations, and also initial guidance for application designers using ALTO.

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 September 3, 2012.

Copyright Notice

Copyright (c) 2012 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

<a href="#">1.</a>	<a href="#">Introduction . . . . .</a>	<a href="#">4</a>
<a href="#">2.</a>	<a href="#">General Considerations . . . . .</a>	<a href="#">5</a>
<a href="#">2.1.</a>	<a href="#">General Placement of ALTO . . . . .</a>	<a href="#">5</a>
<a href="#">2.2.</a>	<a href="#">Relationship between ALTO and Applications . . . . .</a>	<a href="#">7</a>
<a href="#">2.3.</a>	<a href="#">Provided Guidance . . . . .</a>	<a href="#">7</a>
<a href="#">2.3.1.</a>	<a href="#">Keeping Traffic Local in Network . . . . .</a>	<a href="#">8</a>
<a href="#">2.3.2.</a>	<a href="#">Off-Loading Traffic from Network . . . . .</a>	<a href="#">8</a>
<a href="#">2.3.3.</a>	<a href="#">Intra-Network Localization/Bottleneck Off-Loading . . . . .</a>	<a href="#">9</a>
<a href="#">2.4.</a>	<a href="#">Provisioning ALTO Maps . . . . .</a>	<a href="#">11</a>
<a href="#">3.</a>	<a href="#">Deployment Considerations by ISPs . . . . .</a>	<a href="#">12</a>
<a href="#">3.1.</a>	<a href="#">Requirement for Traffic Optimization by ISPs . . . . .</a>	<a href="#">12</a>
<a href="#">3.2.</a>	<a href="#">Considerations for ISPs . . . . .</a>	<a href="#">13</a>
<a href="#">3.2.1.</a>	<a href="#">Very small ISPs with simple Network Structure . . . . .</a>	<a href="#">13</a>
<a href="#">3.2.2.</a>	<a href="#">Large ISPs with layered fixed Network Structure . . . . .</a>	<a href="#">13</a>
<a href="#">3.2.3.</a>	<a href="#">ISPs with Mobile Network . . . . .</a>	<a href="#">15</a>
<a href="#">4.</a>	<a href="#">Using ALTO for P2P . . . . .</a>	<a href="#">17</a>
<a href="#">4.1.</a>	<a href="#">Using ALTO for Tracker-based Peer-to-Peer Applications . . . . .</a>	<a href="#">19</a>
<a href="#">4.2.</a>	<a href="#">Expectations of ALTO . . . . .</a>	<a href="#">24</a>
<a href="#">5.</a>	<a href="#">Using ALTO for CDNs . . . . .</a>	<a href="#">25</a>
<a href="#">5.1.</a>	<a href="#">Request Routing using the Endpoint Cost Service . . . . .</a>	<a href="#">25</a>
<a href="#">5.1.1.</a>	<a href="#">ALTO Topology Vs. Network Topology . . . . .</a>	<a href="#">26</a>
<a href="#">5.1.2.</a>	<a href="#">Topology Computation and ECS Delivery . . . . .</a>	<a href="#">26</a>
<a href="#">5.1.3.</a>	<a href="#">Ranking Service . . . . .</a>	<a href="#">26</a>
<a href="#">5.1.4.</a>	<a href="#">Ranking and Network Events . . . . .</a>	<a href="#">27</a>
<a href="#">5.1.5.</a>	<a href="#">Caching and Lifetime . . . . .</a>	<a href="#">27</a>
<a href="#">5.1.6.</a>	<a href="#">Redirection . . . . .</a>	<a href="#">28</a>
<a href="#">5.1.7.</a>	<a href="#">Groups and Costs . . . . .</a>	<a href="#">28</a>
<a href="#">6.</a>	<a href="#">Advanced Features . . . . .</a>	<a href="#">29</a>
<a href="#">6.1.</a>	<a href="#">Cascading ALTO Servers . . . . .</a>	<a href="#">29</a>
<a href="#">6.2.</a>	<a href="#">ALTO for IPv4 and IPv6 . . . . .</a>	<a href="#">30</a>
<a href="#">6.3.</a>	<a href="#">Monitoring ALTO . . . . .</a>	<a href="#">30</a>
<a href="#">6.3.1.</a>	<a href="#">Monitoring Metrics Definition . . . . .</a>	<a href="#">30</a>
<a href="#">6.3.2.</a>	<a href="#">Monitoring Data Sources . . . . .</a>	<a href="#">31</a>
<a href="#">6.3.3.</a>	<a href="#">Monitoring Structure . . . . .</a>	<a href="#">31</a>
<a href="#">7.</a>	<a href="#">Known Limitations of ALTO . . . . .</a>	<a href="#">33</a>



<a href="#">7.1.</a>	<a href="#">Limitations of Map-based Approaches . . . . .</a>	<a href="#">33</a>
<a href="#">7.2.</a>	<a href="#">Limitations of Non-Map-based Approaches . . . . .</a>	<a href="#">34</a>
<a href="#">7.3.</a>	<a href="#">General Challenges . . . . .</a>	<a href="#">34</a>
<a href="#">8.</a>	<a href="#">Extensions to the ALTO Protocol . . . . .</a>	<a href="#">36</a>
<a href="#">8.1.</a>	<a href="#">Host Group Descriptors . . . . .</a>	<a href="#">36</a>
<a href="#">8.2.</a>	<a href="#">Rating Criteria . . . . .</a>	<a href="#">36</a>
<a href="#">8.2.1.</a>	<a href="#">Distance-related Rating Criteria . . . . .</a>	<a href="#">36</a>
<a href="#">8.2.2.</a>	<a href="#">Charging-related Rating Criteria . . . . .</a>	<a href="#">37</a>
<a href="#">8.2.3.</a>	<a href="#">Performance-related Rating Criteria . . . . .</a>	<a href="#">37</a>
<a href="#">8.2.4.</a>	<a href="#">Inappropriate Rating Criteria . . . . .</a>	<a href="#">38</a>
<a href="#">9.</a>	<a href="#">API between ALTO Client and Application . . . . .</a>	<a href="#">39</a>
<a href="#">10.</a>	<a href="#">Security Considerations . . . . .</a>	<a href="#">40</a>
<a href="#">10.1.</a>	<a href="#">Information Leakage from the ALTO Server . . . . .</a>	<a href="#">40</a>
<a href="#">10.2.</a>	<a href="#">ALTO Server Access . . . . .</a>	<a href="#">40</a>
<a href="#">10.3.</a>	<a href="#">Faking ALTO Guidance . . . . .</a>	<a href="#">41</a>
<a href="#">11.</a>	<a href="#">Conclusion . . . . .</a>	<a href="#">42</a>
<a href="#">12.</a>	<a href="#">References . . . . .</a>	<a href="#">43</a>
<a href="#">12.1.</a>	<a href="#">Normative References . . . . .</a>	<a href="#">43</a>
<a href="#">12.2.</a>	<a href="#">Informative References . . . . .</a>	<a href="#">43</a>
<a href="#">Appendix A.</a>	<a href="#">Contributors List and Acknowledgments . . . . .</a>	<a href="#">45</a>
	<a href="#">Authors' Addresses . . . . .</a>	<a href="#">46</a>



## **1. Introduction**

Many Internet applications are used to access resources, such as pieces of information or server processes, which are available in several equivalent replicas on different hosts. This includes, but is not limited to, peer-to-peer file sharing applications and Content Delivery Networks (CDNs). The goal of Application-Layer Traffic Optimization (ALTO) is to provide guidance to applications, which have to select one or several hosts from a set of candidates, that are able to provide a desired resource. The basic ideas of ALTO are described in the problem space of ALTO is described in [[RFC5693](#)] and the set of requirements is discussed in [[I-D.ietf-alto-reqs](#)].

However, there are no considerations about what operational issues are to be expected once ALTO will be deployed. This includes, but is not limited to, location of the ALTO server, imposed load to the ALTO server, or from whom the queries are performed.

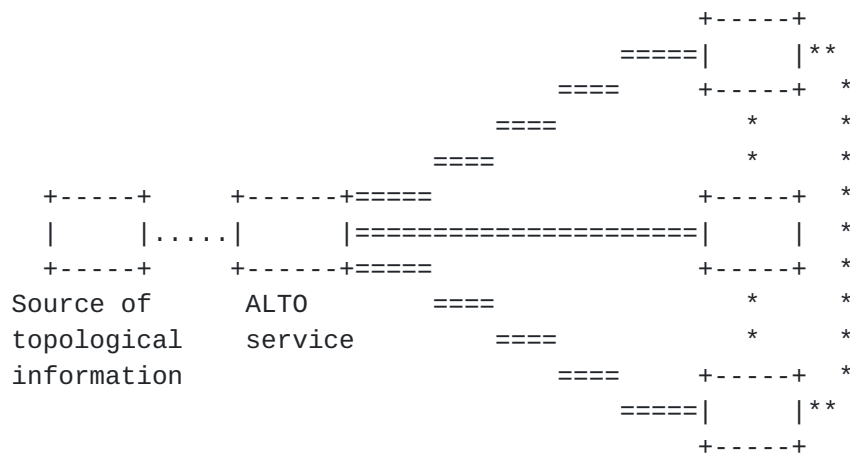
Comments and discussions about this memo should be directed to the ALTO working group: [alto@ietf.org](mailto:alto@ietf.org).



The ALTO server and ALTO clients can be situated at various entities in a network deployment. The first differentiation is whether the ALTO client is located on the actual host that runs the application, as shown in Figure 2, (e.g., peer-to-peer filesharing application) or if the ALTO client is located on resource directory, as shown in Figure 3 (e.g., a tracker in peer-to-peer filesharing).







