OSSP: Overlay Streaming Service Platform

Tein-Yaw Chung[1], Huai-Lei Fu[1], Chih-Chen Wang[1], Yung-Mu Chen[1]

[1]Department of Computer Science and Engineering, Yuan Ze University, Taoyuan, Taiwan, R.O.C.

E-mail: {csdchung, supergothere, swswggg, armor}@netlab.cse.yzu.edu.tw

Abstract

Overlay network had received great attention in recent years. Overlay network usually used the peer-to-peer(P2P) networking paradigm for enabling end systems to share their resources. Recent researches had proposed many interesting applications on overlay network such as files sharing and multimedia streaming service. In this paper, we proposed an overlaid p2p network platform to support modern multimedia applications efficiently. This platform so-called OSSP(Overlay Streaming Service Platform) can be exploited for multimedia services include live streaming services, on-demand real-time streaming services, non-real-time transmission services, computation services and data storage services. The architecture of OSSP can be divided into three planes: transmission plane, streamization plane and management plane. With OSSP, we believed that high efficient and networking will be possible.

Keywords: overlay network, streaming services, VOD, peer-to-peer.

1. Introduction

Recently, lots of P2P systems[1] had been proposed to offer video streaming services such as VOD. VOD is a hot research because of its commercial value. Such systems' performance directly is affected by their low-delay startup latency and continuous playback techniques they use. Low-delay startup latency is associated with the satisfaction of users. Naturally, less startup latency makes users more pleased. Continuous media contents playback is another important issue to users. Consumers always hope their VOD systems can play smoothly and enjoy the high quality media contents. In this paper, we propose a service platform, so-called OSSP (Overlay Streaming Service Platform), which integrates the functionalities of data transmission and network management to achieve that purpose.

OSSP defines three planes to simplify the complexity of streaming services: streamization plane, transmission plane and management plane. Streamization plane makes sure that all peers in OSSP can build and maintain streaming pipes for transferring multimedia contents. OSSP slices the media contents into segments. These segments are stored on peers. When a peer requires a video clip to play, the other peers will send their segments which belong to the same video clip to the required peer. We describe the functionality of streamization plane later. Transmission plane is a SSIF(Self-Stable InFrastructure) P2P engine, which provides a stable P2P environment and organizes its own topology dynamically. SSIF constructs and maintains a locality-aware topology to map the physical position of every peer in OSSF for reducing redundant message routing. With locality-aware topology, the performance of SSIF increases. Management plane borrows some ideas from ORMAN[7] to offer capacities that let OSSP easy to manage. These capacities include data aggregating, admission control, services diagnosis, and load sharing. The services OSSP provides have the following characteristics.

1. Scalable: Be scalable to tens of thousands of end hosts.
2. Adaptable: Be adaptable to end hosts with different capability and platform.
3. Efficient: Improve the resources utilization.
4. Stable and self-organized: To support thousands of end hosts, OSSP must require fixed configuration as less as possible and possess self-organizing property while maintaining stable services. Any intervention from service providers must take effect and is seamlessly to other end hosts
5. Manageable: The aggregated status information of end hosts must be monitored and managed. In particular, it helps to discover peers, schedule tasks and allocate resources due to users' requests.
6. Balanced: With idle end hosts, new incoming processes can be executed on it to balance the loads of entire system. OSSP enables the mechanism to deploy the processes to facilitate efficiency and load sharing.

The remainder of this paper is organized as follows. Section II describes the design of OSSP. Section III concludes this paper and discusses the future works.

2. The Architecture of OSSP

OSSP is designed for the following services: live streaming, VOD streaming, and non-real-time transmission service. OSSP includes three planes: transmission plane, streamization plane and management plane. As shown in Fig. 1. Streamization plane is above transmission plane, and exploits management plane to get information for making decisions. Management plane not only offers monitored information to the other two planes, but also manages SSIF.

