MODIFIED BITTORRENT PROTOCOL AND ITS APPLICATION IN CLOUD COMPUTING ENVIRONMENT

Soumya V L[1] and Anirban Basu[2]

[1]Dept of CSE, East Point College of Engineering & Technology, Bangalore, Karnataka, India
Soumya8.vl@gmail.com
[2]Dept of CSE R&D, East Point College of Engineering & Technology, Bangalore, Karnataka, India
abasu@anirbanbasu.in

ABSTRACT

BitTorrent is widely used as a protocol for file sharing and has gained popularity due to its scalability and performance advantages. This paper proposes a modification to the BitTorrent protocol for Peer-to-Peer communication [1]. The proposed method has been implemented and performance measured on an experimental set up. Results reveal the performance gains that can be obtained. The paper also discusses how the modified BitTorrent protocol can be used to improve Peer-to-Peer communication in a Cloud computing environment.

KEYWORDS

Peer-to-Peer communication, Network traffic, BitTorrent Protocol, Performance.

1. INTRODUCTION

BitTorrent protocol used for Peer-to-Peer file distribution is receiving a lot of attention in computer networking. It enables fast downloading of large files using minimum Internet bandwidth. In traditional Client/Server architecture [6] the Client communicates only with the Server for all information needs but as the number of Clients increases performance degrades due to increase in the Server Load[8]. BitTorrent protocol overcomes this drawback and maximizes transfer speed by gathering pieces of the file required and downloading these pieces simultaneously from Peers who already have them. This process reduces downloading time of very large files, such as videos and television programs more than that possible with other protocols. The paper proposes modification to BitTorrent protocol to make it still faster by allowing them to download files from other Peers without having the complete file. The paper discusses the BitTorrent protocol in Section 2 and its modifications in Section 3. The modified protocol has been implemented and performance measurements carried out on an experimental set up as described in Section 4. The performance improvements are discussed in Section 5 of the paper. Experiments show how the response time progressively decreases with increase in the number of Clients but Server Load remains the same even when the numbers of Clients increase. Section 6 of the paper discusses application of the modified BiTorrent protocol for data transfer in a Cloud environment.

DOI : 10.5121/ijcsea.2012.2503

2. Peer-to-Peer Communication using BitTorrent Protocol

Normally, communication in a network is based on Client/Server architecture, where communication between a Client and the Server follows unicast (point-to-point communication) method i.e., the communication takes place only between the Server and Client as shown in Figure 1.

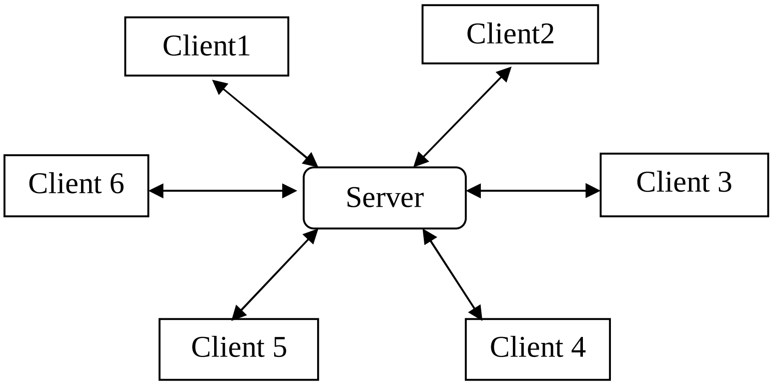


Figure 1. Client/Server Communication

If the network contains a large number of Clients, the Server may fail to respond to the entire Clients' requests due to heavy load on the Server. Moreover when Clients are distributed far and wide, the network traffic increases substantially and becomes a performance bottleneck [8]. As the number of Clients increase, the load at the Server also increases. Inappropriate distribution of resources can lead to increase in traffic resulting in network congestion and possible packet loss [4]. In such cases, Peer-to-Peer communication may be advantageous as all data transfer between Clients need not pass through a central Server.