Legend:

### === ALT0 client protocol

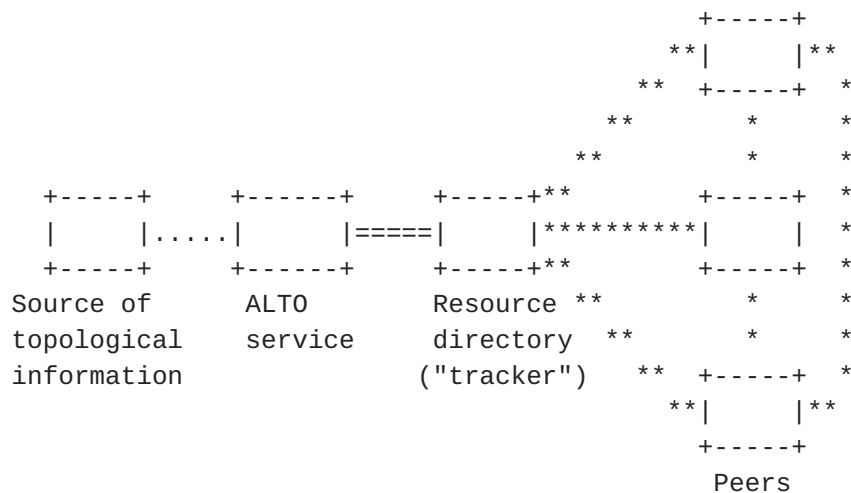
\*\*\* Application protocol

```
... Provisioning protocol
```

Figure 2: Overview of protocol interaction between ALTO elements, scenario without tracker

Figure 2 shows the operational model for applications that do not use a tracker, such as, edonky, or in if the tracker should be the querying party. This use case also holds true for CDNs. The ALTO server can also be queried by CDNs to get a guidance about where the a particular client accessing data in the CDN is exactly located in the ISP's network.





Legend:

=== ALTO client protocol

\*\*\* Application protocol

... Provisioning protocol

Figure 3: Overview of protocol interaction between ALTO elements, scenario with tracker

However, Figure 3 does not denote where the ALTO elements are actually located, i.e., if the tracker and the ALTO server are in the same ISP's domain, or if the tracker and the ALTO server are managed/owned/located in different domains. The latter is the typical use case, e.g., taking Pirate Bay as example that serves Bittorrent peers world-wide.

## 2.2. Relationship between ALTO and Applications

ALTO is intended to be used by a wide-range of applications. However, any application using ALTO must also work if no ALTO servers can be found or if no responses to ALTO queries are received, e.g., due to connectivity problems or overload situation (see also [\[I-D.ietf-alto-reqs\]](#)). (Editor's note: better text needed here!)

## 2.3. Provided Guidance

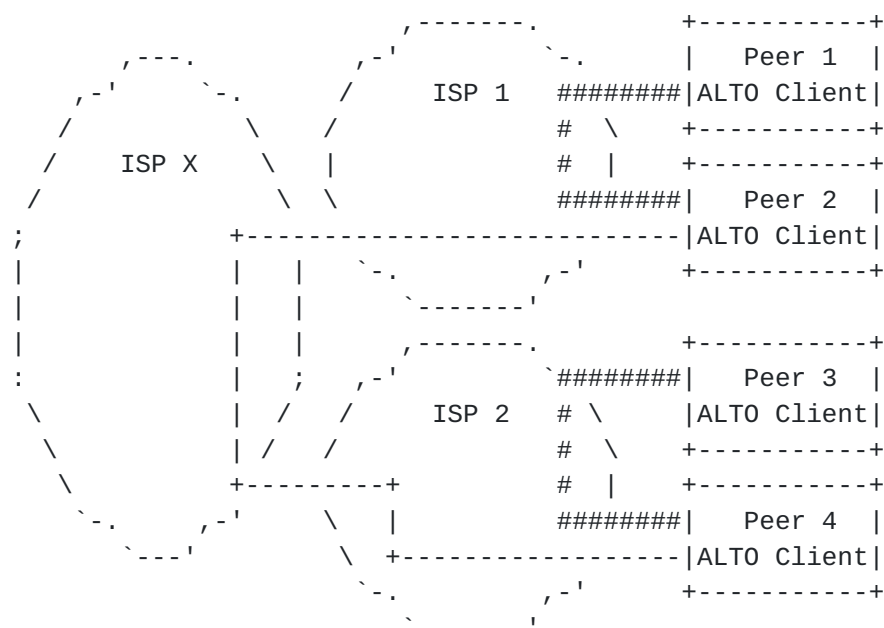
ALTO gives guidance to applications on what IP addresses or IP prefixes, and such which hosts are to be preferred according to the operator of the ALTO server. The general assumption of the ALTO WG is that a network operator would always express to prefer hosts in its own network while hosts located outside its own network are to be avoided (are undesired to be considered by the applications). This might be applicable in some cases but may not be applicable in the general case. The ALTO protocol gives only the means to let the ALTO server operator to express its preference, whatever this preference



is. This section explores this space.

### 2.3.1. Keeping Traffic Local in Network

ALTO guidance can be used to let applications prefer other peers within the same network operator's network instead of randomly connecting to other peers which are located in another operator's network. Figure 4 shows such a scenario where peers prefer peers in the same network (e.g., Peer 1 and Peer 2 in ISP1 and Peer 3 and Peer 4 in ISP2).



Legend:

### preferred "connections"

--- non-preferred "connections"

Figure 4: ALTO Traffic Network Localization

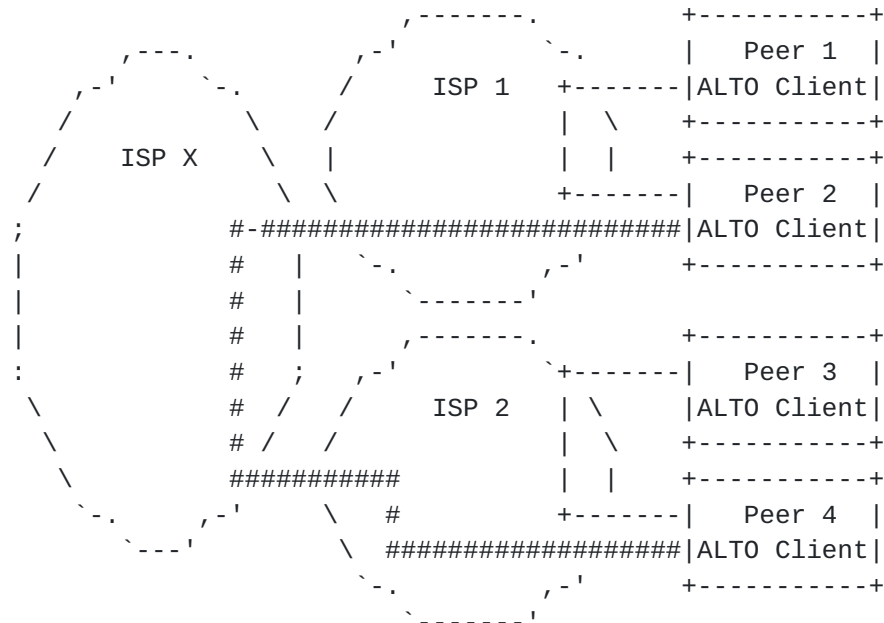
TBD: Describes limits of this approach (e.g., traffic localization guidance is of less use if the peers cannot upload); describe how maps would look like.

### 2.3.2. Off-Loading Traffic from Network

Another scenario where the use of ALTO can be beneficial is in mobile broadband networks, e.g., CDMA200 or UMTS, but where the network operator may have the desire to guide peers in its own network to use peers in remote networks. One reason can be that the wireless network is not made for the load cause by, e.g., peer-to-peer



applications, and the operator has the need that peers fetch their data from remote peers in other parts of the Internet.



Legend:

=== preferred "connections"

--- non-preferred "connections"

Figure 5: ALTO Traffic Network De-Localization

Figure 5 shows the result of such a guidance process where Peer 2 prefers a connection with Peer4 instead of Peer 1, as shown in Figure 4.

TBD: Limits of this approach in general and with respect to p2p. describe how maps would look like.

### 2.3.3. Intra-Network Localization/Bottleneck Off-Loading

The above sections described the results of the ALTO guidance on an inter-network level. However, ALTO can also be used to guide peers on which internal peers are to be preferred. For instance, to guide Peers on a remote network side to prefer to connect to each other, instead of crossing a bottleneck link, a backhaul link to connect the side to the network core. Figure 6 shows such a scenario where Peer 1 and Peer 2 are located in Net 2 of ISP1 and connect via a low capacity link to the core (Net 1) of the same ISP1. Peer1 and Peer 2 would both exchange their data with remote peers, probably clogging the bottleneck link.





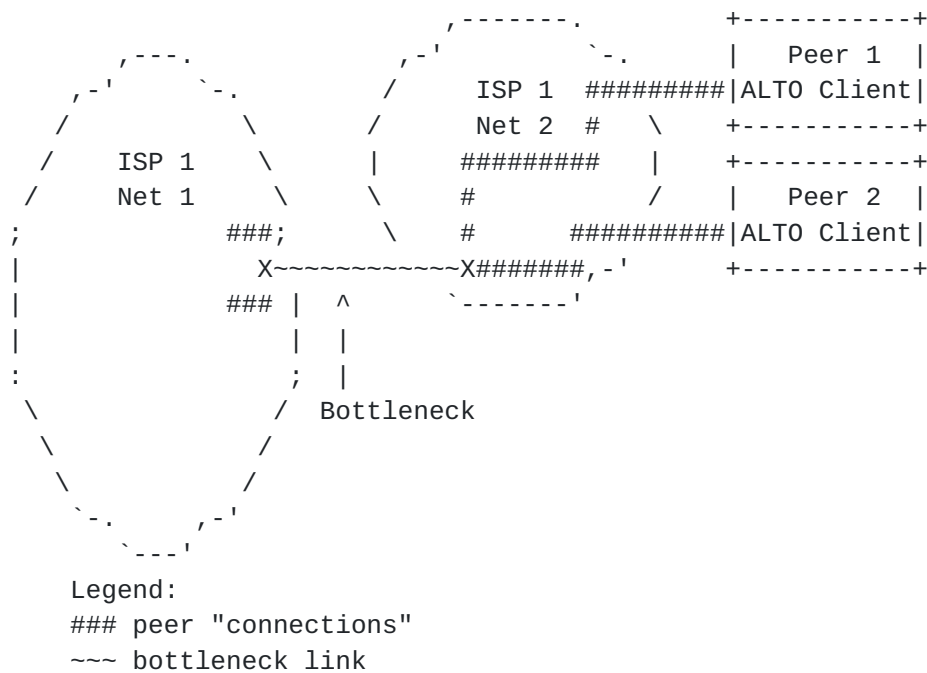


Figure 6: Without Intra-Network ALTO Traffic Localization

The operator can guide the peers in such a situation to try first local peers in the same network islands, avoiding or at least lowering the effect on the bottleneck link, as shown in Figure 7.

