Faculty of Information And Communication Technology

INCREASING SERVICE QUALITY OF MULTIMEDIA STREAMING USING HYBRID PEER-TO-PEER MODEL

AHMAD TAJUDDIN BIN SAMSUDIN

DOCTOR OF PHILOSOPHY

2010

INCREASING SERVICE QUALITY OF MULTIMEDIA STREAMING USING HYBRID PEER-TO-PEER MODEL

AHMAD TAJUDDIN BIN SAMSUDIN

A thesis submitted

in fulfilment of the requirements for the degree of Doctor of Philosophy

Faculty of Information Technology and Communication

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

2010

ABSTRACT

The peer-to-peer file sharing application has become increasingly popular for Internet users since the 1999 introduction of Napster. In the past decade, several more applications have been invented, such as Gnutella, FastTrack, Chord, Freenet, and BitTorrent. However, all of these applications use the downloading method that requires the source to be downloaded from one or more resources to one requesting peer. Now, alternative methods of file sharing delivery have been introduced; some examples include CELL, CollectCast, DAC, and GnuStream dan PAST, all of which use streaming to deliver media content to the user.

Because peer-to-peer file sharing and streaming systems - such as Video on Demand and IPTV - are popular among Internet users, developers and researchers have a tendency to combine the Video on Demand and peer-to-peer topology into one system. This way, the channel program of Video on Demand or IPTV can be inexpensively distributed on the Internet by utilizing the availability of peers; each peer becomes both a receiver and a supplier to another peer.

Many problems are associated with media streaming of file sharing applications, some of which include query-saturated networks, high latency in locating content, attempting to preserve uninterrupted streaming sessions, high peer load, flash crowds, and bottlenecks. As such, proposed system has been introduced in an effort to reduce the maintenance cost of overlay network topology, the routing and access costs of lookup services, and to service costs of streaming sessions. Proposed system is a combination of pure, client-server, and hierarchical peer-to-peer topology, and is categorized as a hybrid peer-to-peer business model.

Five experiments, consisting of four simulations and an expert evaluation, have been executed to determine the performance, scalability, maintenance, reliability and usability. The results reveal that proposed system can improve the quality of file sharing applications by reducing path length, peer load, and total usage while maintaining the overlay network topology in various churn rates, as well as locating a file in lookup services. The results also show that proposed system has sufficient scalability whenever the network size and number of queries increase.

ABSTRAK

Perisian perkongsian data berasaskan nod-kepada-nod amat popular di kalangan pengguna Internet sejak kemunculan Napster tahun. Kemudian pelbagai protokol dicipta seperti Gnutella, FastTrack, Chord, Freenet, and BitTorrent. Kesemua protokol diatas berasaskan penghantaran data secara muat-turun daripada satu atau lebih sumber kepada satu peminta. Selepas itu ia mula diganti dengan kaedah muat-aliran seperti CELL, CollectCast, DAC, GnuStream dan PAST.

Apabila sistem pengkongsian data nod-kepada-nod semakin popular dan pada masa yang sama permintaan sistem video atas permintaan dan IPTV turut meningkat maka wujud kecenderungan para pengkaji untuk membangunkan perisian yang membolehkan nod berkembang sebagai penerima dan penghantar. Eksplotasi terhadap topologi nod-kepada-nod membolehkan program TV dapat disebarkan melalui talian Internet secara meluas dengan kos rendah.

Didapati sistem pengkongsian data sedia ada masih ada beberapa masalah yang antara lain ialah ketepuan network akibat carian data yang meluas, masa menunggu yang tinggi semasa carian data dilakukan, sering kali sesi muat-lairan terganggu akibat kehilangan nod didalam sistem, beban kerja yang tinggi, capaian yang mendadak terhadap satu nod dan kepincangan tugas. Penyelidik mereka-bentuk sistem hibrid nod-kepada-nod dengan mengabungkan nod-kepada-nod tulin, pelayan-pelanggan dan hirarki dengan tutjuan untuk mengurangkan kos baik pulih topologi rangkaian, gelintar-carian dan perkhidmatan muat-aliran dapat dipertahankan.

