

Research and Implementation of a Scalable P2P-C/S hybrid architecture and system

(Work-in-progress)

Peng Jian

School of Computer Science, Sichuan University, chengdu, Sichuan, P.R.China, 610064

penguest@163.com

Zhang Daping

School of Computer Science, Sichuan University, chengdu, Sichuan, P.R.China, 610064

daping_zhang@163.com

Li Zheng

School of Computer Science, Sichuan University, chengdu, Sichuan, P.R.China, 610064

lizheng@cs.scu.edu.cn

ABSTRACT

The scalable P2P-C/S hybrid architecture is put forward to combine the advantage of two models, namely P2P and C/S. By changing the degree of participation of the server and the peers, it can ease the burden of SourceNode or improve the expansibility and stability of the system. BTPublisher is designed to be an effective, reliable and stable file distribution system based on the popular BitTorrent protocol. In BTPublisher, SourceNode is introduced to construct a scalable P2P-C/S hybrid system for file distribution.

Categories and Subject Descriptors

C.2.4 [Computers-Communication Networks]: General-Distributed applications H.3.5 [Information Storage and Retrieval]: Online Information Services-Data sharing

General Terms

Design, Performance, Reliability

Keywords

P2P, C/S, BitTorrent, scalable file distribution, SourceNode

1. Introduction

On the Internet, most data is stored as files, and for file distribution there are two models, that's Client/Server(C/S) and Peer to Peer (P2P). The two models have their own characteristic as well as limitations, and none of them can meet requirements of file distribution perfectly. In this paper, the scalable P2P-C/S hybrid architecture is put forward to combine the advantage of these two models, namely P2P and C/S, by changing the degree of participation of the server and the peers, which can ease the burden of SourceNode or improve the expansibility and stability of the system. BTPublisher is designed to be an effective, reliable and stable file distribution system based on the popular BitTorrent protocol which takes advantage of the ability of locating and

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

INFOSCALE 2007, June 6-8, Suzhou, China
Copyright © 2007 ICST 978-1-59593-757-5
DOI 10.4108/infoscale.2007.948

downloading resource quickly. In addition, in BTPublisher, SourceNode is introduced into traditional P2P network to construct a scalable P2P-C/S hybrid architecture for file distribution.

2. P2P-C/S hybrid architecture

2.1 Overview of the traditional single model

In C/S model, there's a centric server in the system. As a result, the extensibility is limited and the system will be attacked easily, the server will turn out to be the bottleneck of performance. P2P technology will be a candidate in this situation. The decentralization character of P2P system provides infinite expansibility in theory.

The model used by BitTorrent is similar to Napster. Both of them have a centric server (in BitTorrent system it is called tracker), which provides resource location service. As a result, it can be classified as a centralized system. However, BitTorrent is a file distribution system but Napster is a traditional file shared system.

Generally, the BitTorrent provides advantages including swift resource location and high download speed. Whereas, it has some problems as below:

Firstly, all nodes play the same role in P2P network and lack of uniform access control mechanism which make the system difficult to be managed and controlled.

Secondly, P2P system is open and all nodes in the system are heterogeneous and the quality of service (QOS) is not controllable.

Finally, all nodes including the seed node can join or quit the system at any moment which introduces instability to the system and finally lead to broken links. For instance, BitTorrent is effective in resource downloading, but file published turns to be broken link frequently as the resource only be active in a short period.

2.2 The scalable P2P-C/S hybrid architecture

To overcome shortcomings illuminated above, this paper presents a scalable P2P-C/S hybrid architecture. In this model, the system composes of two parts: the server and the peer. The architecture of the system is illuminated as figure 1.

The role of the server is the same as that in C/S model which has high performance and abundant network bandwidth. Besides,

the server will not only be one computer, with CDN and cluster technology, its serving ability can also be expanded arbitrarily.