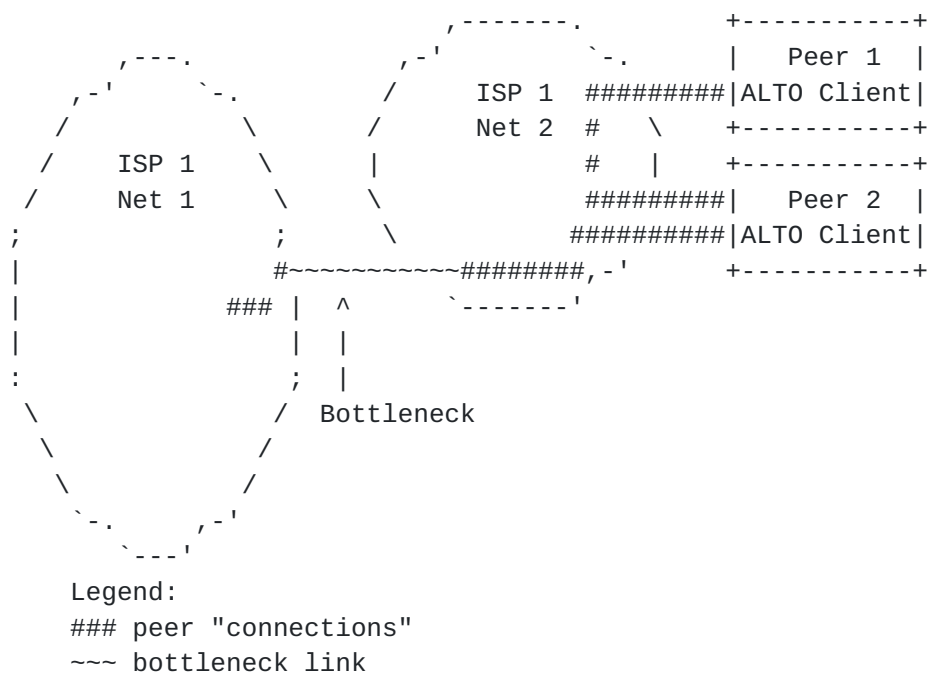


Figure 7: With Intra-Network ALTO Traffic Localization



TBD: describe how maps would look like.

#### **2.4. Provisiong ALTO Maps**

This section will describe how ALTO maps in the protocol can be populated before using them.

### **3. Deployment Considerations by ISPs**

The Internet is a large network constituted of multiple networks worldwide. Numerous of these networks are built by telecom operators or network operators (named ISP in this memo), and these networks provide network connectivity, such as cable networks, 3G and so on. As well as some of networks are built by universities or big organizations themselves, and these networks are used to provide connectivity for research and work. The essence of Internet is its connectivity and sharing capability. However, ISPs emphasize network's manageability and controllability, because ISPs provide public network access service for most person and families, they need to manage, to control and to audit the traffic. Thus, it's important for ISPs to understand the requirement of optimizing traffic, and how to deploy ALTO service in these manageability and controllability networks.

#### **3.1. Requirement for Traffic Optimization by ISPs**

All networks of ISPs are connected to each other through peering points. From view of business mode, the inter-network settlement is needed in traffic exchanging between these ISP's networks. The current settlement can be costly. So to save these cost, the simple and basic method is to decrease the traffic exchange across the peering points and keep the traffic in own network area.

For some large ISPs, their whole network is layered. The upper layer network includes one or several backbone networks, and the lower layer network includes multiple access networks. These access networks are connected to backbone networks, and the exchange traffic with others through backbone network. In this kind of layered network, the bandwidth of backbone network is important and may be scarce. Traffic should be limited to the access networks, so to decrease the usage of backbone as far as possible.

Compared to fixed networks, mobile networks have some special characters, including small link bandwidth, high cost, limited radio frequency resource, and terminal battery. In mobile network, the usage of wireless link should be decreased as far as possible and be high-efficient. For example, in the case of a P2P service, the clients in the fixed network should decrease the data transport from the clients in the mobile networks, as well as the clients in the mobile networks should prefer the data transmission from the clients in the fixed networks.









ALTO service deploying to this kind of large ISP, first the optimization area can be defined according to real network condition. For example, each access network can be defined to be one optimization area. Then cost can be defined according to the optimizing requirement by ISPs. There is one example described below and also shown in Figure 9.

In this example, ISP A has one backbone network and three access networks, named as AN A, AN B, and AN C. A P2P application is used in this example. For the traffic optimization, the first requirement is to decrease the P2P traffic of backbone network in inner ISP A; and the second requirement is to decrease the P2P traffic to outer ISPs. Always, the second requirement is prior to the first one. Also, we assume that the settlement rate with ISP B is lower than with other ISPs. Then ISP A can deploy ALTO service to meet the need of traffic optimization. We will give the detail example of ALTO service definition and configuration according to requirements above.

In inner network of ISP A, we can define each access network to be one optimization area, and assign one PID to every access network, such as PID1, PID2, and PID 3. Because of different settlement with different outer ISPs, we define ISP B to be one optimization area, and assign PID 4 to it, as well as define all other networks to be one optimization area and PID 5.

We assign cost names (C1, C2, C3, C4, C5, C6, C7) as the figure below. C1 is denoted as the link cost in inner AN A, the same as C2 and C3. C4 is denoted as the link cost from PID 1 to PID 2, the same as C5. C6 is denoted as the link cost from the ISP B to ISP A. C7 is denoted as the link cost from other networks to ISP A.

According to discussion of the first requirement and the second requirement above, the relationship of these costs will be defined as:  $(C1, C2, C3) < (C4, C5) < (C6) < (C7)$

This is one very simple example above, in which we do not consider the different link type of access network. In deploying ALTO service in real network, we must consider more real network conditions and requirements. One real example is described in greater detail in [\[I-D.lee-alto-chinatelecom-trial\]](#).



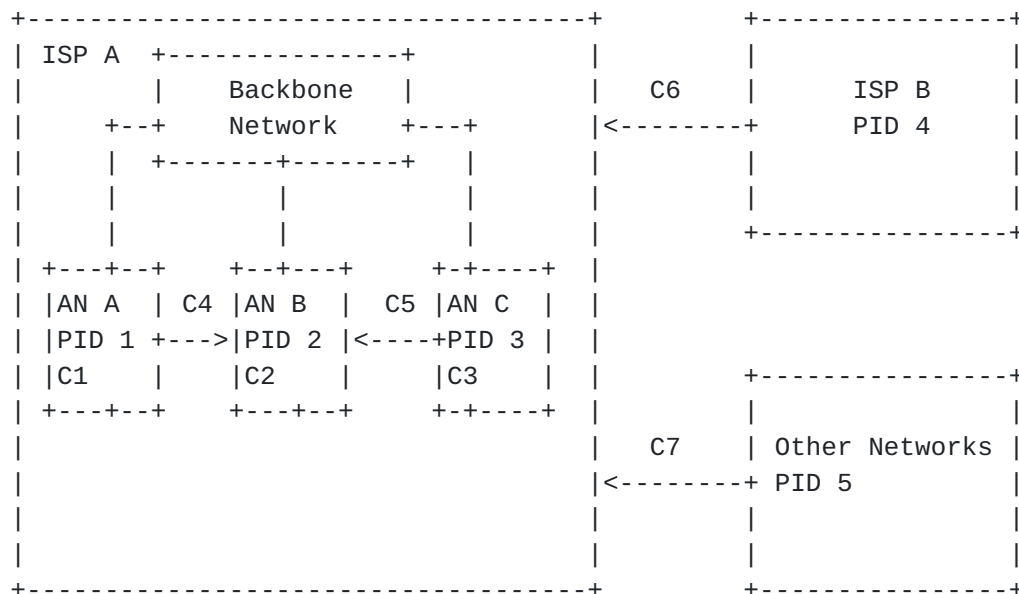


Figure 9: ALTO deployment in large ISPs with layered fixed network structures

### 3.2.3. ISPs with Mobile Network

For ISPs with mobile network and fixed network, the traffic optimizing problems they focus will be optimizing the mobile traffic, except problems on last hop section. Wireless radio frequency resource is scarce and costly in mobile network. The requirement of traffic optimization in mobile network is mainly decreasing the usage of radio resource. The ALTO service can be deployed to meet these needs.

For example in one ISP A as below in Figure 10, there is one mobile network is connected to backbone network. In this kind of network structure, mobile network can be defined as one optimization area, and assigned PID 1. We also define other PID and cost as figure below.

To decrease the usage of wireless link, the relationship of these costs will be defined to:

From view of mobile network: ( $C4 < C1$ ). This means that, the clients in mobile network requiring data resource from clients of the other access networks is prior to clients of mobile network. This policy can decrease the usage of wireless link and power consumption in terminal.

From view of AN A: ( $C2 < C6$ ,  $C5 = \text{maximum cost}$ ). This means that, to



other optimization area, requiring data from mobile network should be avoided.

