Master's Thesis

# Facebook Traffic Data and cacheability

By

Sergi Ibern and Francesc Soler

Department of Electrical and Information Technology
Faculty of Engineering, LTH, Lund University
SE-221 00 Lund, Sweden

# Abstract

Nowadays, saving bandwidth means saving money. Because of that, ISPs and network operators are interested in giving the best service possible to their customers taking advantage of an efficient use of the network and its bandwidth. Due to Facebook is one of the most important social networks and it is also one of the most visited web pages, ISPs and network operators are interested in traffic identification. The main goal is to study how Facebook works, to study FB user behaviour and to cache content in order to save bandwidth.

The knowledge of FB is really useful because if we can identify which kinds of contents are more popular, it would be possible to predict potential cacheable content. For instance, if there is a FB user who every time posts a picture and then a lot of other users download his content, it would be interesting to cache it locally.

In this thesis there are some studies in order to know how Facebook works. First of all we have analyzed Facebook data traffic with Wireshark (only local traffic). It has been useful to create some filtering rules which let us to filter each part in Facebook like Pictures, Update Status, *Likes*, Chat or Videos. After that, we did a 7 days packet dump from a real network using PacketLogic (PL). The last step was creating some Python scripts in order to make statistics with all data.

From the thesis study, it has been found some really interesting results and conclusions about two parts of Facebook: Downloaded Pictures and *Likes*. It has been found that one kind of Downloaded pictures, Profile Pictures in Small size are the most downloaded and that are a potential content to be cached. It has also been possible to calculate the time-life of pictures and to study their timing. It has also been possible to identify the most requested uploader and to study the user behaviour with the most popular tool of Facebook: *the Like Button.*

# Acknowledgments

This Master's thesis would not exist without the support and guidance of our supervisor. We would like to thank Maria Kihl for giving us the opportunity to this thesis in the LTH. We also would like to thank Andreas Aurelius, Jie Li, Viktor Nordell, from Acreo AB, Manxing Du, a Master Thesis student also working with Acreo, and Åke Arvidsson and Lars Westberg, from Ericsson AB, for always giving us feedback and providing necessary information for our thesis.

Sergi Ibern
Francesc Soler

# Terminology and Abbreviations

**ANSI, ISO-8859-1**: ASCII-based standard character encoding created by the American National Standards Institute. It encodes each character as a single eight-bit code value.

**DHCP**: Dynamic Host Configuration Protocol is a network protocol, used to set up network devices so that they can communicate on an IP network.

**DPI**: Deep Packet Inspection, a form of network packet filtering that inspects data and header of packets.

**DSL**: Digital Subscriber Line, network access technology that provides internet access by transmitting digital data using telephone infrastructure.

**FB**: Facebook.

**FB picture ID**: Facebook picture Identifier.

**FB user ID**: Facebook user Identifier.

**FTP**: File Transfer Protocol. It is a standard network protocol used to exchange files through a TCP/IP based network.

**FTTH**: Fiber To The Home, network access technology based on Optical Fiber.

**GET**: HTTP request method. It requests a representation of the specified resource. Requests using GET should only retrieve data and should have no other effect. [16]

**GUI**: Graphical User Interface.

**Host**: a host is a computer connected to a computer network.

**HTTP**: HyperText Transfer Protocol.

**IP**: Internet Protocol. An IP Address is a numerical label assigned to the devices that are part of a computer network that uses Internet Protocol.

**ISP**: Internet Service Provider. Companies that provide Internet access to costumers.

**MAC**: Media Access Control address uniquely identifies a network adapter or an interface card.

**MySQL**: it is an open source relational database management system.

**OSI-7**: Open Systems Interconnection model. Network architecture model based on 7 layers: physical, data link, network, transport, session, presentation and application.

**P2P**: Peer- to-peer. A peer-to-peer network is a network of connected nodes that do not communicate according to the client-server model. All computers can act as a server and a client.

**PCAP FILES**: In the field of computer network administration, pcap (packet capture) consists of an application programming interface (API) for capturing network traffic. Unix-like systems implement pcap in the libpcap library and Windows uses a port of libpcap called WinPcap. [17]

**PDML**: Packet Details Markup Language. It is a very simple language that keeps the information of packets. It is used to create a detailed view of packets.

**PL**: PacketLogic.

**PL Python API**: PacketLogic Python Application Programming Interface is a programming interface provided by Procera Networks to connect to the PacketLogic appliance using Python

**POST**: HTTP request method. It submits data to be processed to the identified resource. The data is included in the body of the request. It may result the creation of a new resource, the update of a already existing resource or both. [16]

**QoE**: Quality of Experience. Subjective measure of a service based on costumer's experience.

**QoS**: Quality of Service. It refers to resource reservation control mechanisms rather than the achieved service quality.

**SNS**: Social Network Sites.

**UNIX Time**: system for describing instances in time, defined as the number of seconds that have elapsed since midnight Coordinated Universal Time (UTC), January 1, 1970, not counting leap seconds. [24]

**URI**: Uniform Resource Identifier. A string of characters used to identify a resource or a name.

**URL**: Uniform Resource Locator. It is a string of characters that is a reference of an Internet Resource.

**WWW**: World Wide Web.

# Table of Contents

# CHAPTER 1

# 1. Introduction

In this first chapter an introduction about this Master Thesis is presented in order to make more understandable its content.

## *1.1 Background*

The following section is an overview of three basic concepts to understand the content of this Master Thesis: Internet, Social Networks and Facebook.

### 1.1.1 Internet

In today's world, Internet has become one of the most important communication ways to share content and to communicate with each other. Since the emergence of Internet, its usage has changed for different reasons. On the other hand, innovation and development are driven by different factors. These factors can be political (laws), commercial (products, patents), social (behaviour), etc.

Broadband access networks introduced new applications and services, so the daily usage has been changed since then. It has gone from the traditional World Wide Web (WWW) usage (web browsing) to triple-play usage, where households have all their communication services (phone, data and TV) through their broadband access connection.

In a recent study [1], it can be observed that for the last 5 years, *file sharing* is always the most relevant traffic, followed by *streaming media* and *web browsing*. Note that file sharing includes Peer to Peer (P2P), but it does not

include Hypertext Transfer Protocol (HTTP)-based direct download *file sharing* traffic offered by Megaupload or Rapidshare. With the closure of Megaupload and other sites like that, P2P traffic may increase. Inside *web browsing* traffic, Facebook is one of the top traffic generators.

Moreover, measuring and analyzing Internet traffic is not an easy job. There are some obstacles and limitations in traffic measurement listed below:

- Internet is a huge and complex system. There are 2.000 millions of users [2], which represent one third of the world population. Although OSI-7 layer model used in Internet nowadays, the variety of protocols adds more complexity to analyze this kind of data.
- Another important point is the user privacy. It is very important to take care with all private information like IP or MAC addresses, as they have to be hashed in order to do not violate any privacy law.
- Although only filtered data are stored, it is a huge amount of data.
- Data analyze and processing requires a lot of time. The execution of the scripts used to parse files can take several days.

### 1.1.2 Social networks

Since some years ago, with the appearance of the online social networks, the concept of Internet has changed. Internet now is more than a communication channel like television or telephone. It has become a tool that allows to Humankind to share all kinds of contents in real-time around the world.

Social networks have introduced a much more human factor to the Internet and its traffic. This traffic contains some people behaviour patterns that are interesting to study in order to have information about human behaviour in front of these social networks. Although there are a lot of online social networks like, for instance, Facebook, Twitter, LinkedIn, MySpace or Google Plus, this project is focused on Facebook.

### 1.1.3 Facebook

Facebook is nowadays the most important social network in the world. Almost all young people use this network as a tool to share contents like pictures, videos and their thoughts and opinions with their Facebook "friends". But Facebook has also become a very important tool for big companies and famous people to share contents and news with Facebook users. This social network is also really important for all kind of companies as a marketing tool. If they know how to penetrate to users (for instance, getting the "Like" in their Facebook profile) these companies will have free marketing campaigns. Hence, when a company wants to announce a new product, thousands of users will receive the announcement without any cost.

Facebook has more than 900 million users [3] around the world: more than 234 millions in North America, almost 225 millions in Europe and more than 224 millions in Asia [4].

Since the birth of this social network in 2004, Facebook popularity has increased more and more every year as it can be seen in Figure 1.1.

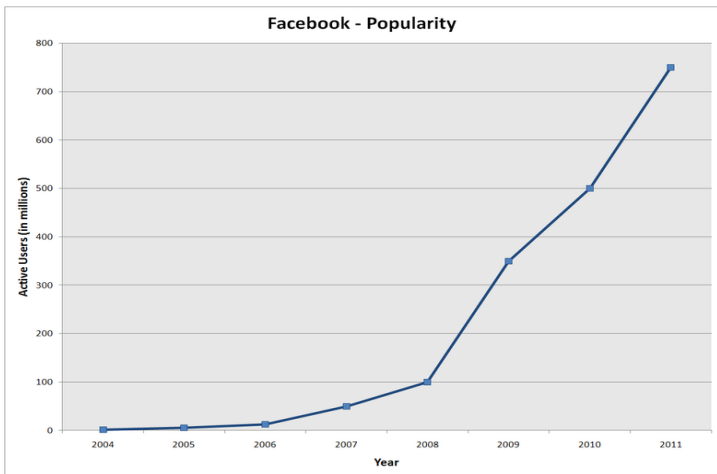


*Figure 1.1. Facebook Popularity (Users-Year) [3]*

That increase of Facebook users means that the amount of data traffic of Facebook website has increased brutally too during the last years, becoming one of the most visited websites.

The main conclusion is that, even Facebook is not responsible of a big percentage of Internet traffic such as other websites (i.e., Youtube) or P2P traffic can be, it is increasing every year and is one of the most visited websites around the world. Facebook traffic is increasing more and more, so ISPs are really interested in how to manage efficiently all this traffic in order to save bandwidth, especially in mobile networks.

## 1.2 Motivations and goals

Bandwidth is an expensive and limited resource. ISPs have to use it in an efficient way. As it has been commented before, Facebook is one of the most visited *web-browsing* websites, so it generates a huge amount of traffic.

If the most popular users can be identified, it is easy to predict that when these users upload some content, this content will be downloaded by a lot of users from the same network. In order to save bandwidth, it can be useful to cache this popular content close to users. It can be cached locally, like in the same device or in a base station.

Besides, it can be useful to know if there are groups of people that always are downloading content from the same people. It also would be interesting to know how Facebook works, how its traffic is and compare traffic patterns of different users in order to know their behaviour.

## 1.3 Limitations

This kind of work has some limitations, because real networks have lots of users, and each user generates a lot of traffic.

All raw data have to be parsed, and processing hundreds (or thousands) of GB requires time as well. Then, when this data is parsed we obtain big text files with sorted data, but this data has to be post processed. It is very difficult to work with full months of raw data, because each script process can take several days. Because of that, the packet dump done for this thesis is a 7 days packet dump (with 5 full days).

Another important point is that results obtained are sorted by MAC addresses and that it has been considered that one MAC address means one user, but this is not always true because one device can be used by more than

one Facebook user. Otherwise, when in this thesis appears “Facebook user identification” (FB user ID) it is totally reliable that this ID belongs to a unique FB user. MAC addresses and FB user ID have been hashed in order to protect users privacy.

## 1.4 Overview of Thesis

This project is called "Facebook Data Traffic and Cacheability" and it has been divided in two different parts.

The first part of this thesis is a research project about Facebook in order to know how this website works and how the data traffic looks like. Facebook website and traffic have been studied deeply in order to find the different important parts that this social network has and to analyse the data traffic of these different parts.

In the second part of this project we use this knowledge of Facebook traffic data to parse and filter a big amount real traffic data of real networks in order to know, first of all, the content that users download and if it would be possible and useful to cache some of this content in order to save bandwidth, and also to know the Facebook users behaviour.

This project has been developed at Lund University in cooperation with Acreo AB, a Swedish research institute, and the company Ericsson AB. It is also a part of a bigger European project called IPNQSIS (IP Network Monitoring for Quality of Service Intelligent Support)[5]. IPNQSIS is a project within the Celtic Research and Development programme. The main goal is to study the behaviour of Quality of Experience (QoE) through the analysis of network and service performance and their impact on end customers. Celtic [6] is a European programme designed to strengthen Europe’s insight and competitiveness in telecommunications. Celtic-Plus is a Eureka ICT cluster and is part of the inter-governmental Eureka network.

# CHAPTER 2

## 2. Previous work

The constant growing of some Internet applications traffic has aroused interest since last years. In [1] it can be seen how residential broadband Internet traffic has increased during the last 5 years. The daily web-browsing traffic volume per end user has increased by about 300%, and this can be, partly, because of Facebook, considering that Facebook is one of the most relevant web-browsing traffic. Because of that, network operators and ISPs are interested in knowing traffic patterns, user characterization and user behaviour in order to manage the network efficiently and to find the best Quality of Service (QoS) and QoE possible to end users.

