Impact of Social Networking Services on the Performance and Scalability of Web Server Infrastructures

Claudia Canali*, Jose Daniel Garcia†, Riccardo Lancellotti*

*University of Modena and Reggio Emilia
{claudia.canali, riccardo.lancellotti}@unimore.it
†University Carlos III of Madrid
josedaniel.garcia@uc3m.es

Abstract

The evolution of Internet is heading towards a new generation of social networking services that are characterized by novel access patterns determined by social interactions among the users and by a growing amount of multimedia content involved in each user interaction. The impact of these novel services on the underlying Web infrastructures is significantly different from traditional Web-based services and has not yet been widely studied. This paper presents a scalability and bottleneck analysis of a Web system supporting social networking services for different scenarios of user interaction patterns, amount of multimedia content and network characteristics. Our study demonstrates that for some social networking services the user interaction patterns may play a fundamental role in the definition of the bottleneck resource and must be considered in the design of systems supporting novel applications.

1 Introduction

The Web evolution is heading towards a radical change in the user behavior, with a growing amount of users that create, share, and distribute content on the Web. The shift towards a "Read-write" paradigm is one of the biggest changes in the Web since its beginning. This paradigm change is driven by the popularity surge of the so called *social networking services* [18] that enable users to build social networks and share their own generated content. Today, the most popular example of Web sites offering social networking services are Weblogs (blogs) [7], where user exchange mostly textual information, sites that allow users to share photos such as Flickr [10, 17] and Facebook [9], and sites where the interaction among users is based on exchanging video clips (e.g., YouTube [11]).

The success of these social networking services is demonstrated by the high number and heterogeneity of participating users. According to Nielsen/NetRatings, social networking sites have seen their traffic growing 47% in the last year, reaching 45% of all active Web users [3]. This confluence of participants is possible because of the low barrier to entry into on-line communities. Many social networking services are designed such that signing up, viewing, and uploading contents are relatively easy. This enables users who may not be technically skilled to participate alongside more experienced users.

The novel characteristics of social networking services concern access patterns and resource working set. The access patterns are characterized by variability and unpredictability due to social interactions (e.g., slashdot effects). Furthermore, the read-write nature of the services leads to a significantly higher amount of upload operations with respect to traditional Web-based services. On the workload side, we observe that social networking services present a wide range of multimedia contents: from mostly-textual blogs, to sites where the user interaction is based on images and video clips, determining a radical shift in the type of data transferred.

The combination of the novel features of high user interactivity and multimedia rich contents introduces new requirements on the underlying server infrastructures that are significantly different from these of traditional Web-based services. While there are several studies on performance and bottleneck analysis for traditional Web applications [20, 4], there have been no substantive studies regarding novel social networking services.

The main contribution of this paper is to investigate how currently available infrastructures can cope with the requirements of novel social networking services. Our analysis aims to identify the bottlenecks that hinder performance in the server infrastructures as a function of the type of supported social networking service, because we claim that understanding the impact of novel services is a fundamental step that will aid in network management, capacity planning, and design of new systems.

In this paper we consider three significant examples of social networking services, namely *Blogging*, *Photo sharing* and *Video sharing*. The services are characterized by different amounts of multimedia contents, and for each service we consider different levels of user interactivity (that is, the read *vs.* write ratio). Furthermore, we consider how the client-side connection may impact on the internal bottlenecks of the server infrastructure. Through simulation we show that the presence of multimedia-rich content in social networking services will cause a major shift in the infrastructure bottlenecks. Furthermore, we show that the level of user interactivity and the network characteristics can play a fundamental role in determining the system bottleneck.

The remaining of the paper is organized as follows: Section 2 describes the main features of social networking services. Section 3 describes the system and workload model used for our analysis. Section 4 presents the results of our experiments, providing an insight on the impact of the service evolution on the underlying infrastructure. Section 5 discusses the related work. Finally, Section 6 provides some concluding remarks.

2 Social networking services

Social networking services are characterized by two main features: the interactivity of the users that access the services and the high degree of multimedia content provided through these services. Both features are becoming more and more prominent as the social networking services unleash their potential of supporting new user interaction models.