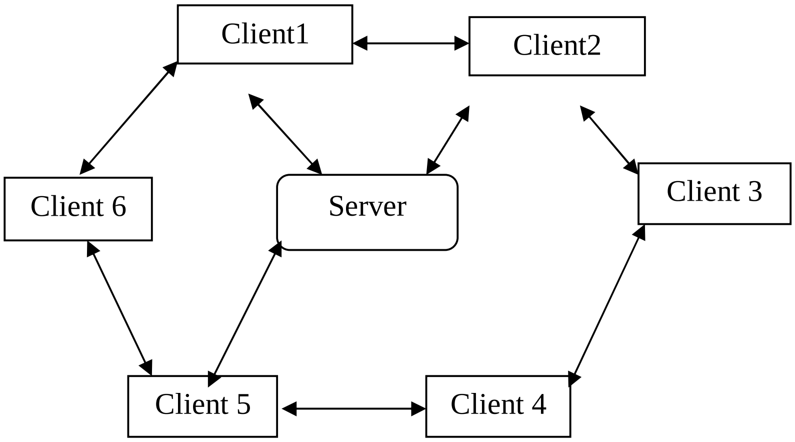


Figure 2. BitTorrent Protocol Scenario

In Peer-to-Peer (P2P) communication, Peers allow its resources, such as processing power, disk storage, network bandwidth, and files to be shared and make them available to other network Clients, without the need for any central coordination [11]. Peers behave both like suppliers and consumers of resources. In contrast to the traditional Client–Server model, BitTorrent protocol establishes communication between Peers. BitTorrent is a fast and efficient protocol for distribution of files over a network and uses file sharing technology to transfer large files directly between Clients' computers. This protocol has the advantage of reducing the traffic on Server and on communication links when distributing large files. Rather than downloading a file from a single Server, the BitTorrent protocol allows Clients to join a cluster of Clients to download files from each other simultaneously. This protocol has advantages over all the other protocols proposed for Peer-to-Peer communication as discussed in [4]. BitTorrent protocol allows Clients in a network with low communication bandwidth to distribute files to other recipients more efficiently.

The important entities involved in BitTorrent Protocol are:

- Peers or Seeders (Clients).
- Leecher
- BitTorrent Tracker
- Server
- Torrent File

Peers: These are Clients which are ready to share files with other Clients in the network. They are also called Seeders.

Leechers: These are the nodes in a network who do not want to share its files with other nodes.

BitTorrent Tracker [9], [10]: The Tracker resides in the Server and contains information that include:

1. The IP addresses of the Peers which are connected to the Tracker.
2. The hash number associated with each file segment or piece. (The file is fragmented into number of segments called "pieces").
3. Hash numbers of the file segments which have already been downloaded from Peers (using SHA-1 Hashing algorithm). This improves security by comparing the hash number of each segment with the original hash number of the entire file.

Tracker keeps track of all the Peers which are connected to Server. Tracker does not allow any unauthorized Client to communicate with Server thereby ensuring security. Tracker also enables the Clients to download files from Server and also from other Clients in the network.

Torrent File: This file contains information about the files to be shared in a network i.e., metadata about the files. This file contains hashes for each segment before sharing it with other Clients. It also contains the information (IP Address) about the Tracker.

3. Modified BitTorrent Protocol

Even though the BitTorrent protocol improves efficiency of a network by sharing files between Peers, it has the disadvantage that a node can share a file with other nodes in a network only when it has the complete file with it, which slows down the performance. To overcome the drawbacks of the BitTorrent protocol it is modified as follows:

1. Incorporating Parallel Downloading: a Client can download segments (of size 1 KB) of a file from a Seeder and from a Server in parallel.

2. Allowing a Peer or Seeder to share a file segment with other Clients in a network even though it does not have the complete file as soon as it downloads the segment from Server.

3. Implementing Centralized Tracker, for enhanced security instead of Decentralized Tracker. In the Decentralized Tracker Peer does not know the identity of the Client authorized to share resource with it and does not provide adequate security.

3.1 Parallel File Sharing using modified BitTorrent Protocol:

The following steps for parallel file sharing using modified BitTorrent protocol is explained for three Clients and a Server:

- Client1 sends a request to the Server for a file. Then torrent file is created for the requested file in the Server and it downloads the torrent file.
- At the same time, if Client2 requests for the same file which is being downloaded by Client1, Client2 starts downloading file segments from both Server and from Client1 in parallel and as Client1 shares its file with Client2 it acts like a Peer or Seeder.
- When Client3 requests for the same file, which is in Client1 and in Client2, it starts downloading simultaneously from Seeder1 (Client1), Server and from Seeder2 (Client2).