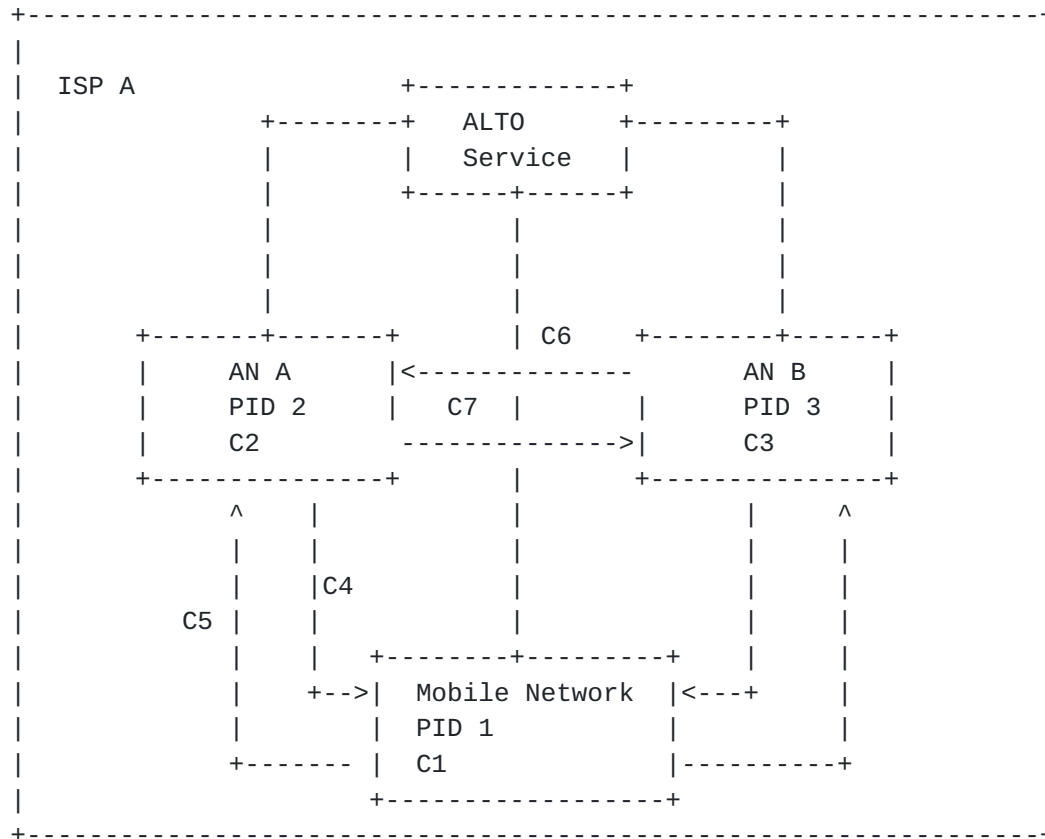
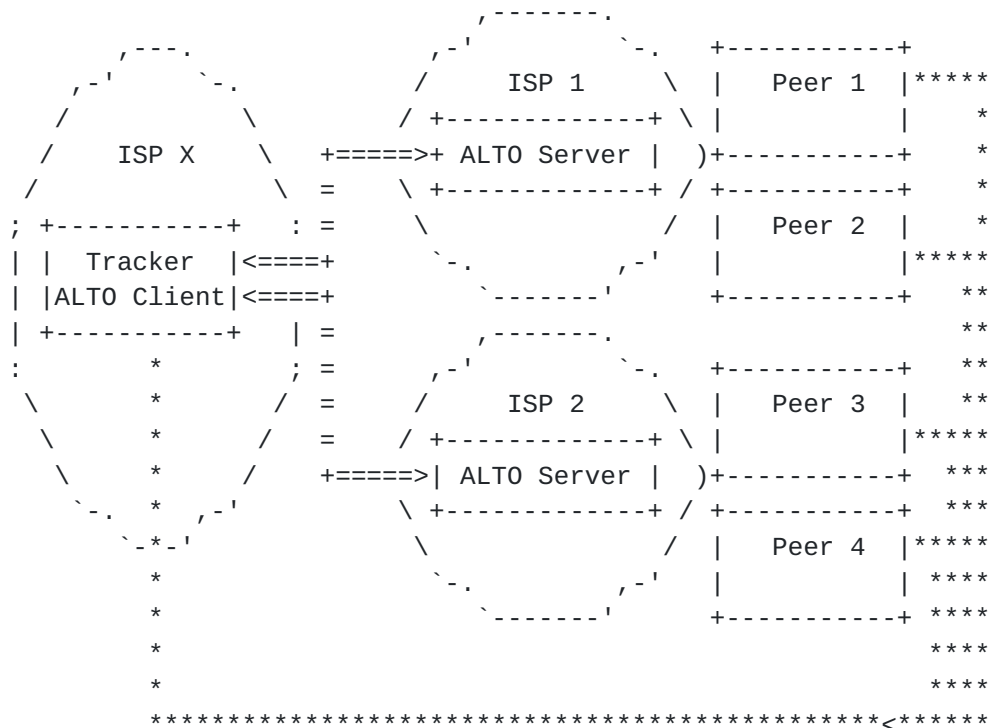


Figure 10: ALTO deployment in ISPs with mobile network



#### 4. Using ALTO for P2P



Legend:

=== ALTO client protocol

\*\*\* Application protocol

Figure 11: Global tracker accessing ALTO server at various ISPs

Figure 11 depicts a tracker-based system, where the tracker embeds the ALTO client. The tracker itself is hosted and operated by an entity different than the ISP hosting and operating the ALTO server. Initially, the tracker has to look-up the ALTO server in charge for each peer where it receives a ALTO query for. Therefore, the ALTO server has to discover the handling ALTO server, as described in [\[I-D.ietf-alto-server-discovery\]](#). However, the peers do not have any way to query the server themselves. This setting allows to give the peers a better selection of candidate peers for their operation at an initial time, but does not consider peers learned through direct peer-to-peer knowledge exchange. This is called peer exchange (PEX) in bittorrent, for instance.





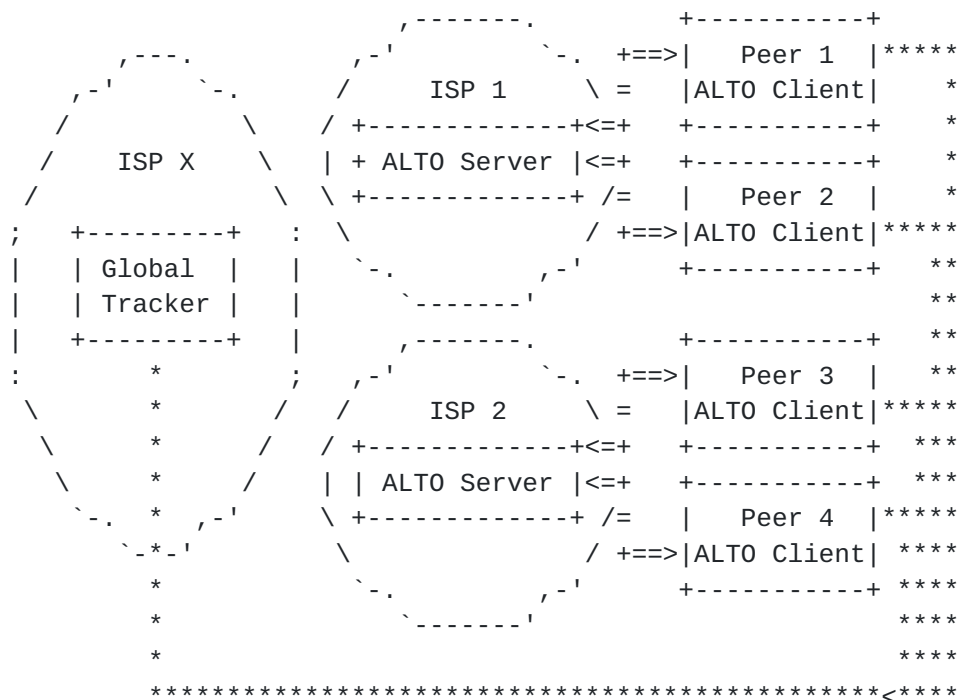


Figure 12: Global Tracker - Local ALTO Servers

The scenario in Figure 12 lets the peers directly communicate with their ISP's ALTO server (i.e., ALTO client embedded in the peers), giving thus the peers the most control on which information they query for, as they can integrate information received from trackers and through direct peer-to-peer knowledge exchange.



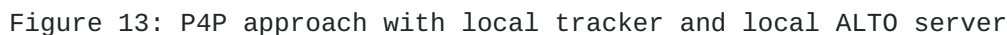


Figure 13 and Figure 11 ostensibly take peers the possibility to directly query the ALTO server, if the communication with the ALTO server is not permitted for any reason. However, considering the plethora of different applications of ALTO, e.g., multiple tracker and non-tracker based P2P systems and or applications searching for relays, it seems to be beneficial for all participants to let the peers directly query the ALTO server. The peers are also the single point having all operational knowledge to decide whether to use the ALTO guidance and how to use the ALTO guidance. This is a preference for the scenario depicted in Figure Figure 12.

The scope of this section is the interaction of peer-to-peer applications that use a centralized resource directory ("tracker"), with the ALTO service. In this scenario, the resource consumer



("peer") asks the resource directory for a list of candidate resource providers, which can provide the desired resource.

For efficiency reasons (i.e., message size), usually only a subset of all resource providers known to the resource directory will be returned to the resource consumer. Some or all of these resource providers, plus further resource providers learned by other means such as direct communication between peers, will be contacted by the resource consumer for accessing the resource. The purpose of ALTO is giving guidance on this peer selection, which is supposed to yield better-than-random results. The tracker response as well as the ALTO guidance are most beneficial in the initial phase after the resource consumer has decided to access a resource, as long as only few resource providers are known. Later, when the resource consumer has already exchanged some data with other peers and measured the transmission speed, the relative importance of ALTO may dwindle.

The ALTO protocol specification [[I-D.ietf-alto-protocol](#)] details how an ALTO client can query an ALTO server for guiding information and receive the corresponding replies. However, in the considered scenario of a tracker-based P2P application, there are two fundamentally different possibilities where to place the ALTO client:

1. ALTO client in the resource consumer ("peer")
2. ALTO client in the resource directory ("tracker")

In the following, both scenarios are compared in order to explain the need for third-party ALTO queries.

In the first scenario (see Figure 15), the resource consumer queries the resource directory for the desired resource (F1). The resource directory returns a list of potential resource providers without considering ALTO (F2). It is then the duty of the resource consumer to invoke ALTO (F3/F4), in order to solicit guidance regarding this list.

In the second scenario (see Figure 17), the resource directory has an embedded ALTO client, which we will refer to as RDAC in this document. After receiving a query for a given resource (F1) the resource directory invokes the RDAC to evaluate all resource providers it knows (F2/F3). Then it returns a, possibly shortened, list containing the "best" resource providers to the resource consumer (F4).





|

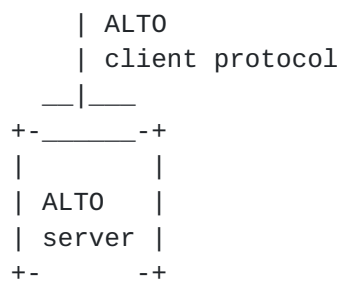


Figure 14: Tracker-based P2P Application with random peer preselection

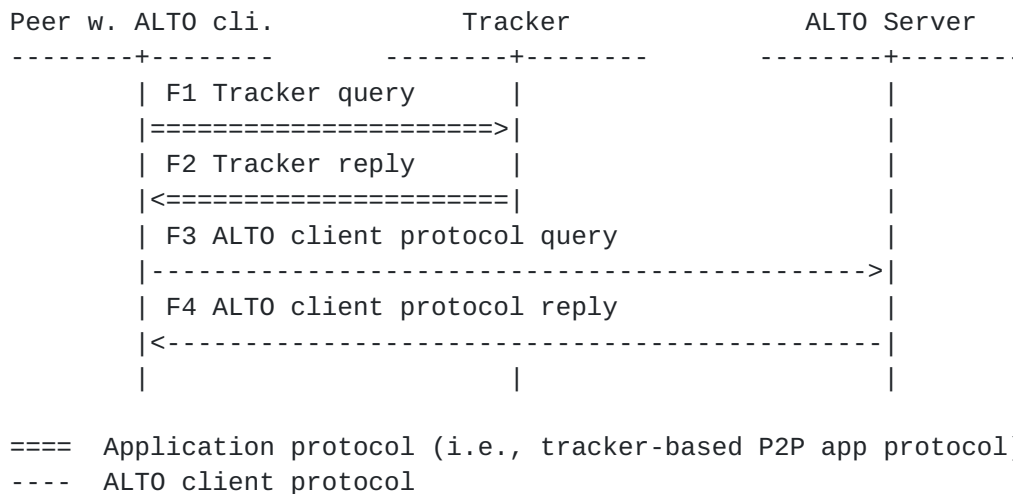


Figure 15: Basic message sequence chart for resource consumer-initiated ALT0 query





