Peer-to-Peer workload characterization: techniques and open issues

Mauro Andreolini
University of Rome “Tor Vergata”
andreolini@ing.uniroma2.it

Michele Colajanni
University of Modena and Reggio Emilia
colajanni.michele@unimore.it

Riccardo Lancellotti
University of Modena and Reggio Emilia
lancellotti.riccardo@unimore.it

Abstract

The popularity of peer-to-peer file sharing networks has attracted multiple interests even in the research community. In this paper, we focus on workload characterization of file-sharing systems that should be at the basis of performance evaluation and investigations for possible improvements. The contribution of this paper is twofold: first, we provide a classification of related studies on file-sharing workload by distinguishing the main considered information and the mechanisms and tools that have been used for data collection. We also point out open issues in file-sharing workload characterization and suggest novel approaches to workload studies.

1. Introduction

The P2P phenomenon has received an increasing amount of attention in the last years. Thanks to their distributed nature [19], these systems represent an innovative and promising paradigm to build scalable and fault tolerant systems.

Multiple applications of peer-to-peer systems have been proposed. Examples include filesystems [10, 3], Web caches [8] and Streaming services [20]. However, the killer application for peer-to-peer systems remains file sharing over a large scale and large dimensions. The popularity of file sharing applications is increasing over time, thanks also to the growth in broadband connections that are available even to the home users. (A significant portion of the traffic on network backbones is related to file sharing activity [11].) For now on, file sharing represents the main test-bench for the scalability and fault tolerance properties of peer-to-peer systems.

There are many goals behind the workload studies of file sharing systems. Let us mention the improvement of peer-to-peer protocols [2, 14], the creation of realistic analytical and simulation models [5], the introduction of caching solutions [11], and the evaluation of the economic impact of file sharing due to copyright infringements [15].

The literature on P2P file-sharing workload characterization is recent, but large already. Each proposed approach has its pros and cons. We propose the first survey that considers main studies in file-sharing workload analysis and classifies them according to two main parameters: techniques used for data collection that is, crawling or traffic interception and information that have been analyzed, such as shared contents, user behavior, structure and performance of the interconnections;

Hence, the contribution of this paper is twofold: we propose a taxonomic scheme for workload analysis classification; our classification allows us to point out discrepancies among different studies and to point out open issues and areas for future research.

The paper is organized as following. Section 2 outlines the main characteristics of file sharing and the main elements of file sharing networks. Section 3 describes the two techniques used for data collection. Section 4 provides an analysis of the state-of-the-art in workload characterization. Finally, Section 5 outlines open issues, future research directions, and provides some concluding remarks.

2. File sharing networks

File sharing networks are essentially peer-to-peer systems designed to allow users to exchange files. The file sharing application inherits from peer-to-peer systems two main characteristics, that is: creation of a so-called *overlay network* and the use of a decentralized approach to network management. Both them are key characteristics for the deployment of a world-wide service such as file sharing.

The basic function of a file-sharing network is to allow a node to advertise the shared files and to carry out a lookup process over the overlay network to find a resource shared by remote node. The lookup process is generally based on *queries* that match the resource characteristics. The most common case is a query that matches filenames based on regular expressions. The lookup process returns the list of resources that match the query and the location of these resources. Once a file is found, a download process can be initiated for the actual file retrieval.

Query and download are the basic operations for the majority of file sharing networks. Although the basic principles are common, there are multiple incompatible networks, each characterized by different protocols. We find useful to focus the analysis on two popular file sharing networks FastTrack [9] and Gnutella [6] because most workload characterization results are based on them.

Some researchers have directed their study also on other file sharing networks such as E-Donkey an DirectConnect [18], however studies taking into account multiple networks are few and their contribution is limited to a partial view of the network characteristics. On the other hand, Gnutella and FastTrack have been studied through different techniques with various features aspects.

The FastTrack network is used by the Kazaa [9] file sharing software. The network uses two protocols. The former is used for network management and for resource lookup; it is characterized by a heavy use of cryptography that hinders its reverse engineering. The latter protocol is used for file download and can be easily analyzed because in practice it corresponds to the standard HTTP protocol integrated with few headers.