End users also change their habits. Some years ago users read newspapers, watched TV or made phone calls, but nowadays they can do all of this via Internet. Besides, the emergence of Internet applications like Facebook, Twitter, BitTorrent, Skype, Voddler, Youtube, Vimeo, Spotify, Megaupload, etc, has forced to adapt networks for this traffic increase.

In [7] there is a study of the daily traffic pattern. The lowest activity was found between 5 a.m. and 6 a.m., and the highest activity was at night, around 9 p.m. File sharing is more or less responsible for the pattern because it represents the 74% of all traffic, followed by streaming media with 7.6%. BitTorrent is the most used file sharing application with 94% of volume ratio.

As it can be seen in [8], HTTP streaming has become one of the most used IP services in Internet. One of the most curious things is that 10% of all users are responsible for 50% of the total HTTP streaming traffic and 50% of the connections. In [9], the Swedish P2P video service Voddler and the Swedish P2P music service Spotify are especially studied. 20% of local hosts use Voddler and 65% of local hosts use Spotify.

Youtube is the largest video sharing site, and it has more than 100 million clips watched per day. In [10] there is a study about how Youtube works and how to cache locally at the client. Local caching at the client is similar to the process of local caching web pages at the local browser's cache. Youtube users watch the same video more than once, so bandwidth can be saved and startup delay and disruption of the video playout can be avoided if the video is already cached. Furthermore, the default cache space for web browsers is on the order of 50 MB and if the average payload size of a video is around 7 MB several videos can be cached.

Online games have been studied in some reports as well. For instance, [11,12] talk about the famous Massively Multiplayer Online Role-Playing Game (MMORPG) called World of Warcraft (WoW). This game has more than 11 million players around the world.

The study of user behaviour is not only useful for networks operators and ISPs. With all social networks in Internet, a lot of companies have seen them as a really important marketing tool. In [13] some networks like Orkut, MySpace, Hi5 and LinkedIn are analyzed.

There are some studies [14,15] which are more social than technological. For instance, in [14] there is an estimation of how many adults and teenagers in America use Social Network Sites (SNS), and one of their results is that half of the adults and three-quarters of the teenagers use SNS. It also talks about the average number of Facebook friends, statistics about "Likes" and so on. [15] has some behaviour studies related with politics, strangers in Facebook friends, private messages, etc, separated by age and sex.

# CHAPTER 3

# 3 Facebook Parts and Facebook filtering rules

This chapter is focused on a first study of Facebook traffic data analyzing its traffic locally. The most important parts of FB have been split and studied separately. In order to split these different parts, filtering rules have been created and tested.

## *3.1. Method to analyze and to study Facebook parts*

In order to know how Facebook traffic data looks like and to identify all the different interesting parts that this online social network has, the program used has been Wireshark. Wireshark is a network protocol analyzer which allows capturing and interactively browsing all traffic running on a computer network [18]. This software is very useful to watch, to study and to parse the data traffic that is sent and received from a computer. The packets captured by Wireshark also give a lot of information of the different OSI layers.

It is possible to extract the MAC address from the information captured from the Data Link Layer, to extract the IP source address and the IP destination address from the Network Layer or to extract the FULL URL of the packets that use HTTP protocol form the Application Layer and also to know the Request Method of the HTTP packets.

During the first part of our analysis, in which the goal is to know how to parse different parts of Facebook and to know how this website works, we have focused in the FULL URL that gives us a lot of information. Furthermore, it has been also really useful to know which kind of HTTP Request Method that is used in different parts (most of times, GET or POST).

Finally, in order to do different tests and captures of Facebook data traffic, we have created two real Facebook accounts. It has been useful to do tests such as uploading pictures, talking to each other via chat, or watching pictures uploaded by the other "unreal" Facebook user.

The tests and captures have been done from one computer connected to Internet and connecting to Facebook website. At the same time we have captured the traffic of that computer with Wireshark, creating *pcap* files. With these files created, then we can create filtering rules and get a lot of information of the packets captured (Figure 3.1.)

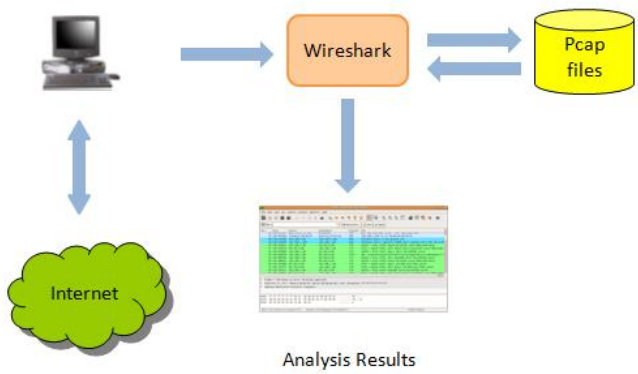


*Figure 3.1. Scheme of capturing method with Wireshark*

## 3.2. Facebook filtering rules

In this section the filtering rules about the principal parts of Facebook SNS are presented. We have divided it in groups or parts that represent an important part of Facebook traffic data: pictures, likes, chat, status updates and videos. Each part of this section includes an explanation about each Facebook part followed by the Wireshark filtering rule created to parse it.

### 3.2.1. Pictures

Pictures are one of the most important and popular parts that can be found in Facebook. There are lots of pictures that appear during a Facebook session. Basically, there are user pictures (pictures uploaded by FB users), advertisements and Facebook icons.

#### 3.2.1.1. Downloaded pictures

Downloaded pictures are the pictures downloaded during an ongoing FB session. Some of these pictures are downloaded automatically when the session is started and some of them are downloaded voluntarily by FB users.

These kinds of pictures are requested by GET requests which contain picture URL. After the request is sent, the picture is received using TCP. In addition the link included in the GET request contains the name of the picture and it can be useful to know how many times a picture appears in a FB session.

In a FB session most of pictures downloaded are pictures from the Main Page including icons, advertisements and pictures from other FB users in small size. These different pictures types are usually stored in different FB servers. The servers where pictures are stored appear in the URL of the GET requests, so it is possible to know them. Besides, most of FB pictures include an extension at the end of their names which provides information about the type of picture. Matching the server and the extension, it has been possible to classify all the downloaded Facebook pictures. This classification is attached in *Annex 1,* where all the pictures types can be observed in detail.

Finally, regarding that downloaded pictures can have four different formats (*jpg, gif, png* and *tif*) and that downloaded pictures information is in GET requests, it has been possible to create a filtering rule which parses these kinds of pictures.

> Downloaded Pictures Filtering Rule: *http.request.method == GET and (http.request.uri contains ".jpg" or http.request.uri contains ".gif" or http.request.uri contains ".png" or http.request.uri contains ".tif")*

#### 3.2.1.2. Uploaded pictures

Uploaded pictures are also important in a Facebook session. FB users usually upload pictures or albums in their profile in order to share them with other users.

When a FB user uploads a picture (profile picture, album picture, etc), it is possible to analyze the traffic created by the upload to the Facebook Server.

There are two ways to upload a picture in Facebook. FB users can upload a single picture from their wall or they can upload a picture creating an album (even though they only upload a single picture).

In both cases the original name of the picture (the name that the picture had in the computer before being uploaded) can be found in a POST request sent by FB user, more specifically in the MIME (Multipart Multimedia Encapsulation) part of this packet.

In order to parse this kind of picture we have created a filtering rule which parses the POST requests that contains the uploaded picture. This filter rule parses the packets that content the Full Request URI:

> Uploaded Pictures Filtering Rule: *http.request.method == GET and http.request.uri contains "photos_upload"*

#### 3.2.1.3. Matching upload and download pictures

Once the picture is uploaded in the Facebook server, name and size of the picture change. If FB users download the picture afterwards, GET requests containing picture information will contain the new name of the picture. This new name is a long name with numbers including a *fbid* that identifies the picture and a FB user ID of the user who posted the picture. This Facebook user ID is unique.

For instance, a picture originally called *example.jpg* (in the computer of FB user who uploaded that picture), after uploading it has the following name in the FB server:

*306252_119960981467414_100003605398234_87717_*

*1745492665_n.jpg*

In the example, the *fbid* would be the second number, *119960981467414,* and the FB user ID would be the third number, *100003605398234.* Note that

all pictures and links used are from the FB accounts we created to do tests and small captures.

There are no packets with both names (the original from the computer and the new one). It has not been possible to filter the packets containing the upload information and the ones containing the download information without knowing both names before filtering them.

Furthermore, when a FB user uploads a picture, after sending the POST request with the real name of the picture, the picture is loaded in small size in the profile, so a GET with the new name of the picture is done just after we upload the picture. The main problem is that between the POST with the original name and the GET can appear packets and other GET or POST request and it has not been possible to create a filtering rule matching them.

In conclusion, it is possible to parse downloaded pictures and uploaded pictures separately but not to know, for instance, if a picture uploaded will be downloaded later, because picture name and size changes.

### 3.2.2. Likes

The *Like* button is a very important and common Facebook tool used among FB users that share different kind of content such as pictures, status or comments. During a short FB session users can click *Like* in many different publications of other user walls.

Each *Like* is always contained in a POST request as a result of 3 Reassembled TCP packets. *Like* POST requests always contain the same Full Request URI: *http://www.facebook.com/ajax/ufi/modify.php?__a=1*.

Based on this characteristic it has been possible to create a filtering rule which parses only *Likes* posted.

> Like filtering rule: *http.request.uri contains "ajax/ufi/modify.php?_a=1" and http.request.method == POST and data-text-lines contains "like" and frame contains "like"*

Note that in the POST packet containing the *Like*, we can know who posts the like but not to which content the *Like* is posted.

### 3.2.3. Chat

Facebook chat is one of the most popular and used parts during a Facebook session. This tool allows FB users to communicate with their FB friends in real time. Even some other parts of Facebook like pictures or videos are bigger in amount of data, chat is important because is one of the most used tools during a Facebook session.

Chat messages are sent in POST requests. In the POST requests can be found the content of the message and some other interesting information. It is possible to identify the user who sent the chat message, the user who receives the message, the full content of the message and other information like the link of the profile picture which appears in the chat window.

It must be said that FB chat uses a codification to send the content of chat messages. For instance, spaces of the messages are shown as "%20" instead of " ". This is because space is an unsafe URL character and these kinds of characters are encoded in ANSI, ISO-8859-1. The "%" indicates encoding and the two following characters indicate the hexadecimal code of the character.

Moreover, it has been possible to find a pattern of how FB chat works. First of all, the encoded message is sent to a FB IP address. Then, Facebook sends the same message sent before but without ANSI, ISO-8859-1 codification. After that, the answer of the other FB user talking in the chat is received from the same Facebook IP address.

In conclusion, the repeated messages that are received without ANSI, ISO-8859-1 codification are the messages that are shown in the chat window.

The filtering rule which makes possible to filter the messages sent and the answers received is presented below:

> Chat filtering rule: *http contains "www.facebook.com" and data-text-lines contains "text"*

### 3.2.4. Status Updates

Status updates is also a tool widely used by FB users. They allow Facebook users to communicate between them and to post their opinion in the FB wall. Status updates are sent in POST requests. These packets contain always a specified URI as Full Request URI. In addition, it is possible to extract the update status because it appears in a field called "xhpc_message_text" included in the POST request. Status updates appear encoded in ANSI, iso-8859-1 and are always truncated to 79 characters. If the message is shorter than 79 characters, it is possible to see the full message. Otherwise, only the firsts 79 encoded characters can be extracted.

The filtering rule obtained and which makes possible to parse the status updates is:

> Status filtering rule: *http contains "www.facebook.com" and data-text-lines contains "xhpc_message_text"*

### 3.2.5. Videos

Another content that can be found in this social network are videos. The videos from Facebook (stored in Facebook servers) are videos uploaded by FB users in the same way they upload pictures.

It must be said that most of the videos that appear in Facebook are not stored in a Facebook server. At the beginning of Facebook, some years ago, there were more videos stored in Facebook servers, but now, most of the videos that can be watched in Facebook are external links from Youtube, Vimeo or other video websites.

Facebook videos are always stored in FB server with the same URL (*http://video.ak.fbcdn.net)* and the format of FB videos is *.mp4*. These two characteristics have been useful to create the filtering rule which parses all the FB videos.

- Video Filtering Rule: *http.request.uri contains ".mp4" and http.request.uri contains "video.ak.fbcdn.net"*

# CHAPTER 4

# 4 Facebook Traffic Measurements

Thus far, we have focused on filtering and the identification of the different parts of FB considering our own traffic. The next step is to capture data from a real network. This chapter explains in detail the network from which data have been extracted, the method and the exact duration of this packet dump.

## *4.1 Overview of the target network*

The network, in which the measurements and captures were done, is a medium-sized municipal network in the south of Sweden [7]. There are approximately 2500 Fiber To The Home (FTTH) households connected to the network and a small number (~100) of Digital Subscriber Line (DSL) and some enterprise and campus end users. It is an open network, so there are some ISPs to choose from, and each ISP offers a set of subscription types with the maximum symmetric access speed at 100 Mbps, so customers can freely choose among the different services offered by the different providers [1].