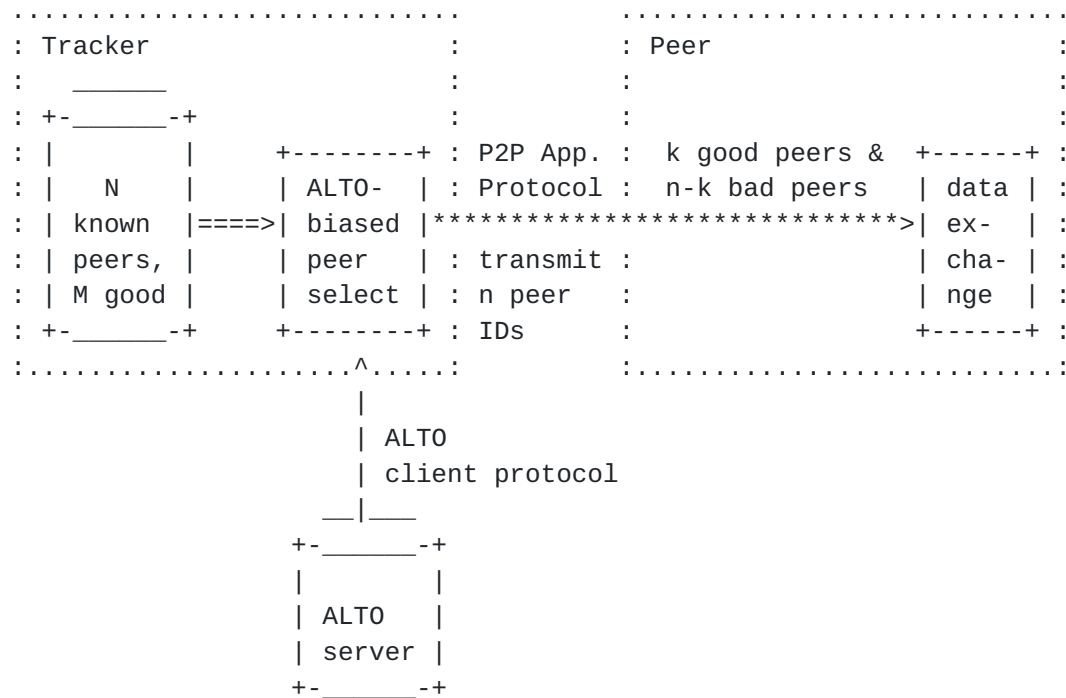


Figure 16: Tracker-based P2P Application with ALTO client in tracker

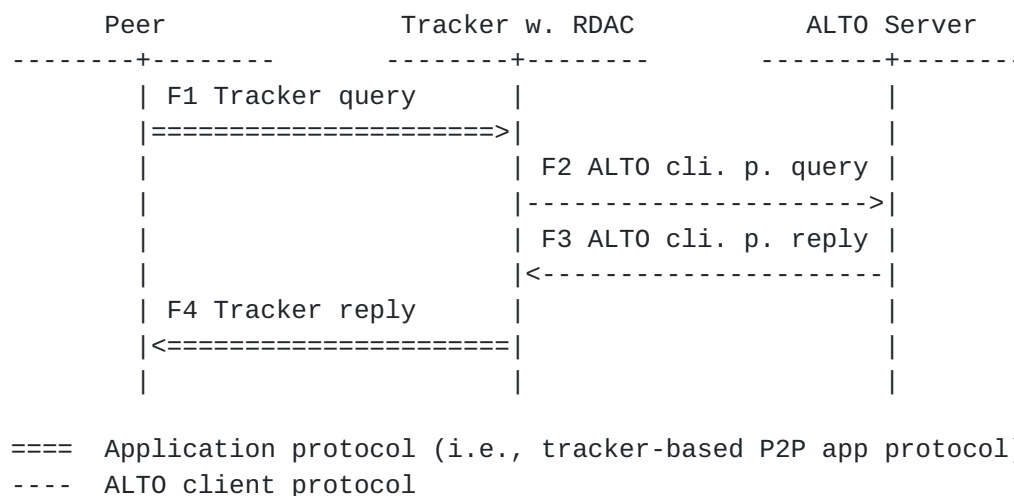


Figure 17: Basic message sequence chart for third-party ALTO query

Note: the message sequences depicted in Figure 15 and Figure 17 may occur both in the target-aware and the target-independent query mode (c.f. [I-D.ietf-alto-reqs]). In the target-independent query mode no message exchange with the ALTO server might be needed after the tracker query, because the candidate resource providers could be evaluated using a locally cached "map", which has been retrieved from the ALTO server some time ago.



The problem with the first approach is, that while the resource directory might know thousands of peers taking part in a swarm, the list returned to the resource consumer is usually shortened for efficiency reasons. Therefore, the "best" (in the sense of ALTO) potential resource providers might not be contained in that list anymore, even before ALTO can consider them.

For illustration, consider a simple model of a swarm, in which all peers fall into one of only two categories: assume that there are "good" ("good" in the sense of ALTO's better-than-random peer selection, based on an arbitrary desired rating criterion) and "bad" peers only. Having more different categories makes the maths more complex but does not change anything to the basic outcome of this analysis. Assume that the swarm has a total number of  $N$  peers, out of which are  $M$  "good" and  $N-M$  "bad" peers, which are all known to the tracker. A new peer wants to join the swarm and therefore asks the tracker for a list of peers.

If, according to the first approach, the tracker randomly picks  $n$  peers from the  $N$  known peers, the result can be described with the hypergeometric distribution. The probability that the tracker reply contains exactly  $k$  "good" peers (and  $n-k$  "bad" peers) is:

$$P(X=k) = \frac{\frac{M!}{k!(M-k)!} \cdot \frac{(N-M)!}{(n-k)!(N-M-n+k)!}}{\frac{N!}{n!(N-n)!}}$$

$$\text{with } \frac{n!}{k!(n-k)!} = \frac{n!}{k! (n-k)!} \quad \text{and} \quad n! = n * (n-1) * (n-2) * \dots * 1$$

The probability that the reply contains at most  $k$  "good" peers is:  
 $P(X \leq k) = P(X=0) + P(X=1) + \dots + P(X=k)$ .

For example, consider a swarm with  $N=10,000$  peers known to the tracker, out of which  $M=100$  are "good" peers. If the tracker randomly selects  $n=100$  peers, the formula yields for the reply:  $P(X=0)=36\%$ ,  $P(X \leq 4)=99\%$ . That is, with a probability of approx. 36% this list does not contain a single "good" peer, and with 99% probability there are only four or less of the "good" peers on the list. Processing this list with the guiding ALTO information will ensure that the few favorable peers are ranked to the top of the



list; however, the benefit is rather limited as the number of favorable peers in the list is just too small.

Much better traffic optimization could be achieved if the tracker would evaluate all known peers using ALTO, and return a list of 100 peers afterwards. This list would then include a significantly higher fraction of "good" peers. (Note, that if the tracker returned "good" peers only, there might be a risk that the swarm might disconnect and split into several disjunct partitions. However, finding the right mix of ALTO-biased and random peer selection is out of the scope of this document.)

Therefore, from an overall optimization perspective, the second scenario with the ALTO client embedded in the resource directory is advantageous, because it is ensured that the addresses of the "best" resource providers are actually delivered to the resource consumer. An architectural implication of this insight is that the ALTO server discovery procedures must support third-party discovery. That is, as the tracker issues ALTO queries on behalf of the peer which contacted the tracker, the tracker must be able to discover an ALTO server that can give guidance suitable for the that respective peer.

#### **[4.2.](#) Expectations of ALTO**

This section hints to some recent experiments conducted with ALTO-like deployments in Internet Service Provider (ISP) network's. NTT performed tests with their HINT server implementation and dummy nodes to gain insight on how an ALTO-like service influence a peer-to-peer systems [[I-D.kamei-p2p-experiments-japan](#)]. The results of an early experiment conducted in the Comcast network are documented here[RFC5632]



## 5. Using ALTO for CDNs

[Section 2](#) discussed the placement and usage of ALTO for P2P systems, but not beyond. This section discusses the usage of ALTO for Content Delivery Networks (CDNs). CDNs are used to bring a service (e.g., a web page, videos, etc) closer to the location of the user - where close refers to shorten the distance between the client and the server in the IP topology. CDNs use several techniques to decide which server is closest to a client requesting a service. One common way to do so, is relying on the DNS system, but there are many other ways, see [[RFC3568](#)].

The general issue for CDNs, independent of DNS or HTTP Redirect based approaches (see, for instance, [[I-D.penno-alto-cdn](#)]), is that the CDN logic has to match the client's IP address with the closest CDN cache. This matching is not trivial, for instance, in DNS based approaches, where the IP address of the DNS original requester is unknown (see [[I-D.vandergaast-edns-client-ip](#)] for a discussion of this and a solution approach).

### 5.1. Request Routing using the Endpoint Cost Service

Alternatively, the Request Router may request the Endpoint service from the ALTO client.

Specifically, the Request Router requests the Endpoint Cost Service in order to rank/rate the content locations (i.e., IP addresses of CDN nodes) based on their distance/cost (by default the Endpoint Cost Service operates based on Routing Distance) from/to the user address.

Once the Request Router obtained from the ALTO Server the ranked list of locations (for the specific user) it can incorporate this information into its selection mechanisms in order to point the user to the most appropriate location.

A Request Router that uses the Endpoint Cost Service may query the ALTO Server for rankings of CDN Node IP addresses for each interesting host and cache the results for later usage.

Maps Services and ECS deliver similar ALTO service by allowing the CDN to optimize internal selection mechanisms. Both services deliver similar level of security, confidentiality of layer-specific information (i.e.: application and network) however, Maps and ECS differ in the way the ALTO service is delivered and address a different set of requirements in terms of topology information and network operations.





