# An Empirical Study of Free-Riding Behavior in the Maze P2P File-Sharing System

| Mao Yang | Zheng Zhang | Xiaoming Li | Yafei Dai |
|---|---|---|---|
| Beijing University | Microsoft Research Asia | Beijing University | Beijing University |
| ym@net.pku.edu.cn | zzhang@microsoft.com | lxm@net.pku.edu.cn | dyf@net.pku.edu.cn |

## Abstract

*Maze[1] is a P2P file-sharing system with an active and large user base. It is developed, deployed and operated by an academic research team. As such, it offers ample opportunities to conduct experiments to understand user behavior. Embedded in Maze is a set of incentive policies designed to encourage sharing and contribution. This paper presents an in-depth analysis of the effectiveness of the incentive policies and how users react to them. We found that in general the policies have been effective. But they also encourage the more selfish users to cheat by whitewashing their accounts as a variation of Sybil attack. We examine multiple factors that may contribute to the free-riding behavior. Our conclusions are that upload speed, NAT and amount of shared files are not the problems, and selfish behavior is demonstrated more by shorter online time. Since free-riders are also avid consumers of popular files, we suggest a two-pronged approach to reduce free-riding further: mechanisms to direct queries to sources that would other wise be free-riders, and policies to encourage users make their resource more available.*

## 1. Introduction

Maze[1] is a peer-to-peer file-sharing application that is developed and deployed by an academic research team. Maze is similar in structure to Napster, with a centralized, cluster-based search engine, but is additionally outfitted with a social network of peers. This hybrid architecture offers exact keyword-based search, simple locality-based download optimizations, and also reduces dependency on the central cluster. Maze has a set of evolving incentive policies which, complemented by direct user feedbacks via forum, discourage free-loading, a problem plaguing many similar networks. More details of the Maze architecture are available in [2][3].

Maze is in its 4th major software release, and is currently deployed across a large number of hosts inside China's internal network. As of October 2004, Maze includes a user population of about 410K users and supports searches on more than 150 million files totaling over 200TB of data. At any given time, there are over 10K users online simultaneously, and over 200K transfers occurring per day.

Maze provides an excellent platform to observe many important activities inside the network and some of our measurement results have been reported in [2]. In this paper, we focus on the reputation and incentive aspects of the Maze architecture. We found that, in general, the incentive policies are effective to encourage contribution. However, one consequence is that free-riders start cheating by account whitewashing. The fact that the free-riders are avid consumers of popular contents should have made them the sources of contributors. However, the slow updating of the Maze central indexing makes it harder to direct queries to these users. Looking at the free-riding behavior further, we found that one of the more direct measurements of the selfish degree is the online session time: free-riding users usually stay only one-third as long as the server-like users. Although 40% of users are behind firewall, NAT is generally not the source to blame, nor is the upload speed. However, high upload speed and not being hindered by firewall are advantageous for motivated users to contribute.

The roadmap of the paper is as follows. Section-2 gives a quick overview of its architecture. Experiment methodology is described in Section-3. Section-4 is the main body of this paper, where we take a closer look at the incentive policies and the free-riding behavior in Maze. Secition-5 contains related work and we conclude in Section-6.

## 2. Maze Architecture Overview

Maze grew out of the need to address the downloading problem of the FTP part of a Web search project called T-net[4]. As the service became popular, the limited number of FTP servers has led to degrading performance. The first step of Maze is to allow parallel downloading from peer users. Each peer will authenticate itself to the Maze central server, and upload the index of the files that it has in its local Maze directory. Each

[1] This work is partially supported by NSFC grant 90412010.

peer sends periodical heartbeats to the central server as well. This allows full-text queries to be conducted over the set of the online peers. Maze fans out the requests to multiple sources to download different chunks of the file, with simple locality hint that gives priority to peers that share more prefix of the initiator's IP address.