A key requirement of user interactivity in social networking services is the support for content upload. Indeed, instead of just consuming content posted by a single administrator, users of social networking services are able to actively create and share their own content as well as view content provided by their peers. Furthermore, some services such as MySpace [14] and Facebook [9] support social interaction by allowing individuals with similar interests to form social groups. Support for content upload changes dramatically the requirements on the underlying server infrastructure for two main reasons. First, asymmetric communication lines, such as ADSL, are designed for a download-oriented usage pattern and the limited upstream bandwidth may lead to poor upload performance. Second, uploading content requires write operations that are more expensive than reads for mass storage devices. Another consequence of social interactions among users is that, whenever a content upload is carried out, it is likely to cause an increase of subsequent upload operations due to comments and replies from other users [8]. This results in a high variability in the read/write ratio that places additional strains on the server storage subsystem.

The second main characteristic of social networking services is the ever increasing amount of multimedia content exchanged among the users. Indeed, the services range from blogging, where users mostly exchange textual content, to multimedia-oriented services that allow users to share photos and even videos. The presence of multimedia content changes significantly the impact of the workload on the infrastructure with respect to traditional Web. Since multimedia rich resources are far larger than traditional Web resources, disks and network links are a critical resource that may become congested, thus hindering the overall system performance.

To carry out this analysis we consider three main types of social networking services, namely *Blogging*, *Photo sharing*, and *Video sharing*, that differ for the workload characteristics. Furthermore, for each service we propose three scenarios with different amounts of user-uploaded content.

Blogging service

The Blogging service allows users to exchange information and replies in textual format. The working set involved in blogging is similar to the traditional Web workload [19], where Web resources are typically in the order of tens of Kbytes. However, the interaction patterns show significant differences if compared to any type of traditional dynamic Web site, due to the social interactions among the users. Indeed, comment posting follows a bursty behavior [8] that may determine usage patterns that resemble a flash crowd with sudden peaks in the disk write/read ratio.

Photo sharing service

The Photo sharing service consists in a public image storing and sharing service that allows users to upload their own photos, view photos created by others and comment on them. We consider the Flickr Web site as a typical example of a Photo sharing service. Besides the complex interaction related to the user behavior in a social networking service [8], Photo sharing is characterized by content of extremely variable sizes. The exchanged images may range from small snapshots taken with a mobile phone camera to high-resolution photos taken with professional-grade equipment. Furthermore, for each image, the site presents multiple thumbnails at different resolutions to allow users to save bandwidth during the preview of a photo album. This results in access patterns for the storage subsystem that are extremely difficult to predict.

Video sharing service

The Video sharing service is one of the most innovative services supporting user interactions. The analysis of video sharing services and its social implications has attracted the interest of the research community [11,5]. The typical trend in video sharing is to exchange short video resources, typically clips of few minutes, and the most widespread approach to data transfer is to use HTTP-based pseudo streaming techniques instead of full streaming protocols. The resulting workload is one of the main challenges for the server storage and for the network infrastructure, due to the size of content being downloaded and uploaded from the client devices.

3 System Model

The model used for our analysis takes into account the server infrastructure, the clients and the network connecting clients and server. We evaluate performance and scalability of Web server infrastructures for social networking services by using a discrete event simulator based on the Omnet++ framework [16], that relies on the Inet package for packet-level simulation. Since a goal of our study is the analysis of how bottlenecks change depending on the type of service, we develop a model that represents accurately the main resources that may hinder the scalability of a server infrastructure.

Recent studies evaluating the potential bottlenecks of dynamic Web systems [2] suggest that, on the server side, the main system resources that may become bottlenecks are typically *CPU* and *Disk*. Furthermore, the type of network connection between clients and server infrastructure may have a significant impact on the resource utilization and, consequently, on the system bottlenecks.

