DECADE
Internet-Draft
Intended status: Informational
Expires: April 21, 2011

L. Chen
H. Liu
Yale University
October 18, 2010

Leveraging In-network Storage in P2P LiveStreaming
draft-chen-decade-intgr-livestr-exmp-00

Abstract

DECADE is an in-network storage infrastructure under discussions and constructions. It can be integrated into Peer-to-Peer (P2P) applications to achieve more efficient content distributions. This document represents a detailed example of integrating DECADE with a dominating P2P application - P2P livestreaming. This document describes a preliminary framework of DECADE client API, P2P live streaming integration, the environment of the test on this integrarion and application performance analysis in the test.

Status of this Memo

This Internet-Draft is submitted to IETF in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

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."

The list of current Internet-Drafts can be accessed at http://www.ietf.org/ietf/1id-abstracts.txt.

The list of Internet-Draft Shadow Directories can be accessed at http://www.ietf.org/shadow.html.

This Internet-Draft will expire on April 21, 2011.

Table of Contents

1. Introduction

DECADE is an in-network storage infrastructure under discussions and constructions. It can be integrated into Peer-to-Peer (P2P) applications to achieve more efficient content distributions.

This draft introduces an instance of application integration with DECADE. In our example system, the core component includes DECADE server and DECADE-aware P2P live stream client. DECADE server running at Linux platform is designed to support data reading and writing for DECADE clients. For DECADE client, we employed a P2P live streaming system called P2PLS (P2P Live Streaming). We utilized a preliminary API (Application Programming Interface) set provided by DECADE to enable P2PLS clients to leverage DECADE in their data transmission. In this draft, we introduce the system structure of the DECADE-P2PLS integration system, the main DECADE related message flow, the environment of the test on this integration, and the system performance in the test.

Please note that P2PLS in this draft only represents the usage case of "live streaming" out of a large number of P2P applications, while DECADE itself can support other applications. The API set of DECADE is only an experimental design and implementation. It is not a standard and is still under development.

2. Concepts

2.1. DECADE Server

DECADE Server is a server implemented DECADE protocols, management mechanism and storage strategies. It is an important element to provide DECADE services. In a DECADE server, we have a number of Data Lockers which are virtual accounts and storage space for applications.

2.2. DECADE Client Plug-in

DECADE Client Plug-in is a plug-in component for application clients to use DECADE server. This plug-in component is served as an application-specific interface between a particular application and DECADE servers. It can be a very simple component of basic DECADE access APIs implemented, or a smart component to integrate application-specific control strategies with DECADE APIs.

2.3. P2P LiveStreaming Client

P2P LiveStreaming Client is our self-maintained version of a native P2P live streaming application.

3. DECADE Client API

In order to simplify the usage of DECADE server for P2P application clients, we provide a set of APIs. On top of these APIs, a P2P application client can develop its application-specific control and data distributed policies to utilize DECADE servers.

There are five basic APIs:

- Get_Object: to get an object from a DECADE server with an authorized token. The get operation can be classified into two categories: Local Get and Remote Get. Local Get is to get an object from a local DECADE server. Remote Get is to use an application's local DECADE server to indirectly get an object from a remote DECADE server. The object will first passed to the local DECADE server, then return to the application client

- Put_Object: to store an object into a DECADE server with an authorized token. An application client can either store an object in its local DECADE server or in other clients' DECADE servers, if it has a authorized token. Both operates are directly store, we don't provide indirectly store (i.e. remote put).

- Delete_Object: to delete an object in a DECADE server explicitly with an authorized token. Know that an object can be deleted implicitly by setting a expired time or a specific TTL.

- Status_Query: to query current status of an application itself, including a list of stored objects, resource usage, and so on. An application cannot query others' status.

- Generate_Token: to generate an authorized token. The token can be used to access an application client's local DECADE server, or passed to other clients to allow others to access its DECADE server.