Penyelidik telah mengadakan empat eksperimen melalui simulasi berkomputer di makmal, dan satu eksperimen berasaskan penilaian oleh para pengguna tegar sistem nod-kepada-nod. Hasil kajian menunjukkan bahawa sistem yang dicadangkan dapat meningkatkan kualiti perkhidmatan kepada perisian pengkongsian data secara muat-aliran dengan mengurangkan panjang-aliran, muatan-beban dan jumlah pengunaan mesej berbanding sistem nod-kepada-nod yang lain semasa proses carian data dan kekerapan keluar-masuk nod dalam rangkaian. Hasil kajian juga mendapati bahawa sistem memiliki keutuhan anjalan yang baik bila bilangan nod dan permintaan carian bertambah.

ACKNOWLEDGEMENT

In preparing this thesis, I was in contact with many people, researchers, academicians, and practitioners. They have contributed towards my understanding and thoughts. In particular, I wish to express my sincere appreciation to my thesis supervisor, Professor Dr. Nanna Suryana Herman, for encouragement, guidance, critics and friendship. Without his continued support and interest, this thesis would not have been the same as presented here.

I am also indebted to Telekom Research and Development (TMR&D) Malaysia for funding my Ph.D. study. Researchers and librarians at TMR&D and Fern-Universitat in Hagen of Germany also deserve special thanks for their assistance in supplying the relevant support and literatures.

My sincere appreciation also extends to all my colleagues and others who have provided assistant at various occasions. Their views and tips are useful indeed. I am grateful to all my family members.

DECLARATION

I declare that this thesis entitled "*Increasing service quality of multimedia streaming using hybrid peer-to-peer model*" is the result of my own research except as cited in the references. The thesis has not been accepted for any degree and is not concurrently submitted in candidature of any other degree.

Signature	:....................................
Name	:....................................
Date	:.....................................

TABLE OF CONTENT

CHAPTER		PAGE

CHAPTER		PAGE

LIST OF TABLES

LIST OF FIGURES

FIGURE	TITLE	PAGE

LIST OF ABBREVIATIONS

ACS	Acting child supervisor
ANSI	American National Standards Institute
API	Application Programming Interface
ARPA	Advanced Research Projects Agency
ARPANET	Advanced Research Projects Agency Network
ASQC	American Society for Quality Control
ASCII	American Standard Code for Information Interchange
ATA	Approximate Text Addressing
BELUMS	Bandwidth, cEntral processing unit power, packet Loss rate, Uptimes, Memory, and Storage
CAN	Content Addressable Network
CMMI	Capability Maturity Model Integration
CPU	Central Processing Power
CS	Child supervisor
DAC_{p2p}	distributed differentiated admission control protocol
DARPA	Defense Advanced Research Projects Agency
DHT	Distributed Hash Table
DLM	Dynamic Layer Management
DNS	Domain Name System
DRAGON	Dynamic Resource Allocation in GMPLS Optical Networks
DRM	Digital rights management
GB	Gigabytes
GHz	Gigahertz
GMPLS	Generalized Multiprotocol Label Switching
GUI	Graphical User Interface
HADAS	Heterogeneity-Aware Distributed Access Structure
HCR	High Churn Rate
ICT	Information and Communication Technology
IP	Internet Protocol
IPTV	Internet Protocol Television
ISO	International Organization for Standardization
JDK	Java Development Kit
JRE	Java Runtime Environment
Kbps	kilobit per second
LCR	Low Churn Rate
LRU	Least Recently Used
LST	Lightweight SuperPeer Topologies
MB	Megabytes
Mbps	megabit per second
MCR	Moderate Churn Rate
MHz	Megahertz