3.1 Workload model

We describe the interaction patterns of a user accessing social networking services according to [8]. The user behavior is based on a session mechanism, where each session is composed of multiple interactions. In particular, we assume three possible user interactions, corresponding to:

- Accessing the main page showing the directory of the shared information (e.g., blog entries, images and videos)
- Accessing a second level page showing only one specific content (e.g., a specific blog entry with its comments, an image or a video clip)
- Uploading new content

A user session may be a *read session* if the user only downloads contents, or a *write session* if the user also uploads contents to the server. In our experiments, we consider three different *upload scenarios* where the percentage of write sessions with respect to all the user sessions is equal to 5%, 10% and 20%, respectively. During a read session, the user accesses the main page, then performs multiple requests for second level pages. Each read session includes 4 requests for second level pages on average. On the other hand, a write session involves an access to the main page, followed by an access to a second level page and a content upload. We use a Gamma distribution (α=0.64, β=12.62) to model the interarrival time of user sessions and a Weibull distribution (α=0.69, β=0.33) for the interarrival time of requests within the same user session [8].

We now describe the content size distributions that we use to model the three social networking services.

For the Blogging service we refer to the workload characterization in [8], where textual resources have an average size of 30KB. The file size distribution is modeled as a Pareto distribution with parameter α equal to 1.17.

Since there are not studies that characterize the resources of the Flickr Web site, we carried out a preliminary evaluation to determine the workload characteristics for this scenario. We downloaded 20 Photo Albums from the Flickr Web site for a total of 13000 images on which we perform statistical analysis. We find that the histogram of photo sizes in our dataset can be fit by an aggregate of a normal distribution, with parameters μ=2.10 and σ=0.68, and a lognormal distribution, with parameters μ=3.91 and σ=0.70, as shown in Figure 1.

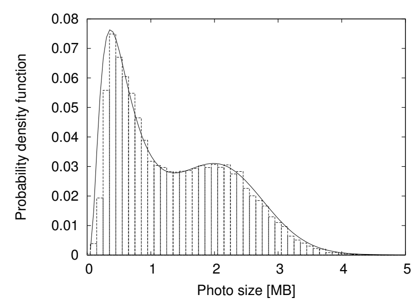


Figure 1. Photo size distribution

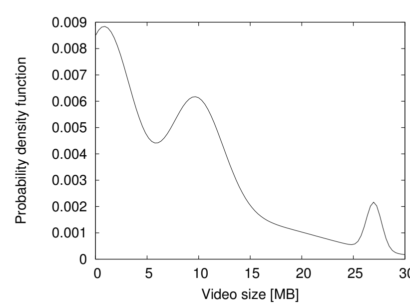


Figure 2. Video size distribution

To model the workload of the Video sharing service, we refer to the video size distribution described in [6]. The size distribution is modeled as an aggregate of four normal distributions, whose parameters are shown in Table 1. The parameters μ and σ determine the mean and the standard deviation, respectively, while r is the weight of each curve in the aggregated distribution, that is shown in Figure 2.

Table 1. Parameters for aggregated distribution of video size.

Parameter	Norm1	Norm2	Norm3	Norm4
μ	0.74	9.63	27.00	13.66
σ	2.87	2.68	0.74	7.96
r	48.6%	26.2%	2.7%	22.5%

3.2 Server model

Since our aim is to evaluate which are the system bottlenecks in a server supporting social network services, we devote special care to model each component of the server infrastructure.

The server supports two main types of requests: read requests, that we consider based on HTTP GETs, and write requests, carried out through HTTP POSTs. We suppose that client requests are processed in parallel using a time-sharing CPU and a disk-based storage unit.

After entering the system, a read request is assigned to a thread and a service time is determined for parsing the request. CPU time is shared among all the active threads using a round-robin CPU scheduler. After this pre-processing time, the Web resource is retrieved from the disk and sent to the client. We consider that, during the service of a request a multimedia resource is not retrieved entirely from the disk, but data is fetched from the disk and delivered to the client in a chunk-by-chunk fashion. This model is supported by analysis on download of multimedia resources: HTTP pseudo-streaming is one of the most popular mechanism for multimedia content download and this transfer mechanism relies on range requests and chuncked encoding [12].