#### **5.1.1. ALTO Topology Vs. Network Topology**

The ALTO server builds a ALTO-specific network topology that represents the network as it should be understood and utilized by the application layer (the CDN). Besides the security requirements that consist of not delivering any confidential or critical information about the infrastructure, there are efficiency requirements in terms of what visibility of the network, and which level of granularity, it is required by the CDN and more in general by the application layer.

The ALTO server builds topology (for either Map and ECS services) based on multiple sources that may include: routing protocols, network policies, state and performance information, geo-location, etc. In all cases, the ALTO topology will not contain any details that would endanger the network integrity and security (e.g.: There will be no leaking of OSPF/ISIS/BGP databases to ALTO clients).

#### **5.1.2. Topology Computation and ECS Delivery**

ECS allows the CDN not to have to implement any specific algorithm or mechanism in order to retrieve, maintain and process network topology information (of any kind). The complexity of the network topology (computation, maintenance and distribution) is kept in the ALTO server and ECS is delivered on demand. Thus ECS is used in order to implement a lightweight integration of ALTO services in the CDN layer. ECS implies an ALTO and CDN implementation with the necessary scalability in order to cope with the amount of transactions that CDN and ALTO server will have to handle (knowing that the CDN is able to cache ALTO ECS results for further use).

The ALTO server delivering ECS may integrate various information sources such as routing topology, policies, state and performance, geo-location, etc, and deliver the ranking service to the CDN upon request. The network topology information is controlled, managed by the ALTO server and the CDN benefits from ranking services in order to optimize application layer mechanisms used for content location selection. This allows the ALTO server to enhance and modify the way the topology information sources are used and combined without requiring any update in the mechanisms the ECS is delivered and do not require any update process between ALTO and the CDN.

#### **5.1.3. Ranking Service**

When a user request a given content, the CDN locates the content in one or more caches and executes a selection algorithms in order to redirect the user to the 'best' cache. In order to achieve that, the CDN issues an ECS request with the endpoint address (IPv4/IPv6) of the user (content requester) and the set of endpoint addresses of the



content caches (content targets). The ALTO server, receives the request and ranks the list of content targets addresses based on their distance from the content requester. By default, according to [[I-D.ietf-alto-protocol](#)], the distance represents the routing cost as computed by the routing layer (OSPF, ISIS, BGP) and may take into consideration other routing criteria such as MPLS-VPN (MP-BGP) and MPLS-TE (RSVP), policy and state and performance information in addition to other information sources (policy, geo-location, state and performance).

Once the ALTO server computed the distance it replies with the ranked list of content target addresses. The list being ranked by distance, the CDN is capable of integrating the rankings into its selection process (that will also incorporate other criteria) and redirect the user accordingly.

#### **5.1.4. Ranking and Network Events**

ALTO server ranks addresses based on topology information it acquires from the network. The different methods and algorithms through which the ALTO server computes topology information and rankings is out of the scope of this document. However, and in the case the rankings are based on routing (IP/MPLS) topology, it is obvious that network events may impact the ranking computation. The scope of the ECS service delivered to a CDN is not to maintain the CDN aware of any possible network topology changes since, due to redundancy of current networks, most of the network events happening in the infrastructure will have limited impact on the CDN. However, catastrophic events such as main trunks failures or backbone partition will have to take into account by the ALTO server so to redirect traffic away from the failure impacted area.

#### **5.1.5. Caching and Lifetime**

Each reply sent back by the ALTO server to the ALTO client running in the CDN has a validity in time so that the CDN can cache the results in order to re-use it and hence reducing the number of transactions between CDN and ALTO server. The ALTO server may indicate in the reply message how long the content of the message is to be considered reliable and insert a lifetime value that will be used by the CDN in order to cache (and then flush or refresh) the entry.

An ALTO server implementation may want to keep state about ALTO clients so to inform and signal to these clients when a major network event happened so to clear the ALTO cache in the client. In a CDN/ALTO interworking architecture where there's a few CDN component interacting with the ALTO server there are no scalability issues in maintaining state about clients in the ALTO server.



#### **5.1.6. Redirection**

When ALTO server receives an ECS request, it may not have the most appropriate topology information in order to accurately determine the ranking. In such case, the ALTO server, may want to adopt the following strategies:

- o Reply with available information (best effort).
- o Redirect the request to another ALTO server presumed to have better topology information (redirection).
- o Doing both (best effort and redirection). In this case, the reply message contains both the rankings and the indication of another ALTO server where more accurate rankings may be delivered.

The decision process that is used to determine if redirection is necessary (and which mode to use) is out of the scope of this document. As an example, an ALTO server may decide to redirect any request having addresses that are located into a remote Autonomous System. In such case the redirection message includes the ALTO server to be used and that resides in the remote AS. Redirection implies communication between ALTO servers so to be able to signal their identity, location and type of visibility (AS number).

#### **5.1.7. Groups and Costs**

An automated ALTO implementation may use dynamic algorithms to aggregate network topology. However, it is often desirable to have a mechanism through which the network operator can control the level and details of network aggregation based on a set of requirements and constraints. IP/MPLS networks make use of a common mechanism to aggregate and group prefixes that is called BGP Communities. BGP is the protocol all SP networks use in order to exchange information about their prefix reachability. BGP Community is an attribute used to tag a prefix so to group prefixes based on mostly any criteria (as an example, most SP networks originate BGP prefixes with communities identifying the Point of Presence (PoP) where the prefix has been originated).

The ALTO server may leverage the BGP information that is available in the SP network layer and compute group of prefixes. By policy, the ALTO server operator may decide an arbitrary cost to set between groups. Alternatively, there are algorithms that allows a dynamic computation of cost between groups.



## 6. Advanced Features

### 6.1. Cascading ALT0 Servers

The main assumptions of ALTO seems to be each ISP operates its own ALTO server independently, irrespectively of the ISP's situation. This may true for most envisioned deployments of ALTO but there are certain deployments that may have different settings. Figure 18 shows such setting, were for example, a university network is connected to two upstream providers. ISP2 if the national research network and ISP1 is a commercial upstream provider to this university network. The university, as well as ISP1, are operating their own ALTO server. The ALTO clients, located on the peers will contact the ALTO server located at the university.

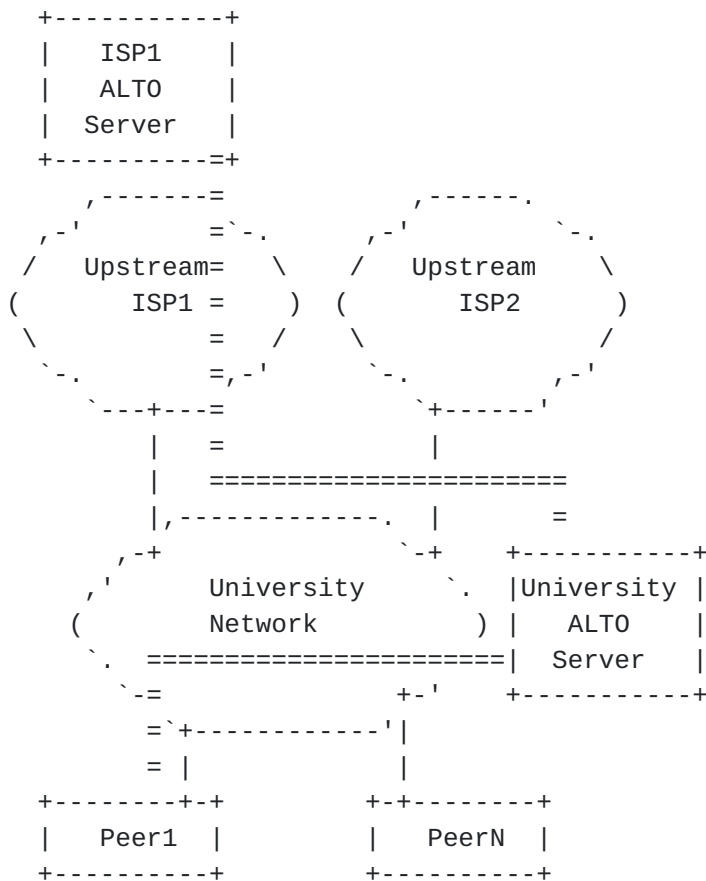


Figure 18: Cascaded ALT0 Server

In this setting all "destinations" useful for the peers within ISP2 are free-of-charge for the peers located in the university network (i.e., they are preferred in the rating of the ALTO server). However, all traffic that is not towards ISP2 will be handled by the





ISP1 upstream provider. Therefore, the ALTO server at the university has also to include the guidance given by the ISP1 ALTO server in its replies to the ALTO clients. This can be called cascaded ALTO servers.

## **6.2. ALTO for IPv4 and IPv6**

TBD

## **6.3. Monitoring ALTO**

In addition to providing configuration, an ISP providing ALTO may want to deploy a monitoring infrastructure to assess the benefits of ALTO and adjust its ALTO configuration according to the results of the monitoring.

To construct an effective monitoring infrastructure, the ISP should (1) define the performance metrics to be monitored; (2) and identify and deploy data sources to collect data to compute the performance metrics. We discuss both below.

[Editor's note: Is there a relationship to the IPPM working group at the IETF?]

### **6.3.1. Monitoring Metrics Definition**

- o Inter-domain ALTO-Integrated Application Traffic (Network metric): This metric includes total cross domain traffic generated by applications that utilize ALTO guidance. This metric evaluates the impacts of ALTO on the inbound and outbound traffic of a domain.
- o Total Inter-domain Traffic (Network metric): This is similar to the preceding but focuses on all of the traffic, ALTO aware or not. One possibility is that some of the reduction of interdomain traffic by ALTO aware applications may (XXX missing words?). This metric is always used with the preceding and the following metrics.
- o Intra-domain ALTO-Integrated Application Traffic (Network metric). (XXX description missing)
- o Network hop count (Network metric): This metric provides the average number of hops that traffic traverses inside a domain. ALTO may reduce not only traffic volume but also the hops. The metric can also indirectly reflect some application performance (e.g., latency).



- o Application download rate (Application metric): This metric measures application performance directly. Download means inbound traffic to one user. Global average means the average value of all users' download rates in one or more domains.
- o Application Client type audit(Application metric): this metric gives the audit of client types in ALTO service. The current types include fixed network client and mobile network client.

### **6.3.2. Monitoring Data Sources**

The preceding metrics are derived from data sources. We identify three data sources.

1. Application Log Server: Many application systems deploy Log Servers to collect data.
2. P2P Clients: Some P2P applications may not have Log Servers. When available, P2P client logs can provide data. This is for P2P application
3. OAM: Many ISPs deploy OAM systems to monitor IP layer traffic. An OAM provides traffic monitoring of every network device in its management area. It provides data such as link physical bandwidth and traffic volumes.

