Unveiling Facebook: A Measurement Study of Social Network Based Applications

Atif Nazir, Saqib Raza, Chen-Nee Chuah

University of California, Davis

{anazir, sraza, chuah}@ucdavis.edu

ABSTRACT

Online social networking sites such as Facebook and MySpace have become increasingly popular, with close to 500 million users as of August 2008. The introduction of the Facebook Developer Platform and OpenSocial allows third-party developers to launch their own applications for the existing massive user base. The viral growth of these social applications can potentially influence how content is produced and consumed in the future Internet.

To gain a better understanding, we conducted a large-scale measurement study of the usage characteristics of online social network based applications. In particular, we developed and launched three Facebook applications, which have achieved a combined subscription base of over 8 million users. Using the rich dataset gathered through these applications, we analyze the aggregate workload characteristics (including temporal and geographical distributions) as well as the structure of user interactions. We explore the existence of 'communities', with high degree of interaction within a community and limited interaction outside the community. We find that a small fraction of users account for the majority of activity within the context of our Facebook applications and a small number of applications account for the majority of users on Facebook. Furthermore, user response times for Facebook applications are independent of source/destination user locality. We also investigate distinguishing characteristics of social gaming applications. To the best of our knowledge, this is the first study analyzing user activities on online social applications.

Categories and Subject Descriptors

C.2.0 [**Computer - Communication Networks**]: General; H.4.3 [**Information Systems Applications**]: Communications Applications

General Terms

Measurement

IMC'08, October 20–22, 2008, Vouliagmeni, Greece.

Keywords

Online Social Networks, Social Games, Facebook, Applications, Characterization

1. INTRODUCTION

Over the past few years, online social networks (OSNs) have attracted a massive following, with close to 90% of undergraduate students in the United States using one or the other social network on a regular basis [6]. As a result, two OSNs (Facebook [21] and MySpace [28]) are now among the top ten visited websites on the Internet [14].

OSNs have an inherent viral property in that applications' user base can undergo exponential growth given the quick spread of information much like real-world social networks. Furthermore, through open developer platforms, large OSNs such as Facebook and MySpace have recently opened their doors to developers across the world, enabling even amateur developers to create applications by leveraging the underlying social graphs. The introduction of these third-party applications has led to even higher traffic on the corresponding social networks. For example, there was 30% increase in Facebook's site traffic in the week following the launch of its developer platform. Given the increasing popularity of these applications, we believe it is important to characterize such social network-based applications as a representative modern class of workload.

This paper presents a detailed study of the usage characteristics and nature of user interactions for three home-grown applications launched using Facebook's pioneering Developer Platform[1] [22]. We believe this is the first analysis of its kind. The key contributions of this paper are summarized as follows:

- We developed and launched three applications using the Facebook Developer Platform. Our applications have been able to realize a combined user base of more than 8 million users, placing them amidst the top 1% of Facebook applications at the time of writing this paper. We used these applications to procure a rich data set on the usage of social network applications, which has been made available to the Internet measurement community[2].

[1]We chose Facebook since it was the pioneer in launching its Developer Platform (in May 2007). Moreover, multi-million dollar investment and Facebook's active development have made Facebook Developer Platform the most evolved third-party application base to date.

[2]Data available at http://www.ece.ucdavis.edu/rubinet/data.html

- We analyze various usage characteristics of our applications, such as geographical distribution of users, user interactions and response times, and how they vary with respect to the application type.

- We use our data set to infer the nature of user interaction through Facebook applications. We model this interaction through *interaction graphs* and show that it exhibits small-world properties. One of our key findings is that application dynamics can significantly affect the structure of interaction graphs, hence weakening the association between them and the underlying real-world (friendship) relationships between users. For example, user interaction graphs for non-gaming applications are shown to contain stronger community structures as compared to gaming applications.

- We also analyze global usage data for a broader set of Facebook applications and show that application popularity is characterized by a power-law distribution with exponential decay, and use our finding to give insights into the underlying mechanism behind application subscription and usage.

The paper is structured as follows. We begin with a brief overview of the related work in Section 2. Section 3 describes our data collection methodology and the design of our applications in detail. We then present high-level characteristics of Facebook applications in Section 4, our findings regarding community structures for our applications in Section 5, and our findings related to user-level behavior for those applications in Section 6. We conclude with a discussion of our results and future work in Section 7.

2. RELATED WORK

Over the past few years there has been a flurry of activities on social network analysis. While some researchers have focused on graph theoretic properties of social networks [7, 9, 10], others have analyzed individual networks' usage patterns [2, 6]. However, there has not been a detailed study of third-party applications developed and launched on OSNs with a massive user base such as Facebook. We believe this paper is the first to measure and characterize this new workload, the user interaction, and its relationship to the underlying social networks.

Facebook has been the focus of a few studies recently. A newly published study on characterization of Facebook applications [5] uses profile crawling to explore the high-level characteristics of application users on Facebook, as well as growth patterns of applications using publicly available usage statistics from Adonomics [16]. We confirm some of the findings of this paper, and go beyond the scope of this study by analyzing activity data from our home-grown applications.

Another important study by Golder et al. [6] on messaging activity inside Facebook highlights Facebook-specific characteristics such as regularities in daily and weekly traffic and its relation to the use of Facebook by a select demographic (college students). The same study found that activity on Facebook seems to be focused on individual 'networks' and is related to temporal usage patterns of those networks. Here, 'networks' refers to Facebook's classification of users into different networks of school, college, work and regional categories. We were able to confirm the findings of [6] with regards to periodicity of traffic on Facebook, as well as extend our understanding of traffic patterns and user behavior to third-party Facebook applications.

Other relevant studies include Newman's work on community extraction algorithms [13] and Liben-Nowell's work on the relationship between geography and online friendships [8]. We utilize results of the former and attempt to extend Liben-Nowell's findings by looking at user interaction on social applications and its relation to users' geographical placement.

Furthermore, a recent study by Mislove et al. [10] focused on the graph theoretic properties of large OSNs such as YouTube [31], Flickr [25], and Orkut [29]. It discussed the existence of small-world and scale-free properties. While we do touch upon similar aspects in this study, note that we focus on a new workload, namely *third-party applications* on OSNs. In our study, we analyzed the actual user interactions through our home-grown applications, rather than focusing on the social networks determined through user friendship profiles.

3. BACKGROUND AND METHODOLOGY

Facebook is a social networking website that has recently gained immense popularity. Part of the reason for Facebook's success is its developer platform, which we shall discuss shortly. A friendship is formed on Facebook when one Facebook user extends a (friendship) invitation to another user. Upon confirmation by the latter, the friendship relationship is formed. Much of the activity on Facebook occurs due to these friendship relationships. However, due to the introduction of the Developer Platform, non-friend interactions are now rising through interaction on social applications. Therefore, it is important to analyze users' interactions through these social applications, beyond the definition of 'friends' through Facebook profiles.

In this section, we provide a brief overview of the Facebook Developer Platform, followed by details of the applications we implemented and a description of the data set used for our study.

3.1 Facebook Developer Platform

The Facebook Developer Platform was launched in May 2007 [22] with little fanfare and only about eight applications in its roster. Over the subsequent months, the Platform experienced phenomenal growth, showcasing more than 35,000 applications by July 2008 [16]. The launch of the Platform also increased Facebook's traffic by about 30% in the opening week, and it has seen overall growth since [30].

Fig. 1 shows aspects of the Facebook Developer Platform's architecture that are relevant to our applications. In this architecture, a user interacts indirectly with the application servers through Facebook's API servers. This enables Facebook to protect users from malicious content that may be embedded in the response data by the application servers, since Facebook can process and strip undesirable content from the server responses before forwarding them to users.

We must note, however, that Facebook has an alternate method for deploying applications on its Platform that enables users to interact directly with the application servers.