The Gnutella network is based on open protocols. Even if the number of nodes and the amount of files shared in the Gnutella network is lower than that in the FastTrack network [19], the open nature of the network makes Gnutella an interesting basis for the study of file sharing characteristics. Gnutella uses two protocols: HTTP for file download and a network-specific protocol for network management and resource lookup. The Gnutella protocol specification is available in two versions: Gnutella v0.4 (the first available standard version of the protocol) and Gnutella v0.6 [6] officially standardized in 2004 and now adopted by most Gnutella servents.

3. Classification of workload analysis according to collection technique

There are two main approaches for data collection, that is: *Active probing (crawling)* and *Passive probing (traffic interception and analysis)*. Active probing is a technique for data collection based on *crawlers*. A crawler is a modified servent that issues queries to inspect the contents and the structure of the peer-to-peer network. Passive probing collects data without issuing explicit queries but intercepting and analyzing actual file sharing traffic.

Fig. 1 represents the crawling approach to data collection. The small monitors are the servents of the file-sharing network, and the clouds represents physical networks that are connected through links shown as thick solid lines. The crawler connects to the overlay network and issues queries (shown as dashed arrows). The crawler creates a snapshot of the overlay network based on the responses to its queries.

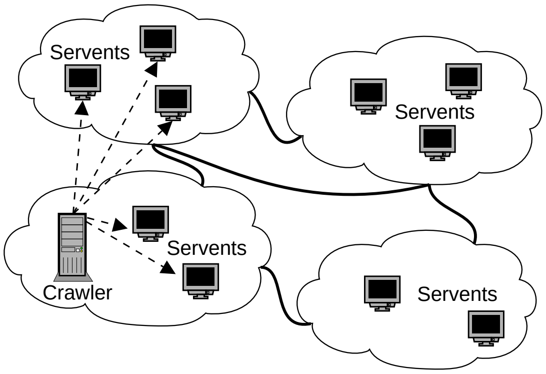

Figure 1. Crawling.

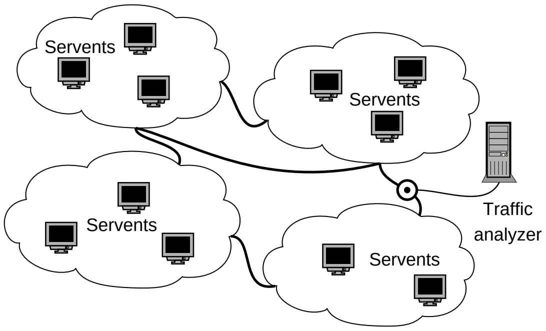

Figure 2. Traffic interception and analysis.

For each file stored in every node a crawler can collect the name (that can be used also to determine the resource type), the size and the file hash value. The latter acts as a content digest (usually computed with the SHA1 algorithm) of the actual file content. Due to the strong non-collision properties of SHA1, the hash code can be assumed as a unique identifier of the file content. This allows a double analysis of resources to detect different files with the same name and identical files stored under different names.

From a technical point of view, crawlers are easy to implement, and their implementation is further simplified when open source servents are available. On the other hand, when the overlay network protocol is not known (for example, in the FastTrack/Kazaa network), the use of a crawler is extremely difficult (and has not yet be done) because it would require a previous step for the reverse engineering of the network protocol. For this reason studies on file-sharing networks based on crawling are carried out basically on the Gnutella network.

Passive probing collects data without issuing explicit queries but analyzing already available file sharing traffic. This analysis is typically carried out by intercepting and analyzing the traffic on a network link. Fig. 2 shows the passive approach to data collection. The traffic analyzer is connected to a link connecting multiple physical networks. The analyzer can gather information on the overlay network only based on the traffic observed over the link.

Traffic interception and analysis introduce two main issues to be addressed: first, the file-sharing traffic on the link under observation must be a significant sample of the overall file-sharing traffic, second, only traffic related to file sharing is significant.

The first issue requires a careful selection of the link to be observed. Analysis carried out on a scarcely popular link can lead to wrong or inaccurate conclusions because the intercepted traffic deviates substantially from the real workload. For this reason studies using a traffic analysis approach takes into account links such as ISP backbones [11, 12] or big organizations (e.g., companies, universities) outbound links [7].

Extracting file sharing packets from the overall traffic requires a classification of the traffic over the link. From the point of view of a file-sharing workload characterization we can recognize three types of traffic: (1) traffic directly related to resource downloading, (2) traffic related to overlay network management and queries, (3) traffic unrelated to file sharing.