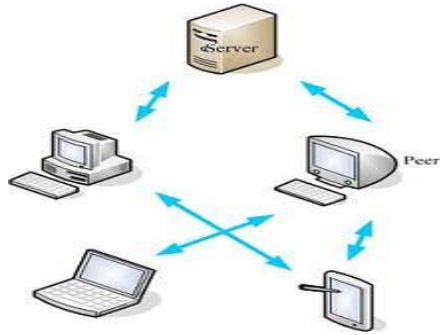


Fig. 1 P2P/CS hybrid model

The role of the peer is the same as the role of the node of the P2P networks, and the nodes communicate with each other with one peculiar protocol, for performing peculiar works such as data transferring. However, as one special node, the server is provided by peculiar people, its performance and bandwidth can be satisfied.

In the hybrid model of P2P/CS, because of the dual system features from both P2P and C/S, there are three statuses in the procedure of system performance based on the participation degree of the server.

- a) The server providing data alone (S_0): When the system starts running, the number of nodes in the system is small, and the scale of the network is very small. Here the system is a typical C/S structure. Because of the small number of the nodes, there is no too much burden to the server.
- b) The server and the peer providing data at the same time (S_1): With the increase of the node number, the scale of P2P network formed with these nodes is increasing too. With communication, the nodes can exchange data with each other. However, because the data from the server has not been dispatched to the system and the only source of the integrated data is the server, so the server must continue the data transferring. In the meantime, both server and peer provide the data in the system, and the system is with hybrid status.
- c) The server does not provide data, and only the peer provides data (S_2): With more and more nodes join in the network, the scale of the network is increasing gradually, and the whole copies of files are completely distributed in peers. The server may not provide data to system, and it can release server resource. Here the system is a typical P2P structure.

The status transferring from S_0 to S_2 , the system transition from C/S structure to P2P structure can alleviate the press of the server effectively. By contraries, the status transferring can also be from S_2 to S_0 which can satisfy the reliability and the stability. The status transferring is illuminated in figure 2.

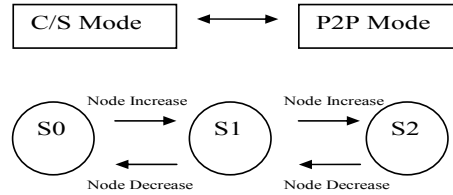


Fig. 2 Status transferring

With the P2P/CS hybrid model, the system utilizes the high extensibility of the P2P network, overcomes the performance bottleneck of the typical C/S structure, and with introducing the server into P2P network, the system obtains the improved reliability and stability. In the mean time, many current applications in Internet are based on typical C/S mode, and the service providers purchase many servers with high performance. Utilizing the P2P/CS hybrid model can make use of these devices, and protect the known investment effectively.

3. The BTPublisher system architecture

3.1 BTPublisher system architecture

As a P2P System, peers in BitTorrent System can arbitrarily join in and quit the network. Using the “least first” algorithm, the system can distribute pieces of file to the network quickly. However, the system can not confirm that all pieces of the file have distributed to nodes in the network. A single lost piece will cause all nodes that don't hold it fail to download the integrated file, and then lead to a broken link.

In addition, when BitTorrent system right after running, only the seed node contains the integrated resource and the download speed will be low due to its low bandwidth and poor performance. This phenomenon is called “slow startup”. Besides, without many nodes in existence at the same time, the advantage of the multi-source download of the BitTorrent system can not be utilized effectively. Furthermore, other nodes in the system provide only limited upload bandwidth. Consequently the download speed is lower than the traditional C/S system.

To solve these problems, this paper presents an effective and stable file distribution system, that's BTPublisher, which is base on the BitTorrent protocols and takes advantage of its abilities of swift resource location and high speed downloading.

BTPublisher consists of three parts, namely Tracker, SourceNode, and Peer, and its architecture is illuminated in figure 3.

- **Tracker**

Just as the one in BitTorrent system, the tracker takes charge of resource locating task for the whole system. In the tracker, a map will keep all the download information of the running system, in which file identifier (FileID) is as the key, and records of basic information (for example, the IP address and the service port) of those nodes downloading the same file is as the value. When the tracker receives a download request, it will return proper information according to FileID posted by the request.