LIST OF ABBREVIATIONS

MP3	Moving Picture Experts Group-1 Audio Layer 3
MSN	The Microsoft Network
NS	Nucleus supervisor
ODIN	Optical Dynamic Intelligent network
ONT	Overlay network topology
OSI	Open Systems Interconnection
OTS_{p2p}	optimal media data assignment algorithm
P2P	Peer-to-Peer
RAM	Random Access Memory
RFID	Radio-Frequency Identification
RTP	Real Time Protocol
RTT	Round-trip time
SHA	Secure Hash Algorithm
SIAM	Security, Intelligent Application and Multimedia
SWAM-V	Voronoid-based Small-World Access Methods
TCP	Transmission Control Protocol
Test ID	Number of Test
TMR&D	Telekom Research and Development Sdn. Bhd.
TTL	Time-to-live
TV	Television
UDP	User Datagram Protocol
UGC	User Generated Content
VOIP	Voice over Internet Protocol
XOR	Exclusive-OR
XML	eXtensible Markup Language

LIST OF SYMBOLS

Ai	access cost
B	bandwidth
b	number of entries
€	central processing power
C_i	total cost
i	the set of numbers
ID	identity
l	level or height
$\log N$	logarithm for base 2.
Mi	maintenance cost
N, n	number of peers
O	big O or Landau notation
P_{i}	scale of rank from 0 to 100
P_{R}	ranking peer
P_{L}	serving limitation
Ptr	pointer list
$P_{\$}$	storage of peer
P_{LR}	packet loss rate
P_{u}	uptime of peer
R_{a}	available bandwidth
Ri	routing cost
R_{o}	upstream or upload or offer rate
S_{a}	active sender
S_i	service cost
S_{s}	standby sender
w	weightage
x	displacement
U	uptime
M	memory

LIST OF APPENDICIES

CHAPTER 1

INTRODUCTION

1.0 Introduction

File sharing becomes trendier as peer-to-peer applications continue to emerge and provide options for the community to share their content, such as KaZaA, BitTorrent, eDonkey2000, Napster, Gnutella and Freenet. Nowadays P2P technology is used not only for downloading but also for streaming. Researchers have been trying to implement media streaming in a P2P environment, such as CELL, CollectCast, SplitStream, ZIGZAG, PeerStream and GnuStream.

This research study focuses on streaming concepts instead of downloading for file-sharing applications. There is a difference between downloading and streaming techniques, as referred to anon. (2006). First, in streaming, the media plays while it is being transferred; whereas in downloading, files from the peer's storage centre are stored directly onto a local storage disk. So, in a download-based system, the user does not need to be ready to watch a video, whereas in a streaming-based system, the user should be prepared to watch it once the content has been transferred.

Secondly, in a streaming environment, the user can determine if the media is what the user wants after transferring only a few seconds of it, whereas in the download method the user can not do that. Furthermore, it can be disappointing to discover that the content is not right after eight days of a download process, as can happen with a particularly large media file.

Last but not least, the download environment allows users to have a local copy to modify, store and copy to different locations, or to share with others, whereas in streaming an owner can protect the copyright. In this proposed design a streaming concept instead of a download concept has been introduced, but the platform still can be used for other P2P applications.

1.1 Background

P2P systems can be based on an overlay network topology, which is structured, as well as on unstructured systems. Keong *et al.* (2005) and Chawathe *et al.* (2003) defined the structured overlay network topology as tightly controlled and characterised routing the query message to overlay paths as easy, whereas the unstructured system is composed of peers joining and leaving the network with some loose rules, but without any prior knowledge of the topology.

In both structured and unstructured P2P overlay network topologies, there are different possibilities of guarantee to locate data. Gnutella cannot guarantee its ability to locate data, although it uses flooding. Gnutella requires $0(N)$ steps with a limited TTL, whereas the centralized servers of Napster, BitTorrent, SPON, and eDonkey2000 should, in theory, be able to provide the location of data. Meanwhile, the majority of structured P2P lookup services offer DHT, in which DHT can guarantee data location within their overlay network topology. The application translates the exact name of the file into a key and performs a query to match a key over DHT routing information; in example, Chord and Viceroy can locate data in small $O(\log N)$, where N is the number of peers in the network.