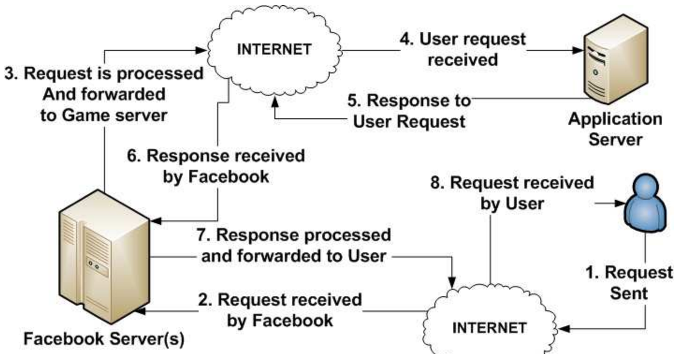

Figure 1: Facebook Developer Platform's architecture used by *Fighters' Club*, *Got Love?* and *Hugged*.

However, the architecture shown in Fig. 1 is the dominant architecture used, primarily due to the ease of displaying content to the application users[3] and the protection of the application servers' identity from the end-users. We adopt this architecture to develop our own Facebook applications to limit the resources required to render content to users.

3.2 Implementation of Social Applications

For the purpose of this study, we implemented three Facebook applications: *Fighters' Club*, *Got Love*, and *Hugged*, which will be discussed below. At the time of writing this paper, these applications had been used by a total of 8.24 million users (7.23 million unique Facebook identities). There are some overlapped users across the three applications. Note that applications on Facebook generally require user subscription (or *installation*). Once a user installs an application, it may provide updates on users' and their friends' activities through the profile page.

3.2.1 Social Gaming: Fighters' Club (FC)

Fighters' Club (*FC*) [24] was launched on Jun 19, 2007 on the Facebook Developer Platform. It is one of the first games to launch on Facebook, and evolved over a period of 9 months to have been played by over 3.44 million users on Facebook. FC allows users to pick virtual fights with their Facebook friends that last from 15 to 48 hours. For the duration of the fight, each player may request support from their Facebook friends, who then help the individual's team defeat the opposing user's team through a series of virtual 'hits' decreasing the (limited) strength[4] of the target opponent(s). The team with the higher cumulative strength at the end of the fight is declared the winner[5].

Users on FC may have one of the following three roles in a given game instance (fight):

Offender: The user instigating the fight is the offender. This user must choose a friend to fight against, provide a reason for picking the fight, and select a fight duration (from 15 to 48 hours).

Defender: The Facebook friend 'picked on' by the *offender* is the defender.

Supporter: The *offender* and *defender* may advertise the fight to their Facebook friends. These friends then pick *one* side (the offender's or the defender's) and *support* the chosen user's team. Supporters may withdraw support from fights or change sides until the last 2 hours of the fight.

The duration of games was fixed to be at least 15 hours due to the wide geographical distribution (of users) possible on social networks, and in order to accommodate users' inability to react instantly when games are formed against them. This delay in reaction is an artifact of OSNs, and is discussed later in Section 4.2.3.

3.2.2 Non-Gaming: Got Love (GL)

Got Love (*GL*) [26] was launched on the Facebook Developer Platform on Nov 27, 2007 and has been used by a striking 4.07 million users since. The purpose of the application is to enable users to pick a set of 'special' friends they admire in order to display them as a distinct set of 'loved' friends on their user profile page.

3.2.3 Non-Gaming: Hugged

The third application, *Hugged* [27], was launched on Facebook on Jan 29, 2008 and has since been used by more than 730,000 users. Like *GL*, *Hugged* is also a simple application where users are able to send virtual 'hugs' to their friends. However, unlike *GL* where a user targets the same friend only once, *Hugged* allows users to send virtual 'hugs' repeatedly to the same friends.

3.3 Data Collection

Data from FC, GL & Hugged: Most of the data analyzed in this paper is from a 3-week trace, starting March 20, 2008, taken at the respective applications' servers. By recording and time-stamping each user request forwarded by Facebook to our application servers, we were able to trace all activities on *FC*, *GL*, and *Hugged* for the 3-week period. Formally, we define activity as an action performed by a subscribing user on *FC*, *GL*, or *Hugged* on another user. More specifically:

- On FC, an activity involves picking a fight with a friend, supporting a fighter in a given fight, and hitting an opponent in a fight. Note that a user may support and hit *non-friends*.
- On GL, an activity occurs when a user A sends 'love' to user B. An individual B may be 'loved' by A only once. In this case, A and B must be Facebook friends.
- On Hugged, an activity occurs when a user A sends a virtual 'hug' to a user B. Individual A may send multiple virtual 'hugs' to B. In this case, too, A and B must be Facebook friends.

Table 1 summarizes our data set from the three applications, along with the following user statistics:

Total Unique Users: Total number of unique Facebook identities that appear in our 3-week long trace.

Total Subscribing Users: Unique Users that had installed our applications on Facebook.

[3] This architecture relieves application servers from the the extra task of downloading and uploading extra content, such as users' real names and graphical content from Facebook.

[4] The measure of strength is a point system ranging from 0 to 5 points for each individual on FC. By default, each user's strength is 3.0, and it increases/decreases as the users win/lose fights, respectively. Individuals with 0.0 strength cannot be targetted in virtual 'hits'.

[5] In cases where both teams have 0.0 cumulative strength, the team making the last 'hit' wins.

Table 1: Data set analyzed in this paper.

	Fighters' Club	Got Love	Hugged
Total Activities	25,911,335	7,196,302	2,146,819
Total Unique Users	154,681	5,376,704	1,322,631
Total Subscribing Users	85,928	1,518,767	408,651
Total Active Users	43,669	642,088	198,379
(Active) Users w/ Geo Info	40,369	97,465	180,216
Users w/ Friendship Data	35,349	72,074	121,389
BW Consumption Info	Dec 15 Onwards	Feb 15 Onwards	Feb 15 Onwards
Google Analytics Data	Dec 15 Onwards	Feb 15 Onwards	Mar 22 Onwards

Total Active Users: Subscribing Users that instigated at least one activity on our applications.

However, since we use the indirection-based platform architecture described previously (Section 3.1), we had to separately capture IP addresses of the users in order to map individual users to different geographical locations. We achieved this by having users' browsers initiate HTTP requests directly to our application servers using FBML[6] IFrames at every visit to the application home page. We were then able to capture users' IP addresses and the respective Facebook user IDs.

In order to map IP addresses to geographical locations (countries), we used longest-prefix matching with the legacy country zones provided in [18]. We were able to track IP addresses for only a portion of the *active* users (see Table 1). Moreover, for users who visited our application sites, we tracked the number of their friends and the subset of their friends who also subscribed to our applications (referred to as 'subscribing friends' in the remaining discussions). This is feasible since Facebook provides each user's friends list data with every request sent to an application server. Note that we gathered IP addresses and friendship data for application users over the period of one week, ending April 1, 2008.

Furthermore, we utilized bandwidth consumption data tracked directly at the application servers, as well as the 'average time spent' metric tracked through Google Analytics[7] [17] for each application. We also acquired daily unique usage activity data for the top 200 applications on Facebook (as of April 22, 2008) from Developer Analytics[8] [19] for the period starting January 29, 2008 and ending April 22, 2008. However, many applications' statistics were missing for days in between. For the analysis in the next section, we selected 160 out of the 200 applications that have clean data for a total of 79 days.

4. HIGH-LEVEL CHARACTERISTICS

4.1 Global Facebook Application Statistics

We use the top applications' data from Developer Analytics to study the daily volume of users that use a particular application. We define the Daily Active Usage (DAU) to be the number of *unique* users that *visit* the application at least once during a given day. Fig. 2 plots the mean DAU for the entire set of the top 160 applications that we selected. The dotted vertical lines delineate weekends over the 79-day period. We see a consistent pattern showing that Facebook applications attract a relatively less number of unique users on weekends as compared to weekdays. Our data also shows that application usage generally peaks on Tuesdays. To show the relative popularity of our three applications (*FC*, *GL*, and *Hugged*), we rank the 160 applications in our data set in decreasing order of their DAU over the 79-day measurement period. We divide the 160 applications into 4 tiers by DAU and plot the mean DAU for each quartile. Fig. 2 shows that the DAU of our three applications are comparable to the mean DAU for the bottom two quartiles of applications (divided DAU-wise). Since Facebook hosts more than 35,000 applications [16], this shows that our applications are comfortably placed within the top 1% of all applications.