Each request for data from the file system is mapped by the file system module in multiple subsequent accesses to data blocks, to model into a deep detail the disk behavior. Disks are modeled considering that currently available storages usually provide caching capabilities at the level of blocks. The actual effectiveness of a disk cache depends on the access patterns. However, due to the highly skewed popularity distributions of social networking services [5], we consider a block cache hit rate in the order of 90%, that is a common value for multimedia oriented file storage [13]. Read requests are mapped by the file system in a sequence of block reads. The requests for the physical disk accesses not satisfied by the cache are queued and executed by the disk module. Write requests are composed of subsequent HTTP POST requests, each containing a data chunk from the client to the server. For each request, the server processes the HTTP message and inserts the uploaded content in a temporary buffer. We consider that the file system adopts a write-back policy for write operations, hence the write of blocks on the disk occurs off-line with respect to the interactions between server and client. It is worth to note, however, that while write operations do not have a direct effect on the execution time of HTTP POST requests, they impact on the queue length of the disk device, and, as a consequence, on the response time for other operations accessing the disk.

3.3 Network model

The network model used in our experiments aims to represent a typical server infrastructure with a centralized server and remote clients. The network model considers all the levels of the TCP/IP stack, as well as physical link characteristics (i.e., bandwidth, delay and error rate) to provide a clear picture of how network may impact on performance. The network models are provided by the Inet package of the Omnet++ simulator [16].

We design two types of scenarios with respect to the network: a *LAN* scenario where all the clients are connected through LAN-grade links to the server and a *DSL* scenario that represents the case of residential clients with an ADSL connection.

For the LAN scenario, each client is connected to an ISP router with a Fast-Ethernet connection. The ISP router is connected through a Gigabit connection to a border router of the autonomous system where the server infrastructure is located, and another Gigabit link connects the server infrastructure with the border router.

In the DSL scenario, each client is connected through a Fast-Ethernet connection to a local router. The client router is connected to an ISP router using an asymmetric PPP link. The asymmetric link is characterized by a downstream bandwidth of 20 Mbit/s with a delay of 2 ms and an uplink bandwidth of 1 Mbit/s per second with a delay of 60ms, which are common parameters for ADSL links. We suppose that the overall client population is equally distributed among a set of multiple ISPs, and each ISP is connected to the server router using a Gigabit connection. Finally, the server is connected to the border router using the same Gigabit link of the LAN scenario.

4 Experimental results

In this section we evaluate how the type of service, the user behavior and the network characteristics affect the performance of social networking services. We devote special attention in finding for each scenario which is the system bottleneck. We consider the three social networking services, Blogging, Photo sharing and Video sharing, and for each service we evaluate three different upload scenarios, where the percentage of write sessions is equal to 5%, 10% and 20%, and two network scenarios, namely *LAN* and *DSL*, as described in Section 3. For each scenario we consider the following metrics:

Throughput: the amount of requests satisfied in the time unit (minute) by the server.

CPU utilization: the fraction of time where the CPU is busy with respect the experiment duration.

Disk utilization: the fraction of time where the disk is busy with respect the experiment duration.

We choose these metrics because they allow us to identify which resource represents the bottleneck of the system at the peak throughput for each considered scenario. For all the sets of experiments, we simulate the system for 10 hours. For each simulation we discard the first 3000 seconds to avoid transient effects in our system evaluation. The simulation results refer to data averaged over 10 runs.

4.1 Blogging service

Figure 3 shows the throughput for the Blogging service as a function of the number of clients in the *LAN* scenario. The peak throughput is similar for any upload scenario, showing that the performance of the system is not affected by the percentage of write sessions.

We analyze the utilization of the system resources to identify which resource represents the system bottleneck at the peak throughput. Table 3 presents the CPU and disk utilization at the peak throughput for all the upload scenarios. We observe that the CPU is the bottleneck resource for any upload scenario. On the other hand, the disk utilization never exceeds 0.05 even for frequent content uploads. This result is due to the small size of the textual information exchanged among the users of the Blogging service, that does not involve expensive operations on the storage device even when the amount of write operations grows. This allows the system to serve an increasing number of concurrent requests. The CPU utilization increases linearly with the number of clients, until it almost reaches the 100%.