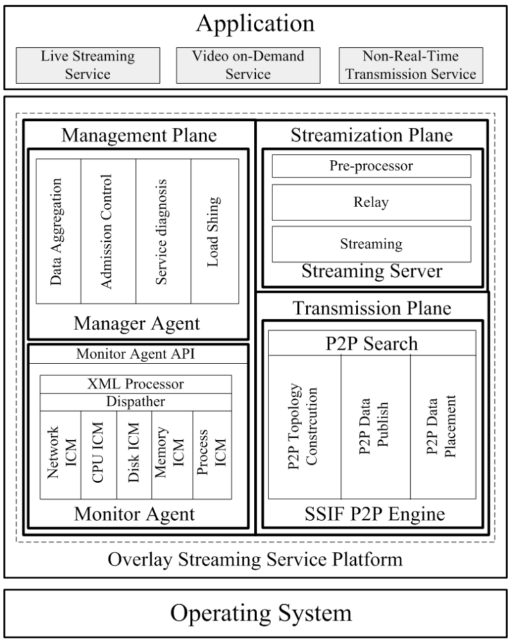

Fig. 1:The architecture of OSSP

A. Transmission Plane

Transmission plane bases on a P2P engine, so-called SSIF (Self-Stable Infrastructure) P2P engine. SSIF is just like CBT[2] in order to achieve locality-aware clustering. Further, SSIF makes an improvement on locality. The dynamic hybrid network positioning(DHNP) mechanism in SSIF, which is a positioning mechanism like GNP[5], makes the peers' locality more precisely. As shown in Fig. 2, a three layered hierarchical topology structure is build by SSIF. The peers in SSIF are classified into regular peers, super peers and cluster leaders. A cluster leader is a powerful host which maintains the inner operations in a cluster. All cluster leaders compose the topology backbone for data and control messages transmission. The super peer is in charge of forwarding messages, which generated by regular peers behind the firewall or in the same LAN. Cluster leaders treat super peers as regular peers, and super peers hide all associated regular peers away cluster leaders.

The data publish subsystem in SSIF uses two kind of content deployment polices: passive deployment and active deployment. The passive deployment policy is the general data sharing model in the P2P system. Peers download the files from other peers and then share those files. In the active deployment policy, the files' owner deploys the data by itself.

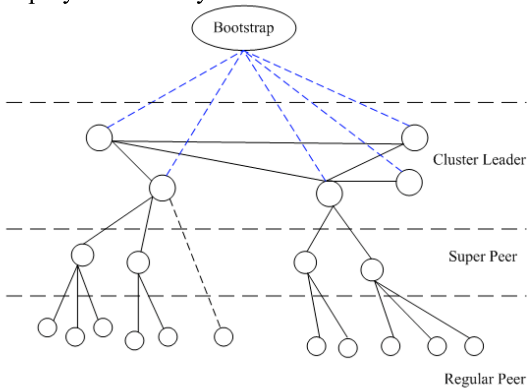

Fig. 2: Three layer hierarchical topology structure in SSIF

The data placement subsystem in SSIF supports four service scheduling schemes: First-Come-First-Serve(FCFS), Capacity Amplification(CA), High-Credit-First(HCF) and Capacity Amplification with Penetration(CAP) is for passive content deployment and active content deployment. In CA scheme, the supplying peer schedules the services by their required capacities such as bandwidth and serving time. In HCF scheme, the usage of reputations or credits encourages peers to share more files and stay online for a longer duration. The supplying peer schedules the service order by their credits. Summarily, transmission plane constructs a robust P2P topology to provide scalable, efficient, stable and self-organized structure. The plane offers critical functions, such as topology construction, P2P searching, data publishing and data placement.