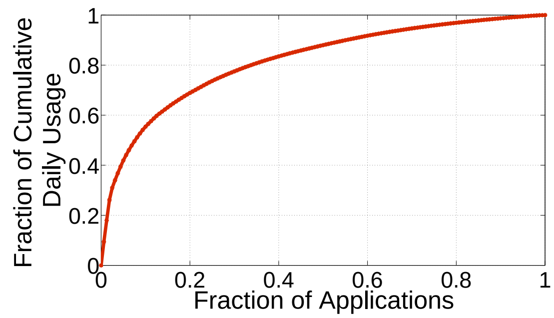

Figure 3: Cumulative distribution of DAU with applications sorted by descending order of average daily active users over 79 days.

We also looked at the distribution of DAU across our set of the 160 top applications. Fig. 3 plots the fraction of the sum of DAU values averaged over the 79-day period that is accounted for by the top x percent of the applications. The Pareto principle or the 80-20 rule is evident in that 20% of the most popular applications account for approximately 69% of the daily active users. Fig. 4 compares the distribution of average DAU across the 160 top applications in greater detail against the best-fit power-law and exponential curves. It suggests that application popularity follows a power-law distribution with an exponential cutoff, which is characterized by an exponential decay term that domi-

[6]Facebook Markup Language.

[7]While it is well-known that 'average time spent' is web-session based, exact details regarding this metric have not been made publicly available by the website at the time of writing this paper.

[8]Developer Analytics is a popular metric measurement site focused on Facebook applications. We infer the reliability of its measurements through anecdotal evidence as well as validation using data collected from our applications.

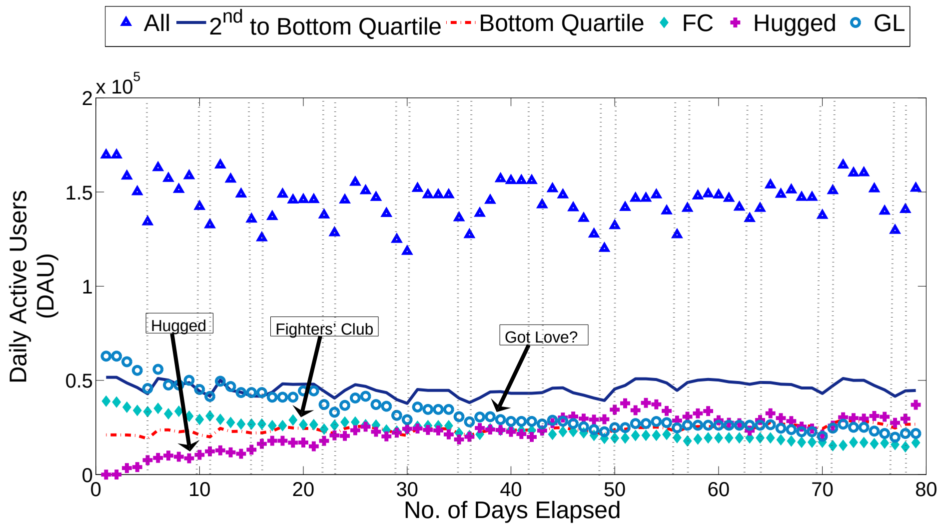

Figure 2: Daily Active Users (DAU) across time for top 160 applications on Facebook.

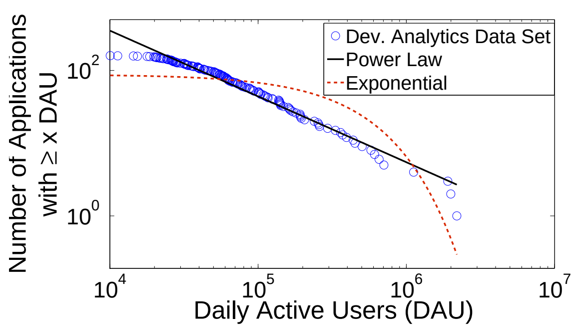

Figure 4: Empirical Distribution of daily active usage for top 160 applications on Facebook.

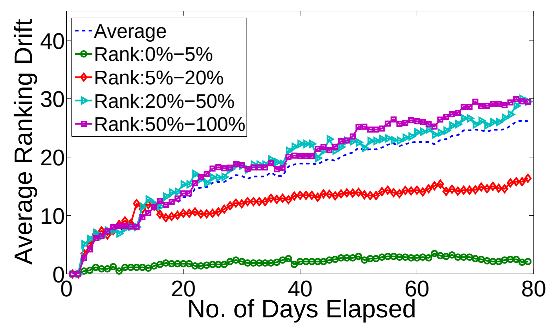

Figure 5: The change in applications' popularity ranks.

nates the power-law behavior after a certain threshold. [2] showed the popularity distribution of user generated video from sites such as YouTube [31] and Daum [20] exhibit a similar structure. Power-law popularity and usage distribution have also been observed in a wide-array of cases from web-references to real-world social networks [12].

The most straightforward explanation for the existence of the power-law is that the preferential attachment process (generally seen on social networks) generates power-law distributions. In our context, preferential attachment would imply that the probability of a new user subscribing to an application is proportional to the number of the application's existing users. We would expect to observe such phenomenon since Facebook maintains a bulletin board ('news feed') that updates Facebook users about their friends' activities. This serves as an advertising mechanism to promote applications that have an existing subscription base. Moreover, users can also explicitly advertise or engage their social network friends in applications they use.

The exponential cutoff to the classical power-law distribution has been studied before [12]. We consider a few plausible explanations. [1, 4] showed how preferential attachment with aging and/or fertility results in power-law with exponential cutoff. Fertility implies that applications have a minimum number of initial subscribers before preferential attachment gets triggered. This may apply to Facebook applications since the utility of these applications depends upon social networking. Hence, there can exist applications that require a certain quorum to be reached before subscribers could realize its full potential. Aging implies that after a certain time applications become obsolete. Another explanation given in [11] is information filtering. [2] also considered this to be a plausible explanation for video popularity on YouTube and Daum. The argument put forth is that given finite space, for example in the YouTube homepage or the friends' activity bulletin board in Facebook, information about less used applications gets filtered. Hence a classical power-law distribution is not achieved.

Furthermore, we found that applications' maintenance of global rank depends on how popular applications are. To study this, we divided the top applications on Facebook into four tiers and measured their *ranking drift* for each day since Jan 29, 2008. For an application, we define Ranking Drift on Day X as $|(RankonDay0) - (RankonDayX)|$, and plot the average drift values (per day) for each tier in Fig. 5.

Table 2: Mean and St. Deviation Values

	Mean	St. Dev	$\frac{Mean}{St.Dev}$
Quartile 1 (Top)	695,354	1,219,396	0.570
Quartile 2	106,171	261,921	0.405
Quartile 3	26,947	92,588	0.291
Quartile 4 (Bottom)	13,003	34,983	0.372

It can be seen that the lowest drift is observed for the top 5% applications, and this drift increases for lower quartiles. Furthermore, we provide the variance in DAU for each quartile in Table 2. Table 2 provides an intuitive argument for the former: since DAU numbers are closely clustered for applications in the lower quartiles, small changes in DAU lead to large changes in an application's rank. Similarly, since DAU numbers are farther apart for higher quartiles, even fairly large drops in application usage tend not to affect application ranks in the short term.

4.2 Global Usage Patterns of *FC*, *GL*, and *Hugged*

4.2.1 Geographical Distribution of Users

Facebook launched in May 2004 as primarily an OSN for college students across the United States. Since then, Facebook has expanded its reach to other geographical regions as well. By tracking the IP addresses of users accessing our three applications, we were able to map a subset of active users to different countries (see Table 1). We plot the resulting geographical distribution of users for *FC*, *GL*, and *Hugged* in Fig. 6a,6b,6c. As seen here, most of the applications' users reside in United States, United Kingdom, and Canada. Note also that user contribution from other countries varies for different applications, with Australia and South Africa being the dominant countries among the lower contributors in all three applications.