4. DECADE Integration of P2P LiveStreaming Client

We integrate DECADE client API with a P2P live streaming application, in order that clients of this P2P live streaming application can easily utilize DECADE server in their data transport processes.

4.1. DECADE Client Plug-in

DECADE Client Plug-in is a plug-in component, on top of DECADE client APIs, designed for application clients to use DECADE server. This plug-in component serves as an application-specific interface between a particular application and DECADE servers. In this test, we employed P2P live streaming application to leverage this plug -in.

4.2. DECADE Integration Architecture

The architecture of the P2P live streaming and DECADE integration is in Figure 1:

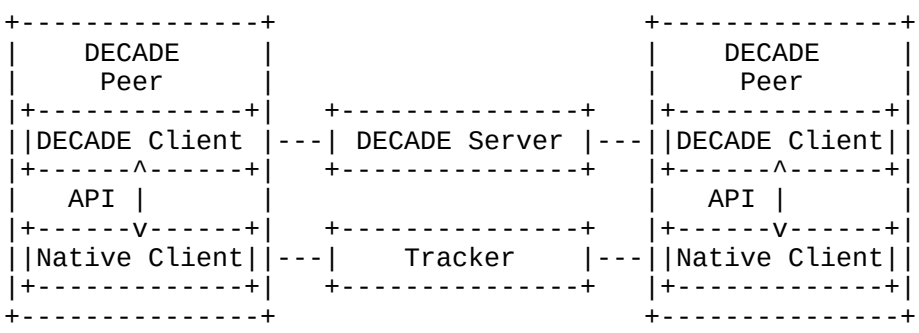


Figure 1

A DECADE integrated P2P live streaming client uses DECADE client plug-in to communicate with its DECADE server and access data between itself and DECADE servers. It uses an additional modified message protocol, as well as its original P2P message protocols to connect with other peers, exchange control messages.

4.2.1. Data Access

DECADE client plug-in is called whenever the application want to get data objects from (or put data objects into) DECADE servers. Every data object transferred between DECADE server and original P2P live streaming client must go through this plug-in. Neither the DECADE server and the original client knows each other. A Data object is a transfer unit of data, between DECADE servers and application clients. It the size of data object can be application-customized, according to variable requirements of performance or sensitive factors (e.g. low latency, high bandwidth utilization).

4.2.2. Message Control

Control and Data plane decoupling is a design principle of DECADE. Control messages are propagated in an original P2P way. Besides, original P2P control messages, a modified message protocol is needed for DECADE authorized token delivery. By exchanging DECADE authorized tokens, P2P live streaming clients can retrieve or store data objects into or from other clients' DECADE servers.

5. Test Environment and Settings

In order to demo our DECADE integrated P2P live streaming application, we conduct an test in Amazon EC2 and PlanetLab to show the performance of the system. For comparison, we conduct two tests, DECADE integrated P2P live streaming and native P2P live streaming, in the same environment use the same setting.