Napster is a pioneer in introducing a centralized server. The server will collect all meta-data and provide a location of data to peers. SPON, proposed by Riley and Scheideler (2004a), and Kothapalli and Scheideler (2005), is a tree-based network that manages group updates and supports efficient broadcasting. SPON uses a supervisor peer to maintain the network during node arrivals and departures and routes broadcasts using direct connections between nodes.

BitTorrent has a meta-data collector server, which stores *.torrent* files, as well as a tracker server, which keeps track of the peer activity (in example, download and upload) and availability of the pieces of files. In this P2P system, the response time of lookup services is constant, and the resulting set of queries are in multiple files and peers (online and offline). Unfortunately, the BitTorrent server does not take care of any contents deleted from the system. In other words, the central server still stores the *.torrent* file, although that particular file does not exist anymore in the network.

It is no wonder that eDonkey2000 also implements the idea of a centralized server. Unfortunately, on September 28, 2005, eDonkey2000 officially closed its doors due to illegal downloading. eDonkey2000 received a cease-and-desist letter from the Recording Industry Association of America as a result of the June 2001 Supreme Court ruling that makers of software that facilitates copyright infringement are liable for that infringement (Veiga, 2006).

The P2P researchers believe that, in order to be a success in P2P applications, systems need to protect the content, the content owner, the peer itself and also sharing activities. Researchers should consider implementing the appropriate protection mechanism into P2P file-sharing applications.

Many previous studies show that researchers tried to take advantage of the different characteristics of peers such as offer rate, available bandwidth, buffer, storage, memory, uptime, etc., to offer better quality of downloading or streaming process.

Gnutella proposed Ultrapeer topology, in which the peer itself becomes a server. Ultrapeers perform extra work, however, compared to a normal peer. They have to receive multiple meta-data and query requests from connected peers. To become an ultrapeer, the peer should be able to meet the constraints; not be firewalled, and have a suitable operating system as well as sufficient bandwidth, uptime, RAM and CPU speed.

KaAza also introduced a similar ultrapeer called the SuperPeers. The difference is that the peers can nominate themselves as SuperPeers, although their connection speed is 128kbps or the CPU speed is 445 MHz! Kleis *et al.* (2005a; 2005b) propose LST which is designed to have lower complexity and management overhead than structured P2P overlay network topology. Another study conducted by Xiao *et al.* (2005) shows that the quality of a SuperPeer system is significantly improved under the DLM algorithm. In DLM, every peer decides to be a SuperPeer or leaf-peer independently without global knowledge.

There are a few systems proposed that use multicast technology to support streaming to a large number of receivers simultaneously, and their intention is to stream from one single-source to many peers. This technique can be used to improve the quality of media streaming, such as IPTV and video on demand. Xu *et al.* (2002), ZIGZAG (Tran *et al.*, 2003), GridMedia (Zhang *et al.*, 2005), and Split-Merge (Kulkarni and Markham, 2005) are using peer-to-peer topology, in which these systems utilize the existence of nodes to receive and distribute the packet stream to other nodes. ZIGZAG is concerned about the unpredictable nature of peer behaviours and resource limitations at the receiver peer. In Split-Merge, they take into consideration the upstream capacity.

Cai *et el.* (2005) proposes CELL mechanism to utilize the peer buffers, in which the supplying and requesting peer will determine whether the requesting peer should be able to cache some of the segmented video files. CELL uses Gnutella-like lookup services and stops the query once the system finds one caching host. Freenet (Clarke *et al.*, 2000) also implemented caching; if a file is found during lookup service, and the file is successfully retrieved by the original requester, then the file will cache on sequence upstream requester.