Table 2. Average resource utilization at the peak throughput for Blogging service

Resource	Write sessions		
	5%	**10%**	**20%**
CPU	0.997	0.995	0.994
Disk	0.022	0.030	0.047

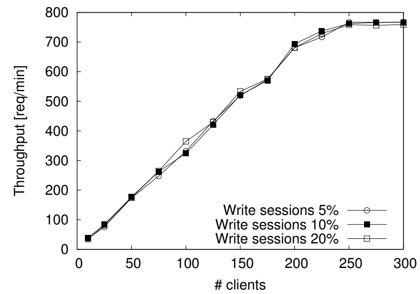


Figure 3. Throughput for Blogging service

4.2 Photo sharing service

Figure 4 shows the throughput for the Photo sharing service as a function of the number of clients in the *LAN* scenario. As expected, for each considered upload scenario the overall throughput is lower than that achieved for the Blogging service. This is due to the larger size of the exchanged contents that involves more expensive operations. We observe that the peak throughput for the Photo sharing service differs depending on the considered upload scenario. Specifically, a high amount of write operations causes a significant degradation of the system performance, with a 34% decrease of the peak throughput as the percentage of write sessions changes from 5% to 20%.

To investigate the reasons of this behavior, we consider the average CPU and disk utilization for the different upload scenarios, that are shown in Figure 5. It is interesting to observe that the resource bottleneck at the peak throughput depends on the upload scenario. For a limited amount of write sessions (5% in Figure 5(a)) the CPU is the bottleneck; on the other hand, when users update contents more frequently (Figures 5(b) and 5(c)) the disk becomes the bottleneck of the system. The shift in the resource bottleneck is due to the cost of write operations, that are more expensive for the disk than read operations. When 95% of the operations just require reading from the disk, the CPU is

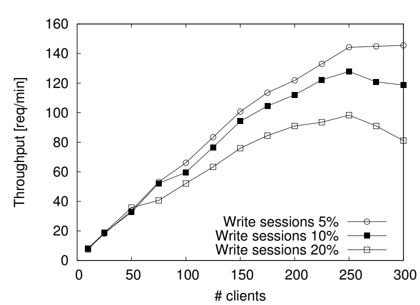


Figure 4. Throughput for Photo sharing service

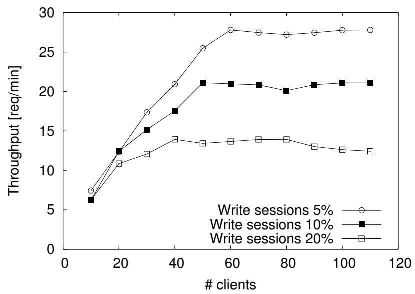


Figure 6. Throughput for Video sharing service

the most loaded resource and high volumes of concurrent requests saturate its capacity. On the other hand, when the percentage of write sessions grows to 10%, the utilization of disk increases faster than the CPU utilization as the number of clients grows (Figures 5(b)).

If the number of write operations further increases, the capacity of the disk saturates for a lower amount of concurrent requests. In comparison, the cost of handling the requests on the CPU is small (Figure 5(c)).

4.3 Video sharing service

We now analyze the results for the Video sharing service in the *LAN* scenario. Figure 6 shows the throughput for different upload scenarios as the number of the clients increases. We observe that the amount of content uploads significantly affects the peak throughput, that is more than halved when the percentage of write sessions changes from 5% to 20%.

To understand why the system performance drops much more significantly for higher percentages of write sessions, we analyze CPU and disk utilization. Table 3 shows the disk utilization, that is close to 1 for all upload scenarios, while the CPU utilization never exceeds 0.16. We can conclude that the disk is the bottleneck resource for all upload scenarios. This result is quite intuitive, since the large size of video content (median size equal to 8.4 MBytes) requires operations on the storage device much more expensive than CPU-related operations. The drop in throughput is motivated by the disk-bound nature of requests and by the high cost of write operations. Our experiments show that the disk-bound nature of requests in the Video sharing service may halve the system throughput when the amount of write sessions grows by a factor of 4.

4.4 Impact of network scenario

We now evaluate whether the presence of a DSL connection between clients and server (described in Section 3.3)

Table 3. Average resource utilization at the peak throughput for Video sharing service.