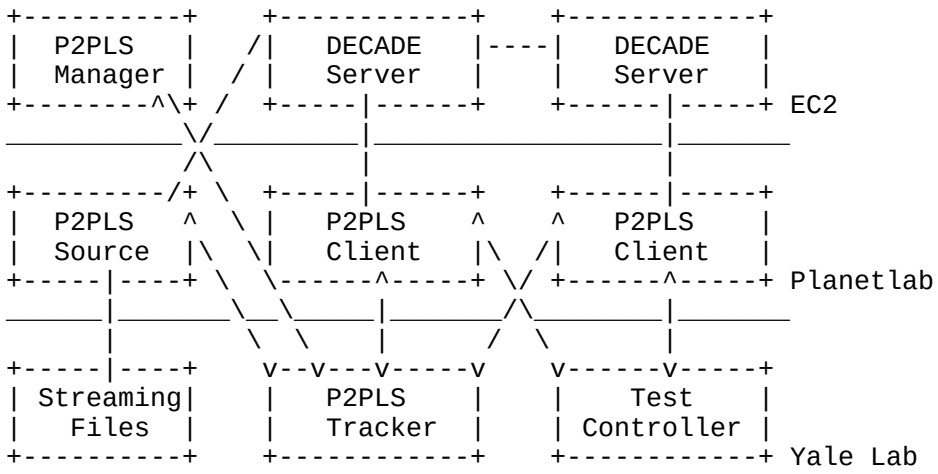
5.1. Test Settings

Our test ran on a wide-spread area and a diverse platforms,including a famous commercial cloud platform and a well-known testbed PlanetLab. The environment settings are as following:

- EC2 Regions: we setup DECADE servers in Amazon EC2 cloud, including all four regions around the world, US east, US west, Europe and Asia.
- PlanetLab: we run our P2P live streaming clients (both DECADE integrated and native clients) on PlanetLab of a wild-spread area.
- Arrival pattern: we let all the clients join at the same time to simulate a flash crown scenario.
- Total Bandwidth: for fair comparison, we set the system's total supply bandwidth to be exact the same in both test.

5.2. Platforms and Components

In our test, we have different functional components running in different platforms, including DECADE servers, P2P live streaming clients(DECADE integrated or Native), Tracker, Source server and Test Controller, as shown in Figure 2.



P2PLS represents to P2P LiveStreaming.

Figure 2

5.2.1. EC2 DECADE Server

DECADE Servers ran on Amazon EC2 small instances, with bandwidth constraint.

5.2.2. PlanetLab P2P LiveStreaming Client

Both DECADE integrated and Native P2P live streaming clients run in planetlab spreading in different locations in the world. The DECADE integrated P2P live streaming clients connect to a closest DECADE server according to its Geo-location distance of the servers. DECADE integrated P2P live streaming clients use their DECADE servers to upload to neighbors, instead of their own "last-mile" bandwidth.

5.2.3. Tracker

A native P2P live streaming tracker ran in our lab to serve for both DECADE integrated clients and native clients during the test.

5.2.4. Source Server

A native P2P live streaming source server ran in our lab to serve for both DECADE integrated clients and native clients during the test. The capacity of source is equivalently constrain for both cases.

5.2.5. Test Controller

Test Controller is a manager to control all machines behaviors in both EC2 and PlanetLab during the test.

6. Performance Analysis

During the test, DECADE integrated P2P live streaming clients achieve a better performance in the results.

6.1. Performance Metrics

- Startup Delay: the time from a peer joins the channel to the moment it starts to play.
- Piece Missed Rate: number of pieces a peer missed in the playing buffer to total number of pieces.
- Freeze Times: number of a peer freeze during playing.
- Average Peer Uploading Rate: Average uploading bandwidth of a peer per second.

6.2. Result and Analysis

- Startup Delay: In the test, DECADE integrated P2P live streaming clients startup around 35˜40 seconds. some of them startup at about 10 seconds. Native P2P live streaming clients startup around 110˜120 seconds. less than 20% of them startup within 100 seconds.
- Piece Missed Rate: In the test, both DECADE integrated P2P live streaming clients and native P2P live streaming clients achieved a good performance in pieces missed rate. Only about 0.02% of total pieces missed in both cases.
- Freeze Times: In the test, native P2P live streaming clients suffered from 40% more freeze than DECADE Integrated P2P live streaming clients.
- Average Peer Uploading Rate: In the test, according to our settings, DECADE integrated P2P live streaming clients had no upload in their "last-mile" access network. More than 70% of peers uploaded much more than streaming rate. That is to say, much uploading bandwidth are waste during data transport.

7. Security Considerations

This document does not contain any security considerations.

8. IANA Considerations

This document does not have any IANA considerations.

9. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

Authors' Addresses

Lijiang Chen
Yale University

Email: lijiang.chen@yale.edu

Hongqiang Liu
Yale University

Email: hongqiang.liu@yale.edu