Furthermore, as shown above and supported by [23], the majority of the users on Facebook are based in the United States, United Kingdom, and Canada, which affects traffic patterns observed on (all) our indigenous applications, and on Facebook in general [6]. For example, Fig. 7 shows a clear diurnal pattern observed in a 24-hour snapshot of traffic on *FC*, where bandwidth consumption rises around the start of working hours in the United States and Canada (9AM CDT) and falls sharply at the end of working hours (5PM CDT). We observed similar daily traffic on *GL* and *Hugged*. Moreover, around special days observed in these three regions, especially Christmas, Thanksgiving, New Years Eve, and Valentine's Day, even weekday Facebook traffic falls quite sharply, even more so than observed on regular weekends.

4.2.2 User Interactions and Power Laws

Consider an activity graph that consists of a node for every user on an application. An edge exists between two nodes A and B if A and B interacted directly with each other (i.e., performed an activity directly on each other) through the application. We consider the *degree* of a user A as the number of *distinct* users A interacts with directly using an application. Fig. 8 shows log-log plots of the degree distribution for each application's activity graph. It can be seen that user interaction on (all) our three applications follows a power-law distribution. However, the power-law distribu-

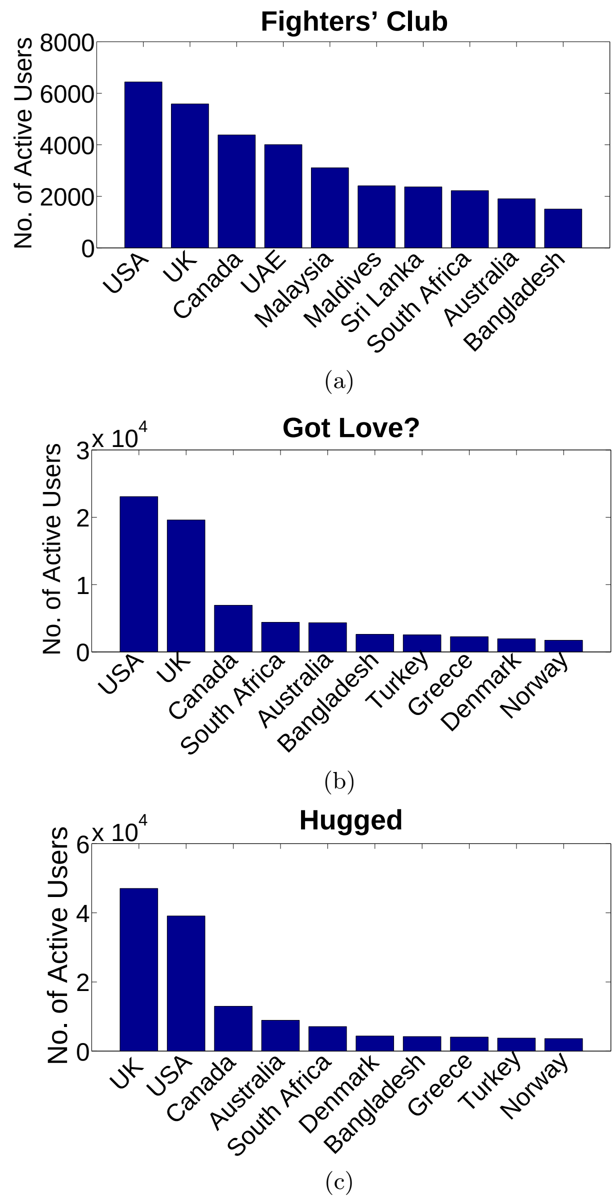

Figure 6: Geographical spread for unique *active* users on *FC*, *GL* and *Hugged*. Only the top 10 contributing countries are shown above.

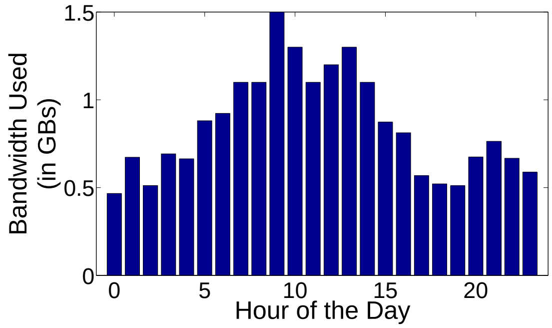

Figure 7: Bandwidth consumption of *FC* across 24-hours. The time of day is reported in CDT.

tion in *FC* is clearer than for *GL* and *Hugged* due to a denser number of degrees in the *FC* activity graph.

This result implies that a small number of 'power users' on Facebook dominate user interaction on platform applications, and as a consequence, generate the bulk of traffic or activities. We believe these power users are the driving force for the success of an application and responsible for sustaining the application's daily usage numbers in the long term.

4.2.3 Gauging User Response Times on Facebook

An aspect of social networks particularly important for application development (especially social gaming) is the delay in user response per activity initiated. Let 'user response' denote the time it takes for a target user to respond to an activity initiated through an application. For example, for the *Hugged* application, this would mean the time (number of seconds) elapsed between sending of a 'hug' request, and its reception by the target user. We say a target user 'receives' a request once they accept/confirm the request. Note that we tracked both when a user sent a particular 'hug' request to a target, as well as when the target user confirmed the request in order to calculate user response delays. This data was gathered for a total $684,505$ requests sent using *Hugged* over 3 weeks. We use this data as a representative sample of user response times for (all) our three applications, since application-to-user communication generally occurs mainly through the same channel(s) as employed in *Hugged*[9].

A CDF of user response times collected from *Hugged* is shown in Fig. 9. Since OSNs allow geographically remote users to maintain friendships online, activity often takes place between users in varied geographical locations, and large user response delays are to be expected. We found the average user response time was 16.52 hours, with the longest response times taking up as much as 567 hours (approximately the length of the trace)! However, as can be seen through Fig. 9, the probability of user response beyond 48 hours is considerably small and decreases noticeably after the 24-26 hour mark. We observe similar response time

[9]Facebook has introduced an in-beta AJAX-based method for 'live' communication between users on an application (LiveMessage). This is the same technology as used in the built-in live chatting application available to all users on Facebook. While we expect LiveMessage to alter response times especially for social gaming applications on Facebook, no data is currently available to gauge the differences.

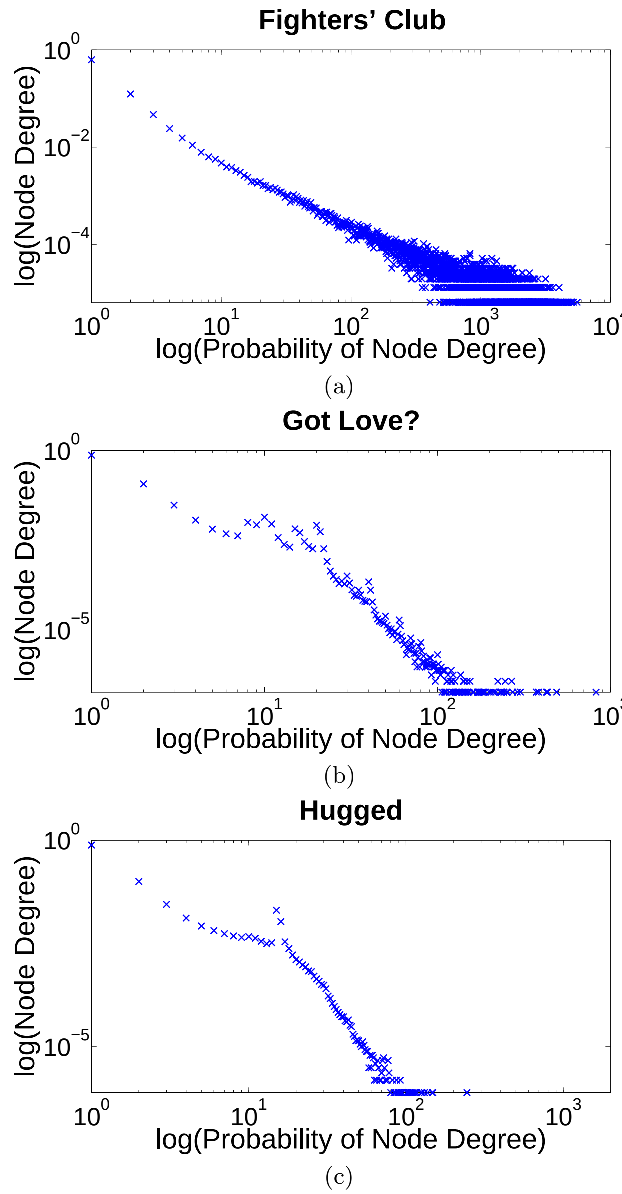

Figure 8: The log-log plots of the degrees of user interaction on *FC*, *GL* and *Hugged*. It can be seen that user-user interaction due to all three applications follows a power-law.