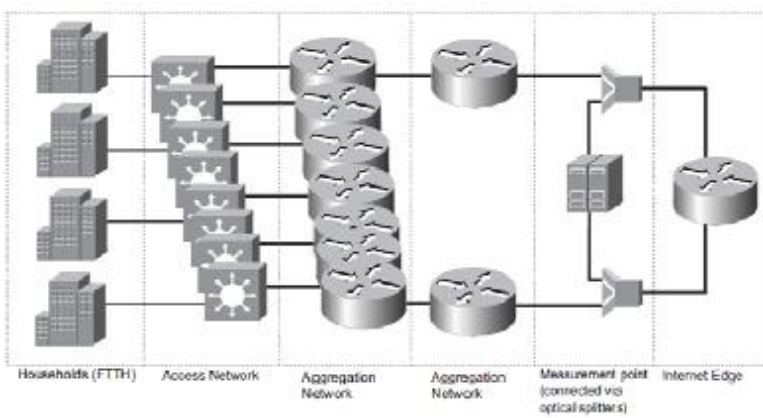

*Figure 4. 1. Scheme of the Swedish municipal network where the packet dump has been done. Thick lines denote 1 Gbps line rates and thin lines denote 100 Mbps line rates [9]*

## 4.2 Measurement procedures (Packet Logic)

The measurements were performed using a traffic data collection tool called PacketLogic (PL)[20]. This tool is a commercial real-time hardware/software used for traffic surveillance, traffic shaping or as a firewall. In PL, traffic is identified based on packet content (deep packet inspection and deep flow inspection) instead of port definitions.

The PacketLogic client is an application where a user can connect to PacketLogic server through a Graphical User Interface (GUI). When user is connected, is possible to get a detailed view of the network (in real time or from previous measurements). All measurements can be grouped based on hosts application signatures or protocols. This tool can identify more than 1000 Internet application protocols, and the signature database is continuously updated. PL uses the traffic in both directions in the identification process, and records all traffic (inbound and outbound) that pass through it every five minutes in the form of the traffic volume, the traffic application, the actual timing and the IP address for each traffic record during this 5 minutes time period. It also has a statistics database that records the average amount of traffic in the inbound and outbound directions.

The measurement equipment was connected to this municipal network via optical 50/50 splitters as we can see in Figure 4.1. It splits the optical signal in 2 exact signal copies, so the traffic in the network is not affected by the measurement device. The measurement point is the Internet Edge (IE) aggregation point, where ISPs are connected to the network [8].

## 4.3 The traffic database

In order to study the target network traffic statistics, a MySQL database was established to store the original traffic collected by PL using the Python API.

Data from the PacketLogic is stored in a table called PL. The information in PL is composed by date and time (in 5 minutes intervals), hashed IP address (in order to keep privacy and ensuring that no violation of integrity or law is done), application, protocol, inbound traffic and outbound traffic.

It also stores in the same database the log data of the DHCP server of the network, that contains information as date and time, hashed IP address, broadband service subscription type, access switch and access port.

Since the network uses DHCP, an IP address cannot directly be related to a household, because IP addresses are dynamic so they can change over time (with a minimum lease time of 20 minutes) and a household can use more than one IP address. Because of that, it is necessary to match hashed IP address with timestamps in both tables in order to know the hashed MAC address.

## *4.4 Packet dump and data structure*

We created a filter rule with the PL client in order to filter only the Facebook traffic data from the municipal network commented above. This filter rule filtered all traffic which contains **facebook.com** and **fbcdn.net** in the URL.

We did a big packet dump that started on 22nd of May 2012 at 13:57:19 and finished on 28th of May of 2012 at 09:03:19, so it was 5 full days that allow us doing some statistics per day and per hour. It was a total of 234.3GB (11205 *pcap* files). After that, we downloaded this raw data (*pcap* files) from a FTP server and we stored it in an external hard drive for further analysis.

# CHAPTER 5

# 5 Facebook data Traffic Analysis

The content of this chapter is the explanation of the methodology and steps followed to get results, to analyze and to parse the big amount of data obtained from the South Network (*see Chapter 4, Section 4.1*.).

## *5.1 Analysis tools and methodology*

Once we have created a suitable PacketLogic filtering rule to parse all the Facebook traffic data and we have obtained the packet dump of almost a week, the next step is to analyze, to parse and to extract valuable information. In order to parse the *pcap files* containing the Facebook data traffic, we have created scripts using *Python 2.7* as programming language in Linux OS.

Python is a remarkably powerful dynamic programming language that is used in a wide variety of application domains and fields [21]. This programming language features a fully dynamic type system and automatic memory management.

We have chosen Python because all the libraries and APIs are under open license (they are freely usable). It also means that a lot of free information and documentation can be easily found. Furthermore, Python uses clear and readable syntax and is often used as scripting language. Besides, Python can be executed in Windows and Linux without changing any lines of the codes. It has been really helpful because we have based in some scripts created in Windows and we have used and executed them in Linux without any problem. Moreover, Python includes multiprocessing module that makes

possible to scale Python programs to multiple cores. Regarding a big amount of data is going to be parsed, is very useful to be able to execute some parts of the scripts concurrently.

The second important software used has been Tshark. Tshark is a network protocol analyzer included in Wireshark that works by command line. Tshark is also a network protocol analyzer that enables the capture of packet data from live network or to read packets from a previous saved captured (*pcap*) file. Tshark is able to detect, read and write the same capture files that are supported by Wireshark and is able to parse the captures files with the same filtering rules than Wireshark [22].

Tshark has been used in the Python scripts created in order to open and read the *pcap* files in PDML format and then to parse and extract the appropriate information. It has been really important to be able to read the capture files in this format to extract in an easy way the data needed for the different parsing we have done.

It must be said that only the scripts which parse information directly from the captured files use Tshark and that all the scripts created give as result text files with different information.

Finally, the third important tool used to analyze and get results from the captured files has been Microsoft Excel.

Microsoft Excel is a software program developed by Microsoft Corporation. It is a spreadsheet program that allows users to organize, format and calculate data with formulas using a spread system broken up by rows and columns.

Although, there is other similar open software, we have used Excel because we have a broad knowledge about it and because it is a powerful tool to calculate and create statistics and ranking of a big amount of data.

## 5.2 Facebook parsed parts

This section explains the steps followed to parse and obtain information from the big packet dump done. Using Python scripts it has been possible to extract information useful to do statistics and obtain results about two major parts of Facebook: *Downloaded Pictures* and *Likes*.

### 5.2.1 Downloaded Pictures

Facebook pictures constitute a very large part of the FB traffic. During a Facebook session many pictures of different types and sizes are downloaded. Besides, FB pictures can give a lot of important information about FB user behaviour.

First of all, in order to parse all the pictures that appear in Facebook data traffic, the first step has been to create a Python script.

This script opens and reads the *pcap* files obtained from the South Network and parses all the packets that contain Facebook pictures. To do that, the script opens the *pcap* files with Tshark and applies the filtering rule seen in *Chapter 3, section 3.2.1.* This filtering rule parses all GET request packets which contain the URL of the pictures downloaded by Facebook users. In these packets there is a lot of information that needs to be extracted and written in the result text file. Figure 5.1. shows the part of the script where the filtering rule is applied.

```
def xml_pcap(pcap_file):
    proc_play = subprocess.Popen(["tshark", "-r", pcap_file,
"-T", "pdml", 'http.request.method == GET and
(http.request.uri  contains ".jpg" or http.request.uri
contains ".gif" or http.request.uri contains ".png" or
http.request.uri contains ".tif")'], stdout=subprocess.PIPE)
    output_play, errors_play = proc_play.communicate()

    if output_play.rfind("</pdml>") == -1:
        return output_play + "</pdml>"
                    return output_play
```

*Figure 5.1. Part of the script that opens Tshark and applies the filtering rule of pictures*

On one hand, only in the URL, there is a lot of important information about the picture. Picture names are the last part of the URL link. In addition, the Facebook User pictures name contains Facebook uploader identification. Extracting this FB user ID, it is possible to know who uploaded the picture. Besides, the URL includes the server where the picture is stored. As it was explained in *Chapter 3, section 3.2.1*, with the server (and also with the

picture extension) is possible to classify the different type of Facebook pictures.

On the other hand, there is a lot of other useful information of the packets which can be extracted from the pictures filtered. The created scripts extract information about the downloaded picture such as the IP, the device where picture has been downloaded and the browser used by the downloader. We have also extracted the time stamp, in UNIX time format. In order to identify the downloaders it is necessary to get the MAC address too. These addresses cannot be extracted directly from the packet. In this case, Acreo AB provided us the MAC addresses matching the time stamp and the IP addresses. Once all this information about the picture and the downloader is parsed, it is written in a text file (one text file for each *pcap* file). This resulting text file contains all the Facebook Pictures downloaded by all the Facebook users from the South Network during almost seven days.

In addition, in order to classify all pictures and to get statistics of each kind of picture, a script very similar to the first one has been created. This has been done directly from the *pcap* files and it gives as result a text file with the number of each type of picture based of the classification that can be seen in *Annex 1*.

### 5.2.2 User pictures

From the text file which contains all the pictures it is possible to parse only the User Pictures. User Facebook pictures are the pictures uploaded by a FB user and then downloaded by other users. As it has been seen in *Chapter 3, section 3.2.1.*, user pictures include a FB user ID. So, in order to parse only user pictures, a second script has been created parsing only the pictures with a correct FB user ID.

The next step was to parse and filter the large text file generated with different scripts in order to obtain different results. Figure 5.2 is a small example of the most popular pictures sorted by number of requests and the time-life of each picture (time between the first and the last download). The format of this text file is:

*Number of downloads / picture name anonymized / time-life (seconds)*

```
3204    _q.jpg  500228.913001
2641    _n.jpg  459536.067
2638    _q.jpg  459536.256999
2624    _q.jpg  499659.229
2190    _s.jpg  498839.581
2133    _s.jpg  396203.623
1823    _q.jpg  466472.115
1812    _q.jpg  499346.405
```

*Figure 5.2. Small example of the text file obtained sorting the most popular pictures.*

### 5.2.3 Likes

*Likes* are also a really important tool used in Facebook. When a user shares some content like pictures, comments or status, the user's FB friends can post a *Like* on them clicking the content. *Like* is one of the "special" tools that this social network offers, it is interesting to get some real statistics and numbers about this Facebook tool in order to know also users behaviour.

To parse all *Likes* done during a week, a script has been created using the filtering rule found in *Chapter 3, section 3.2.2*. As it has been done with pictures, the script parses the time stamp of the POST *Like* request and also IP address, device and browser.

```
def xml_pcap(pcap_file):
    proc_play = subprocess.Popen(["tshark", "-r", pcap_file,
"-T",        "pdml",        'http.request.uri        contains
"ajax/ufi/modify.php?__a=1" and http.request.method == POST
and data-text-lines contains "like" and frame contains
"like"'], stdout=subprocess.PIPE)
 output_play, errors_play = proc_play.communicate()
    if output_play.rfind("</pdml>") == -1:
        return output_play + "</pdml>"
 return output_play
```

*Figure 5.3. Part of the script that opens Tshark and applies the filtering rule of Likes.*

Finally, the MAC address is obtained from the Acreo AB database in the same way as with pictures. After that, another script has been created which

adds all the repeated MAC addresses, obtaining as result a text file with the number of *Likes* posted by each MAC address and time stamps (time when the *Like* was posted) following the next format:

*Hashed MAC Address / number of Likes / epoch times (seconds)*

A small example of this text file can be observed in Figure 5.4.

```
12b3263ecec53ca1b1c6d6a037c3a57d      8
        1337688734.223520000   1337703954.152524000
        1337703956.935520000   1337719276.402521000
        1337806335.200523000   1337889066.669520000
        1337963028.667520000   1338047229.192527000
d923f884b6b68b6b6799f9e319a4dc3c      5
        1337689176.286526000   1337715231.573523000
        1337854049.810521000   1337854291.718522000
        1337854510.003525000
```

*Figure 5.4. Small example of the text file showing the number of Likes for each hashed MAC address.*

# CHAPTER 6

# 6 Results and analysis

In this chapter we present the results and all the analysis done with the data obtained until this point. The results presented are all from the packet dump done from the South Network during the last week of May 2012, during approximately 7 days (see details in *Chapter 4, section 4.4*). As it has been explained in previous chapter, the results are focused on two really important parts of this social network: pictures, because they correspond to most of the traffic volume and can give a lot of information about Facebook, and *Likes*, a very important and used tool in Facebook.

## *6.1 Overview of South Network in May 2012*

During the month of May it has been possible to get some global statistics about the websites visited by all users of the South Network. It has been useful to have a global idea about which websites generate more traffic and in which positions are Facebook and Facebook servers in this ranking of websites.

The first classification is focused on the website domain. In this classification the most important domains are ".com" and ".net". The domain ".com" is the most important domain and is responsible of 12.9 Tebibytes (TiB) (inbound traffic plus outbound traffic). In second position appears ".net" which generates a total of 1.6 TiB.