### **6.3.3. Monitoring Structure**

As discussed in the preceding section, some data sources are from ISP while some others are from application. When there is a collaboration agreement between the ISP and an application, there can be an integrated monitoring system as shown in the figure below. In particular, an application developer may deploy Monitor Clients to communicate with Monitor Server of the ISP to transmit raw data from the Log Server or P2P clients of the application to the ISP.



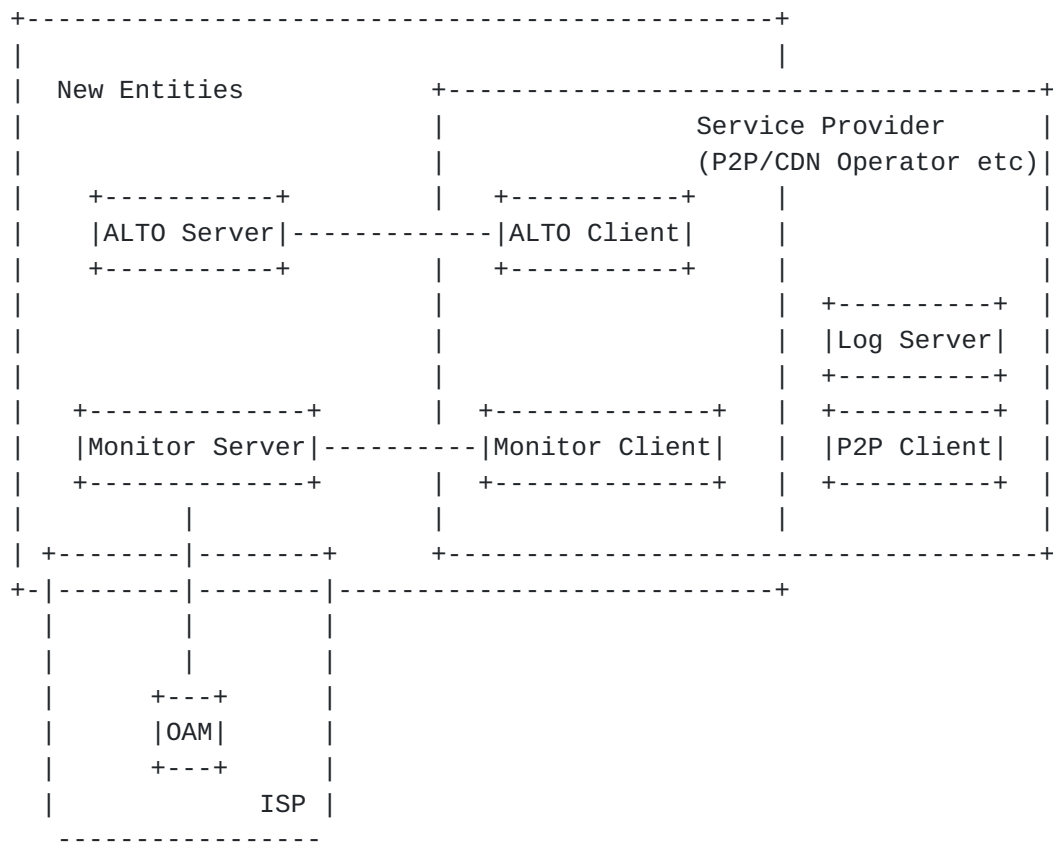


Figure 19: Monitoring Structure



## **7. Known Limitations of ALTO**

This section describes some known limitations of ALTO in general or specific mechanisms in ALTO.

### **7.1. Limitations of Map-based Approaches**

The specification of the ALTO protocol [[I-D.ietf-alto-protocol](#)] uses, amongst others mechanism, so-called network maps. The network map approach uses Host Group Descriptors that group one or multiple subnetworks (i.e., IP prefixes) to a single Host Group Descriptor. A set of IP prefixes is called partition and the associated Host Group Descriptor is called partition ID. The "costs" between the various partition IDs is stored in a second map, the cost map. Map-based approaches are chosen as they lower the signaling load on the server, as the maps have only to be retrieved if they are changed.

The main assumption for map-based approaches is that the information provided in these maps is static for a longer period of time, where this period of time refers to days, but not hours or even minutes. This assumption is fine, as long as the network operator does not change any parameter, e.g., routing within the network and to the upstream peers, IP address assignment stays stable (and thus the mapping to the partitions). However, there are several cases where this assumption is not valid, as:

1. ISPs reallocate IPv4 subnets from time to time;
2. ISPs reallocate IPv4 subnets on short notice;
3. IP prefix blocks may be assigned to a single DSLAM which serves a variety of access networks.

For 1): ISPs reallocate IPv4 subnets within their infrastructure from time to time, partly to ensure the efficient usage of IPv4 addresses (a scarce resource), and partly to enable efficient route tables within their network routers. The frequency of these "renumbering events" depend on the growth in number of subscribers and the availability of address space within the ISP. As a result, a subscriber's household device could retain an IPv4 address for as short as a few minutes, or for months at a time or even longer.

Some folks have suggested that ISPs providing ALTO services could sub-divide their subscribers' devices into different IPv4 subnets (or certain IPv4 address ranges) based on the purchased service tier, as well as based on the location in the network topology. The problem is that this sub-allocation of IPv4 subnets tends to decrease the efficiency of IPv4 address allocation. A growing ISP





that needs to maintain high efficiency of IPv4 address utilization may be reluctant to jeopardize their future acquisition of IPv4 address space.

However, this is not an issue for map-based approaches if changes are applied in the order of days.

For 2): ISPs can use techniques, such as ODAP (XXX) that allow the reallocation of IP prefixes on very short notice, i.e., within minutes. An IP prefix that has no IP address assignment to a host anymore can be reallocate to areas where there is currently a high demand for IP addresses.

For 3): In DSL-based access networks, IP prefixes are assigned to DSLAMs which are the first IP-hop in the access-network between the CPE and the Internet. The access-network between CPE and DSLAM (called aggregation network) can have varying characteristics (and thus associated costs), but still using the same IP prefix. For instance one IP addresses IP11 out of a IP prefix IP1 can be assigned to a VDSL (e.g., 2 MBit/s uplink) access-line while the subsequent IP address IP12 is assigned to a slow ADSL line (e.g., 128 kbit/s uplink). These IP addresses are assigned on a first come first served basis, i.e., the a single IP address out of the same IP prefix can change its associated costs quite fast. This may not be an issue with respect to the used upstream provider (thus the cross ISP traffic) but depending on the capacity of the aggregation-network this may raise to an issue.

## **7.2. Limitiations of Non-Map-based Approaches**

The specification of the ALTO protocol [[I-D.ietf-alto-protocol](#)] uses, amongst others mechanism, a mechanism called Endpoint Cost Service. ALTO clients can ask guidance for specific IP addresses to the ALTO server. However, asking for IP addresses, asking with long lists of IP addresses, and asking quite frequent may overload the ALTO server. The server has to rank each received IP address which causes load at the server. This may be amplified by the fact that not only a single ALTO client is asking for guidance, but a larger number of them.

Caching of IP addresses at the ALTO client or the usage of the H12 approach [[I-D.kiesel-alto-h12](#)] in conjunction with caching may lower the query load on the ALTO server.

## **7.3. General Challenges**

An ALTO server stores information about preferences (e.g., a list of preferred autonomous systems, IP ranges, etc) and ALTO clients can retrieve these preferences. However, there are basically two



different approaches on where the preferences are actually processed:

1. The ALTO server has a list of preferences and clients can retrieve this list via the ALTO protocol. This preference list can be partially updated by the server. The actual processing of the data is done on the client and thus there is no data of the client's operation revealed to the ALTO server .
2. The ALTO server has a list of preferences or preferences calculated during runtime and the ALTO client is sending information of its operation (e.g., a list of IP addresses) to the server. The server is using this operational information to determine its preferences and returns these preferences (e.g., a sorted list of the IP addresses) back to the ALTO client.

Approach 1 (we call it H1) has the advantage (seen from the client) that all operational information stays within the client and is not revealed to the provider of the server. On the other hand, does approach 1 require that the provider of the ALTO server, i.e., the network operator, reveals information about its network structure (e.g., AS numbers, IP ranges, topology information in general) to the ALTO client.

Approach 2 (we call it H2) has the advantage (seen from the operator) that all operational information stays with the ALTO server and is not revealed to the ALTO client. On the other hand, does approach 2 require that the clients send their operational information to the server.

Both approaches have their pros and cons and are extensively discussed on the ALTO mailing list. But there is basically a dilemma: Approach 1 is seen as the only working solution by peer-to-peer software vendors and approach 2 is seen as the only working by the network operators. But neither the software vendors nor the operators seem to willing to change their position. However, there is the need to get both sides on board, to come to a solution.



## **8. Extensions to the ALTO Protocol**

### **8.1. Host Group Descriptors**

Host group descriptors are used in the ALTO client protocol to describe the location of a host in the network topology. The ALTO client protocol specification defines a basic set of host group descriptor types, which have to be supported by all implementations, and an extension procedure for adding new descriptor types. The following list gives an overview on further host group descriptor types that have been proposed in the past, or which are in use by ALTO-related prototype implementations. This list is not intended as normative text. Instead, the only purpose of the following list is to document the descriptor types that have been proposed so far, and to solicit further feedback and discussion:

- o Autonomous System (AS) number
- o Protocol-specific group identifiers, which expand to a set of IP address ranges (CIDR) and/or AS numbers. In one specific solution proposal, these are called Partition ID (PID).

### **8.2. Rating Criteria**

Rating criteria are used in the ALTO client protocol to express topology- or connectivity-related properties, which are evaluated in order to generate the ALTO guidance. The ALTO client protocol specification defines a basic set of rating criteria, which have to be supported by all implementations, and an extension procedure for adding new criteria. The following list gives an overview on further rating criteria that have been proposed in the past, or which are in use by ALTO-related prototype implementations. This list is not intended as normative text. Instead, the only purpose of the following list is to document the rating criteria that have been proposed so far, and to solicit further feedback and discussion:

#### **8.2.1. Distance-related Rating Criteria**

- o Relative topological distance: relative means that a larger numerical value means greater distance, but it is up to the ALTO service how to compute the values, and the ALTO client will not be informed about the nature of the information. One way of generating this kind of information MAY be counting AS hops, but when querying this parameter, the ALTO client MUST NOT assume that the numbers actually are AS hops.
- o Absolute topological distance, expressed in the number of traversed autonomous systems (AS).