As for crawling, the traffic analysis requires open and well-documented protocols. Download is carried out through the HTTP protocol in both the Gnutella and the FastTrack/Kazaa network. Download analysis is hence straightforward for both networks.

Traffic analysis introduces also significant technical issues. From Section 2 we know that multiple protocols are used for overlay network management and signaling, hence the traffic analyzer must be able to recognize specific traffic signatures [18]. An alternative approach is to rely only on specific well-known ports. However, this solution is not reliable because file sharing servents can be configured to use non-standard ports. This behavior has become more popular since firewalls are configured to hinder the diffusion of file sharing by blocking protocol-specific ports.

4. Survey of workload analysis

We classify the literature on workload analysis in the last years into three broad categories:

Characterization of the resource working set: it focuses on what resources are shared over the network. For example, we can study the number of shared files and their popularity distribution. These analyses allow the evaluation of the caching potential of file-sharing traffic and provide an evaluation of the magnitude of the file sharing phenomenon. Other interesting studies are finalized to classify the wide heterogeneity of shared files into a few profiles, generally based on the MIME type.

Analysis of the user behavior: it is mainly related to the dynamic aspects of the network. A non exhaustive list of user behavior studies includes analysis on download starts and abortions, time related patterns in the population of users, such as the download session duration, frequency of servent joins and leaves. Time stability of these patterns has also been taken into account.

Characterization of the servents and of the overlay network investigates on connection characteristics of the servents belonging to the network. Moreover, some researches have focused on the relationship between the overlay network and the physical network topologies.

4.1. Characterization of the resource working set

Analysis on working set have focused on two main topics that is, resource popularity and size.

Studies on file popularity. If we consider the studies on popularity, we have three main analyses that have been carried out: popularity analysis on global resource set, based on resource type and as a function of time.

In [11], Leibowitz *et al.* found through traffic analysis a very skewed popularity curve in which 80% of downloads is referred to 20% of the resources. The same authors confirmed the observation in a subsequent study [12]. In [1], through crawling, Andreolini *et. al* found that the popularity of shared resources follows a Zipf law. On the other hand, another study of Gummadi *et al.* [7] suggests that if we focus on file downloads popularity distributions can be better described through truncated-Zipf curves. This difference between the results of [1] and [7] is due to the different data collection strategies. However, the topic seems interesting and worth of further studies.

Another interesting study is the analysis on the file types popularity. Two analyses [11, 1] have addressed this issue and their conclusions are the same. Fig. 3 shows the number of shared files according to its type. We aggregated the MIME types into four groups: audio, video, archives (corresponding to archival data) and documents (e.g. PDF, text, postscript files). All studies confirm that audio clips are the most popular files, accounting for nearly 50% of files, followed by archives, video and documents, with the latter being rather uncommon.

A final study on popularity is how popularity rank changes over time. This analysis have been carried out by Leibowitz *et al.* in [12] by studying variations the popularity rank of the 400 most popular files. The study identified two categories of resources: a small group of resources (nearly 20% of the working set) that are characterized by stable popularity rang and the remaining 80% of the resources that is subject to fast changes in popularity.

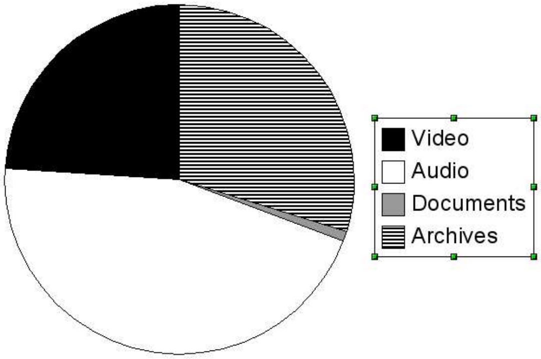

Figure 3. File type popularity (# of files).