It is really interesting to focus on these two domains because, besides being the domains which generates more traffic, all the Facebook traffic data is in these domains. Facebook website is *www.facebook.com* so a lot of traffic from Facebook comes from this URL, but there are a lot of URLs from Facebook servers (most of them store FB users pictures) which contain "fbcdn.net" in their URL.

About the ".com" domain, during the month of May, the website which has generated the most traffic has been *www.youtube.com* followed by *www.apple.com*. The domain *www.facebook.com* appears in 7th position and has generated a total of 0.2 TiB. Note that the most visited website in the world [25], *www.google.com,* appears in the 8th position with 0.1 TiB generated. It means that this ranking is focused on the traffic data generated and for this reason the most visited website in the world appears in 8th position and in first place there is "Youtube" a video streaming website which generates much more traffic.

About the ".net" domain, the URLs finished by "fbcdn.net" are the most important ".net" traffic generators with a total of 256 Gigibytes (GiB). Some examples of ".net" Facebook URLs that appear in the classification are "ak.fbcdnt.net", "hphotos-ash1.fbcdn.net" o "s-hprofilie-ash2.fbcdn.net". The most relevant is "ak.fbcdn.net" and the others can be negligible because of the very small percentage that they represent.

The main conclusion is that Facebook website and Facebook servers are very relevant in terms of traffic generators and that Facebook servers generate more or less the same amount of traffic as *www.facebook.com*. This is understandable because these servers store lots of user pictures, so they store lots of data. With these general statistics, the next step is to present the results obtained about Facebook traffic from the South Network.

## *6.2 Downloaded pictures results*

In this section we present the results obtained from Facebook pictures which were downloaded during the period of the packet dump. In the first part we have done a classification of all the downloaded pictures in order to know the different kinds and percentages of downloaded pictures. In the second part, we present a detailed ranking and timing about Facebook downloaded pictures.

### 6.2.1 Global pictures classification

We start by presenting an overview of what kind of pictures are the most downloaded during all days of the packet dump. With this, we have studied which kinds of pictures are generating more traffic.

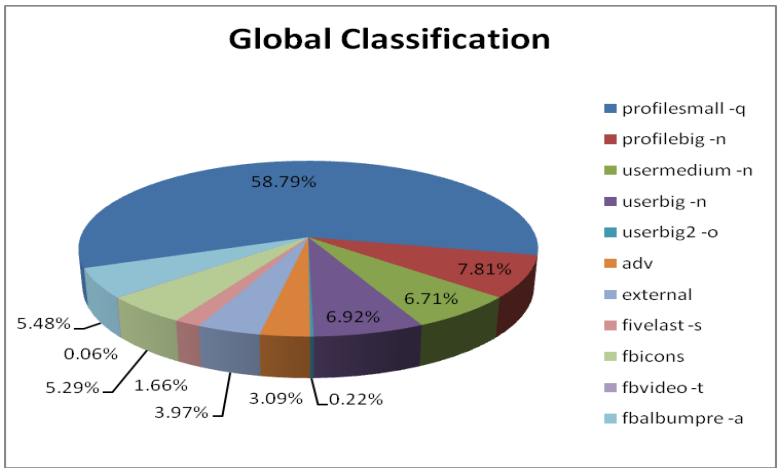


*Figure 6.1. Global Pictures Classification*

Figure 6.1 shows the global classification of all pictures filtered from the packet dump. A detailed description of each type of picture can be checked in *Annex 1*. More than half of all downloaded pictures are *profilesmall_q*. It is completely understandable because during a Facebook session a lot of profile pictures are downloaded although users do not want to download them. It is because if a user is on Facebook and one friend updates his status, or he uploads a picture, or his friend is connected to the chat, this user will download automatically his friend's profile picture in small size. *profilebig_n* pictures are downloaded every time one user has clicked on a profile, so it more or less corresponds to how many profiles have been visited. *userbig_n* pictures are pictures in big size, so user wants to see these kinds of pictures clicking on them.

*fivelast_s,* which represent a very small percentage of downloaded pictures, will disappear in a short period of time because with new Time-Line profile configuration, these pictures will not appear any more, and Facebook is trying to force all users to change their profiles to Time-Line.

Regarding to *fbvideo_t,* it is normal this low percentage, because usually videos are stored in external servers like Youtube or Vimeo (not in Facebook servers), so their frame previews are counted as "external".

In order to see better all other types of pictures, Figure 6.2. is the same as Figure 6.1. without considering *profilesmall_q*.

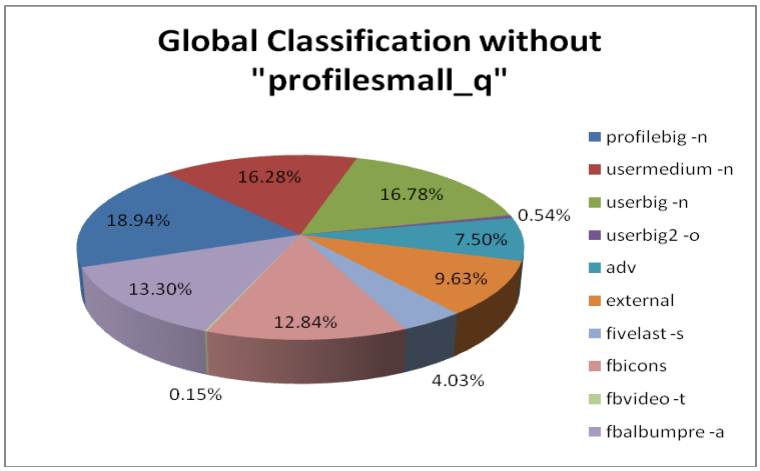


*Figure 6.2. Global classification without "profilesmall_q"*

In this case, it is easy to see that *profilebig_n*, *usermedium_n* and *userbig_n* are generating more or less the same number of requests and these 3 have the half of total requests without considering small profile pictures. *fbalbumpre_a* and *fbicons* are relevant as well, but the rest (*adv*, *userbig2_o*, *external*, *fivelast_s*, *fbvideo_t*) are only a 1/5 of the total.

The next step is to focus on which percentage of all pictures is generated by users. We have considered as user pictures *profilesmall_q*, *profilebig_n*, *usermedium_n*, *userbig_n*, *userbig2_o*, *fivelast_s*, *fbvideo_t* and *fbalbumpre_a*. No-users pictures are *adv*, *external* and *fbicons*.

These percentages are presented in Figure 6.3 where it can be seen that almost 90% are user pictures. This is due to the big percentage that small profile pictures, *profilesmall_q*, represent.

Even though no-user pictures represent less than the 13% of total pictures it is interesting to analyze their characteristics. *fbicons* are most requested no-user's pictures (5.29% of total pictures, Figure 6.1). Every time a Facebook user is loading a Facebook webpage a lot of icons are downloaded. The *adv* pictures are advertisements, so Facebook decides when users will see these pictures according to their preferences and likes, but *external* pictures depend on user's friends because they are external links. For instance, if an important event has happened, probably a lot of users will share links announcing news, so they will be generating a lot of *external* pictures.

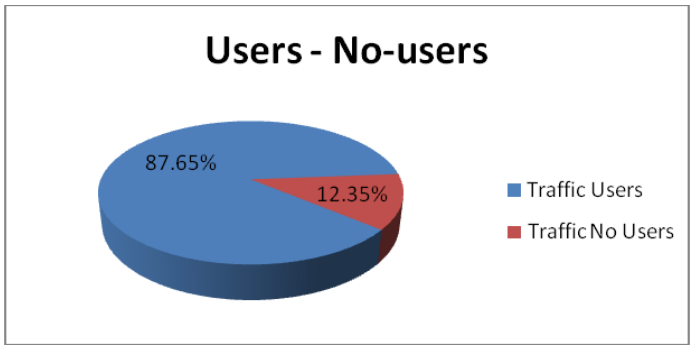


*Figure 6.3.Traffic User/ No-User*

Finally, the relation about clicking and not clicking in order to download user pictures has been investigated. *Click pictures* are those user pictures that need to be clicked by FB users to see them. *No-click pictures* are pictures downloaded automatically without clicking on them.

*Click pictures* are *userbig_n* and *userbig2_o*, although *userbig2_o* is irrelevant. *No-click pictures* are *profilesmall_q*, *profilebig_n*, *usermedium_n*, *fivelast_s*, *fbvideo_t* and *fbalbumpre_a*. This classification can be observed in Figure 6.4.

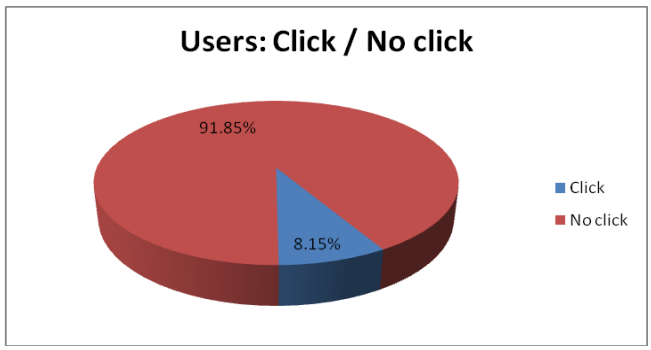


*Figure 6.4. Users:Click – No click*

In order to understand this classification, a brief description of the different kinds of pictures considered as *Click* or *No-click pictures* is included:

- *profilebig_n* pictures are downloaded when a FB user is visiting a profile, but he/she does not click to see this picture in big size.

- *usermedium_n* pictures are previews of an uploaded picture (in the wall), but they are shown without clicking on it.
- *fbalbumpre_a* pictures are shown when a FB user clicks on a album, and then FB user downloads all preview pictures.
- *userbig_n*: These pictures are in big size and it always means clicking on another picture. There are some ways to download a *userbig_n* picture. FB users can click on different kinds of *No-click pictures (usermedium_n, profilebig_n, fbalbumpreview_a* or *fivelast_s)* obtaining a big size user picture (*userbig_n*).

Although there are different ways to download a *Click picture,* in Figure 6.4. it can be seen that only 8.15% of all user pictures downloaded in Facebook are downloaded because FB users click on them. It means that users see most of pictures in previews, without clicking on them to see pictures in big size.

### 6.2.2 Downloaded Facebook users pictures statistics and results

In this section we present the results related with the most downloaded FB user pictures. The downloaded pictures with more than 50 requests have been evaluated in order to study their characteristics and their time-life. We have also studied the timing of pictures in order to know when FB users download pictures.

It must be said that we have anonymized all the pictures names. We have done statistics about downloaded pictures but we have not had access to the pictures links because all the pictures names parsed during this part have been hashed.

#### 6.2.2.1 Ranking and time-life of pictures

First of all, a ranking of the most downloaded pictures has been done. This ranking includes all the downloaded Facebook user pictures with 50 or more requests. This rank is made with a total of 10320 pictures and is sorted by number of downloads that each picture has had during the last week of May. Figure 6.5. shows the ranking of the downloaded pictures.

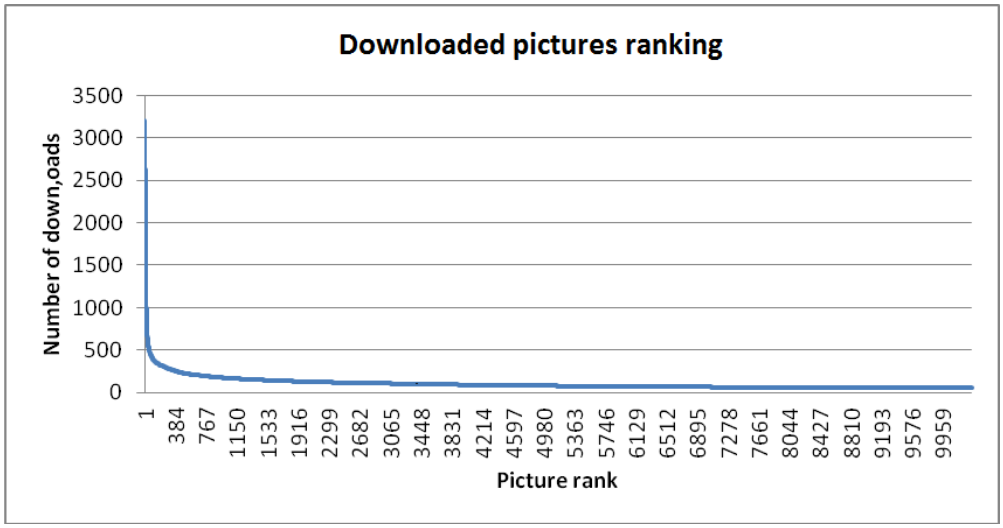


*Figure 6.5.Popularity ranking of pictures downloaded more than 50 times*

As it can be seen in the Figure 6.5., the curve is very abrupt. Only a small group of pictures are downloaded more than 500 times, and most of pictures are downloaded between 500 and 50 times. We have classified the pictures in groups by number of downloads and the result is the Table 6.1.

| Interval | # pictures | % of total pictures | # downloads | % of total downloads |
|---|---|---|---|---|
| **>1000** | 16 | 0.16 | 29652 | 2.80 |
| **1000-501** | 37 | 0.36 | 25342 | 2.39 |
| **500-101** | 3288 | 31.86 | 529766 | 50.02 |
| **100-50** | 6979 | 67.63 | 474436 | 44.79 |
| **TOTAL** | 10320 | 100.00 | 1059196 | 100.00 |

*Table 6.1. Number of pictures classified by number of total downloads.*

This table shows how only a 0.52% of pictures are downloaded more than 500 times (0.15% more than 1000 times and 0.36% between 1000 and 500).