Recognizing that we eventually need to reduce the dependencies upon the central server, Maze in addition let each peer to have several peer lists. The first is the "friend-list," which is bootstrapped from the central server with a list of other peers when the user first registered, and can be modified later on. Frequently, the user adds those who have satisfied her queries before. The second is the "neighborhood-list," which contains a set of online peers that share the B-class address. Finally, Maze gives a small list of peers who currently have high reputation scores as an incentive to reward good sharing behaviors. A peer can recursively browse the contents of the Maze directories of any level of these lists, and directly initiate downloading when they find interesting contents. These lists form the bases over which we plan to add P2P search capabilities.

A NAT client can download from a non-NAT client, or another NAT client behind the same firewall. However, it can not download from a NAT user behind a different firewall.

Maze has an evolving set of incentive policies designed to discourage free-loadings. This is the subject of the rest of the paper and therefore we defer its discussion until then.

Maze also has an associated online forum. This is where many discussions among the users take place, and is also the venue that Maze developers gather feedbacks. Our experience has proven that this forum is invaluable.

## 3. Experiment Methodology

The Maze5.04 release we issued on September 26th has a component to report their download behavior includes the source or sink of the transfer, file type, file size, file signature (MD5) and the transfer bit rate. The central servers also log the following information per client: online time, IP address and network information (such as NAT or non-NAT), the files shared, the change of the user's reputation point, and finally the register information. Table 1 gives the summary of the logs. Unless otherwise stated, results are analyzed using logs from 9/28 to 10/28. We use mysql to process these logs.

**Table 1: summary of log information (9/28~10/28)**

| | |
|---|---|
| Log duration | 30 days |
| # of active users | 130,205 |
| # of NAT users | 51,613 |
| # of transfer files | 6,831,019 |
| Total transfer size | 97,276GB |
| Average transfer file size | 14,240KB |
| Average transfer speed | 327,841 bps |
| # of unique transfer files | 1,588,409 |

## 4. Reputation and Incentive Mechanism

In this section, we will start by describing the Maze incentive policies, and then look at its overall impact overtime. Then we will focus on the free-riders, followed by a more detailed analysis of possible courses of free-riding.

### 4.1 The Maze Incentive Policies

In Maze, we use an incentive system where users are rewarded points for uploading, and expend points for successful downloads. The rules are:

1. New users are initialized with 4096 points.
2. Uploads: +1.5 points per MB uploaded
3. Downloads:
   - -1.0/MB downloaded within 100MB
   - -0.7/MB per additional MB between 100MB and 400MB
   - -0.4/MB between 400MB and 800MB
   - -0.1/MB per additional MB over 800MB
4. Download requests are ordered by $T = requestTime - 3\log P$, where $P$ is a user's point total.
5. Users with $P < 512$ have a download bandwidth of 200Kb/s.

This point system was discussed in the MAZE forum and agreed-upon before implemented. It was designed to give downloading preference to users with high scores. These users add to their request time a negative offset whose magnitude grows logarithmically with their score. In contrast, a bandwidth quota is applied to downloads of users with lower scores (<512). Although the quota seems to be high, it is consistent with our observation that a large number of users have access to high-bandwidth links. Finally, while we encouraged uploads and deducted points for downloads, we recognized that the majority of bytes exchanged on Maze were large multimedia files, and made the download point adjustment graduated to weigh less heavily on extremely large files. For instance, the user will spend all the start points if she downloads 4K MB files for 1MB size files, 5.3K MB files for 400MB files, or 7K MB files for 800MB files.

Our policies award at least 50% more points for uploading than downloading. This is based on our belief that the contributing users should earn more rights to download. For instance, when a user has uploaded 267MB files, he will earn enough points to download 628MB files. Therefore, those who contribute contents shall see their points increase quickly. On the other hands, if a user downloads more than uploads, his score will decrease over time, and will eventually drop to so low as he will be deprived of the right of conducting any downloading. Since the number of downloads and uploads are equal, the total points of the entire Maze population will grow. For the time being, we do not believe this is an issue.