4. Test Bench for Performance Analysis

Performance Analysis of the modified BitTorrent Protocol for Peer-to-Peer communication was carried out by implementing on a Test Bench with one Server and three Clients (Client1, Client2, and Client3) connected in a LAN. This Test Bench differs from a normal LAN environment, as Clients do not belong to the same domain. In the setup, a Client can share files with another Client only when sharing between Clients is enabled by the Network Administrator. The experimental setup of Test Bench consisting of Clients (Peers), Tracker, and Server is shown in Figure 3.

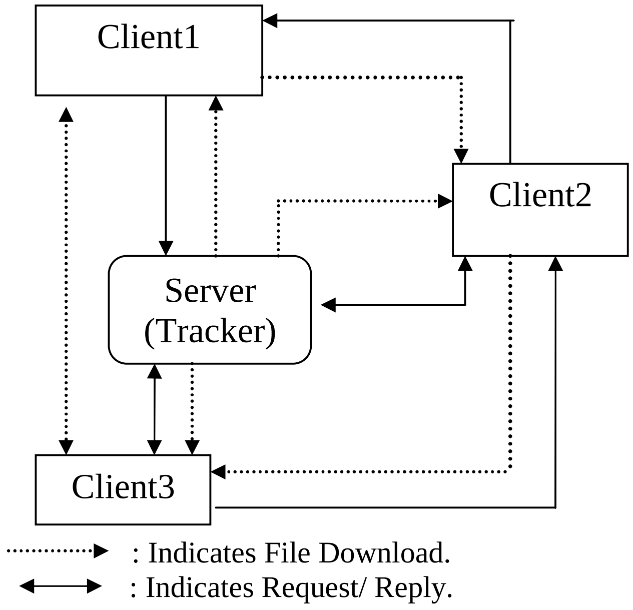
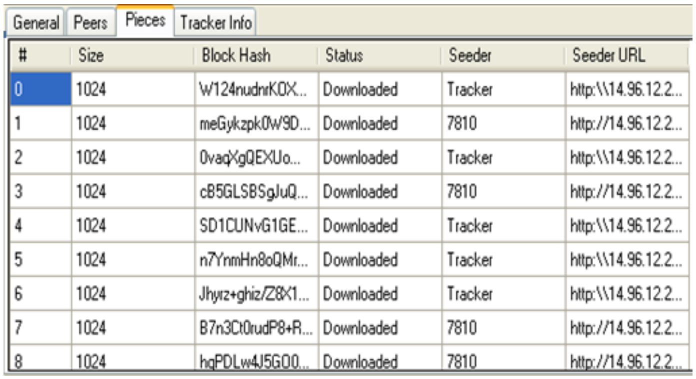


Figure 3. Experimental Set up

Figure 3 shows Client2 and Client3 downloading file segments in parallel from a Server and from other Peers. Initially Client1 gets file from Server and when it is downloading the segments, it can share these segments with other Clients who are requesting for these. Initially Client2 requests for a file from Server, and then Server sends details of the other Peer(s) who are having the same file. Then Client2 starts downloading segments from both the Server and Peer simultaneously. In this set up Client2 gets details of Client1 before it starts downloading segments from Client1 as well as from Server. The same procedure is followed for Client3 which downloads file segments from Client1, Client2 and also from Server in parallel.

5. Experimental Results

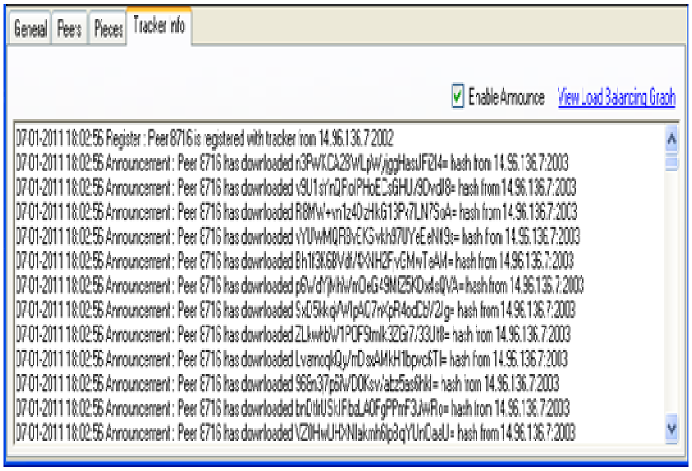
Experiments conducted on modified BitTorrent Protocol are discussed in this section using My_Music.torrent file, a medium size C# file.



