

ALTO
Internet-Draft
Intended status: Standards Track
Expires: April 18, 2013

D. Wang
H. Song
N. Zong
Huawei
Oct 15, 2012

ALTO and DECADE Integration
draft-wang-alto-decade-integration-00

Abstract

While DECADE architecture is going to provide a framework to use in-network storage with standard interface for content distribution, there are still cases when many in-network storage servers provide the same content. Then traffic optimization mechanism is needed to select one best replica for the user request. This document is going to discuss the integration of Application Layer Traffic Optimization (ALTO) protocol into DECADE so as to optimize network traffic.

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 18, 2013.

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

1.	Introduction	3
2.	Terminology	3
3.	Integration Architecture	3
3.1.	Function Entity	3
3.2.	Integration Architecture	3
3.3.	Process Flow	5
4.	Use cases	6
4.1.	P2P File Sharing	6
4.2.	Offline P2P live streaming	7
5.	ALTO Protocol Considerations	9
6.	Security Considerations	10
7.	IANA Considerations	11
8.	References	11
8.1.	Normative References	11
8.2.	Informative Reference	11
	Authors' Addresses	11

1. Introduction

In-network storage is popular for content distribution, but existing framework either keeps applications out of the control loop or need to support variety of application protocol and implementations. DECADE architecture provides the general content distribution framework with standard interface for resource control and data transport. But either using the DECADE compatible in-network storage servers as personal storage or using it as content distribution platform for applications, there are possibilities that multiple servers provide the same content, which would a good scenario to consider using Application Traffic Optimization (ALTO) Protocol to select a better-than-random server.

2. Terminology

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](#)].

This document reuses the terminology defined in [I-D.[draft-alto-problem-statement](#)] and [I-D.[draft-decade-problem-statement](#)].

3. Integration Architecture

3.1. Function Entity

ALTO Server: The logical entity that provides interfaces to the queries to the ALTO service [[RFC5693](#)].

DECADE Server: A DECADE Server stores DECADE data inside the network, and thereafter manages both the stored data and access to that data [I-D.ietf-decade-arch-02].

DECADE Client: A DECADE Client stores and retrieves data at DECADE Servers [I-D.ietf-decade-arch-02].

App Tracker: Index Server that stores current peer list which is sharing certain network resource, such as movies or music, and so on.

3.2. Integration Architecture

Figure 1 shows the basic integration architecture framework of integrating DECADE and ALTO. It indicates that integrating DECADE and ALTO can provide optimized network traffic by selecting serving

peers in a better-than-random model.