In CollectCast applications, the Hefeeda *et al.* (2003) use offer rate and available bandwidth as an indicator of peer goodness. They utilise modified Tapestry lookup services, and return one or more supplier peer. Then, CollectCast does a peer selection based on offer rate and available bandwidth over topology-aware selection techniques. Thus, two groups of peers will be created: active sender and standby sender. CollectCast dynamically switches active senders and standby senders, so that the collective network performance out of the active senders remains satisfactory.

In this part of the research, the researcher focuses on the distributed and supervised peer, how to rank the peers, file delegation and dynamic standby peers. The proposed design uses the advantages of the server-client business model to search and retrieve information. To make sure there is no flash crowd or excessive access by users, the proposed design needs to improve the delegation policy. To provide a dynamic standby peer, the system uses a content hash key value as an indicator to search for an identical content in the P2P network. Proposed designs consider the selection of the best peer to serve a streaming session as a crucial step in providing a good quality video stream.

1.2 Problem statement

1.2.1 Traffic, peer load and latency

The main problem of pure P2P systems is to assign and locate resources among peers. P2P systems differ in the way in which they construct the overlay topology and distribute queries from node to node. The advantage is that they can easily accommodate a highly transient node population. The disadvantage is that it is hard to find the desired files without distributing queries widely.

In an unstructured and pure P2P, when one node joins or leaves the network, does it have to alert the other peers about its activities? If not, how do the other peers know this peer is in their group?

If one closely looks into the majority of structured and pure P2Ps, system provide DHT, from which a much better lookup service is provided within their overlay network topology, whereas in hybrid P2P, the challenge is to figure out the mechanism for organizing the peers that allows them to cooperate, provide a metadata index and, finally, enable any peer to resourcefully locate contents.

However, structured designs are likely to be less resilient in the face of a very transient user population, precisely because it is hard to maintain the structure of overlay network topology required for routing to function efficiently when nodes are joining and leaving at a high rate.

Kothapalli and Scheideler (2005) said that server/client-based systems can provide guarantees and are, therefore, preferable for critical applications that need a high level of reliability; but the peer-to-peer systems can scale to millions of sites with low-cost hardware whereas the classical approach of using server-client systems does not scale well unless powerful servers are provided. BitTorrent is one of an example of the server-client search approach, where the latency of lookup services is constant and the peer load is minimal. Unfortunately, BitTorrent does not take care of any deleted contents from the system or the status of the peer itself. The server also seems to work alone and not to communicate or replicate with other servers.

1.2.2 Preserve quality streaming session

The majority of P2P applications do not deploy a proper delegation process. For example, in BitTorrent, every peer has a right to search for particular data in P2P networks. The requesting peer has a right to get all results without any filtering. If that peer shares a popular file, then the owner will face a problem and his computer will always be connected with other computers. As a consequence, his available bandwidth will drop, peer load will increase and, finally, it will not be able to preserve the quality of media while streaming.

CollectCast does a peer selection based on the offer rate and available bandwidth over topology-aware selection technique. Thus, two groups of peers; for example, active and standby senders will be performed. System dynamically switches active senders and standby senders, so that the collective network performance out of the active senders remains satisfactory. However, the standby sender in CollectCast is static. If a new peer joins the network and has that file, it will not take part as a standby sender.

1.2.3 Bottleneck and flash crowd

If a server in a hybrid P2P system is down due to maintenance or bottleneck, the system faces the problem of assigning and locating the resources. This problem also occurs in a server/client-based system. Sometimes, when a thousand clients rapidly try to connect to the server for media downloading, the flash crowd symptom always arises (in example, the CNN portal on September 11, 2001).

Actually, pure P2P also exhibit the flash crowd when without a proper delegation, for example a thousand requesting peers try to download a popular file from the same source. Sometime it also occurs at node within the chain of lookup service especially system used uni-path random walk such as Freenet.