#	Size	Block Hash	Status	Seeder	Seeder URL
0	1024	W124nudnrKOX...	Downloaded	Tracker	http:\\14.96.12.2...
1	1024	meGykzpk0W9D...	Downloaded	7810	http://14.96.12.2...
2	1024	0vaqXgQEXUo...	Downloaded	Tracker	http:\\14.96.12.2...
3	1024	cB5GLSBSgJuQ...	Downloaded	7810	http://14.96.12.2...
4	1024	SD1CUNvG1GE...	Downloaded	Tracker	http:\\14.96.12.2...
5	1024	n7YnmHn8oQMr...	Downloaded	Tracker	http:\\14.96.12.2...
6	1024	Jhyrz+ghiz/Z8X1...	Downloaded	Tracker	http:\\14.96.12.2...
7	1024	B7n3Ct0rudP8+R...	Downloaded	7810	http://14.96.12.2...
8	1024	hqPDLw4J5G00...	Downloaded	7810	http://14.96.12.2...

Figure 4. Parallel Download of file segments by Peers

Figure 4, shows the information displayed on the screen of the Test Bench during parallel downloading of file segments by different Clients in the network. It displays hash numbers of the file segments which are downloaded by different Peers along with their corresponding URL. It also shows the status of download, and size of each file segment. Parallel download is carried out to reduce the download time at the Clients.



☑ Enable Announce

```
07-01-2011 18:02:56 Register : Peer 8716 is registered with tracker from 14.96.136.7:2002
07-01-2011 18:02:56 Announcement : Peer 8716 has downloaded n3PwKCA28vLpWjggHasJFiZl4= hash from 14.96.136.7:2003
07-01-2011 18:02:56 Announcement : Peer 8716 has downloaded v9U1sYnQFolPHoECsGHU/9Dvdl8= hash from 14.96.136.7:2003
07-01-2011 18:02:56 Announcement : Peer 8716 has downloaded R8MW+vn1z4OzHkG13Px7LN7SoA= hash from 14.96.136.7:2003
07-01-2011 18:02:56 Announcement : Peer 8716 has downloaded vYUwMQR8vEKGvkh97UYeEeNI9s= hash from 14.96.136.7:2003
07-01-2011 18:02:56 Announcement : Peer 8716 has downloaded 8h1f3K68Vdl/4XNH2FvGMvTeAM= hash from 14.96.136.7:2003
07-01-2011 18:02:56 Announcement : Peer 8716 has downloaded p6WdYjMhWnOxG49NIZ5KDx4s0VA= hash from 14.96.136.7:2003
07-01-2011 18:02:56 Announcement : Peer 8716 has downloaded SxD5kkqWlpAC7nKpR4odCb/2Jg= hash from 14.96.136.7:2003
07-01-2011 18:02:56 Announcement : Peer 8716 has downloaded ZLkwkbW1POF9mlk32Gi7/33Ut8= hash from 14.96.136.7:2003
07-01-2011 18:02:56 Announcement : Peer 8716 has downloaded LvamoqkQyJnDsxAMkH1bpvc6TI= hash from 14.96.136.7:2003
07-01-2011 18:02:56 Announcement : Peer 8716 has downloaded 96Sn37p6WD0Ksv/abz5as6hkl= hash from 14.96.136.7:2003
07-01-2011 18:02:56 Announcement : Peer 8716 has downloaded bnDtkUSklFbdLA0FgPPmF3JwRo= hash from 14.96.136.7:2003
07-01-2011 18:02:56 Announcement : Peer 8716 has downloaded VZ0HwUHXNlakmh6p3qYUnCasU= hash from 14.96.136.7:2003
```

Figure 5. Tracker information

As shown in Figure 5, the Tracker gives information of the file which is downloaded by the Client. It also gives the IP address of the Client computer from where files are downloaded.