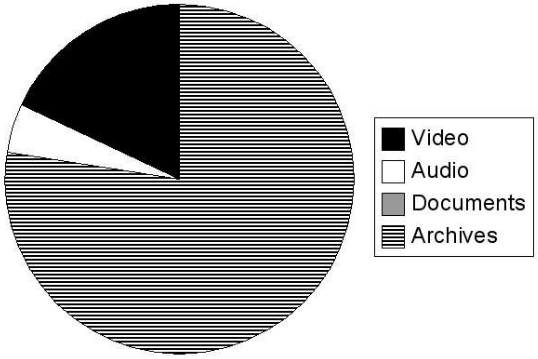

Figure 4. File types popularity (size).

Studies on working set size. We can distinguish three main studies carried out on the working set and resource size of file sharing networks: resource size on the global working set, resources shared by each node, and resource size according to its type.

In [11], Leibowitz *et al.* provided a histogram of the file size for the working set of the FastTrack/Kazaa network. The study shows that a high number of shared files have sizes of nearly 5 MB. This is consistent with the previously reported results on audio files popularity.

A Further analysis related to file sized has been carried out through crawling by Andreolini *et al.* [1] and provides an analytical model for the resources shared by each node. The study suggests that the number bytes shared by each node follow a distribution with a lognormal body and a Pareto tail.

A final analysis on working set size is the relationship between file MIME type and its size. Leibowitz *et al.* found a that file size is strongly correlated to its type. For example, audio clips tend to be rather small (a few MB in size), while video and archive files are at least an order of magnitude bigger. If we consider Fig. 4 we see that archives accounts for more than 75% of the global working set size, while audio clips accounts for less than 10% of it. These results are particularly interesting when compared with Fig. 3: audio files are the most common resource, but its contribution to the working set size is very small. On the other hand, archives are the main contributors to the working set size, even if their number is reduced. These results are confirmed in [1]. This latter study provides also an analytical model for file size according to file type using lognormal and Pareto distributions.

4.2. Analysis of the user behavior

Studies focusing on user behavior belongs to two categories: studies aiming to define a “file sharer user profile” and studies aiming to characterize the user activity cycles.

Definition of user profile. User profile has been described focusing on particular behaviors (considered antisocial or dangerous for the network), taking into account the time required for file download, and evaluating time-related modifications in user behavior.

A first important contribution focusing on user profiles aims to address the issue of *freeloaders* that is, users downloading resources without sharing any file. The common beliefs is that these users have an antisocial behavior, wasting resources (mainly bandwidth) available in the file sharing network. A study [5] based on analytical models, however, suggests that freeloaders are not necessarily a negative aspects of the network because, while they are connected to the network they contribute in routing query messages. Moreover, partially downloaded files are made available to the network and provide additional replicas of the file being downloaded.

A further contribution to describing the file sharing network user is the study of Gummadi *et al.* [7]. The study provides an interesting characterization of Kazaa users by analyzing the file sharing traffic in a university campus. A first finding of the study is that “users are patient”: the researchers found that even for small files (less than 10 MB, typically audio files), 30% of the downloads take more than an hour and for 10% of the resources the download takes nearly a day. For large requests (more than 100 MB), less than 10% take less than one hour, 50% take more than one day and 20% of the users wait for a week to complete their download.

The same study outlines also an interesting *aging* effect on the user. As users gets accustomed to the Kazaa tool (i.e. after 3-4 weeks), the number of downloads is nearly halved and the amount of data download is reduced to one fourth respect to new users.

User activity characterization User activity characterization can be described based on *download session length*, that is the time during which the user is downloading at least a resource and *activity fraction*, that is the fraction of time

	median	90-percentile
Activity fraction [7]	66%	100%
Download session length [7]	2.40 min	28.33 min
Session length [17]	60 min	300 min

Table 1. User activity parameters.

spent by users downloading files from the network.

In [7] Gummadi *et al.* studied both metrics through traffic analysis. The results of this study are shown in Tab. 1: the activity fraction tends to be high, with a median value of two-third of time spent in downloads. On the other and, each download session tends to be rather short (lasting only a few minutes). This suggests that one download is typically split into multiple small chunks that are downloaded separately.

A similar analysis carried out by Saroiu *et al.* [17] suggests much longer sessions. However, this latter study is carried out with a crawler. As a consequence, the session length is not referred to the download activity, but to the standard join/leave cycle of a servent in a file sharing network. Moreover, the study described in [17] is carried out on the Gnutella Network, while the analysis described in [7] is based on the FastTrack/Kazaa network. These considerations can explain the different results of the two studies. Our conclusion is that this results discrepancy is worth of further investigation.