For convenience of discussion, we will define the *server-like* and *client-like* users for those users whose points are above and below their initial point (4096), respectively. As of 10/28, the ratio between these two classes of users is 4.4:1. We found that client-like users are responsible for 51% downloads but only 7.5% uploads. These statistics suggest the existence of free-loading. Figure 1 depicts the CDF curves of number of upload and download activities against user reputation scores. Our reputation metrics has reflected the user behavior in general.

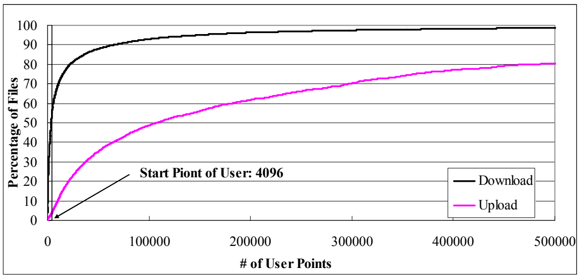

**Figure 1: CDF distribution of uploads and downloads against user reputation scores up to 500000.**

One of the top score users that we interviewed share out many popular course materials, which are video files of various formats and are large enough to earn points quickly. The motivation for a Maze user to be server-like is primarily to gain social status in the community, rather than earning points to download. There seems to exist a self-enforcing cycle that propels the riches get richer.

A set of good incentive policies should have the net effect of moving users towards more sharing behaviors. Since its very first release, Maze has the incentive policies in place. However, before 5/20 of 2004 (the release date of Maze3.02), the policies are quite different. For each MB of transfer, a download will deduct one point, whereas an upload will add one point. Furthermore, the scores are not used in anyway as to enforce the QoS measures that this new set of policies do. The new policies were extensively discussed in the Maze online forum, and officially launched in May. Over the period of several months, we are able to gather the scores and observe the effects.

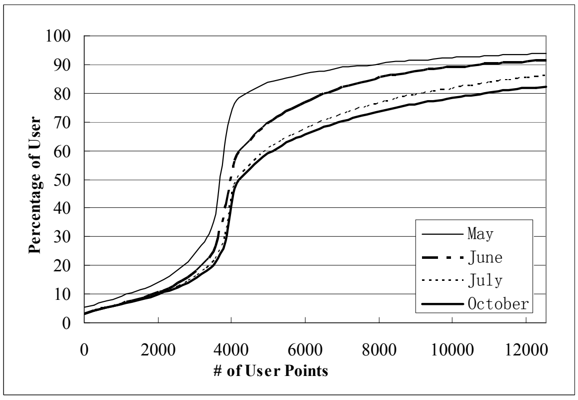

**Figure 2: CDF distribution of user points up to 12000 for the month of May, June, July and October.**

The impact of the policies is best understood with the upload and the download logs, which are only available after the Maze5.04 release of 9/26. The complete information that we have are the reputation scores, which are kept on the Maze central server. Figure 2 shows the changes of the reputation CDFs from May till October. There are around 20~30% of users who stay at their starting points (4096); these are the registered but inactive users. These set of curves are difficult to analyze because, as we mentioned earlier, the total point of the system continue to increase and thus the “center of the gravity” shall move towards right unless there are absolutely no activities. However, we do believe that the policies are effective to some extent. For instance, the proportion of client-like users decreases from 93 % in May, to 30% in June, to 22% in July and finally to 19% in October. Also, if the policies were ineffective to change user behavior, the client-like users shall see their point totals drop quite rapidly. This does not happen. In the future, we will collect more statistics to study this aspect.