The fourth column of Table 6.1 shows the number of downloads for each group of pictures. Comparing the number of pictures and the numbers of downloads, it can be seen that the Pictures downloaded more than 1000 times represent a 2.8% of the total downloads. It means that 16 pictures, a

0.16% of the total pictures downloaded, represent almost the 3% of total downloads.

Pictures downloaded between 1000 and 100, the 32.22% of total pictures downloaded, are responsible of the 52.32% of the total downloads and pictures downloaded between 500 and 50 times, more than 67.63% of the total pictures, represent a 44.79% of the total downloads.

For pictures downloaded 500 or more times, a total of 53 pictures (0.52% of total pictures captured), it is interesting to investigate which kind of Facebook pictures they represent.

After analyzing, the conclusion is that most of pictures are *profilesmall_q*, pictures with the extension *_q.jpg* (more detailed in *section 6.2.1*). A total of 41 of 53 pictures, the 77.36% of pictures downloaded 500 times or more are Profile pictures in small size.

It is also important to know if, considering all the downloaded pictures, the percentage of Small profile pictures represents also a very important percentage as in the case of the pictures downloaded 500 times or more. In order to know this, we have classified the pictures downloaded more than 50 times. The result is the Figure 6.6, in which the percentage of each type of picture can be observed.

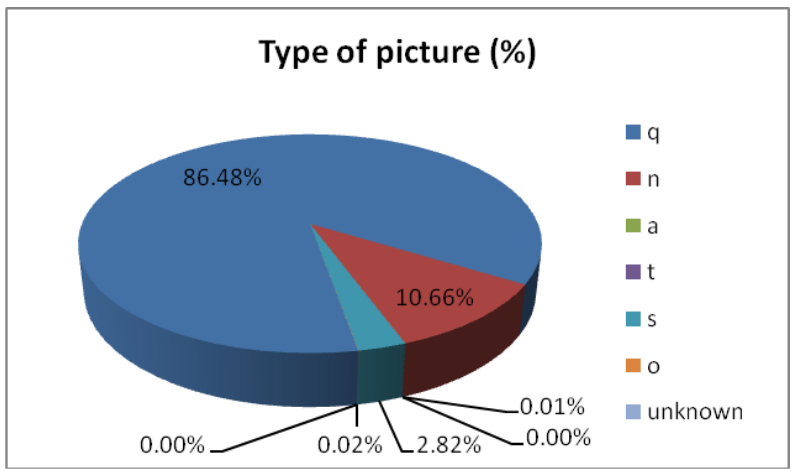


*Figure 6.6. Percentage of pictures downloaded more than 50 times*

After observing last figure, the conclusion is that Small Profiles Pictures (represented as *q* in the graph, *profilesmall_q*) are the 86.48% of all pictures

downloaded more than 50 times. In second position, with a 10.66% of all downloaded pictures, appears *n* extension, which includes *profilebig_n*, *usermedium_n* and *userbig_n*.

The next step of the downloaded pictures analysis is related with the time-life of Facebook User Pictures. We have defined the time-life as the interval of time between the first and the last download of each picture. It can be useful to know the time period during which the pictures are downloaded by Facebook users in order to decide if this content may be cached locally, for example.

First of all, we have done a ranking of time-life of the user pictures downloaded 50 or more times. Figure 6.7. represents the ranking of pictures time-life. Observing it, it can be seen that clearly most of the pictures, the 77.94% of total pictures, have a time-life between 139 and 120 hours.

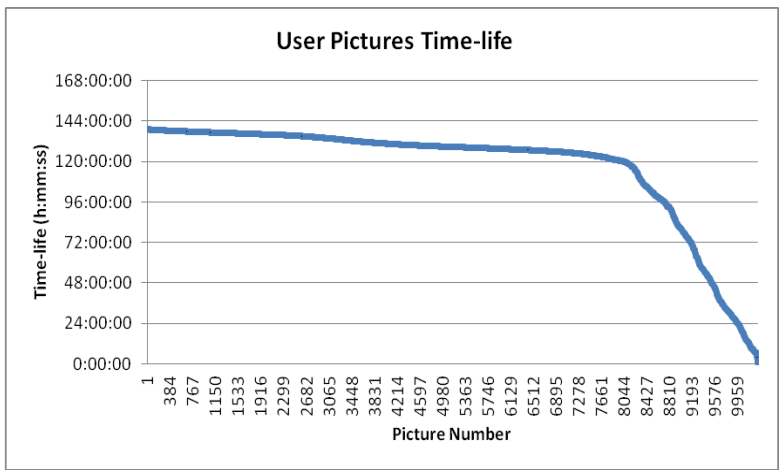


*Figure 6.7. Time-life evolution for user pictures downloaded 50 or more times*

This result is due to two different factors. On one hand, the packet dump was done during 7 days, but it was started at 13:57:19 of the first day and it was finished at 09:03:19 of the seventh day. It means that pictures were downloaded during a total of 139:07:00 hours.

On the other hand, as it was said before, the 86.4% of the pictures downloaded more than 50 times are *profilesmall_q* pictures and these kinds of pictures are static pictures, pictures that appear during a Facebook session because are the profiles pictures of the other FB users (most of them

Facebook friends pictures). Usually FB users do not change their Profile Picture often.

In order to have reliable results, first of all, we should have a longer packet dump and secondly, we should discard the Small profile pictures. The first thing has been impossible to do, due to the limitations we described in *Chapter 1, Section 1.3*, but it has been possible to discard Small profile pictures in order to see if the graph changes and if the time-life of the rest of User pictures can be more trustable.

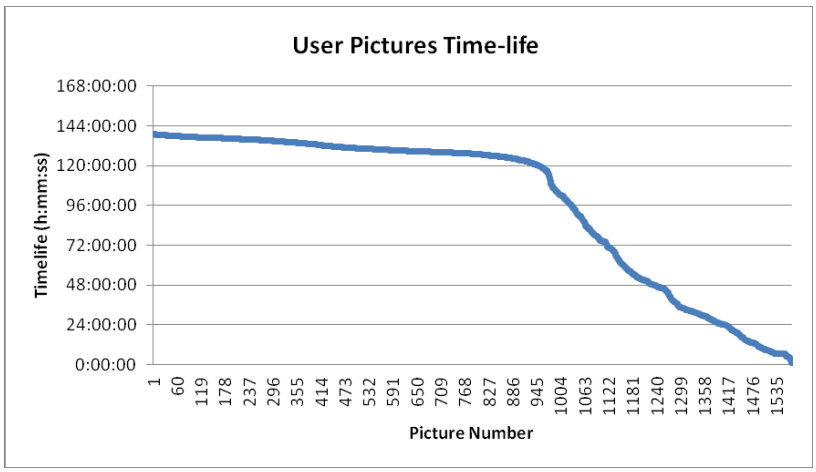


Figure 6.8. *Time-life evolution for user pictures (without small profile pictures) downloaded 50 or more times*

In Figure 6.8. it can be observed that the form of the graph changes in comparison with Figure 6.7. In this case, the 60.33% have a time-life of 120 h or longer. Regarding that, discarding small profile pictures, the predominant extension of pictures is *n*, which includes *userbig_n*, *profilebig_n and usermedium_n*, pictures time-life is now more realistic because when these pictures are uploaded, they appear on the FB new's wall during a brief period of time.

Even though, to conclude that the time-life of pictures is totally trustable we should have a longer packet dump.

### 6.2.2.2 Timing of downloaded Facebook User Pictures

In order to understand better the user behaviour regarding downloaded user Facebook pictures, a deep study about the time of these downloads has been done.

The results obtained are focused on the 20 most downloaded pictures including small profile pictures (*profilesmall_q*). Considering this type of pictures in this case is useful because it also shows when users have more connections.

We have chosen this "Top 20" because it includes all the FB user pictures with at least 1000 downloads. Regarding that the packet dump includes 7 days (5 entire days), the important results about timing have been obtained from these 5 entire days.

The first step has been to sort the time of downloads of the 20 most downloaded pictures by intervals of two hours. It must be said that we have chosen the 20 most downloaded pictures in all the packet dump (of 7 days), and that for this reason they are sorted by the total downloads, even if the downloads done during the 5 entire days may be less.

| 5 days | Name | 0:00-1:59 | 2:00-3:59 | 4:00-5:59 | 6:00-7:59 | 8:00-9:59 | 10:00-11:59 | 12:00-13:59 | 14:00-15:59 | 16:00-17:59 | 18:00-19:59 | 20:00-21:59 | 22:00-23:59 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 2853 | Top1 | 166 | 71 | 34 | 45 | 108 | 218 | 354 | 310 | 395 | 410 | 409 | 333 |
| 2241 | Top2 | 210 | 55 | 22 | 27 | 112 | 155 | 232 | 179 | 263 | 247 | 353 | 386 |
| 2239 | Top3 | 217 | 50 | 22 | 26 | 116 | 155 | 235 | 173 | 255 | 248 | 348 | 394 |
| 2282 | Top4 | 143 | 68 | 26 | 38 | 107 | 139 | 228 | 213 | 270 | 296 | 409 | 345 |
| 1866 | Top5 | 118 | 36 | 26 | 42 | 77 | 133 | 236 | 184 | 253 | 270 | 282 | 209 |
| 2084 | Top6 | 81 | 48 | 19 | 37 | 92 | 149 | 199 | 212 | 263 | 322 | 405 | 257 |
| 1645 | Top7 | 106 | 13 | 7 | 21 | 66 | 124 | 229 | 134 | 228 | 206 | 333 | 178 |
| 1545 | Top8 | 91 | 40 | 20 | 30 | 61 | 129 | 156 | 166 | 216 | 225 | 246 | 165 |
| 1632 | Top9 | 104 | 13 | 7 | 19 | 61 | 121 | 229 | 134 | 229 | 202 | 334 | 179 |
| 1294 | Top10 | 99 | 69 | 72 | 86 | 76 | 103 | 109 | 108 | 106 | 133 | 188 | 145 |
| 1275 | Top11 | 71 | 23 | 11 | 28 | 73 | 90 | 153 | 128 | 158 | 211 | 183 | 146 |
| 1107 | Top12 | 60 | 34 | 14 | 16 | 40 | 86 | 122 | 117 | 173 | 169 | 164 | 112 |
| 1072 | Top13 | 43 | 20 | 5 | 16 | 58 | 82 | 88 | 83 | 100 | 185 | 231 | 161 |
| 959 | Top14 | 43 | 15 | 9 | 17 | 62 | 74 | 106 | 86 | 124 | 147 | 148 | 128 |
| 1119 | Top15 | 29 | 18 | 5 | 8 | 28 | 36 | 97 | 114 | 232 | 182 | 222 | 148 |
| 918 | Top16 | 36 | 19 | 5 | 16 | 58 | 80 | 84 | 70 | 86 | 138 | 182 | 144 |
| 871 | Top17 | 36 | 10 | 15 | 13 | 21 | 67 | 96 | 81 | 132 | 149 | 170 | 81 |
| 830 | Top18 | 50 | 12 | 8 | 27 | 39 | 55 | 76 | 97 | 94 | 112 | 173 | 87 |
| 895 | Top19 | 141 | 4 | 1 | 2 | 2 | 92 | 153 | 72 | 102 | 118 | 168 | 40 |
| 952 | Top20 | 26 | 15 | 2 | 8 | 21 | 31 | 80 | 100 | 202 | 158 | 184 | 125 |
| 29679 | Total | 1870 | 633 | 330 | 522 | 1278 | 2119 | 3262 | 2761 | 3881 | 4128 | 5132 | 3763 |

*Table 6.2. Top 20 of Facebook user pictures downloaded, and downloads in 2 hours intervals during the 5 entire days*

In Table 6.2, the Top 20 is shown separating downloads by hours of the 5 entire days. It means that all pictures downloaded between 0:00 h and 1:59

of the 5 entire days, for instance, are counted in the column “00:00 -1:59” of each picture. The last row shows the number of total downloads of the 20 most downloaded pictures in every interval. It has been used to make Figure 6.9, where it can be seen the distribution of downloads during the day.