4.3. Characterization of the servents and of the overlay network

An interesting aspect takes into account the network and the servents. Two main topics have been analyzed: network topology and servent connectivity.

Studies on network topology. The main issues addressed by studies on network topology are the relationship between overlay network topology and physical IP network and the structure of the overlay network.

In [16], Ripeanu *et al.* use a crawler to demonstrate that the Gnutella network topology is completely different from the physical network topology. The authors argue that this makes the Gnutella file sharing system inefficient. The same study suggests that the network can be described as a *power-law* network. This means that the Gnutella network is composed by a reduced number of nodes with a high out-degree and multiple nodes with a reduced number of connections. Further study carried out by Saroiu *et al.* through crawling [17] confirms this observation. The same study analyzes the impact of the power-law structure on the network resilience and concludes that the network is highly resilient to random node failure. On the other hand removing just a small amount (less than 5%) of the best connected nodes can lead to network partitioning. In [13], a detailed study of the Kazaa network topology is given. The authors also try to deduce the behavior of supernodes by injecting their clients into the Fasttrack network. In particular, they have found that supernodes tend to select the least loaded neighbors.

	median	90-percentile
Latency	100 ms	900 ms
Bandwidth	1Mb/s	20 Mb/s

Table 2. Servent connectivity parameters.

Characterization of servent connectivity. Studies on servent connectivity have mainly focused on two main parameters, that is available bandwidth and network latency.

The main results addressing this problem are by Saroiu *et al.* [17]. The authors carried out a crawling on the Gnutella network and for each servent collected the servent-advertised bandwidth. An analysis of latency and bottleneck bandwidth has been carried out for every discovered servent and the data have been compared with the advertised information. The authors point out that advertised data tends to underestimate the actual network connectivity. Tab. 2 shows the latency and bandwidth found in [17]. As we can see most users are characterized by DSL-class networking, as testified by the median bandwidth and latency.

The authors combine the bandwidth and latency data with node availability information and define two peer profiles: a “Server” profile characterized by high bandwidth, low latency and high availability, and a “Client” profile with reduced connectivity and availability. The study suggests that Gnutella is not a real peer-to-peer network because less than 15% of the nodes fit in the “server” profile and the large majority of peers are mainly “clients”.

5. Open issues and conclusions

In this paper we proposed an analysis of the state of the art in file sharing workload analysis. We described the two main approaches to data collection and we provide a taxonomic classification of the literature on peer-to-peer workload analysis according to three main categories, that is analysis of file-sharing working set, characterization of user behavior and analysis on network structure and characteristics.

There are many open issues in workload characterization of file sharing.

Experimental results and conclusions should use multiple data collection techniques. In multiple analysis (e.g., session duration, resource popularity) the use of just a crawler or a traffic analyzer leads to quite different results.

This suggests that only a combination of the two data collection approaches can take into account different aspects of the same phenomenon and can provide additional insight.

A second interesting problem is the lack of geographic-related analysis on file sharing download. The peer-to-peer systems aim to build a world-wide network. On the other hand, the same overlay network is deployed over a physical geographic network and user behavior is related to its geographical position (mainly due to timezones). It seems interesting to analyze what location-related aspects of user behavior are visible in a global distributed systems, such as a file-sharing network.

A final issue that needs additional efforts is the implementation of new traffic analysis techniques. For example, IP packet capture and analysis over high capacity links is required to obtain significant information for workload characterization. Packet capture and off-line analysis is not an option because the amount of storage required would be unacceptably high. Hence we need tool that are able of analyzing peer-to-peer packets on-the-fly. On the other hand, we also need to carry out complex signature matching to provide a sound traffic analysis, as pointed out by Sen *et al.* [18]. A typical issue of Intrusion Detection System, where complex matching has to be carried out on high traffic links is packet loss due to overload. It seems necessary to address the trade-off between accuracy in traffic analysis and computational load. An interesting contribution in this direction derives from Internet traffic analysis where statistical models are used to extract information from a subset of the whole traffic. This approach has been suggested in the NetScope project [4], but it seems that no similar efforts have been directed towards peer-to-peer traffic analysis.