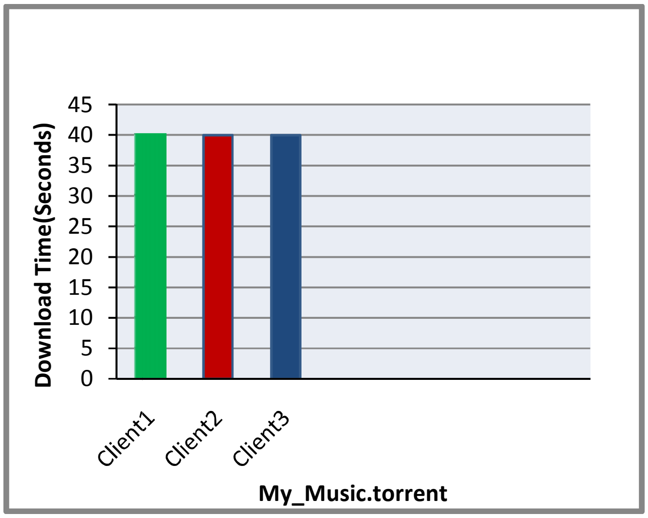


Figure 6. Download time of Clients using BitTorrent Protocol

Figure 6, shows the download time of all Peers/Clients using existing BitTorrent protocol. For downloading the file My_Music.torrent file Client1, Client2, and Client3 takes the same time i.e. 40 seconds each. Even after implementing Peer-to-Peer communication, download times of all Clients remain the same, but it reduces the Server load. When Client2 and Client3 request for the file from Client1, load at Client1 increases as it has to provide file to both the Clients requesting at the same time. Congestion at that node may occur, resulting in performance degradation.

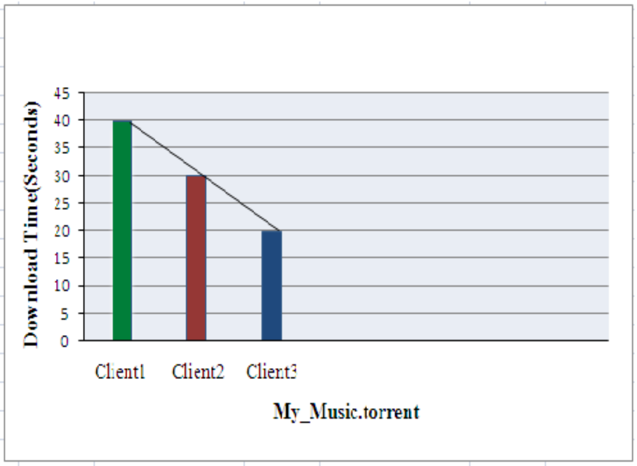


Figure 7: Download time of Clients using Modified BitTorrent protocol.

Figure 7 shows how the Download Time is reduced by using modified BitTorrent Protocol in Peer-to-Peer communication. Client1, Client2, and Client3 take 40, 30, and 20 seconds respectively. This is because, Client2 starts downloading file segments from Client1 as well as from Server in parallel. Therefore Client2's download time is less compared to Client1. The same is the case for Client3. It is found that with the application of modified BitTorrent Protocol download time of Clients reduces with the increase in the number of Clients. It does not cause congestion at a particular Client (here Client1).

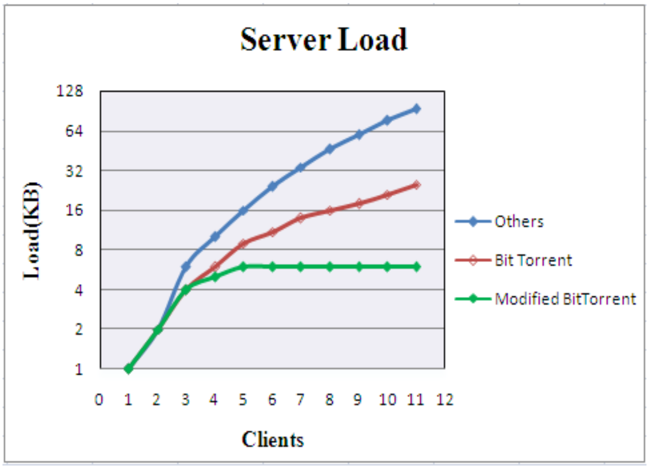


Figure 8: Server Load

Figure 8, shows the effect of application of Modified BitTorrent protocol on Server load. Initially, the load at Server increases gradually and then remains constant throughout the communication even with increase in number of Clients wanting to communicate with the Server. But in other methods the load at Server increases with increase in number of Clients.