Resource	**Write sessions**		
	5%	**10%**	**20%**
CPU	0.153	0.123	0.094
Disk	0.940	0.947	0.985

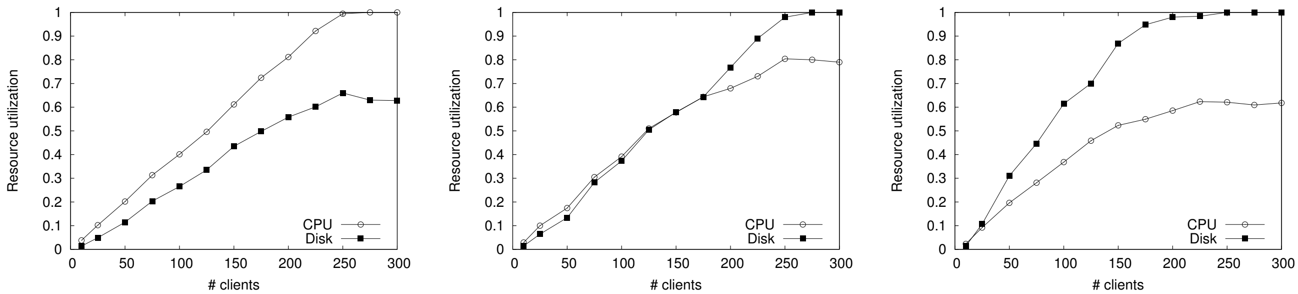
may affect the system scalability and the presence of bottlenecks for the three considered social networking services.

For space reasons, we report results only for the Photo and Video sharing services, that show the most interesting changes with respect to the *LAN* scenario. Indeed, the presence of a DSL connection does not significantly affect the overall system performance for the Blogging service due to the small size of textual contents exchanged. For this service, the factor that limits the overall system throughput remains the concurrency of the requests, which exhausts the capacity of the CPU, as it happens in the *LAN* scenario.

Table 4 shows the peak throughput and the average resource utilization for the Photo sharing service in the *LAN* and *DSL* scenarios. Bold text is used to evidence the utilization of the bottleneck resource at the peak throughput. As regards the peak throughput, we observe a significant reduction with respect to the *LAN* scenario. This result is expected, because each request lasts longer due to the network delays introduced by the *DSL* scenario. However, we notice that the throughput reduction depends on the upload scenario, ranging from almost 30% to 17% as the percentage of write sessions changes from 5% to 20%. To explain the reasons of this difference, we evaluate the utilization of CPU and disk at the peak throughput.

The *DSL* scenario places a higher demand on the CPU with respect to the *LAN* scenario, because the longer duration of client requests significantly increases the CPU utilization due to the higher request concurrency. Hence, the CPU becomes a bottleneck earlier with respect to the LAN scenario, thus resulting in a throughput reduction. Unlike the Blogging scenario, the throughput reduction is significant because images are medium-sized contents, and the DSL link delay has considerable consequences. This effect is also evident when the percentage of write sessions increases to 10%: the system bottleneck at the peak throughput shifts from the disk (LAN case) to the CPU (DSL case). On the other hand, for a 20% of write sessions the disk remains the bottleneck resource even in presence of a DSL connection. In this case the peak throughput experience a minor decrease with respect to the other upload scenarios because the disk utilization is not increased by the presence of a DSL connection as much as the CPU utilization.

Table 5 reports the peak throughput and the average resource utilization for the Video sharing service in *LAN* and *DSL* scenarios. The throughput is reduced with respect to



(a) Write sessions 5% (b) Write sessions 10% (c) Write sessions 20%

Figure 5. Average resource utilization for Photo sharing service

the *LAN* scenario, however the throughput reduction (from 12% to 13%) is smaller than that observed for the Photo sharing service (from 17% to 29%). While the former result is expected due to the longer duration of the requests, the latter result may surprise because video contents are larger than images, and consequently more expensive to serve. From Table 5 we observe that the disk is always the bottleneck resource for the Video sharing service. As for the Photo sharing service, when the throughput is limited by the disk performance the effect of DSL connections is much less evident than when the bottleneck is the CPU.