B. Streamization Plane

The major function of Streamization plane is to generate streaming for multimedia content services (videos, audio, documents and other useful resources) in upper-level. The streaming services on P2P network usually suffer from the heterogeneity. For example, the problem arised in [4] refers that the different buffer restricted size in variant peers could cause some unpredictable exception. In OSSP, we introduce a segments-based algorithm into Streamization plane. OSSP divides a video into the proper amount of segments in advance. According to the smoothing algorithm based on [5] and [6], we design a scheme to arrange the transmission order of segments from variant source peer to make sure playing the video streaming smoothly. This scheme is constructed on clustered P2P network. All peers in OSSP are easy to gather these segments and forward segments which are kept in other peers. These features make that every peer in OSSP is a relay node and the workload of video servers is distributed evenly.

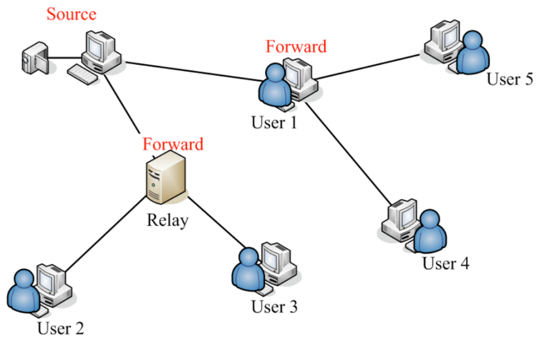

Fig. 3 The relay diagram

As shown in Fig. 3, "User 1" is a forwarding peer to transmit segments from source peer to others. Certainly, we can deploy a dedicated node to take the forwarding job, such as "Relay" in Fig. 3. In OSSP, every peer downloads segments from other peers, but not in the same time. This means each forwarding peer can only store some segments of entire video clip to distribute the workload of video server.

C. Management Plane

In management plane, OSSP defines the two components: manager agent (MA) and monitor agent (MO). MO is required in every member of OSSP to collect monitored data such as CPU utilization and free physical memory size. On the other hand, MA is in charge of aggregating, evaluating, and publishing monitored data from MOs. Namely, MO is a data collector, offering MA the status data of local hosts. MA is an interface to provide applications and administrators for querying and searching functionalities. Also, MA is a key role of maintaining the performance of OSSP. We exploit some concepts from ORMAN[7] to implement MO and MA.

To reduce the management cost and accommodate various heterogeneous operating environments like Linux and Microsoft Windows series, we expect MA and MO have the following features which have been discussed in ORMAN:

1. XML-based Data Processing: this feature enables intuitive data presentation and easy data transformation across various OS platform. Thus, all transmitted messages over OSSP are encapsulated in XML-formatted document. The benefits of XML technology can be found in [8].
2. Modular Architecture: we separate the programming codes which actually collect monitored data into modules called ICM (Information Collector Module). With modular architecture, administrators can build their own module to monitor their services. And, allow the administrator install monitoring modules on-line for new rising services.
3. Remote Control and Deployment Mechanism: with this feature, administrators can control MO and deploy monitoring modules remotely via MA. Thus, MA must trace every associated MO and provide authentication mechanism to filter malicious operations out.

In OSSP, how to decide which peer has a MA is easy. We define each peer has a MO, but only each management peer such as cluster leader or super peer has a MA. A management peer's MA manages MAs and MOs which belong to its associated peers. Besides, a manager peer's MO is controlled by its own MA.

As mentioned above, OSSP uses ICMs to collect peers' status data. In our design, ICMs are components of MO. As shown in Fig. 1, the MO has many ICMs. Each ICM takes charge of collecting different status data of local host. In practice, every ICM should be built for a specified service, not a monitored item. For instance, a webpage ICM which monitors the status of web server process and the dataflow rate of http requests.

Every application based on OSSP can ask the MA providing some useful information to maintain its efficiency. For example, a peer-to-peer VOD (Video On Demand) application needs finding which peer is more powerful to become a relay peer. Namely, MA provides peers' metadata to applications so that applications exploit these data achieving better performance. Generally speaking, MA provides the following services:

1. Data Aggregation: MA gathers and arranges monitored data form MOs, these data can be used in other three services.
2. Admission Control: discriminate which request is admitted for getting service of a specified application.
3. Service Diagnosis: evaluate the state of a specified application service whether healthy or not.
4. Load Sharing: evenly distribute the workload to peers which have unutilized capacity.