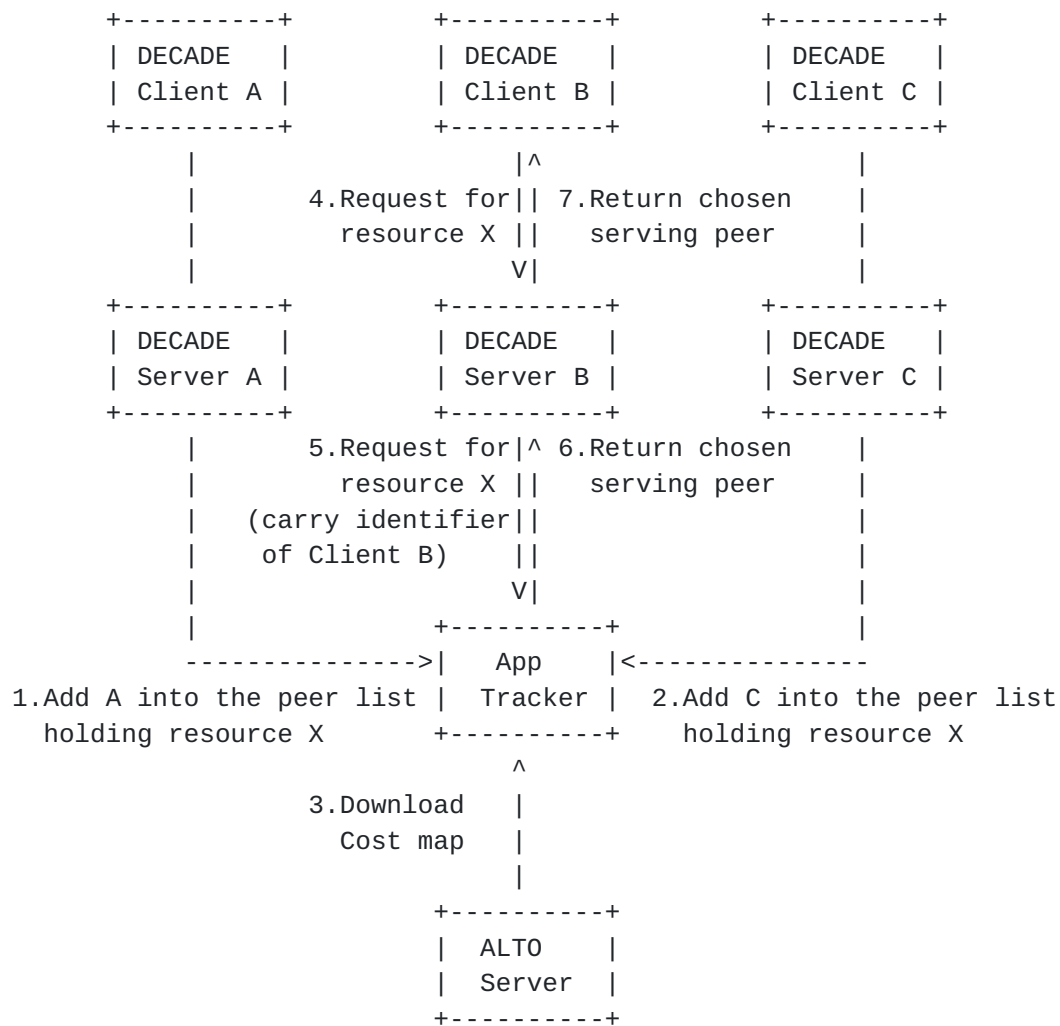


Figure 1 ALTO and DECADE Integration Architecture Framework

In this framework, DECADE Client requests certain network resources from its in-network storage (DECADE Server). If its DECADE Server contains such resources, the request would be satisfied immediately and DECADE Client can download the resource directly. Otherwise, its DECADE Server will query the App Tracker for the best serving peer which contains those resources, carrying the identifier of the requesting DECADE Client in the request. Each time when App Tracker receives DECADE Server's request for certain resources, it queries for all current serving peers that contain such resources. It then looks up the network map it already downloaded from the ALTO Server in advance. The cost between the requesting DECADE Client and each serving peer in the peer list will be calculated. Finally, the best serving replica at the lowest cost will be returned and the requesting resources can be downloaded from it. This scheme embodies the idea of integration of DECADE and ALTO, which may bring many

benefits, such as saving network bandwidth.

In this framework, the cost between every two serving peers can be looked up from ALTO Server, however, the cost between a DECADE Client and any other Client's DECADE Server cannot be obtained. Therefore, it'd better to pre-store network map with cost between each DECADE Client and its DECADE Server in App Tracker. In this way, combining with cost map provided by ALTO Server, the cost between a DECADE Client and any other Client's DECADE Server can be calculated, and the lowest cost can be decided.

3.3. Process Flow

Figure 2 shows the process flow of integrating DECADE and ALTO .

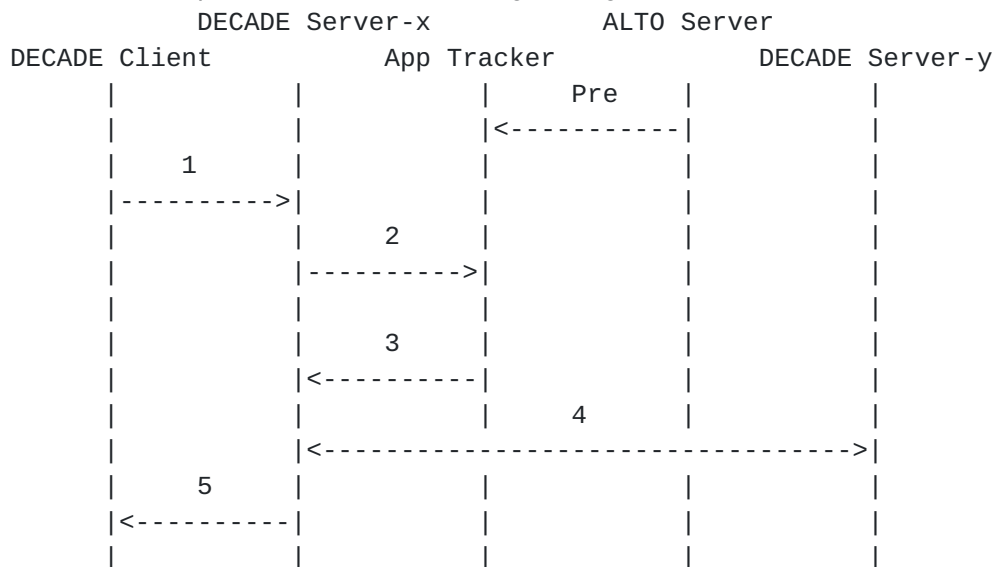


Figure 2 Process Flow of Integrating DECADE and ALTO

Pre: App Tracker downloads cost map from ALTO Server in advance. This step can happens any time before step 3.