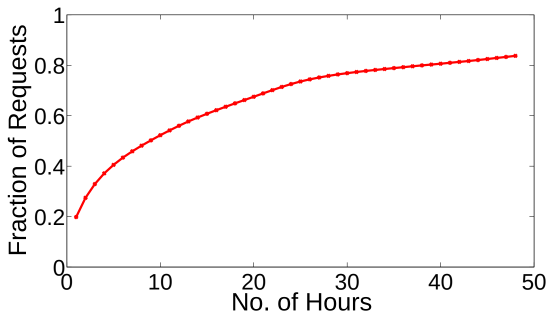

Figure 9: The CDF of user response time grouped into number of hours (e.g., response times ranging from 0 to 60 minutes were grouped as 1-hour, 60 to 120 minutes as 2-hour, and so on).

across our three Facebook applications, which is not surprising since they employ the same methods for communications through OSNs (i.e., e-mail, Facebook notifications, and invitation requests). However, we expect to see drastically different response times for other Facebook applications that involve real-time user interactions.

One may speculate that user response times would be different for requests sent to target users in the same locality (country) versus targets in foreign localities. However, our measured response times show this was not the case. The CDF plot for local requests' response times and that for foreign requests' response times were nearly the same as those shown in 9, with negligibly small differences. Furthermore, the average response times for foreign and local requests were comparable as well: the average response time was 14.8 hours for $383,397$ foreign requests, and 15.1 hours for $219,195$ local requests tracked[10].

5. COMMUNITY STRUCTURES

Development of popular applications for a broad user base poses challenges due to the viral nature of information spread on social networks. Scalability was one major challenge we faced developing our three Facebook applications. For example, within a month of launching *FC*, our application servers encountered 50-55 requests/sec. This, coupled with enormous storage, retrieval, and processing of data soon rendered cheap server solutions inadequate. Furthermore, user experience began to be affected, e.g., *FC* users complained of experiencing large delays when trying to meet game instances' deadlines[11].

Like *FC*, social games (due to their relatively higher engaging nature) often achieve high bandwidth consumption even at low DAU numbers. Realizing the trend on Facebook (especially) toward social gaming applications and considering the viral nature of information spread on social networks in general, we expect scalability for social applications to be a top concern for developers today. We believe that measurement results presented in this section provide crucial insights into addressing the scalability issues in developing social applications for a massive user base.

An important consideration in alleviating scalability concerns is the segregation of data into different locations for faster processing. Towards this end, we analyze *interaction* activities on our applications, as described next.

5.1 Definitions

In order to derive the results presented next, we analyzed the *interaction graphs* for *FC*, *GL*, and *Hugged*. We say that two (unique) users A and B *interact* on an application if: either A performs an activity on B or vice versa, or they *both* perform an activity on a common friend C (*GL* and *Hugged*)[12], or they perform an action in the same game instance (*FC*). Given this definition of interaction, we define the interaction graph $G = (V, E)$ such that for all unique users performing activities on a specific Facebook application, there $\exists v \in V$, and an undirected edge $(x, y) \in E$ for each interacting pair of users x, y. Additional concepts needed for the analysis presented next are:

Component: Two nodes x and y belong to the same *component* if $\exists\ (x, y) \in E$. A component's nodes are only connected with other nodes in the same component.

Clustering Coefficient: The *clustering coefficient* of a node $v \in V$ is the ratio of number of edges between neighbors x of v (such that $\exists\ (x, v) \in E$) and the total number of edges possible between those neighbors. The *clustering coefficient* of a *graph* is the average of individual nodes' clustering coefficients.

Community: A *community* in a graph is a set of nodes such that the ratio of edges between these nodes, and edges from these nodes to nodes outside of this community is 'high' [13]. That is, a community is a densely connected subgraph of G.

Structure Coefficient: Let e_{ij} be the fraction of (total) edges in the graph that connect vertices in community i to vertices in community j, and let $a_i \equiv \sum e_{ij}$. Then the *structure coefficient* of the graph is $\sum (e_{ii} - a_i^2)$. Community structure in a network is said to be *strong* if the *structure coefficient* is more than 0.3 [13].

5.2 Results

For the results discussed below, we used 1-week data (subset of the 3-week data) starting April 4, 2008 gathered through *FC*, *GL*, and *Hugged*. This was done primarily due to computationally expensive algorithms needed for the results produced here[13].

Table 3 shows the number of unique *interacting* nodes and edges in the interaction graphs for the 1-week trace. Our first consideration in attempting to relieve the scalability concerns was to extract disconnected components from

[10] Note that we were only able to geographically map a *subset* of the actual active users, as discussed in Section 3.

[11] Due to the fixed length of games on *FC* (15 to 48 hours), our users focus on entering game instances as close to the games' end as possible. This is done in order to prevent opponents from gaining points by hitting back in the game instance. Due to the competitive nature of the game, this translates into following gaming deadlines very closely, often down to the last second.

[12] Individuals A and B performing activity on the same *user* are said to *interact* as this is important with regards to scalability concerns: the same data point (C) is being accessed by the active individuals.

[13] The graph sizes for the 3-week trace had at least twice as many nodes *and* edges as compared to the 1 week trace for our applications. Due to CPU and memory limitations, we focused on a smaller (representative) trace for the experiments performed.