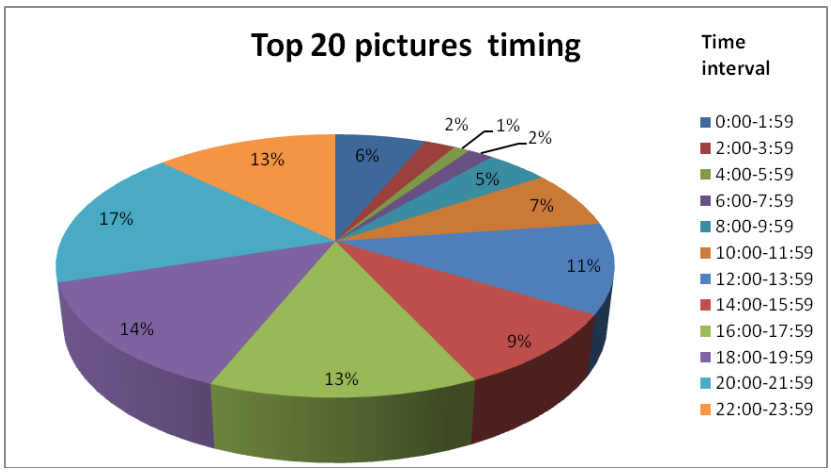


*Figure 6.9. Total downloads of the 20 most downloaded User Facebook pictures divided by hours (in intervals of two hours).*

Observing Figure 6.9, it is easy to conclude that most pictures are downloaded during the second half of the. During these 5 days, the interval with most downloads is between 20:00 and 22:00h (17% of total downloads), but there are no big differences between the intervals from 16:00 to 23:59 h (percentages from 13% to 17% during these 8 hours). During night intervals from 22:00, the percentage decreases a lot, with the minimum percentage of downloads from 4:00 until 5:59 h (1% of total downloads).

The main conclusion of this graph is that during the day, especially from 12:00h until 23:00h, is the time period when most downloads are done by FB users.

In order to know if by discarding small profile pictures the results of the downloads timing change, the same table has been derived, selecting the 20 most downloaded FB user pictures without considering the *profilesmall_q* pictures.

| 5 days | Name | 0:00-1:59 | 2:00-3:59 | 4:00-5:59 | 6:00-7:59 | 8:00-9:59 | 10:00-11:59 | 12:00-13:59 | 14:00-15:59 | 16:00-17:59 | 18:00-19:59 | 20:00-21:59 | 22:00-23:59 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 2241 | Top1 | 210 | 55 | 22 | 27 | 112 | 155 | 232 | 179 | 263 | 247 | 353 | 386 |
| 1866 | Top2 | 118 | 36 | 26 | 42 | 77 | 133 | 236 | 184 | 253 | 270 | 282 | 209 |
| 2084 | Top3 | 81 | 48 | 19 | 37 | 92 | 149 | 199 | 212 | 263 | 322 | 405 | 257 |
| 1632 | Top4 | 104 | 13 | 7 | 19 | 61 | 121 | 229 | 134 | 229 | 202 | 334 | 179 |
| 1107 | Top5 | 60 | 34 | 14 | 16 | 40 | 86 | 122 | 117 | 173 | 163 | 164 | 112 |
| 959 | Top6 | 43 | 15 | 9 | 17 | 62 | 74 | 106 | 86 | 124 | 147 | 148 | 128 |
| 1119 | Top7 | 29 | 18 | 5 | 8 | 28 | 36 | 97 | 114 | 232 | 182 | 222 | 148 |
| 918 | Top8 | 36 | 19 | 5 | 16 | 58 | 80 | 84 | 70 | 86 | 138 | 182 | 144 |
| 830 | Top9 | 50 | 12 | 8 | 27 | 39 | 55 | 76 | 97 | 94 | 112 | 173 | 87 |
| 819 | Top10 | 33 | 13 | 11 | 8 | 23 | 68 | 97 | 80 | 124 | 129 | 163 | 70 |
| 510 | Top11 | 41 | 1 | 3 | 9 | 24 | 49 | 56 | 56 | 85 | 80 | 67 | 39 |
| 479 | Top12 | 20 | 22 | 10 | 8 | 27 | 43 | 52 | 40 | 58 | 87 | 76 | 36 |
| 339 | Top13 | 13 | 2 | 3 | 8 | 12 | 18 | 27 | 36 | 39 | 48 | 76 | 57 |
| 368 | Top14 | 22 | 6 | 4 | 6 | 12 | 25 | 11 | 16 | 7 | 7 | 140 | 112 |
| 314 | Top15 | 17 | 5 | 3 | 7 | 22 | 22 | 30 | 36 | 40 | 44 | 46 | 42 |
| 359 | Top16 | 4 | 1 | 2 | 0 | 2 | 115 | 83 | 65 | 40 | 23 | 10 | 14 |
| 342 | Top17 | 14 | 5 | 3 | 1 | 7 | 13 | 14 | 59 | 62 | 64 | 73 | 27 |
| 301 | Top18 | 15 | 2 | 0 | 1 | 10 | 29 | 27 | 30 | 42 | 39 | 61 | 45 |
| 259 | Top19 | 48 | 7 | 8 | 17 | 26 | 29 | 13 | 17 | 19 | 23 | 33 | 19 |
| 320 | Top20 | 51 | 44 | 14 | 11 | 58 | 98 | 44 | 0 | 0 | 0 | 0 | 0 |
| 17166 | Total | 1009 | 358 | 176 | 285 | 792 | 1398 | 1835 | 1628 | 2233 | 2333 | 3008 | 2111 |

*Table 6.3. Top 20 of Facebook user pictures downloaded without profile pictures, and downloads in 2 hours intervals during the 5 entire days*

Table 6.3 shows the same kind of data as before but now discarding small profiles pictures. The same graph, based on the total number of downloads in each interval is presented in Figure 6.10.

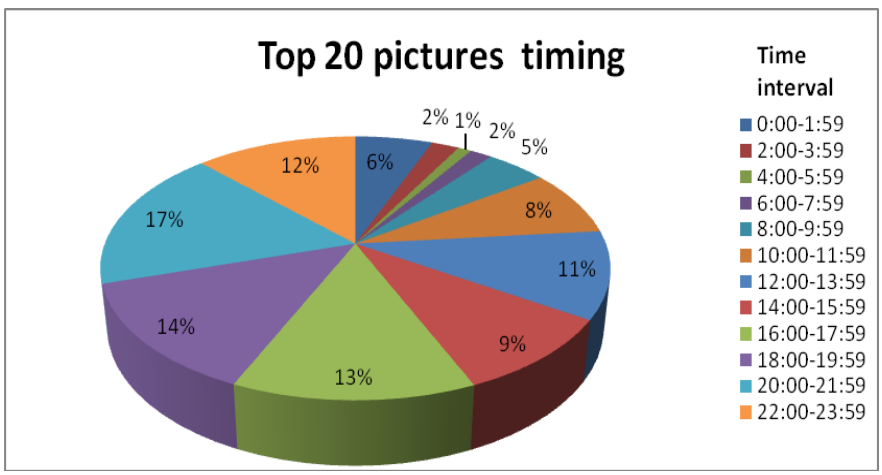


*Figure 6.10. Total downloads of the 20 most downloaded User Facebook pictures (without small profile pictures) divided by hours (in intervals of two hours).*

As can be seen in the graph, the results are exactly the same as in Figure 6.9., except for two intervals. Considering profile pictures, time period from 22:00 to 23:59 h produced the 13% of the downloads, and now, without considering them, it corresponds to the 12% of the downloads. The other interval that changes is the one from 10:00 to 11:59 h which has a 1% decrease.

The conclusion is that the percentages are really similar. It could be noted that maybe FB users have short FB sessions (short connections in which they only check their New's Wall) during several hours of the day, but they download big and medium size pictures (longer sessions in which FB users visit other friends Walls and Albums) in other hours of the day.

Comparing the two graphs, we can conclude that user pictures are downloaded during the same time considering small profile pictures (which are downloaded automatically when a FB user starts a FB session) and without considering Small Profile Pictures.

To complete this part, a new graph has been done (Figure 6.11), in which the number of downloads in each interval with and without small profiles pictures are shown. Comparing the two cases, we can affirm that user pictures are downloaded during the same time period, even if we do not consider the small profile pictures that are downloaded automatically when users connect to Facebook most of the times without clicking on them.

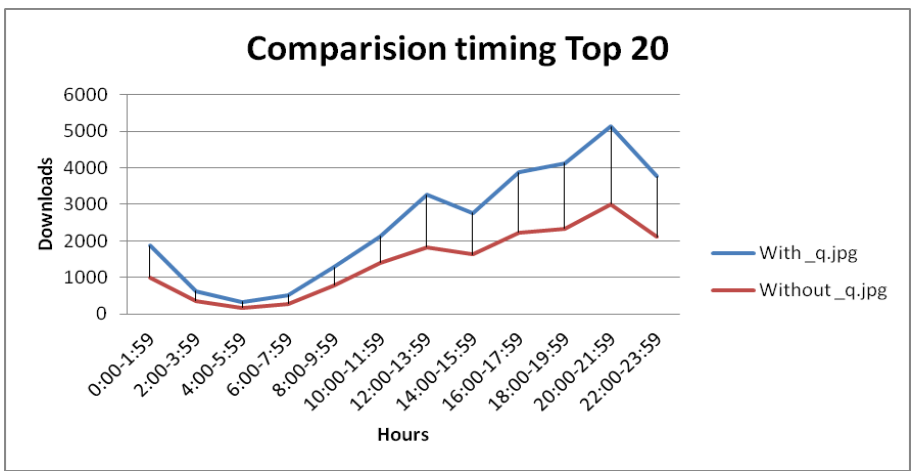


*Figure 6.11. Comparison of donwloads timing, considering and not considering small profiles pictures*

The following step has been to divide the pictures in intervals of two hours, each picture separately, in order to know if the most downloaded pictures are downloaded during the same hours of the day. This has been done for the 5 most downloaded pictures and for each of the 5 entire days.

Figure 6.12. shows the timing of the top 5 pictures (considering small profile pictures) during the 5 entire days of the packet dump.

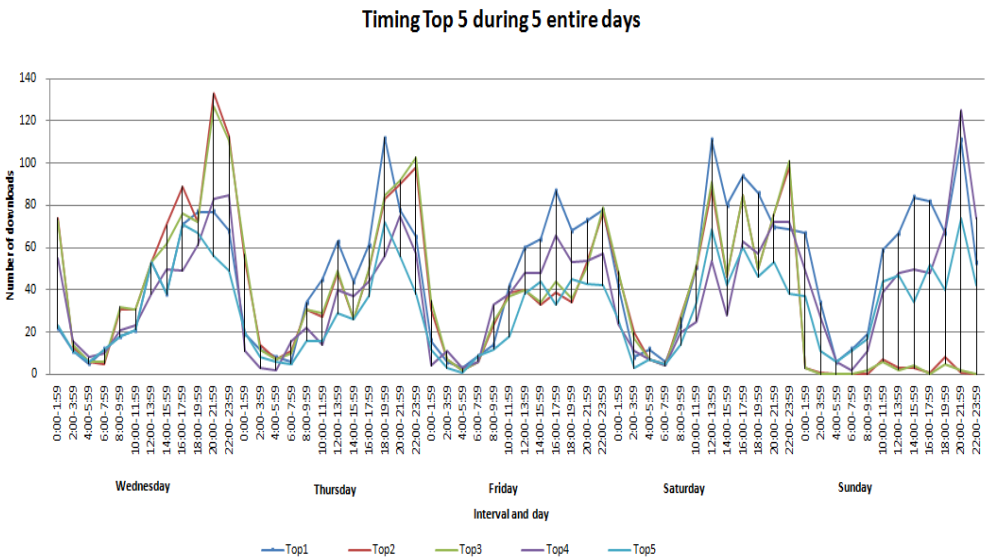


*Figure 6.12. Evolution of top 5 pictures during the 5 entire days*

Observing Figure 6.12, it can be seen how the evolution of downloads is quite different for the 5 most downloaded pictures. The maximum number of downloads appears always during the second half of the day (from 12:00 to 23:59 h) but during different intervals depending on the picture and on the day of the week. It can be observed how during Wednesday and Thursday the maximum number of downloads are in the intervals from 20:00 to 21:59h or from 22:00 to 23:59 h. But this behaviour changes if we look at the three last days of the week. During Friday, for instance, Top1, Top4 and Top5 have the maximum of downloads from 16:00 to 17:59h. During Saturday this change can be appreciated even more, where the maximum number of downloads appear in the interval from 12:00 to 13:59h for the

Top1 and Top5 pictures. On Sunday, the behaviour is closer to Wednesday and Thursday, where the maximum number for Top1, Top4 and Top5 appear during the interval from 20:00 to 21:59h.

Finally, the last step about timing has been to see how downloads are distributed during different days. To do that, we have classified the total downloads of the Top 20 most downloaded pictures by downloads per day (see Table 6.4 below).

| Total Wednesday | Total Thursday | Total Friday | Total Saturday | Total Sunday |
|---|---|---|---|---|
| 6061 | 5897 | 5349 | 6458 | 5914 |
| 20% | 20% | 18% | 22% | 20% |

*Table 6.4. Total Top 20 most downloaded pictures sorted by downloads per day*

As it can be observed, the number of total downloads per day, in percentage, is very similar during the 5 entire days of the week that we have, where the day with most downloads Saturday, with the 22% of the downloads, and Friday the day with less downloads, with the 18% of the total downloads. Wednesday, Thursday and Sunday represent each one the 20% of total downloads.