1. DECADE Client requests for certain network resources from its DECADE Server (here we name it DECADE Server-x).
2. If DECADE Server-x doesn't include the requested resources, it will ask for App Tracker's suggestion. The identifier of DECADE Client will be sent to the App Tracker.
3. APP Tracker looks up all current serving peers that contain such resources. It then calculate the cost between the requesting Client and each serving peers based on the cost map it downloaded from ALTO Server in advance and the pre-stored network map with cost between each DECADE Client and its DECADE Server. Finally, the most suitable serving peer (here we name it DECADE Server-y) at the lowest cost

will be picked up and returned to DECADE Server-x.

4. DECADE Server-x downloads the data object from DECADE Server-y.
5. DECADE Client downloads the data object from its DECADE Server (DECADE Server-x).

4. Use cases

4.1. P2P File Sharing

Related Applications: There are many P2P file sharing applications, e.g., BitTorrent, Shareaza, Ares. The most widely used is BitTorrent.

Original setting: In BitTorrent without DECADE and ALTO, a client uploads to other clients using its last mile. This consumes expensive last-mile uplink bandwidth.

BitTorrent with DECADE: A client first uploads files to a DECADE Server, who can then serve other clients.

BitTorrent with DECADE and ALTO: While two DECADE Servers hold the same file that a client wants, integration DECADE with ALTO can help the client decide which DECADE Server is the best replica to get this file. And in this scheme, it may bring many benefits, such as saving network bandwidth.

Figure 3 shows an example. In this example, DECADE Client B wants to get a file which both DECADE Server A and DECADE Server C hold. In the original setting without integrating DECADE with ALTO, DECADE Client B could randomly get the file by its DECADE Server (DECADE Server B) from either DECADE Server A or DECADE Server C. In this case, DECADE Client B would consult App Server which DECADE Server is the most suitable serving peer. In this process, ALTO Server provides cost map, and the network map with cost between each DECADE Client and its DECADE Server is pre-stored in App Tracker. Based on those two maps, the cost between each serving DECADE Server and the requesting DECADE Client can be calculated, finally the serving peer at the lowest cost would be selected.