### 4.2 The Free-Riders

For simplicity of discussion, we will call the client-like users the free-riders. When a free-rider sees her point drops, she has several choices. For instance, she may start to aggressively promote himself. Indeed, we have found that once a request for content was posed on the forum, it is soon followed by many invitations – typically from those with low points – to advise the availability of the content. There are several things a user can do to cheat the system. One route he might pursue, for instance, is simply to leave the system and re-enter with a different Maze user ID. These are the *whitewashers*. Whitewashers[2] can be detected, but we currently do not ban them. If a user has several Maze accounts, he can mount the more elaborate Sybil's attack [5] by downloading among these accounts to earn credits for each one of them. We know for a fact that these behaviors exist, and are investigating how much fraction they account for.

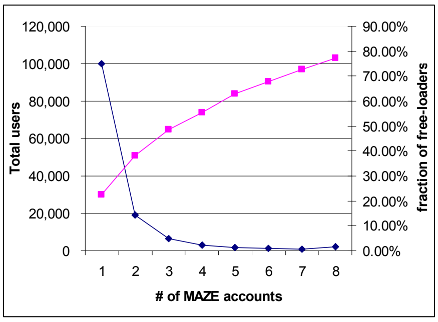

**Figure 3: The distribution of user over number of accounts they have owned and the proportion of free-riders.**

Figure 3 plots the breakdown of user population according to how many different user IDs they own from the time that they first registered. The last bracket includes whitewashers with 8 or more. We are surprised to find that a tiny fraction of whitewashers went so far as to own up to 23 different user IDs, although the majority of the Maze user have only one user ID (75%). We have verified that those who have owned multiple IDs typically spend their points completely before registering a new ID.

One would expect that there is a strong correlation between owning multiple IDs and free-riding behaviors. Our result shows that this is indeed the case. The percentage of free-riders increases steadily with number of user IDs. Within the category of one user ID, there are only 22% free-riders, whereas for those that owns 8 IDs or more, this percentage increases to 77%.

### 4.3 Understanding the Source of Free-Riding

The only way that the free-riders can survive the Maze system *without* cheating is through contribution. Since the free-riders account for the majority of download activities, they will quickly own many of the popular items as well. For the period of 9/28~10/28, we found that the top 10% popular files account for more than 98.8% of total transfer traffic, and over half of which were downloads from the client-like users. Therefore, they can easily make back their deficits provided that 1) the Maze system can quickly direct queries to them and 2) their contents are available.

The first factor is hindered by one of the artifacts that challenges the scalability of Maze recently. Because the Maze central server has limited power, as more and more contents become available, we have to slow down the indexing process. On October 8th, 4 out of the 10 Maze central index servers were decommissioned because of bad hard disks. This exacerbates the situation even further. Currently, new content of a peer does not make into the index until a few days later. Complemented with friend/neighborhood-lists and the high-reputation users that Maze recommends, this has not made searching for popular items too difficult. It is difficult to quantify how this affects the low-point peers to earn back their scores until we perform detailed simulation to see how many free-riders can become download targets if the index is always up to date. However, we believe that this is indeed a factor. We are replacing the bad indexing servers. Still, a more complete solution is to implement the P2P searching in the future releases. Since popular contents spread out quickly, P2P search will allow more download sources to be discovered at a timely fashion.

Even if a user downloaded a popular object, he may choose to move the file out of his Maze partition. The study in [6] shows that 70% of Gnutella

[2] This is not entirely true, especially for users who have one home PC and one work PC. Because the difficulty of merging the contents of these two sources, some users elect to register once for each PC that they own.

users do have any files to share. This is clearly *not* the case in Maze. Figure 4 shows the distribution of total files shared out versus users' reputation score. In fact, the average number of shared files of client-like users is 491, versus 281 of the server-like users. It is logical to infer that these users also contain a good portion of interesting files.