6. APPLICATION OF MODIFIED BITTORRENT PROTOCOL IN CLOUD ENVIRONMENT

Although Cloud computing is rapidly emerging as a new paradigm for delivering IT services on “pay -per -use” basis, successful implementation of Cloud computing needs solution to many performance related issues. The important entities involved in Cloud computing are:

- Cloud Server: Cloud Server (CS) or Cloud Service Provider (CSP) which is a pool of resources in a data centre and provides huge processing power and variety of services to the Clients.

- Clients (Peers): They demand services from CS/ CSP. The services may include computing power, storage resources, applications, and processes.

While Cloud computing looks very promising, providing the services from the Cloud Server to the all Clients within a short time continues to be a challenge, as Cloud environment can have a large number of Clients. Performance degradation may occur due to heavy load at Server. To overcome this problem, modified BitTorrent Protocol discussed in this paper can be used to share large number of files including video, and audio files among peers. When a Client requests for a particular file, the Cloud Service Provider normally provides the requested file, but sometimes CSP may fail to provide requested service due to heavy congestion. In such a situation, Peer-to-Peer communication can be used and Client can seek a file from another Client/Peer instead of from CSP. Taking an entire file from a single Peer can increase load at Peer, hence parallel downloading is preferred, so that, some of the file segments can be downloaded from a Peer and some from CSP in parallel. Download time of each Client can be reduced substantially. Figure 9, shows how Peer-to-Peer Communication works using modified BitTorrent protocol in a Cloud Environment.

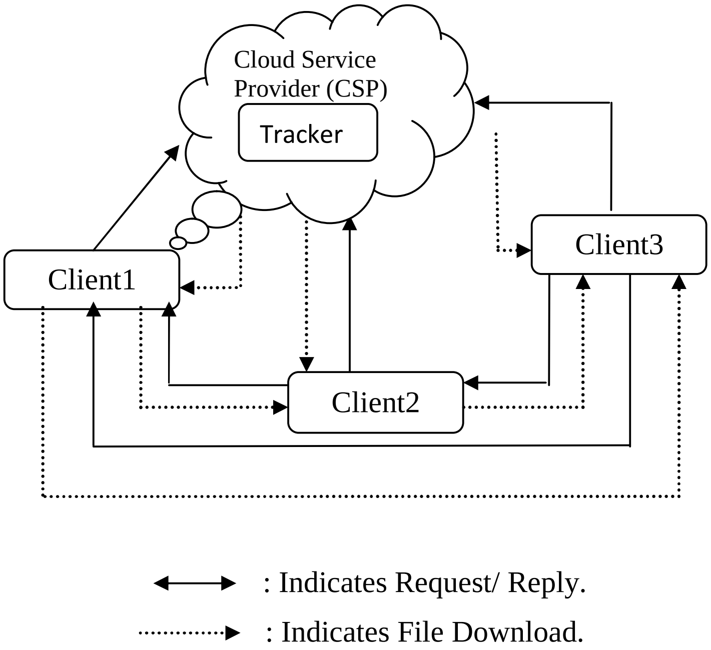


Figure 9. Peer-to-Peer Communication Using Modified BitTorrent in Cloud Platform.

Modified BitTorrent Protocol can be used in Cloud Environment to distribute the Virtual Machine Images efficiently without causing any congestion in the compute nodes. The torrent file is created in the Tracker. A Client/ Peer which needs VM Image starts downloading from the Tracker as well as from other Peers which is having VM Image in Parallel. Hence VM Image can be distributed to the Clients within a short period of time, without substantial increase in network traffic and packet loss.

7. CONCLUSIONS

In this paper, BitTorrent protocol which is widely used in network communication has been discussed and modifications suggested for speeding up file transfer. The modified BitTorrent protocol has been implemented on a test bed and performance analyzed which shows improvement that can be obtained. The paper has also discussed the advantages of using the modified BitTorrent protocol in a Cloud Computing platform to improve the performance.