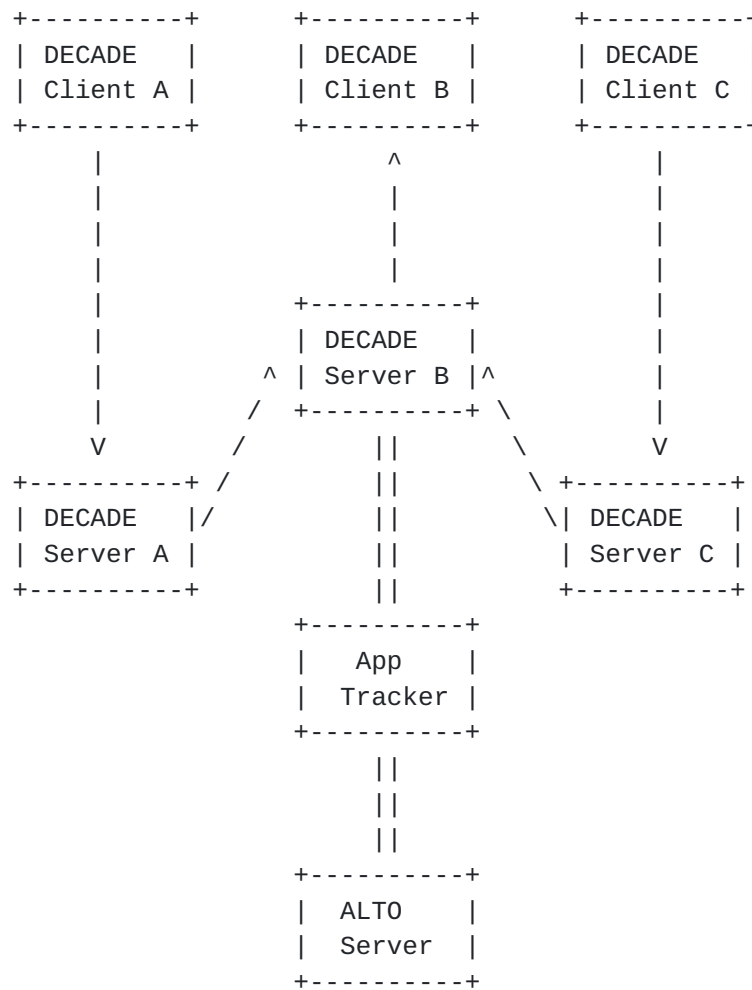


Figure 3 P2P File Sharing

Benefits: DECADE Client can get file by its DECADE Sever from a serving peer at the lowest cost, which may saving traffic. Of course, last mile uplink bandwidth is saved, which is expensive.

4.2. Offline P2P live streaming

Related Applications: There are many P2P live streaming applications, e.g., PPLive, PPStream.

Original setting: In P2P live streaming without DECADE and ALTO, clients may get a bad user experience when they watch the streaming after broadcasting at a later time, since there will be not enough audiences to watch it after living broadcast.

PPLive with DECADE: By using DECADE, a client can watch a live streaming at a later time, and still have a good video quality.

PPLive with DECADE and ALTO: While a client wants to watch a live

streaming at a later time, integrating DECADE and ALTO can help it decide a DECADE Server which hold the content and also at the lowest cost. The client can also enjoy a good user experience, instead of having a bad video quality.

Figure 4 shows an example. In this example, Client B is broadcasting a live streaming. Client A is online and watching the live streaming, and DECADE Server A downloads this streaming from DECADE Server B in real time. Client C is offline and wants to watch this streaming at a later time. DECADE Server C can download this streaming from DECADE Server A and DECADE Server B after comparing their cost by consulting App Tracker and ALTO Server. Then DECADE Server C will hold the living streaming data for Client C. After Client C once is online, it can get the streaming directly for its DECADE Server C.