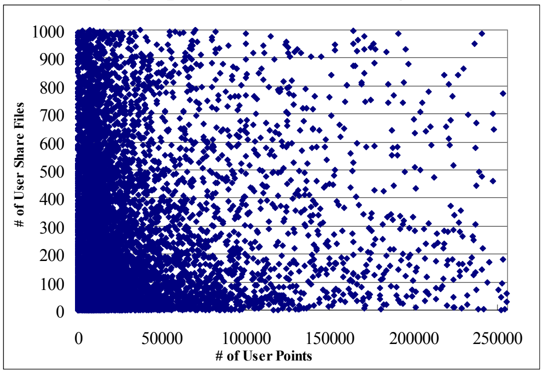

Figure 4: number of shared files versus points

Besides the fact that the slow updating of the central index delays queries to be sent to these potential sources, there are many other reasons. For instance, the user may choose to turn off the Maze server or shut down the machine, either due to resource constraints or selfish behavior. Figure 5 depicts the correlation of the user session time and users reputation of 65K randomly picked users. Overall, users with positive point changes have longer session time, on average 2.89 times more than those with negative point changes (218 minutes versus 75 minutes). The figure also shows that there are users who have earned high points and then stopped contributing and only perform downloading.

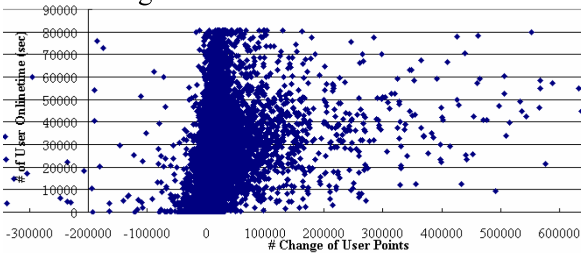

Figure 5: online session time versus point changes

Even when queries have been directed to a Maze user, there are other factors that could make her earn points less aggressively. A Maze client employs parallel downloading from all sources that the index server advices. A source with higher upload bandwidth (and machine power as well) will account for higher proportion of the file being downloaded, and hence is advantageous to earn more points. Figure 6 draws the scatter graph of the effective upload speed versus the change of reputation points. The effective upload speed is the average upload speed weighted over the transfer size.

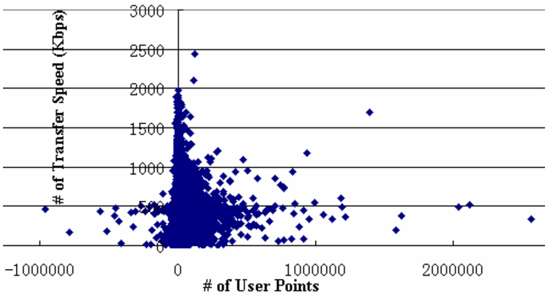

Figure 6: upload speed versus point change

The users with negative point changes and those with positive point changes up to 30K have similar upload speed around 310kbps. However, those with changes above 30K have upload speed more than 400kbps. Thus, upload speed makes a difference for those users want to earn high points, but is not a significant factor for the free-riders.

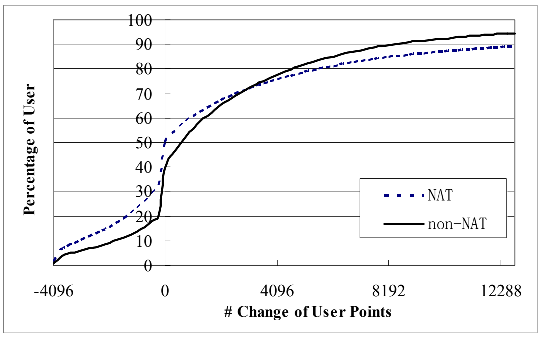

Figure 7: CDF of point changes for NAT and non-NAT users.

There is also the problem of NAT. We have found that 40% of Maze users are behind firewall. Thus, 16% of upload can not happen because the source and the sink are behind different firewalls. This does not in general make the NAT problem an issue, since there are still 84% of chances that a NAT user can upload to a non-NAT user and therefore earn points. In fact, when we look at the point change distribution of the NAT versus non-NAT users (Figure 7), we found that there are more low score NAT users than low score non-NAT users. It is true, however, that the non-NAT users are easier to earn higher points. In other words, although there is significant number of NAT users, firewall traversal is an important but not deciding factor in the free-riding behavior. Notice the sharp drop of both curves at the -4096 point, these correspond to users that have spent all their initial quota and either leave the system or enter again by re-registration.