8. REFERENCES

[1] Kasame Tritrakan and Veera Muangsin, “Using Peer-to-Peer Communication to Improve the Performance of Distributed Computing on the Internet”, Proceedings of the 19th IEEE International Conference on Advanced Information Networking and Applications (AINA’05), 2005.

[2] Armbrust M, A. Fox, R. Griffith, A. D. Joseph, R. H. Katz, A. Konwinski, G. Lee, D. A. Patterson, A. Rabkin, I. Stoica, and M. Zaharia, “Above the Clouds: A Berkeley view of Cloud computing”, Technical Report No. USB-EECS-2009-28, Feb 2009. pp 1-23.

[3] Bin Fan, John C. S. Lui,, and Dah-Ming Chiu,” The Design Trade-Offs of BitTorrent-Like File Sharing Protocols”, IEEE/ACM Transactions on Networking, Vol. 17, No. 2, April 2009. pp 365-376.

[4] Matthias Schmidt, Niels Fallenbek, Matthew smith, Bernd Freisleben, “Efficient Distribution of Virtual Machines for Cloud Computing”, 18th Euromicro conference on parallel, Distributed and Network-based Processing 2010. pp 568-574.
[5] Peeyush Mathur, Nikhil Nishchal, “Cloud Computing: New challenge to the entire Computer Industry”, Proceedings of the 1st International Conference on Parallel, Distributed and Grid Computing (PDGC - 2010), pp 223-228.
[6] Raymond Lei Xia and Jogesh K. Muppala, “A Survey of BitTorrent Performance”, IEEE Communications, Surveys and Tutorials. Vol. 12, No. 2, Second Quarter, 2010. pp 140-158.
[7] Shuai Zhang, Shufen Zhang, Xuebin chen Huo, Xiuzhen Huo, “Cloud computing Research and Development Trends”, IEEE Second International Conference on Future Networks, 2010. pp 93-97.
[8] Rajiv Ranjan, Liang Zhao, Xiaomin Wu, Anna Liu, Andres Quiros and Manish Parashar, “Peer-to-Peer provisioning: Service Discovery and Load Balancing”, Springer-Verlag London Limited 2010, pp 197-217.
[9] http://www.cs.uiowa.edu/~ghosh /bittorentprotocol.com.
[10] Soumya V L, Test Bench for Performance Measurement using Modified BitTorrent protocol, EPCET Internal Report, May 2011.
[11] Kai Hwang, Geoffrey C Fox, Jack J Dongarra, (2012) “Distributed and Cloud Computing”, Elsevier India Pvt. Ltd..Chapter 8

AUTHORS

Soumya V L is working as a Lecturer in Department of Information Science & Engineering, in Sir M Visvesvaraya Institute of Technology, Bangalore. She is an M.Tech (Software Engineering) from East Point College of Engineering & Technology, She has presented papers in various conferences and her research interests are in Cloud Computing and IOT. Ms. Soumya V L is a Life Member of Computer Society of India and is currently pursuing PhD research

AnirbanBasu is an MTech in Electronics and a Masters and PhD in Computer Science with more than 30 years experience in Academia, advanced R&D, Software Industry, Consultancy and Corporate Training. Presently he is Professor and Head, CSE-PG Studies in East Point College of Engineering and Technology, Bangalore and a PhD Supervisor of several universities and guides research on Software Engineering and Cloud Computing. He also consults to IT companies on Software Engineering and Software Quality. He started his career in Indian Statistical Institute in 1979, and joined CDAC in 1989 and played a key role in developing India’s first Supercomputer PARAM. From 1995 to 2002, he worked in Siemens Information Systems Ltd., Computer Associates etc. in senior/ top management positions. Dr Basu is a recipient of a number of awards, which includes the 1985 Canadian Commonwealth Scholarship in Computer Science, Computer Engineering Division Gold Medals of The Institution of Engineers (India) etc.. He has published about 80 research papers in reputed national and international journals and in conference proceedings and has co-authored five books on Advanced Computing, Parallel Processing, Technology Policy, Software Quality and Cloud Computing. Dr AnirbanBasu is a Fellow of The Institution of Engineers (India) and a Senior Member of IEEE (USA), Senior Member of Computer Society of India.