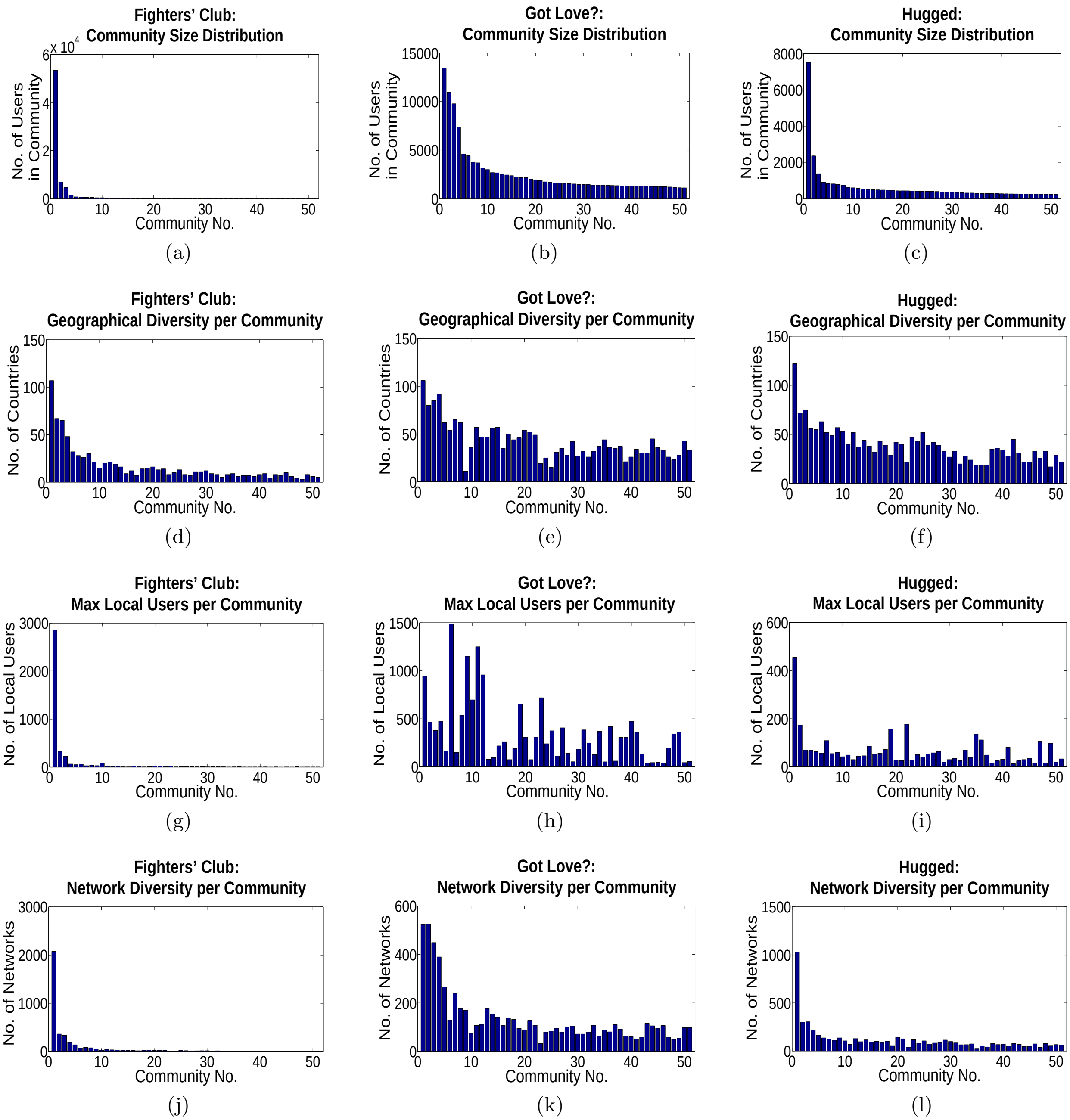

Figure 10: Community size distributions, geographical diversities, maximum number of users in same locality, and network diversities for *FC*, *GL* and *Hugged*. We only display the results for the 51 largest communities in the figures above since *FC*, which has the lowest number of communities, has only 51 communities. Within the results for each application, the community IDs on the x-axes indicate measurements for the same communities.

Table 3: Community Structures on Applications

	Fighters' Club	Got Love	Hugged
No. of Edges in Graph	16.8M	617,864	116,376
No. of Unique Users	73,300	277,540	51,343
Percentage of Users in Largest Component	91%	92.1%	86.7%
No. of Components	29	13,461	4,018
No. of Communities	51	1,951	521
Structure Coefficient	0.03	0.64	0.74
Max Size of Community	53,359	13,435	7,496
Max Geo Diversity	107	106	122
Max Network Diversity	2,858	2,285	1,084
Max Local in Community	2,852 (5.3%)	1,485 (34%)	455 (6.0%)
Clustering Coefficient	0.81	0.31	0.41
Diameter	10	45	29
Average Erdos-Renyi Clustering Coefficient	0.0062	0.000016	0.000085

the interaction graphs on all applications. Our results show that activity was structured on our applications such that most users were part of a single connected component, meaning component-wise segregation of activity data cannot, on its own, provide a solution for scalability. Furthermore, we found the percentage of total users in the largest component to be proportional to the number of users in each application's trace. This implies that as we consider larger data sets for analysis of component sizes, more and more nodes fall into the largest component. As remarked previously in [10], this lop-sided distribution of component sizes is an artifact of social networks generally, and since this result holds regardless of application nature in our data, we expect our results hold for interaction on social applications as well.

Community-wise division of data seems attractive in that OSNs have previously been shown to exhibit strong community structure [10]. However, interaction on social applications, since it goes beyond the underlying social network's friendship graph, may result in distinct community characteristics. Furthermore, we wish to gauge the extent to which metrics such as geographical classification of users might capture community structures of the *interaction graph* on social applications.

To this end, we extracted community structures from the interaction graphs derived from our applications using Newman's Leading Eigenvector community extraction algorithm implemented in the iGraph library for manipulating graphs [15]. The reader is referred to [13] for details on this algorithm[14]. The results of community extraction on *FC*, *GL* and *Hugged* are shown in Table 3. We found that although *Hugged* and *GL* show very strong community structure, *FC* lacks this property. This is indicated by the low structure coefficient for *FC* (which is a lot less than 0.3). As a result, the maximum-size community for *FC* accounts for 72.6% of the users, whereas that for *GL* and *Hugged* accounts for less than 10%. Furthermore, we plot the community size distributions for *FC*, *GL*, and *Hugged* in Figures 10a, 10b, and 10c to show that the distribution of community sizes is quite biased for *FC*, while it exhibits a wider spread for both *GL* and *Hugged*. This result, then, clearly distinguishes *FC*, a social game, from *GL* and *Hugged*. We conjecture that different Facebook applications will exhibit different community structures based on the nature of user interactions (e.g., social gaming vs. non-gaming applications). We discuss the reason for the lack of community structure (in *FC*) later in Section 6.

For each application, we also measure the number of distinct geographical locations (countries) whose users constitute one community. We call this the *geographical diversity of communities*. For the sizable communities extracted, lower geographical diversity may hint at a possible solution with regards to scalability (e.g., by having distributed local servers). Figures 10d, 10e, and 10f show the geographical diversity for the 51 most sizable communities for *FC*, *GL*, and *Hugged*, respectively.

We found that instead of being geographically-focused, communities on our applications consisted of users in many diverse regions, and that there is a lack of relationship between the community sizes and number of contributing countries. On a related note, we also measured the number of users belonging to the *same* country for each community (termed 'local users'), and report the maximum number of local users per community for the 51 largest communities in Figures 10g, 10h, and 10i. Our results show that although for some communities the proportion of users with the same locality might be high, the proportion seems to vary considerably across communities for *GL* and *Hugged*. We may not, however, infer the same results for *FC*, where community sizes excluding the largest community are much smaller.

Considering Facebook-network locality per community, we found that Facebook's definition of networks[15], too, does not capture community structures on social applications, as can be seen by the large number of contributing networks per community in Figures 10j, 10k, and 10l. This is contrary to our expectations that users who are on the same, say, work-related network have a relatively higher degree of real-world interaction, which is expected to translate into online interaction, especially on OSNs. Furthermore, more users belonging to the same network are expected to be mutual friends than users belonging to different networks. This observation is supported by [6].

[14]Detection of community structure is an optimization problem for finding a division of vertices so that the resulting *structure coefficient* for the graph G is maximized.

[15]'Networks' in this context refer to Facebook's classification of users into networks of school, college, work, and regional categories.

With regards to scalability, an important metric for gauging the extent of grouping of users on our applications is the *clustering coefficient*. In particular, we look for the existence of high clustering of nodes on our applications, which may result in possible ways for segregated processing of data. We report the clustering coefficients for *FC*, *GL*, and *Hugged* in Table 3, and report clustering coefficients of Erd os-Rényi random graphs (with same number of nodes and edges) alongside for comparison. It can be seen that as compared to random graphs, clustering in interaction graphs for *FC*, *GL*, and *Hugged* is very high. High clustering combined with the low diameters for all three graphs (Table 3) means interaction networks on our applications are actually *small-world* networks [12]. While this may hint at solutions to scalability issues on social applications, we do not explore this issue further in this paper.

6. DISTINGUISHING GAMING APPLICATIONS

As mentioned before, social gaming is fast becoming a major category of applications on social sites such as Facebook. This is due to high user engagement resulting from the use of gaming applications in general. In order to gauge the reasons for differences in characteristics such as differing community structures for our gaming application, *FC* (as compared to *GL* and *Hugged*), we need to consider user behavior.