4.5 Summary of results

The main messages from our experiments can be summarized as follows. As expected, our results show that the workload has a significant impact on the system performance and on the system bottlenecks. Performance in the delivery of textual resources is mainly affected by CPU power, while multimedia-rich resources such as videos place the highest demand on the system storage. Future network-based services, characterized by large multimedia resources will require the server infrastructures to place significant effort in providing adequate speed in the access to data, for example by improving disk caching techniques, or using faster storage technologies.

An interesting result is that the level of user interactivity plays a key role in determining which is the bottleneck resource. As user interactivity grows, the high amount of write operations increases the disk utilization and may cause a shift of the system bottleneck. To cope with highly variable user behavior, server infrastructures may exploit self-adaptive techniques that allow the system to dynamically adapt and reconfigure itself at runtime in response to changing levels of user interactivity.

Finally, the presence of a low speed connection may have significant consequences on the system performance. Furthermore, low connection speed determines a growth in CPU utilization that may even change the bottleneck resource in the server infrastructure. Self-adaptive techniques may be also exploited to address these issues.

Table 4. Resource utilization for Photo sharing service at peak throughput [req/min]

Write	LAN scenario			DSL scenario		
sessions	**Peak**	**CPU**	**Disk**	**Peak**	**CPU**	**Disk**
5%	144	**0.993**	0.660	103	**0.994**	0.620
10%	128	0.803	**0.987**	93	**0.991**	0.811
20%	98	0.620	**0.990**	81	0.870	**0.974**

Table 5. Resource utilization for Video sharing service at peak throughput [req/min]

Write	LAN scenario			DSL scenario		
sessions	**Peak**	**CPU**	**Disk**	**Peak**	**CPU**	**Disk**
5%	28	0.993	**0.660**	24	0.421	**0.990**
10%	21	0.803	**0.987**	18	0.339	**0.992**
20%	14	0.620	**0.990**	12	0.287	**0.995**

5 Related work

Social networking services has recently received an increasing attention [7, 6, 11, 5] with the aim of characterizing the typical workload of these services in terms of access patterns, object types and object sizes. For example, Duarte *et al.* focus on the blogging service and provide models of social interactions to shape the workload of Blogs [7], while the studies in [6, 11, 5] characterize the workload for multimedia-oriented social networking services, such as photo and video sharing. These studies point out that the social networking services present workload characteristics that significantly differ from “traditional” Web-based services applications and will require the adoption of new strategies at the network and server level to be supported. Our study represents a step further, because it exploits these workload characterizations with the aim to evaluate how the social networking services affect the scalability of the systems and how they affect the bottlenecks in the underlying server infrastructure.

Performance analysis and bottleneck identification have been widely investigated in literature [20, 1]. Some studies compare the performance of the technologies available to provide Web-based services [1] or provide an analysis of the bottlenecks of Web systems [20]. However, these studies rely on standard benchmarks, such as TPC-W and RUBiS, that are limited to e-commerce oriented scenarios. Hence, these studies do not capture the disruptive potential of the innovative services based on social networking. On the other hand, our study explicitly considers the impact on scalability and performance of user interactivity and frequent uploads of multimedia objects, that may easily exhaust network and server resources.

A further contribution of our study is the analysis of system bottlenecks and performance for several workload scenarios that aim to capture the evolutionary trend of social networking services. Previous studies on the impact of evolutionary technology trends on the performance of the underlying infrastructure [2, 4] were focusing on e-commerce services, without considering the social-oriented factors of workload evolution. Other studies on the impact of different network scenarios, such as [15], were even more limited because they focused on low-level details such as TCP congestion control, while we extend the analysis to the main system resources such as network file descriptors, CPU and disk utilization.

6 Conclusions

The growing popularity of social networking services is changing the Web workload. User interactivity and presence of multimedia-rich content place on the underlying server infrastructures a different demand with respect to traditional Web-based services.

We carry out a comprehensive study with different services, levels of user interactivity and client connections. Our results show that system performance and bottlenecks are dependent on the type of service, on the user behavior and on the network characteristics. To support social networking services, future server infrastructures will need to provide high performance storage to support the growth of multimedia-rich content. Furthermore, the high variability of user behavior and network characteristics suggests that future infrastructures may exploit adaptivity to cope with extremely heterogeneous demands on the system resources.