As the file-sharing is a relatively recent application, multiple issues in workload characterization are yet to be addressed. In particular we outlined the following three interesting fields that are worth additional study in the future:

- analysis of file sharing workload carried out combining both crawling and traffic analysis.
- analysis of location-related aspects in file sharing workload.
- Improvement of traffic analyzers used for file sharing study.

References

[1] M. Andreolini, R. Lancellotti, and P. S. Yu. Analysis of peer-to-peer systems: workload characterization and effects on traffic cacheability. In *Proc. of MASCOTS 2004*, Volendam, NL, Oct. 2004.

[2] Y. Chawathe, S. Ratnasamy, L. Breslau, and S. Shenker. Making gnutella-like p2p systems scalable. In *Proc. of ACM SIGCOMM 2003 Conference*, Karlsruhe, Germany, Aug. 2003.

[3] F. Dabek, M. F. Kaashoek, D. Karger, R. Morris, and I. Stoica. Wide-area cooperative storage with CFS. In *Proc. of SOSP '01*, Chateau Lake Louise, Banff, Canada, Oct. 2001.

[4] R. C. et al. Measurement and analysis of ip network usage and behavior. *IEEE Communications Magazine*, May 2000.

[5] Z. Ge, D. R. Figueiredo, S. Jaiswal, J. Kurose, and D. Towsley. Modeling peer-peer file sharing. In *Proc. of INFOCOM 2003*, San Francisco, CA, Apr. 2003.

[6] Gnutella protocol development specification v0.6. Draft, 2004. – http://rfc-gnutella.sourceforge.net/src/rfc-0_6-draft.html.

[7] K. P. Gummadi, R. J. Dunn, S. Saroiu, S. D. Gribble, H. M. Levy, and J. Zahorjan. Measurement, modeling, and analysis of a peer-to-peer file-sharing workload. In *Proc. of SOSP-19*, Bolton Landing, NY, Oct. 2003.

[8] S. Iyer, A. Rowstron, and P. Druschel. Squirrel: A decentralized, peer-to-peer web cache. In *Proc of PODC 2002*, Monterey, CA, Jul. 2002.

[9] Kazaa, 2004. – http://www.kazaa.com.

[10] J. Kubiatowicz, D. Bindel, Y. Chen, P. Eaton, D. Geels, R. Gummadi, S. Rhea, H. Weatherspoon, W. Weimer, C. Wells, and B. Zhao. Oceanstore: An architecture for global-scale persistent storage. In *Proceedings of ACM ASPLOS*. ACM, Nov. 2000.

[11] N. Leibowitz, A. Bergman, R. Ben-Shaul, and A. Shavit. Are file swapping networks cacheable? characterizing p2p traffic. In *Proc. of WCW '02*, Boulder, CO, USA, Aug. 2002.

[12] N. Leibowitz, M. Ripeanu, and A. Wierzbicki. Deconstructiong the kazaa network. In *Proc. of WIAPP '03*, San Jose, CA, USA, Jun. 2003.

[13] J. Liang, R. Kumar, and K. Ross. Understanding kazaa. In *(Submitted for pubblicaiton)*, Brooklyn, NY, 2004.

[14] S. Ratnasamy, S. Shenker, and I. Stoica. Routing algorithms for dhts: Some open questions. In *Proc. of IPTPS '02*, Berkeley, CA, Feb. 2002.

[15] RIAA, 2004. – http://www.riaa.com.

[16] M. Ripeanu, I. Foster, and A. Iamnitchi. Mapping the gnutella network: Properties of large-scale peer-to-peer systems and implications for system design. *IEEE Internet Computing Journal*, 6(1), 2002.

[17] S. Saroiu, P. K. Gummadi, and S. D. Gribble. A measurement study of peer-to-peer file sharing systems. In *Proc. of MMCN '02*, San Jose, CA, USA, Jan. 2002.

[18] S. Sen, O. Spatscheck, and D. Wang. Accurate, scalable in-network identification of p2p traffic using application signatures. In *Proc. of WWW2004*, New York, NY, May 2004.

[19] D. Towsley. Peer-to-peer and application level networking. In *Proc. of Performance 2002*, Rome, Italy, Sep. 2002.

[20] D. Tran, K. Hua, and T. Do. Zigzag: An efficient peer-to-peer scheme for media streaming. In *Proc. of INFOCOM 2003*, San Francisco, CA, Apr. 2003.