As it has been done in other parts of this section, we have discarded the small profile pictures to see if the results change or not.

| Total Wednesday | Total Thursday | Total Friday | Total Saturday | Total Sunday |
|---|---|---|---|---|
| 3194 | 3514 | 3440 | 3519 | 3499 |
| 19% | 20% | 20% | 21% | 20% |

*Table 6.5. Total Top 20 most downloaded pictures without considering profile pictures sorted by downloads per day*

Table 6.5. shows the downloads of the new Top 20 if we discard small profile pictures.

As before, the percentages are very similar during all the days (the difference between the maximum and the minimum is only a 2% of the total downloads). In this case we can observe that the maximum is on Saturday with a 21% of the total downloads (the same day than we considered small profile pictures) and the minimum is on Wednesday with a 19% of the total downloads (with small profile pictures the minimum was on Friday).

During the rest of the days, Thursday, Friday and Sunday, each day has 20% of the total downloads.

### 6.2.3 Uploaders

In this section we have focused on the users who upload pictures, the "uploaders". As it has been explained in *Chapter 3, section 3.2.1.2,* it has been possible to parse the FB users ID of the pictures uploads.

We present a ranking of uploaders and a comparison between the numbers of downloads they have and the different downloaders. It must be said that we have considered each MAC address as a different downloader.

#### 6.2.3.1 Ranking of Uploaders

First of all, a ranking of uploaders sorted by the number of downloads is presented. We have truncated all uploaders with less than 50 downloads on their content, because of that appear 14767 different uploaders. This ranking can be observed in Figure 6.13.

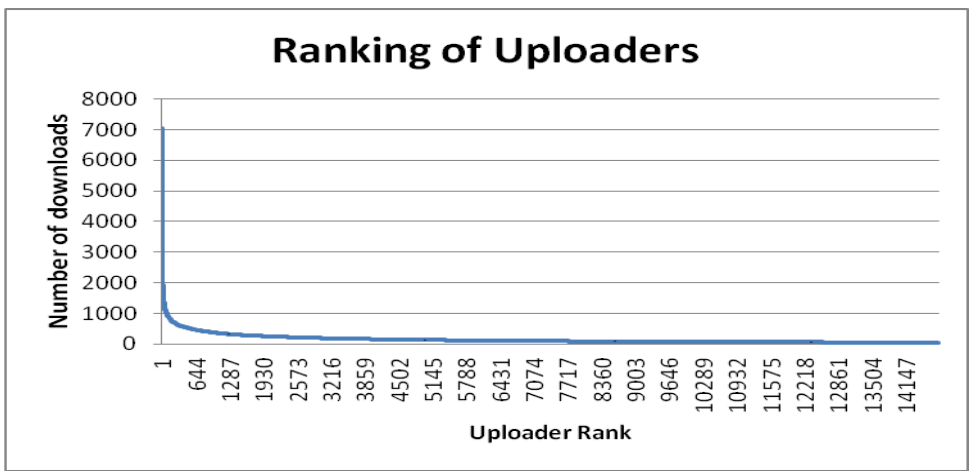


*Fig 6.13. .Ranking uploaders*

As with downloaded pictures, the curve is clearly decreasing. There is a small number of uploaders who are much more popular than others and, although they represent a small percentage of all uploaders considered, they are responsible for a very important number of downloads.

In Table 6.6 it can be seen how many uploaders there are in each range of number of downloads. The table shows how only a few uploaders have a huge number of downloads, and more than the 80% of all uploaders have only between 50 and 200 downloads.

| #downloads | #uploaders | % |
|---|---|---|
| >1000 | 95 | 0,64 |
| 1000-901 | 27 | 0,18 |
| 900-801 | 50 | 0,34 |
| 800-701 | 61 | 0,41 |
| 700-601 | 113 | 0,77 |
| 600-501 | 209 | 1,42 |
| 500-401 | 313 | 2,12 |
| 400-301 | 601 | 4,07 |
| 300-201 | 1397 | 9,46 |
| 200-101 | 4174 | 28,27 |
| 100-50 | 7727 | 52,33 |

*Table 6.6. Number of uploaders for each range of downloads*

The main conclusion is that it there may be a gain to cache content because if there are only a few uploaders who have a huge amount of downloads, it can be easy to identify these uploaders and to predict that this content will be downloaded by a lot of users.

### 6.2.3.2 Relation between Uploaders and downloaders

It has been studied the number of downloads for each uploader and the number of different downloaders that have downloaded from this uploader. Downloaders with less than 1000 downloads have been truncated in all packet dump. Ranking of uploaders has been done in order to know how many uploaders that have all downloaders in common, or groups of downloaders because it can help to cache content from popular uploaders.

| FB user ID (Uploader) | Total downloads: | # different downloaders: |
|---|---|---|
| **Uploader A** | **7046** | **723** |
| Uploader B | 6874 | 345 |
| **Uploader C** | **5300** | **640** |
| Uploader D | 4174 | 184 |
| **Uploader E** | **3904** | **675** |
| Uploader F | 3892 | 380 |
| **Uploader G** | **3373** | **592** |
| **Uploader H** | **3242** | **771** |
| Uploader I | 3062 | 19 |
| Uploader J | 2451 | 117 |

*Table 6.7. FB user ID sorted by number of downloads*

Table 6.7. shows the ranking of the 10 most important uploaders sorted by number of downloads. Most of users in this TOP10 have a relevant number of different downloaders.

Table 6.8. shows the ranking of 10 most important uploaders sorted by the number of different downloaders.

| FB user ID (uploaders) | Total downloads: | # different downloaders: |
|---|---|---|
| **Uploader H** | **3242** | **771** |
| **Uploader A** | **7046** | **723** |
| **Uploader E** | **3904** | **675** |
| **Uploader C** | **5300** | **640** |
| **Uploader G** | **3373** | **592** |
| Uploader K | 2441 | 550 |
| Uploader L | 2144 | 516 |
| Uploader M | 1952 | 450 |
| Uploader N | 1655 | 428 |
| Uploader O | 1054 | 422 |

*Table 6.8. FB user ID sorted by number of different downloaders*

FB user IDs marked in bold that appear in both tables are uploaders who are in TOP10 most downloaded uploaders and who are also in TOP10 uploaders with the highest number of different downloaders. Observing the two last tables can be seen that 5 of the TOP10 uploaders appear in both. More specifically, the first 5 FB user ID sorted by number of different

downloaders (Table 6.8) are also between the 10 most requested uploaders (Table 6.7).

The main conclusion is that the most requested uploaders usually have a huge number of different downloaders, so there is not a small group of downloaders who download a lot of pictures from the same uploader.

### 6.2.4 Relation between Pictures and downloaders:

In this section we have related pictures with all users who have downloaded these pictures. Pictures names and MAC addresses have been hashed in order to anonymize all content. After running some scripts, we have a sorted list with number of downloads of a picture, picture name, all MAC addresses and how many times each MAC address has downloaded this single picture.

The first 2 results are special because they show who have the most downloads and there is only one MAC address for each picture, so one MAC address has downloaded this picture thousands times. There are a few pictures where the same thing happens, but they are not relevant to this study so we have discarded them.

For the 30 most downloaded pictures, there is an average of 569 different downloaders for each picture. It means that usually most downloaded pictures are posted by famous people or Facebook pages with a lot of followers, because common FB users have around 190 Facebook friends [26].

Figure 6.14 shows the downloaders who download more different pictures with the total number of downloads for each downloader.

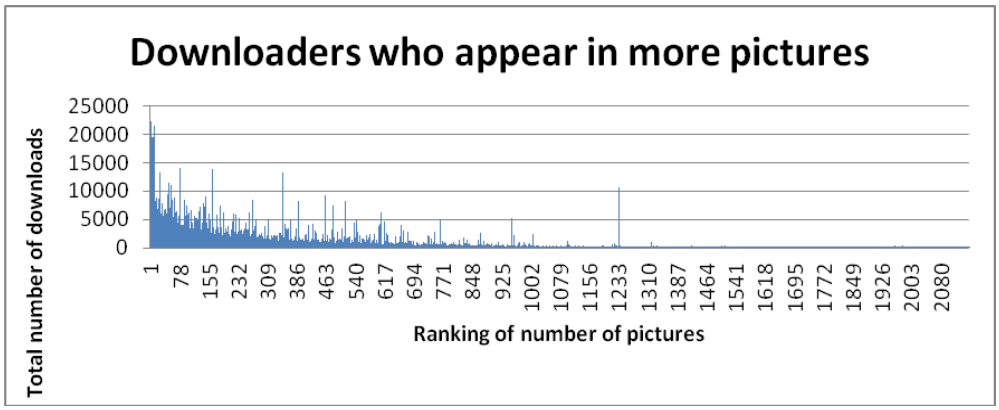


*Figure 6.14. Downloaders sorted by number of different downloaded pictures*

It is necessary to give a more detailed explanation about this graph. Total number of downloads are all requests done by one downloader, but MAC addresses are sorted depending on the number of different pictures downloaded. For instance, the first MAC address (ranking 1), has done 22319 requests, but it has only downloaded 8871 pictures, so a lot of pictures have been downloaded more than once. There are other cases where total number of downloads and total pictures have the same value, so this users have only downloaded each picture once. It is easy to see that people will have usually more requests than the number of different pictures. For instance, if a user is in the main Facebook page and one friend posts something, this user will download his profile picture, and then if this friend is connected to chat or posts another comment/picture, this user will download the same profile picture again.

A remarkable result is that there is 18% MAC addresses that only download each picture once, and it can be caused by short Facebook sessions. Note that the first MAC address which has the same number of requests and the same number of different pictures is the TOP 789, so it is not a very active user.

## 6.3 Facebook Likes

Until now, all the results presented in this chapter have focused on the Facebook pictures. Although pictures represent the most important part of Facebook traffic, we present, in this section, the results obtained about another very important tool used by Facebook users and that is a symbol of this social network: *Like*.

In this section we have considered each MAC address (anonymized) as a FB user. Besides, the statistics have been done about the number of *Likes* posts but we have not had access to the content they like.

### 6.3.1 Ranking of Likes

As it was explained in *Chapter 3*, *section 3.2.2*, it has been possible to parse *Likes* posted by Facebook users. In order to know how many *Likes* are posted and how they are distributed, we have done a ranking of Users, considering that each MAC address corresponds to a user, sorted by the number of *Likes* posted during the seven days (Figure 6.15).

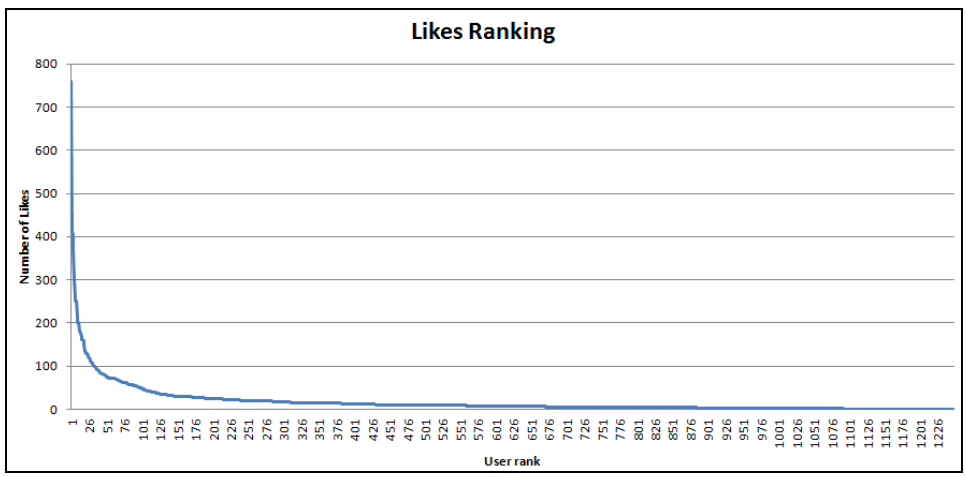


*Figure 6.15. Ranking of "Likes"*

Observing the last graph, it can be seen that the curve is very similar to the curve of downloaded pictures (Figure 6.5) , with an abrupt descendant form.

It means that a small number of users are responsible of most of *Likes* posted on Facebook during the week. The number of *Likes* posted by the user who posts the most *Likes* is 760. In order to know better the behaviour of users with this FB tool, a table is presented, sorting it by intervals of the number of *Likes* done by FB users.

| # Likes | Users | Users (%) | Total likes | Total likes (%) |
|---|---|---|---|---|
| **>200** | 10 | 0.80 | 3415 | 15.38 |
| **200-101** | 23 | 1.84 | 3177 | 14.31 |
| **100-51** | 64 | 5.13 | 4448 | 20.03 |
| **50-11** | 368 | 29.49 | 7737 | 34.84 |
| **10-1** | 783 | 62.74 | 3431 | 15.45 |
| **TOTAL** | 1248 | 100.00 | 22208 | 100.00 |

*Table 6.9. Number of users by number of Likes and total likes*

As it can be observed in the table only 10 users ( 0.8% of the total users that post at least one *Like*) posted more than 200 *Likes* during the week. It is interesting to see how this small percentage of users is responsible of the 15.38% of the total *Likes* posted during that week.