References

[1] C. Amza, A. Chanda, E. Cecchet, A. Cox, S. Elnikety, R. Gil, J. Marguerite, K. Rajamani, and W. Zwaenepoel. Specification and implementation of dynamic Web site benchmarks. In *Proc. of Fifth Annual IEEE International Workshop on Workload Characterization*, Austin, TX, Nov. 2002.

[2] M. Andreolini, M. Colajanni, and R. Lancellotti. Impact of technology trends on the performance of current nd future Web-based systems. *Int'l Journal of Web Services Practices*, 1(1–2):121–132, Dec. 2005.

[3] S. Bausch and L. Han. Social Networking Sites Grow 47 Percent, Year over Year, Reaching 45 Percent of Web Users. Nielsen//NetRatings Research Report. May, 2007.

[4] E. Cecchet, A. Chanda, S. Elnikety, J. Marguerite, and W. Zwaenepoel. Performance comparison of middleware architectures for generating dynamic Web content. In *Proc. of ACM/IFIP/USENIX International Middleware Conference (Middleware 2003)*, Rio de Janeiro, Brazil, June 2003.

[5] M. Cha, H. Kwak, P. Rodriguez, Y.-Y. Ahn, and S. Moon. I tube, you tube, everybody tubes: analyzing the world's largest user generated content video system. In *Proc. of the 7th ACM SIGCOMM Conference on Internet Measurement (IMC'07)*, San Diego, CA, Oct. 2007.

[6] X. Cheng, C. Dale, and J. Liu. Understanding the Characteristics of Internet Short Video Sharing: YouTube as a Case Study. *ArXiv e-prints*, 707, July 2007.

[7] E. Cohen and B. Krishnamurthy. A short walk in Blogistan. *Computer Networks*, 50(5):615 – 630, Apr. 2006.

[8] F. Duarte, B. Mattos, A. Bestavros, V. Almeida, and J. Almeida. Traffic Characteristics and Communication Patterns in Blogosphere. In *Proc. of International Conference on Weblogs and Social Media*, Seattle, WA, Apr. 2007.

[9] Facebook, 2008. – http://www.facebook.com.

[10] Flickr, 2008. – http://www.flickr.com/.

[11] P. Gill, M. Arlitt, Z. Li, and A. Mahanti. YouTube traffic characterization: A view from edge. In *Proc. of Internet Measurement Conference*, San Diego, CA, Oct. 2007.

[12] L. Guo, S. Chen, Z. Xiao, and X. Zhang. Analysis of multimedia workloads with implications for Internet streaming. In *WWW '05: Proc. of the 14th International Conference on World Wide Web*, May 2005.

[13] Y. Hu, T. Nightingale, and Q. Yang. RAPID-Cache-A Reliable and Inexpensive Write Cache for High Performance Storage Systems. *IEEE Trans. on Parallel and Distributed Systems*, 13(3):290–307, 2002.

[14] Myspace, 2008. – http://www.myspace.com.

[15] E. M. Nahum, M.-C. Rosu, S. Seshan, and J. Almeida. The effects of wide-area conditions on WWW server performance. *SIGMETRICS Perform. Eval. Rev.*, 29(1):257–267, 2001.

[16] OMNeT++ Discrete Event Simulation System, 2008. – http://www.omnetpp.org.

[17] N. Van House. Flickr and public image sharing: distant closeness and photo exhibition. In *Proc. of ACM SIGCHI Conference on Human Factors in Computing Systems (CHI'07)*, San Jose, CA, Apr. 2007.

[18] A. C. Weaver and B. B. Morrison. Social Networking. *IEEE Computer Magazine*, 41(2):97 – 100, Feb. 2008.

[19] A. Williams, M. Arlitt, C. Williamson, and K. Barker. Web Workload Characterization: Ten Years Later. *Web content delivery*, 2005. (Tang, Xu, Chanson eds.), Springer.

[20] Q. Zhang, A. Riska, E. Riedel, and E. Smirni. Bottlenecks and their performance implications in e-commerce systems. In *Proc. of 9th International Workshop on Web Content Caching and Distribution*, Beijing, China, Oct. 2004.