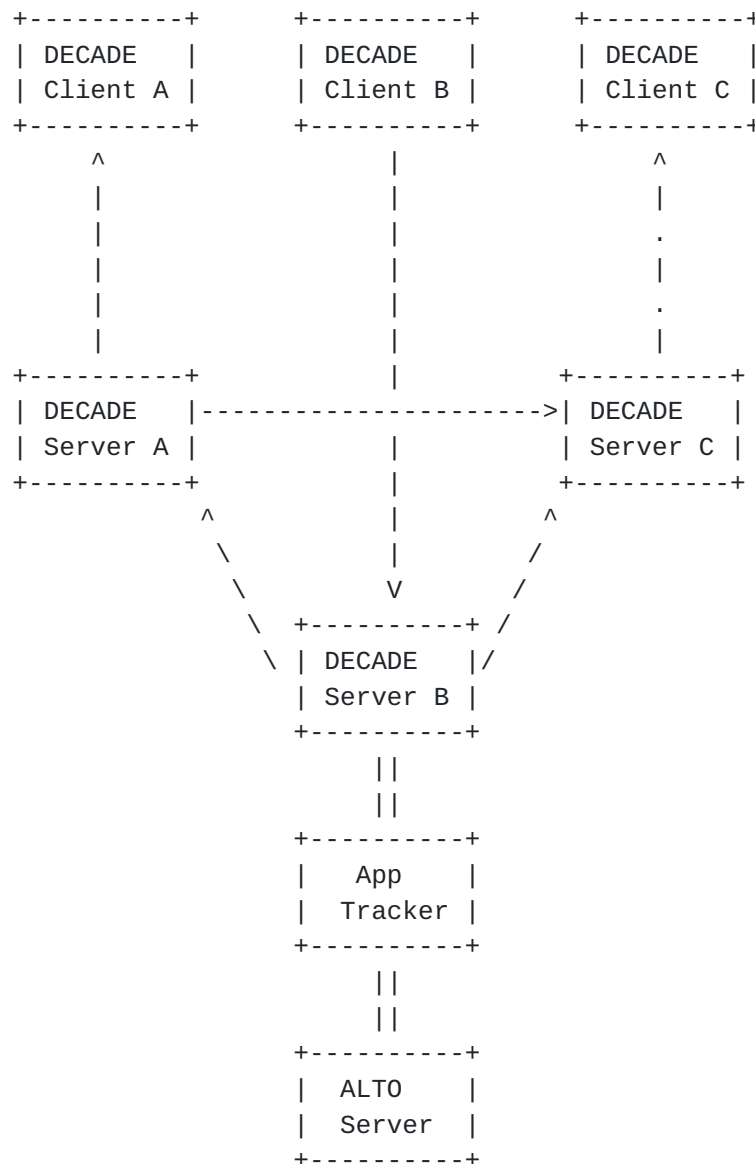


Figure 4 Offline P2P Live Streaming

Benefits: In original setting, clients missed live streaming broadcast, may get a bad video quality when they watch it at a later time. While in integrating DECADE with ALTO, the video quality is also good since Client C downloads it directly from it DECADE Server. Also, the new setting can help save network traffic as well as last mile uplink bandwidth in this use case.

5. ALTO Protocol Considerations

In this context, App Tracker may currently need a variety of information to perform serving peer selection, for example, ALTO Cost Map from ALTO Server. Via embedding ALTO Client in App Tracker, App

Tracker can obtain and locally store ALTO Cost Map from ALTO Server by ALTO Protocol, and benefit from this ALTO information to enable network-efficient traffic patterns and improve serving performance. Figure 5 shows the interactions among the main entities in our integration framework where App Tracker is an ALTO Client and applies cost map when selecting the best serving peer.

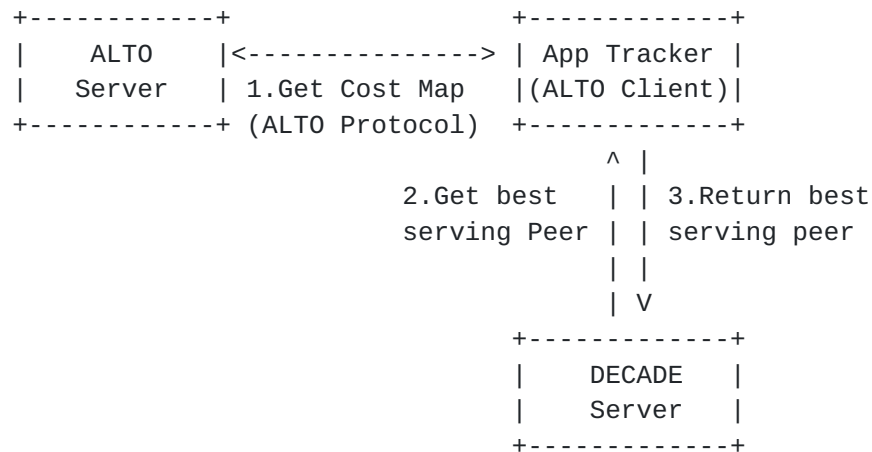


Figure 5 ALTO Client Embedded in App Tracker

1. The App Tracker requests the Cost Map amongst all PIDs from the ALTO Server by ALTO Protocol.
2. A DECADE Server requests current serving peer at lowest cost from the App Tracker.
3. The App Tracker returns current best serving peer, which is computed based on the ALTO Cost Map as well as other information pre-stored in the App Tracker.

Since App Tracker acts as the ALTO Client in our scheme, while applying ALTO Protocol, "Endpoint" in the protocol is used to carry/define information of the App Tracker, including address type, address of App Tracker, et, al. ALTO Cost Map is obtained from ALTO Server via an HTTP GET method without any input parameters. The media type is "application/alto-costmap+json". And ALTO Server would return cost map or an error code to the App Tracker after receiving such request.

6. Security Considerations

This document does not consider security issues. Security will be considered in further version of this draft.

7. IANA Considerations

There is no IANA consideration for this document.

8. References

8.1. Normative References

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", March 1997.
- [RFC5693] Seedorf, J. and E. Burger, "Application-Layer Traffic Optimization (ALTO) Problem Statement", October 2009.

8.2. Informative Reference

- [I-D.ietf-decade-arch]
Alimi, R., Yang, Y., Rahman, A., Kutscher, D., and H. Liu, "DECADE Architecture", July 2011.
- [I-D.ietf-decade-problem-statement]
Song, H., Zong, N., Yang, Y., and R. Alimi, "DECoupled Application Data Enroute (DECADE) Problem Statement".
- [I-D.[draft-alto-problem-statement](#)]
Seedorf, J. and E. Burger, "Application-Layer Traffic Optimization (ALTO) Problem Statement", March 2009.

Authors' Addresses

Danhua Wang
Huawei

Email: wangdanhua@huawei.com

Haibin Song
Huawei

Email: haibin.song@huawei.com

Ning Zong
Huawei

Email: zongning@huawei.com