Data aggregation is a essential service. It plays a data source. Applications and the other MA's service can get raw data form data aggregation service with their purpose.

Admission control and service diagnosis are both diagnosticians for performance evaluation. Applications import their metrics to MA for activating service diagnosis or admission control service. The main objective of admission control is protecting the reasonable QOS (Quality Of Service) from new incoming requests robbing the resource. When an application gives its metrics to MA for admission control, MA will evaluate the state of this application and determine it whether accepts another request with its metrics. The result will be return to the application. Applications can follow this result to deny the incoming request (if the result is no), or just ignore it (always accept) or record it into logs.

Service diagnosis is a kind of monitoring service. Applications could startup this service to oversee its own status with giving metrics. When MA detects this application is dieing or down, MA will send a notification to the administrator. The way how to notify the administrator is a preconfigured parameter which may be an e-mail address or an ICQ number. MA provides continuous service diagnosis which means keeping monitoring the application, and discrete service diagnosis which means the application must request first for startup diagnosing every time.

Currently, the implementation of OSSP provides Service-scoped load sharing service. Applications overlaid on OSSF could register themselves to MA, and then start service-scoped load sharing service helping them performing efficiently.

3. Conclusion

We don't address some important issues in this paper, like Digit Right Management (DRM) and security issues. For DRM, Microsoft has the solution[9]. Security issues also have solved in JXTA platform[10]. At least, we have implemented a platform built on overlay network to provide streaming service. A user easily put his video on the platform and the others can gain the video conveniently and quickly. The idea of overlay streaming in our platform is the segments-based algorithm. This technology not only serves more peers, but also reduces the video server's load. Then the platform will be more scalable. Management plane monitors peers' status and make the system more efficiently via collaborative mechanisms. In the future, we will construct the DRM mechanism in our platform and inspire commercial applications of OSSP.

4. Reference

[1]. C.K. Yeo, B.S. Lee, and M.K. Er "A survey of application level multicast techniques," *Comput. Commun.*, vol. 27, no. 15, pp. 1547-1568, Sept. 2004

[2]. T. Y. Chung, et al., "Architecture and Implementation of Cluster-based Peer-to-Peer Topology and Its Application in Search", to appear in *J. of Internet Technol.* (JIT), 2005.

[3]. Richard Hsiao and Sheng-De Wang. "Jelly:A Dynamic Hierarchical P2P Overlay Network with Load Balance and Locality." *IEEE ICDCS Workshops 2004*, pp. 534-540, 2004.

[4]. C.L. Chan, S.Y. Huang, et al., "An Application-Level Multicast Framework for Large Scale VOD Services," Proceedings of *the 2005 11th International Conference on Parallel and Distributed Systems* (ICPADS'05)

[5]. S. Sen, J. L. Rexford, J. K. Dey, J. F. Kurose, and D. Towsley, "Online smoothing of Variable-Bit-Rate Streaming Video," in *IEEE Trans. on Multimedia*, Vol. 2, No. 1, pp. 37–48, March 2000.

[6]. L. Gao, Z. –L. Zhang, and D. Towsley, "Proxy-Assisted Techniques for Delivering Continuous Multimedia Streams," in *IEEE/ACM Trans. on Networking*, vol. 11, no. 6, pp. 884–894, Dec. 2003.

[7]. C. C. Wang et al, "An Overlay Resource Monitor System," in *ICACT06*, vol. 3, pp. 1875-1879, Korea, Feb. 2006.

[8]. T. Klie and F. Strauβ, "Integrating SNMP Agents with XML-Based Management Systems", *IEEE Commun. Mag.*, pp. 76-83, July 2004.

[9]. Microsoft DRM, http://www.microsoft.com/windows/windowsmedia/drm/default.mspx

[10]. JXTA , http://www.jxta.org