We performed a comparative analysis of daily bandwidth, DAU, and daily average time spent by users on site. Figure 11 shows a 60-day long snapshot of bandwidth consumption, daily unique users, and average time spent on site from *FC*, *GL*, and *Hugged*.

One can see that for *FC*, significant drops in DAU resulted in no significant decrease in bandwidth consumption for the 60-day period tracked. Plotting daily average time on site for *FC* shows that with a decline in the number of unique active users, the average time users spent on *FC* increased. Note that *GL* does not exhibit this characteristic[16].

Another distinguishing characteristic of *FC* comes to light if we consider the fraction of a user's friends that subscribe to applications. Users may appear in our trace either if they perform activities on other users through the applications, or other users perform activities targetting them. We plot the frequency of occurrence of users in our 3-week trace against the subscribing fraction of friends in Fig. 12. It can be seen that frequency of occurrence is related to the subscribing percentage of friends *only* for *FC*, a gaming application.

The last result highlights an important feature of *FC* (and perhaps social gaming applications in general): the probability of activity for a given user depends on the subscribing fraction of friends. This has implications in that social games may require higher 'warm up' time as compared to other social applications before 'achieving popularity'. However, since gaming activity is related to friends' subscription to the application, this also implies that users with many subscribing friends may find it more difficult to stop or even lower their involvement in the game. Considering the high average number of subscribing friends for *FC* in Figure 13, this, in fact, helps explain our previous result from Fig. 11.

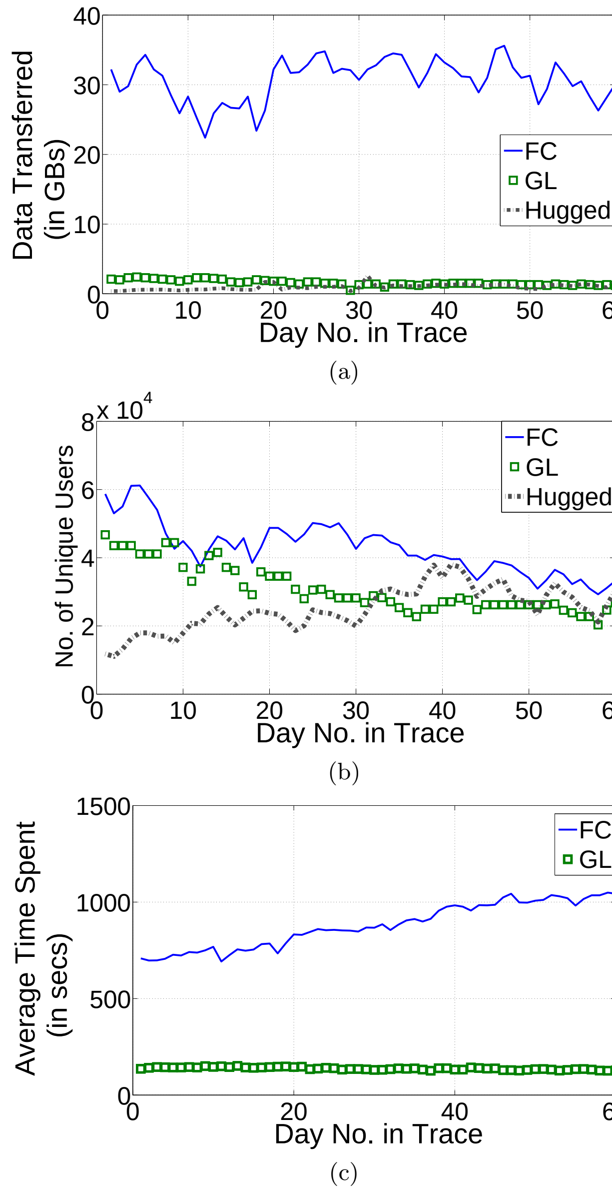

Figure 11: A two-month long snapshot of bandwidth and unique users for all applications, and average time spent on site for FC and GL.

Fig. 13 shows the average number of activities per user, average number of subscribing friends per user, and average number of total friends on Facebook (per user) for the subscribing users of *FC*, *GL*, and *Hugged*. We see that the average number of activities for *FC* is much higher than that for *Hugged* and *GL*. This makes intuitive sense since *FC* is an interactive multiplayer gaming application, and playing a game calls for multiple activites on part of a user. The

[16] Data for average time spent on site for *Hugged* was unavailable for the entire 60-day snapshot and hence is not shown in Fig. 11. However, the average time spent for *Hugged* is ∼2min 40s based on the data that *is* available. This matches average time spent on *GL* (∼2min 20s). Therefore, we expect the results for *GL* to be representative of *Hugged*.

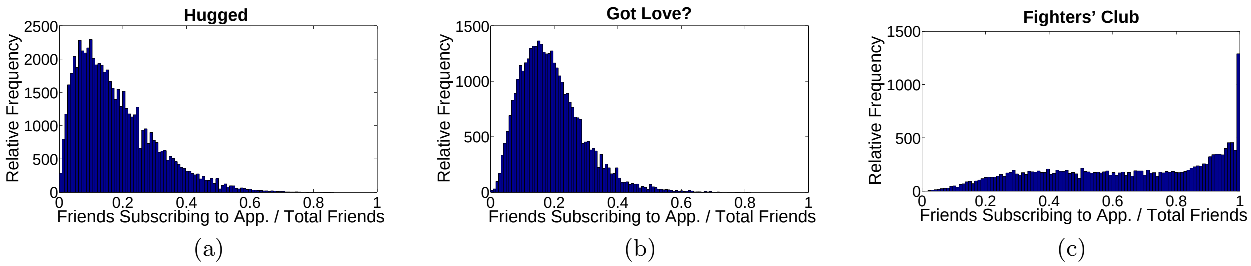

Figure 12: Frequency of occurrence in traces against percentage of friends subscribing to the same application.

somewhat surprising result, however, is that *FC* users on average have a greater number of friends overall, as well as have a greater number of friends *subscribing* to the application. Furthermore, the number of friends subscribing to the application relative to the total number of friends is greater for *FC* than for our non-gaming applications. We discuss this further in relation to community structures in Section 7.

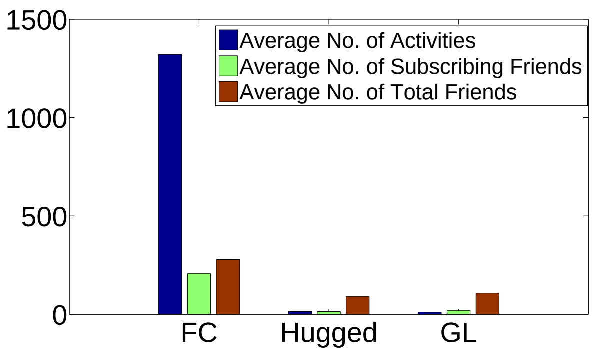

Figure 13: Aggregate averages for number of activities, number of total friends and number of subscribing friends for *FC*, *GL* and *Hugged*.

6.1 Comparing Social Games to Other Online Games

On the topic of distinguishing social gaming applications like *FC*, we noted the high probabilistic activity of users and its relation with the proportion of subscribing friends for the application. We further remarked at how usage might persist over time for individuals with a high proportion of subscribing friends online. Intuitively, we imagine the same phenomenon should occur for other online multiplayer games as well. In particular, online multiplayer games such as CounterStrike seem close in comparison to online gaming on social networks. However, there are major differences in the deployment of the two types of games, and these differences provide for some limitations for gaming on social networks given the status of developer platforms today.

One of the major differences between social games such as *FC* and games in CounterStrike's multiplayer genre is that having been developed specifically for OSNs, social games rely on HTTP/TCP traffic, whereas CounterStrike and the like employ UDP to cater to the bursty nature of the gameplay seen in online multiplayer games. It should be kept in mind that (at least with *FC*) we still see bursts of Internet traffic, especially when sizable games (i.e., games with high number of participating individuals) approach the end of their durations. Furthermore, traffic on social games is pull-based, whereas other online multiplayer games (for e.g., CounterStrike and Half-Life) use a push-based virtual broadcasting approach to disseminate information to players [3]. This clearly limits social games to non-realtime gameplay.