Besides, the tracker in BTPublisher will communicate with SourceNode to obtain its real-time status information, with which the tracker decides that it should return

information of the SourceNode to request nodes or not and finally achieve the dynamic balance of reducing SourceNode's loading and augmenting data accessibility.

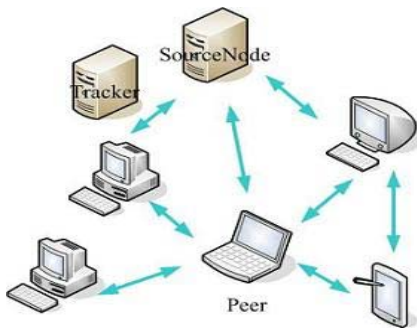


Fig. 3 BTPublisher Architecture

- **SourceNode**

It is the first seed of the system. The most difference between BTPublisher and BitTorrent is that the former has a dedicated publisher, that's the maintainer of the system, but the latter often has multiple publishers at the same time. Moreover, In BTPublisher, resource is published by the maintainer, which uploads the resource to the SourceNode and meta-information file, with which other nodes can initialize the downloading.

There will be multiple nodes directly connect to the SourceNode during the progress of downloading. The SourceNode provides data as well as logs information of the downloading progress, such as which piece has been downloaded by which node.

In addition, there is a module to monitor the status of SourceNode, which provides real-time information to the Tracker.

- **Peer**

Peer in BTPublisher system use the "least first" algorithm to select the file pieces to download and use block algorithm to maintain the justice of the system. When a peer downloads a piece of file completely, it can provide upload service to other nodes. After a node downloads the file completely, it will become a seed as long as it continues uploading, which is encouraged by the BTPublisher system.

Whereas, peer in BTPublisher can not publish the resource initially, that is, the peer will never to be the first seed of the system.

3.2 The running phase of system

As a file distribution system, BTPublisher users can be divided into two categories, that's file downloader and file publisher.

- **File publish**

As the system starting, file publisher has two tasks to do: firstly, file publisher needs to upload the data to be published to the SourceNode, which then becomes the first seed of the system.

Secondly, file publisher should produce a meta-information file (.torrent) according to the data file and store it in a web server to make it available to other peers. Besides, file publisher must register corresponding information to the Tracker.

- **Nodes join**

Peer downloads the meta-information file (.torrent) from the web server, which contains the URL of the Tracker and other information. Peer connects to the Tracker, gets the nodes list which contains all nodes downloading the same file. And then, peer can connect to the nodes in the list and communicate with each other and download data file.

- **Nodes quit**

When a peer quits the system, nodes that are connecting to it, can detect this event and delete the information of node that just quit from its node list. When the Tracker detects there are no seeds in the system (that is, the SourceNode is invalidation), the file information will be deleted form the record list.

4. The design and implement of the BTPublisher system

4.1 The design of system function module

As the three main entities in the system, the Tracker, the SourceNode and the normal Peer may play different roles and functions, but they usually work under the similar workflow model, which is receiving messages, parsing messages, processing messages, and returning the response, that is, the operations of the whole system are driven by messages. Therefore, the system can be partitioned into the several essential modules as followed shown in Figure 4: master control module, message management module, network module, user interface module, file read-write module, performance-monitoring module.

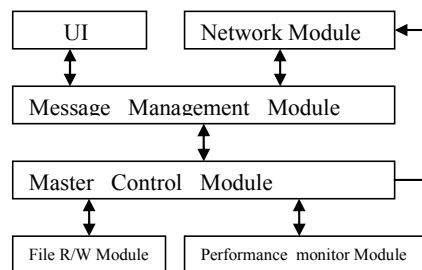


Fig. 4 system module

- **Master control module**

Master control module is the key point to the operations of the BTPublisher, is the most important portion of the Tracker, the SourceNode and the Peer in the system.

Established as the chief control center, the module takes the responsibility for collecting and recording the peer's status, then selectively invoking the corresponding module to process the receiving message and to complete the intended task.

The main function of the control module in the tracker is recording the status of the SourceNode in time. Once receiving the Announce message from the peer, the tracker will select the optimal algorithm. The module makes a decision about the download status, and realizes the conversion of the system download models from P2P model to C/S model.