In summary, the selfish behavior has been demonstrated more by reducing online session time than other factors. In particular, neither the upload speed nor being behind the firewall can be legitimate excuses. On the other hand, high upload speed and/or not being hindered by the NAT issue are necessary for motivated users to contribute.

## 5. Related Work

There are many works on incentive policies. Due to space constraint, we can not include all of them. Many of the works [9][10] focus on modeling, for which the empirical data we obtained would be useful. In terms of measurement studies, [6] was the first study that pointed out the degree of free-riding in Gnutella. Our data confirms the effect but shows that free-riding is not as pronounced in Maze. Our incentive policies could be one of the reasons.

John Douceur [5] proved that if distinct identities for remote entities are not established either by an explicit certification authority or by an implicit one, these systems are susceptible to Sybil attacks. We believe that incentive policies will not remove these attacks. Quite the contrary, it might actually encourage that, as proven by the whitewashing behavior in Maze, simply because this is an easier way out for the selfish users. The centralized registration in Maze makes it possible to counter these attacks.

Several measurement studies have characterized the properties of peer-to-peer file-sharing systems [7][8]. Some of our other experiment results match what these studies have found. However, this paper focuses on free-riding and the contributing factors.

## 6. Conclusion and Future Work

This preliminary study on the free-riding behavior in the Maze system has yielded a few interesting insights. First of all, the incentive policies have been effective in general, but they are circumvented by free-riders using account whitewashing. We have examined several factors that could contribute to the free-riding behavior.

We are reasonably confident to reduce the free-riding behaviors further. Since popular contents dominate the sharing activities, we should be able to devise mechanisms and policies to spread the load more easily. As we discussed earlier, this entails two different aspects: direct queries to sources that would otherwise become free-riders, and to ensure that contents are available when queries do arrive. The first is the responsibility of the query and search mechanism, and we can accomplish it by installing P2P searching mechanism and/or increase the frequency of updating the central index. The second is simply human nature, and the only way to influence that is through more savvy incentive policies (e.g. encourage people to increase their online session durations).

## References

[1] http://maze.pku.edu.cn.

[2] Mao Yang, Ben Y. Zhao, Yafei Dai and Zheng Zhang. “Deployment of a large scale peer-to-peer social network”, Proceedings of the 1st Workshop on Real, Large Distributed Systems

[3] Hua Chen, Mao Yang, et al. “Maze: a Social Peer-to-peer Network”. The International Conference on e-Commerce Technology for Dynamic e-Business (CEC-EAST’04). Beijing, China. September, 2004.

[4] http://e.pku.edu.cn.

[5] John Douceur. “The Sybil Attack”. In Proceedings of the 1st International Workshop on Peer-to-Peer Systems, pages 251–260, Boston, MA, USA, March 2002.

[6] E. Adar and B. Huberman. “Free Riding on Gnutella”. October, 2000.

[7] S. Saroiu, P. K. Gummadi, and S. D. Gribble. “A measurement study of peer-to-peer file sharing systems”. In Proceedings of Multimedia Computing and Networking (MMCN) 2002.

[8] Krishna P. Gummadi, Richard J. Dunn and et al. “Measurement, Modeling, and Analysis of a Peer-to-Peer File-Sharing Workload”. Proceedings of the 19th ACM Symposium on Operating Systems Principles (SOSP-19), Bolton Landing, NY.

[9] http://p2pecon.berkeley.edu.

[10] C. Buragohain, D. Agrawal, and S. Suri. “A game theoretic framework for incentives in p2p systems”. In Proc. 3rd Intl. Conf. on Peer-to-Peer Computing, 2003.