The study in [3] has highlighted the non-varying 24-hour pattern of traffic seen at gaming servers for CounterStrike, Half-Life, etc. This is not the case with social games, in that traffic is very closely determined by the OSN's geographical spread of users and their internet activity (i.e., we see a diurnal pattern, rather than a non-varying temporal pattern, of traffic for online social games). Furthermore, given the nature of social games and the wider audience of social networks, the geographical spread of users in social games is higher than that for other online multiplayer games such as CounterStrike.

7. CONCLUDING DISCUSSION

This paper presents a first look at the usage and subscription characteristics of OSN-based applications as well as the nature of interaction between users in the context of such applications. The data we gathered by launching our three indigenous applications using the Facebook Developer Platform represents a valuable resource since it provided us with first-hand information about such applications launched on an OSN.

Our analysis of global application usage data shows that applications, once popular, remain strong and tend to retain their rankings. It is natural for users to stumble upon more popular applications more often than, say, up-and-coming applications. This has implications for newcomers, most important of which is that as time passes and more applications amass popularity, it becomes increasingly harder for smaller applications to achieve similar popularity, i.e., an application tends to attract users based on how many existing users subscribe to/actively use it. We did, however, see exceptions where newly launched applications quickly gained popularity to climb rankings.

In our study, we specifically looked at interaction as seen on applications when considering formation of communities online. We saw that interaction graphs on *FC* do not exhibit community structure, while those for *GL* and *Hugged* exhibit strong community structures. One reason for the lack of community structure on *FC* is the tendency for FC users to form friendships with 'strangers' using the application. This has been witnessed on especially fighting-oriented social gaming applications, and we believe this is the primary reason for the high average number of subscribing friends (for *FC*) seen in Fig. 13. We believe this behavior distorts the underlying friendship graph of the social network, resulting *at least* in loss of (natural) community structure.

We also found that even though applications' interaction graphs had high clustering of nodes, the community structures extracted were formed of many diverse geographical locations. We showed that Facebook's definition of networks, which attempts to capture community structures, fails to lend meaning to communities (at least) on applications. Through anecdotal evidence, we also highlighted the fact that with high level of activity and the traffic generated, especially on viral social gaming applications such as *FC*, scalability issues come to light fairly early in the application's lifetime.

One way to alleviate scalability problems for high traffic web applications is to segregate information into non-overlapping (or minimally overlapping) chunks. These chunks can then be placed into different locations (for example, in a distributed database), primarily to increase speed of processing requests. Our results show that separating data geographically or network-wise does not help social applications. However, the existence of strong clustering in online activity hints at the existence of a possibility for reaching scalability through data segregation. Further study is needed to explore how exactly this can be achieved.

We believe our three applications provide a reasonably rich set of activity data for a meaningful analysis of social application usage. *FC*, *GL*, and *Hugged* are inherently different in the nature of their user interactions. For example, *GL* and *Hugged* only contain friend-to-friend interactions while *FC* involves interactions with non-friends as well. We also expect other social gaming applications (especially competition-based social games) to have characteristics that are more similar to *FC* than *GL* or *Hugged*. Recognizing that online social applications span a wide variety of categories, further studies are needed to investigate the degree to which our preliminary findings hold generally, as social networking paradigms continue to evolve. Furthermore, the degree to which our results apply to applications on OSNs with demographics different from Facebook (such as MySpace) also remains to be seen.

In summary, we feel this study is an important first step towards exploring a modern class of fast growing Internet applications and means of communication.

ACKNOWLEDGEMENTS

We thank Dr. Zartash Afzal Uzmi for motivating the initial development of the Facebook applications, Irma Qadir for the useful discussions on expected user behavior, Social Gaming Network (SGN) for allowing use of application data, and Developer Analytics for sharing the global application data. We also thank our shepherd, Dr. Stefan Saroiu, Dr. Balachander Krishnamurthy, and anonymous reviewers for their feedback.

8. REFERENCES

[1] N. Berger, C. Borgs, J. T. Chayes, and R. D. Kleinberg R. M. D'Souza and. Competition-induced preferential attachment. In *International Colloquium on Automata, Languages and Programming*, pages 208–221, 2004.

[2] Meeyoung Cha, Haewoon Kwak, Pablo Rodriguez, Yong-Yeol Ahn, and Sue Moon. I Tube, You Tube, Everybody Tubes: Analyzing the World's Largest User Generated Content Video System. In *Proc. Internet Measurement Conference*, 2007.

[3] Wu chang Feng, Francis Chang, Wu chi Feng, and Jonathan Walpole. A Traffic Characterization of Popular On-line Games. In *IEEE/ACM Trans. Networking*, Jun 2005.

[4] Raissa M. D'Souza, Christian Borgs, Jennifer T. Chayes, Noam Berger, and Robert D. Kleinberg. From the Cover: Emergence of tempered preferential attachment from optimization. *Proceedings of the National Academy of Sciences*, 104(15):6112–6117, 2007.

[5] Minas Gjoka, Michael Sirivianos, Athina Markopoulou, and Xiaowei Yang. Poking Facebook: Characterization of OSN Applications. In *Workshop on Online Social Networks*, 2008.

[6] Scott Golder, Dennis Wilkinson, and Bernardo A. Huberman. Rhythms of Social Interaction: Messaging within a Massive Online Network. In *International Conference on Communities and Technologies*, 2007.

[7] M. Granovetter. The strength of weak ties. *American Journal of Sociology*, 1973.

[8] David Liben-Nowell, Jasmine Novak, Ravi Kumar, Prabhakar Raghavan, and Andrew Tomkins. Geographic Routing in Social Networks. In *Proc. National Academy of Sciences*, 2007.

[9] S. Milgram. The small world problem. *Psychology Today*, 1967.

[10] Alan Mislove, Massimiliano Marcon, Krishna P. Gummadi, Peter Druschel, and Bobby Bhattacharjee. Measurement and Analysis of Online Social Networks. In *Proc. Internet Measurement Conference*, 2007.

[11] S. Mossa, M. Barth'el'emy, H. E. Stanley, and L. A. N. Amaral. Truncation of power law behavior in scale-free network models due to information filtering. In *Phys. Rev. Lett., (13)*, 2002.

[12] M.E.J. Newman. The Structure and Function of Complex Networks. In *SIAM Review*, 2003.

[13] M.E.J. Newman. Finding Community Structure in Networks Using the Eigenvectors of Matrices. In *Physical Review*, 2006.

[14] comscore. http://www.comscore.com/, Apr 2008.

[15] The igraph library for complex network research. http://cneurocvs.rmki.kfki.hu/igraph/, Apr 2008.

[16] Adonomics. http://www.adonomics.com, May 2008.

[17] Google analytics. http://analytics.google.com/, Apr 2008.

[18] Blackholes us. http://www.blackholes.us/, Apr 2008.

[19] Developer analytics. http://www.developeranalytics.com/, Apr 2008.

[20] Daum ucc. http://ucc.daum.net/, Apr 2008.

[21] Facebook. http://www.facebook.com/, Apr 2008.

[22] Facebook developer platform. http://developer.facebook.com/, Apr 2008.
[23] Facebook statistics. http://www.facebook.com/press/info.php?statistics, Apr 2008.
[24] Fighters' club. http://apps.facebook.com/fightersc/, Apr 2008.
[25] Flickr. http://www.flickr.com/, Apr 2008.
[26] Got love. http://apps.facebook.com/lovability/addlove/, Apr 2008.
[27] Hugged. http://apps.facebook.com/huggees/start, Apr 2008.
[28] Myspace. http://www.myspace.com/, Apr 2008.
[29] Orkut. http://www.orkut.com/, Apr 2008.
[30] Site snapshot: Facebook. http://siteanalytics.compete.com/facebook.com/, Apr 2008.
[31] Youtube. http://www.youtube.com/, Apr 2008.