In the SourceNode, the function of control module is that, when the node receives the request message from the peer, immediately deciding whether the node returns the data to the request peer according to the distribution status of the pieces of

file. If the SourceNode decides to return data, it will invoke the file read-write module to get the corresponding file data.

For normal node, the function of control module is, to register the status information about the downloading file (such as pieces have been downloaded and download speed, etc.) and to get the piece information in the remote nodes. Then it can schedule according to these information using right arithmetic.

- File write/read module

The goal of the file distribution system is to copy files in the SourceNode to other peers, so the write/read module is important. Especially when popular peers download the files, they also provide the upload service for other peers. Therefore, how to store files efficiently affects the system performance directly. In addition, the buffer mechanism can decrease the read-write frequency, and then can make the hard disk hurt least.

4.2 The implementation of the system

The system BTPublisher chooses C++ as the implementation languages. WinSock 2.0 API was used as network program API in the BTPublisher system, the asynchronous socket model provided by WinSock 2.0 API is more fit for message-driven system, and BTPublisher uses the third party code tinyxml to decode and encode the XML files.

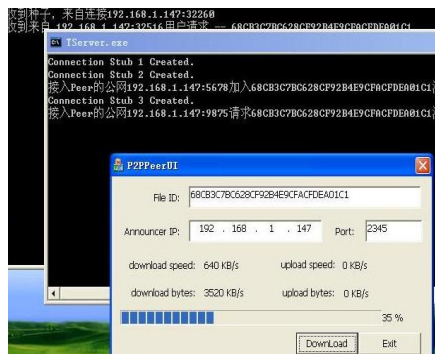


Fig. 5 single peer to download file

After filling the Announcer (provide the meta-information data for the peer) and the IP and ports of the Tracker in, users can

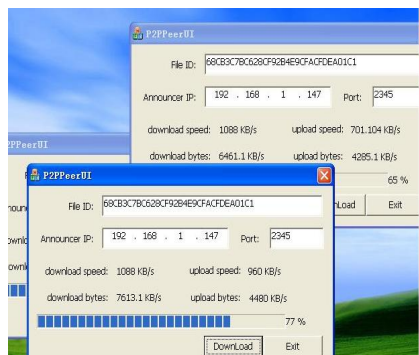


Fig.6 multi peer to download file

publish the file. If published successfully, there would be corresponding record in Announcer and Tracker. The IP in this instance is 192.168.1.147.

After published the file successfully, the user can download the file by using the downloading client, as shown in Fig. 5.

It can be seen, at the very beginning, with the increasing of the normal download peers, peers can exchange data each other, to increase the download speed notably as shown in the Fig. 6. Furthermore, as not all the data provided by the SourceNode peer, it decrease its load obviously.

5. Summary

This paper discussed and realized the BTPublisher system with the scale P2P-C/S hybrid architecture, which is improved on the basis of the popular BitTorrent system which represents the file distribution of the P2P model. Because the BTPublisher is improved on the basic of the popular BitTorrent by which the ability of quick resource location and download is provided, and the SourceNode is introduced in the system, it can effectively improve the stability of the system, increase the speed of the download during the initialization, and enhance user's experiments.

6. REFERENCES

- [1] V. Padmanabhan, H. Wang, P. Chou, and K. Sripanidkulchai, "Distributing streaming media content using cooperating networking," in NOSSDAV, 2002.(CoopNet)
- [2] Rob Sherwood, Ryan Braud, and Bobby Bhattacharjee, "Slurpie: Acooperative bulk data transfer protocol," in Infocom, Hong Kong, HK, 2004, IEEE..
- [3] D. Kostic, A. Rodriguez, J. Albrecht, and A. Vahdat, "Bullet: High bandwidth data dissemination using an overlay mesh," in Symposium on Operating Systems Principles (SOSP), 2003.
- [4] Christos Gkantsidis and Pablo Rodriguez Rodriguez. "Network Coding for Large Scale Content Distribution" in Infocom 2005, IEEE