- o Absolute topological distance, expressed in the number of router hops (i.e., how much the TTL value of an IP packet will be decreased during transit).
- o Absolute physical distance, based on knowledge of the approximate geolocation (continent, country) of an IP address.

#### **8.2.2. Charging-related Rating Criteria**

- o Traffic volume caps, in case the Internet access of the resource consumer is not charged by "flat rate". For each candidate resource provider, the ALTO service could indicate the amount of data that may be transferred from/to this resource provider until a given point in time, and how much of this amount has already been consumed. Furthermore, it would have to be indicated how excess traffic would be handled (e.g., blocked, throttled, or charged separately at an indicated price). The interaction of several applications running on a host, out of which some use this criterion while others don't, as well as the evaluation of this criterion in resource directories, which issue ALTO queries on behalf of other peers, are for further study.

#### **8.2.3. Performance-related Rating Criteria**

The following rating criteria are subject to the remarks below.

- o The minimum achievable throughput between the resource consumer and the candidate resource provider, which is considered useful by the application (only in ALTO queries), or
- o An arbitrary upper bound for the throughput from/to the candidate resource provider (only in ALTO responses). This may be, but is not necessarily the provisioned access bandwidth of the candidate resource provider.
- o The maximum round-trip time (RTT) between resource consumer and the candidate resource provider, which is acceptable for the application for useful communication with the candidate resource provider (only in ALTO queries), or
- o An arbitrary lower bound for the RTT between resource consumer and the candidate resource provider (only in ALTO responses). This may be, for example, based on measurements of the propagation delay in a completely unloaded network.

The ALTO client MUST be aware, that with high probability, the actual performance values differ significantly from these upper and lower bounds. In particular, an ALTO client MUST NOT consider the "upper





bound for throughput" parameter as a permission to send data at the indicated rate without using congestion control mechanisms.

The discrepancies are due to various reasons, including, but not limited to the facts that

- o the ALTO service is not an admission control system
- o the ALTO service may not know the instantaneous congestion status of the network
- o the ALTO service may not know all link bandwidths, i.e., where the bottleneck really is, and there may be shared bottlenecks
- o the ALTO service may not know whether the candidate peer itself is overloaded
- o the ALTO service may not know whether the candidate peer throttles the bandwidth it devotes for the considered application
- o the ALTO service may not know whether the candidate peer will throttle the data it sends to us (e.g., because of some fairness algorithm, such as tit-for-tat)

Because of these inaccuracies and the lack of complete, instantaneous state information, which are inherent to the ALTO service, the application must use other mechanisms (such as passive measurements on actual data transmissions) to assess the currently achievable throughput, and it MUST use appropriate congestion control mechanisms in order to avoid a congestion collapse. Nevertheless, these rating criteria may provide a useful shortcut for quickly excluding candidate resource providers from such probing, if it is known in advance that connectivity is in any case worse than what is considered the minimum useful value by the respective application.

#### **8.2.4. Inappropriate Rating Criteria**

Rating criteria that SHOULD NOT be defined for and used by the ALTO service include:

- o Performance metrics that are closely related to the instantaneous congestion status. The definition of alternate approaches for congestion control is explicitly out of the scope of ALTO. Instead, other appropriate means, such as using TCP based transport, have to be used to avoid congestion.



## **9. API between ALTO Client and Application**

This sections gives some informational guidance on how the interface between the actual application using the ALTO guidance and the ALTO client can look like.

This is still TBD.

## **10. Security Considerations**

The ALTO protocol itself, as well as, the ALTO client and server raise new security issues beyond the one mentioned in [[I-D.ietf-alto-protocol](#)] and issues related to message transport over the Internet. For instance, Denial of Service (DoS) is of interest for the ALTO server and also for the ALTO client. A server can get overloaded if too many TCP requests hit the server, or if the query load of the server surpasses the maximum computing capacity. An ALTO client can get overloaded if the responses from the sever are, either intentionally or due to an implementation mistake, too large to be handled by that particular client.

### **10.1. Information Leakage from the ALTO Server**

The ALTO server will be provisioned with information about the owning ISP's network and very likely also with information about neighboring ISPs. This information (e.g., network topology, business relations, etc) is consider to be confidential to the ISP and must not be revealed.

The ALTO server will naturally reveal parts of that information in small doses to peers, as the guidance given will depend on the above mentioned information. This is seen beneficial for both parties, i.e., the ISP's and the peer's. However, there is the chance that one or multiple peers are querying an ALTO server with the goal to gather information about network topology or any other data considered confidential or at least sensitive. It is unclear whether this is a real technical security risk or whether this is more a perceived security risk.

### **10.2. ALTO Server Access**

Depending on the use case of ALTO, several access restrictions to an ALTO server may or may not apply. For an ALTO server that is solely accessible by peers from the ISP network (as shown in Figure 12), for instance, the source IP address can be used to grant only access from that ISP network to the server. This will "limit" the number of peers able to attack the server to the user's of the ISP (however, including botnet computers).

On the other hand, if the ALTO server has to be accessible by parties not located in the ISP's network (see Figure Figure 11), e.g., by a third-party tracker or by a CDN system outside the ISP's network, the access restrictions have to be more loose. In the extreme case, i.e., no access restrictions, each and every host in the Internet can access the ALTO server. This might no the intention of the ISP, as the server is not only subject to more possible attacks, but also on



the load imposed to the server, i.e., possibly more ALTO clients to serve and thus more work load.

### **10.3. Faking ALTO Guidance**

It has not yet been investigated how a faked or wrong ALTO guidance by an ALTO server can impact the operation of the network and also the peers.

Here is a list of examples how the ALTO guidance could be faked and what possible consequences may arise:

**Sorting** An attacker could change to sorting order of the ALTO guidance (given that the order is of importance, otherwise the ranking mechanism is of interest), i.e., declaring peers located outside the ISP as peers to be preferred. This will not pose a big risk to the network or peers, as it would mimic the "regular" peer operation without traffic localization, apart from the communication/processing overhead for ALTO. However, it could mean that ALTO is reaching the opposite goal of shuffling more data across ISP boundaries, incurring more costs for the ISP.

**Preference of a single peer** A single IP address (thus a peer) could be marked as to be preferred all over other peers. This peer can be located within the local ISP or also in other parts of the Internet (e.g., a web server). This could lead to the case that quite a number of peers to trying to contact this IP address, possibly causing a Denial of Service (DoS) attack.

This section is solely giving a first shot on security issues related to ALTO deployments.



## **11. Conclusion**

This is the first version of the deployment considerations and for sure the considerations are yet incomplete and imprecise.



## **12. References**

### **12.1. Normative References**

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", [BCP 14](#), [RFC 2119](#), March 1997.
- [RFC3568] Barbir, A., Cain, B., Nair, R., and O. Spatscheck, "Known Content Network (CN) Request-Routing Mechanisms", [RFC 3568](#), July 2003.

### **12.2. Informative References**

- [I-D.ietf-alto-protocol]  
Penno, R., Alimi, R., and Y. Yang, "ALTO Protocol", [draft-ietf-alto-protocol-10](#) (work in progress), October 2011.
- [I-D.ietf-alto-reqs]  
Kiesel, S., Previdi, S., Stiernerling, M., Woundy, R., and Y. Yang, "Application-Layer Traffic Optimization (ALTO) Requirements", [draft-ietf-alto-reqs-13](#) (work in progress), January 2012.
- [I-D.ietf-alto-server-discovery]  
Kiesel, S., Stiernerling, M., Schwan, N., and M. Scharf, "ALTO Server Discovery", [draft-ietf-alto-server-discovery-02](#) (work in progress), September 2011.
- [I-D.kamei-p2p-experiments-japan]  
Kamei, S., Momose, T., Inoue, T., and T. Nishitani, "ALTO-Like Activities and Experiments in P2P Network Experiment Council", [draft-kamei-p2p-experiments-japan-06](#) (work in progress), September 2011.
- [I-D.kiesel-alto-h12]  
Kiesel, S. and M. Stiernerling, "ALTO H12", [draft-kiesel-alto-h12-02](#) (work in progress), March 2010.
- [I-D.lee-alto-chinatelecom-trial]  
Li, K. and G. Jian, "ALTO and DECADE service trial within China Telecom", [draft-lee-alto-chinatelecom-trial-03](#) (work in progress), October 2011.
- [I-D.penno-alto-cdn]  
Penno, R., Medved, J., Alimi, R., Yang, R., and S. Previdi, "ALTO and Content Delivery Networks",



[draft-penno-alto-cdn-03](#) (work in progress), March 2011.

[I-D.vandergaast-edns-client-ip]

Contavalli, C., Gaast, W., Leach, S., and D. Rodden,  
"Client IP information in DNS requests",  
[draft-vandergaast-edns-client-ip-01](#) (work in progress),  
May 2010.

[RFC5632] Griffiths, C., Livingood, J., Popkin, L., Woundy, R., and  
Y. Yang, "Comcast's ISP Experiences in a Proactive Network  
Provider Participation for P2P (P4P) Technical Trial",  
[RFC 5632](#), September 2009.

[RFC5693] Seedorf, J. and E. Burger, "Application-Layer Traffic  
Optimization (ALTO) Problem Statement", [RFC 5693](#),  
October 2009.



## **Appendix A. Contributors List and Acknowledgments**

This memo is the result of contributions made by several people, such as:

- o Xianghue Sun, Lee Kai, and Richard Yang contributed [Section 3](#) and [Section 6.3](#).
- o Stefano Previdi contributed Section [Section 5](#) on "Using ALTO for CDNs".

Martin Stiernerling is partially supported by the COAST project (Content Aware Searching, retrieval and sTreaming, <http://www.coast-fp7.eu>), a research project supported by the European Commission under its 7th Framework Program (contract no. 248036). The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of the COAST project or the European Commission.



## Authors' Addresses

Martin Stiemerling (editor)  
NEC Laboratories Europe  
Kurfuerstenanlage 36  
Heidelberg 69115  
Germany

Phone: +49 6221 4342 113  
Fax: +49 6221 4342 155  
Email: martin.stiemerling@neclab.eu  
URI: <http://ietf.stiemerling.org>

Sebastian Kiesel (editor)  
University of Stuttgart, Computing Center  
Allmandring 30  
Stuttgart 70550  
Germany

Email: ietf-alto@skiesel.de

Stefano Previdi  
Cisco Systems, Inc.  
Via Del Serafico 200  
Rome 00191  
IT

Email: sprevidi@cisco.com