In addition, 783 users (users who posted between 10 and 1 *Like*), more than 62% of all users, are responsible of almost the same percentage of *Likes* than the 10 users who post the most *Likes* (a 15.45% of the total *Likes* posted).

### 6.3.2 Timing of Likes

In order to know the behaviour of users who post more *Likes*, a timing study has been done. Like in the case of the downloaded pictures, we have sorted *Likes* posts by the time they were done in intervals of two hours during the five entire days of the Packet dump.

The first result is about the timing of the 20 most active users. Figure 6.16 shows the percentage of *Likes* posted during the day by the 20 users during the 5 entire days of our packet dump.

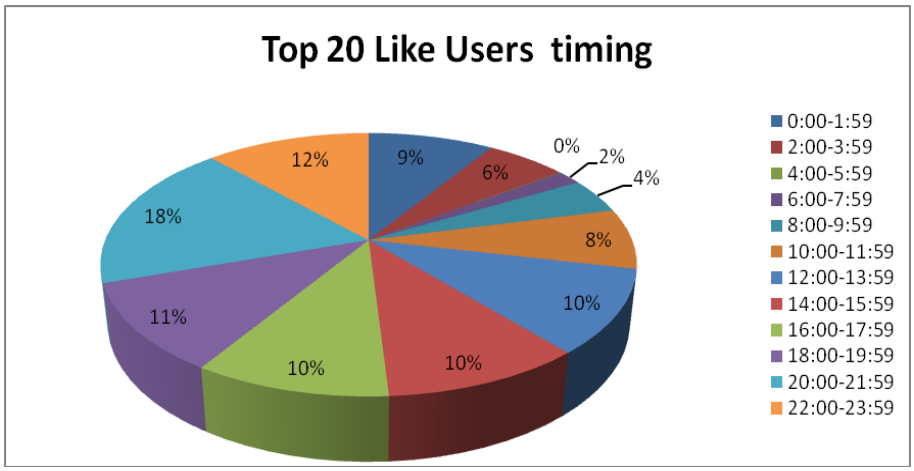


*Figure 6.16. Total downloads of the TOP 20 "Like" users divided by hours (in intervals of two hours).*

Observing Figure 6.16., it can be seen that several intervals represent similar percentages of total *Likes* posted. The minimum is between 4:00 and 5:59 h with less than a 1% of the total *Likes*. Then the percentage starts increasing each interval. There is a big increase in the interval between 10:00 and 11:59 h (8%). And from 12:00-13:59 to 16:00-17:59 h, the percentages are 10%. Then, from 18:00 to 19:59 the percentage increases in one point (11%). The maximum percentage of *Likes* is from 20:00 to 21:59 h with 18% of total.

It is interesting to compare these results with the ones obtained doing the same with the most downloaded pictures (*Figure 6.9*). Both cases are very similar (with small differences of percentages) and have the same general behaviour. Most of downloaded pictures and *Likes* are done during the second half of the day. In addition, the maximum of downloads and *Likes* are produced in the same interval: from 20:00 to 21:59 h. Time period with the minimum of downloads and *Likes* is the same too in both cases (from 4:00 to 5:59 h).

The main conclusion is that users post *Like* and download pictures during the same parts of the day that are basically during the second half of the day (from 12:00 to 23:59 h) and that the maximum and minimum of downloads and *Likes* are in the same intervals.

The next step is to see how the most popular *Like* users (users responsible for most *Likes*) post them individually dividing it by intervals of two hours and for each of the 5 entire days. In this part the 5 most active *Like* users have been considered. The results obtained are shown in Figure 6.17.

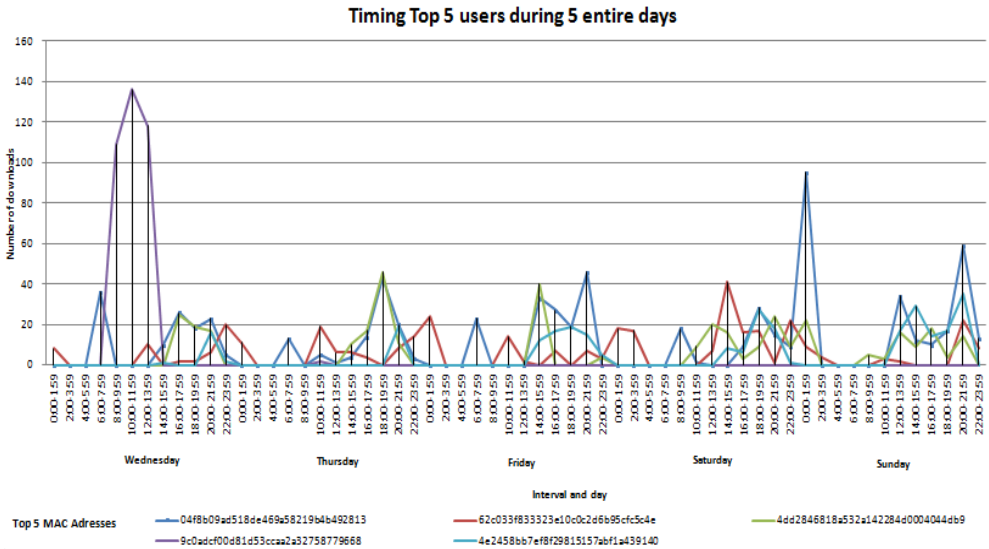


*Figure 6.17. Evolution of top 5 "Like" users during the 5 entire days*

Figure 6.17. shows the individual behaviour of the 5 users who post the most *Likes* during the week. The graph created has been done with the 5 most active users to see their behaviour during the 5 days. In the graph can be seen that the behaviour is not the same depending on the user and on the week-day.

The most surprising case is the *Purple User*. Wednesday is the day when the *Purple User* (the fourth most active user) post most of the likes and that he/she does it during the morning, having the maximum of *Likes* posted (almost 140) between 10:00 and 11:59 h and posting also a big amount of likes (more than 120) from 8:00 to 9:59 h and from 12:00 to 13:59 h. The rest of the days the *Purple User* does not post any *Likes*, except on Thursday, when he/she posts 2 *Likes* between 10:00 and 11:59h.

The rest of the users are more regular, posting *Likes* every day in a more regular way. It can be observed too that a lot of *Likes* are posted during the afternoon-night and that, as it happened in the case of downloaded pictures,

this behaviour is a bit different during Friday and Saturday, when a lot of likes are posted also during the morning – midday. The exception on Saturday night (beginning of Sunday) is the *Blue User* (the most active *Like* user) that posts a big amount of *Likes* (more than 100) from 0:00 to 1:59 h, when the rest of users post *Likes* during the day of Saturday and not in the night (we have considered that Sunday from 0:00 to 1:59 h can be considered as Saturday night too).

The main conclusion is that *Likes* timing depend totally on the individual behaviour and that some of the most active users connect maybe once a week and do a lot of *Likes* during a short period of time.

The last step with *Likes* has been to sort the most active users (Top 20) by days of the week in order to see if during some days users post a bigger amount of *Likes*.

In Table 6.10., can be seen the number of *Likes* posted by the 20 most active users and sorted by day of the week.

| Total Wednesday | Total Thursday | Total Friday | Total Saturday | Total Sunday |
|---|---|---|---|---|
| 911 | 721 | 922 | 706 | 1118 |
| 21% | 16% | 21% | 16% | 26% |

*Table 6.10. Total Top 20 most downloaded pictures without considering profile pictures sorted by downloads per day*

Besides, previous table also shows the percentages of total *Likes*. It can be seen how considering the top 20 most active users, the day with the most *Likes* posted is Sunday with a 26% of the total *Likes* posted. On Wednesday and Friday the 21% of the total *Likes* are posted both days and on Thursday and Saturday the 16% are posted (both days too).

It is interesting to compare these results with the case of downloaded pictures. In that case the percentage were more similar during the 5 entire days and the maximum of downloads was on Saturday. Now, in the case of *Likes* the results are different having more irregularities and becoming Sunday the day when users post more *Likes.*

# CHAPTER 7

# 7 Conclusions and future work

In this thesis a deep study of Facebook Data traffic and Facebook user behaviour has been done. We have created filtering rules and used Wireshark, PL and Python scripts to parse and to obtain the results presented in this report.

First part of this thesis consisted in the study of the FB data traffic locally. It has been possible to parse the different parts of Facebook such as Pictures, Status Updates, Likes, Chat and FB videos.

In the second part of the thesis, we have focused on Facebook pictures and *Like* tool. From the packet dump done during the last week of May 2012, it has been possible to extract and obtain some very interesting results. It must be noted that in the future ***it* could be really interesting to study and to get real data and results about other Facebook parts like Chat, Status Updates or Facebook videos.**

Regarding downloaded pictures, we have done a detailed classification of the different kinds of pictures. A very important result is that a big percentage, 86.48% of the total downloaded pictures, are Small Profile Pictures (*profilesmall_q*). This is due to the fact that Small Profile pictures are downloaded automatically when a FB user starts a session and loads the main page which contains News Wall.

The first important conclusion is that, **considering the large percentage of total pictures that Small Profile Pictures represent, it would be really interesting to cache them locally**. Regarding Small Profile Pictures downloaded are different for each FB user (profile pictures about FB friends), **they should be cached in the same device where the Facebook**

**session takes place**. **The best way to cache this content could also become a good future work.**

The second important conclusion about downloaded pictures is about timing. FB users download pictures during the second half of the day, the interval where they download a bigger number of pictures is from 20:00 to 22:00h and the weekday when more pictures are downloaded is on Saturday. In addition, we have compared this time classification considering Small Profile Pictures and discarding them. The results are practically the same, from where can be concluded that user pictures are downloaded in the same intervals considering small profile pictures (which are downloaded automatically when a FB user starts a FB session) and without considering Small Profile Pictures. **It means that all FB user pictures are downloaded in the same schedules, independently if they are pictures downloaded automatically at the beginning of a Facebook session or if they are pictures that are posted by other users and pictures that FB users download voluntarily.**

We have also studied the downloaded pictures time-life. Our definition of time-life is the time between the first and the last download of a picture. We have seen that most of pictures have a time-life equal to the total period of the packet dump, even discarding the Small Profile Pictures that have a longer time-life because they are not changed very often. **The main conclusion here is that to know the exact time-life of downloaded pictures it would be necessary to do a longer packet dum**p. We can affirm that **a huge number of downloaded pictures have a time-life longer than the period of our packet dump** (139:07:00 hours).

Last part of the study about downloaded pictures has been to relate them with uploaders and with potential downloaders. It has been possible to match the most important FB uploaders with the downloders who download pictures from these uploaders. The main conclusion is that the most requested uploaders have a large number of different downloaders. It means that **most important uploaders in number of downloads are also the most popular uploaders, with more different downloader**s. In addition, we have matched downloaders with all the pictures they download and the times when they download each picture. It has been found that **a few number of downloaders are responsible for the largest portion of downloads**. Besides, **some FB users download an important number of pictures, but**

**they only download each picture once** (the 18% of MAC addresses studied).

Finally, about *Likes*, we have done a ranking and a timing about the *Likes* posts. As it happens with downloaded pictures, a small percentage of FB users are responsible for a big percentage of *Likes.* Besides, about Likes timing, we have studied it in the same way than downloaded pictures and we have compared both cases. The conclusion is that, in general, ***Likes*** **are posted during the same hours as pictures are downloaded.**

# References


[1] [J. Li, A. Aurelius, V. Nordell, M. Du, Å. Arvidsson, M. Kihl, "A five year perspective of traffic pattern evolution in a residential broadband access network", Future Network & Mobile Summit 2012, Berlin, Germany, July 4th, 2012]

[2] [WolframAlpha, "http://www.wolframalpha.com", retrieved June 30th, 2012]

[3] [Wikipedia, "http://en.wikipedia.org/wiki/Facebook", retrieved June 30th, 2012]

[4] [SocialBakers, "http://www.socialbakers.comcountries/continents", retrieved June 30th, 2012]

[5] [IPNQSIS, "http://www.celtic-initiative.org/Projects/Celtic-projects/Call7/IPNQSIS/ipnqsis-default.asp", retrieved June 30th, 2012]

[6] [Celtic, "http://www.celtic-initiative.org", retrieved June 30th, 2012]

[7] [A. Aurelius, M. Kihl, C.Lagersted and P. Ödling, "Traffic analysis and characterization of Internet user behaviour", International Congress on Ultra Modern Telecommunications and Control Systems, Moscow, Russia, October 20th, 2010]

[8] [C.Lagersted, M. Kihl, A. Aurelius, A. Berntson and T. Westholm, "Measuring and Modeling HTTP streaming in IP Access Networks", Technical Report, Acreo, 2008]

[9] [A. Aurelius, C. Lagerstedt and M. Kihl, “Streaming media over the Internet: Flow based analysis in live access networks”, IEEE International Symposium on Broadband Multimedia Systems and Broadcasting, Nuremberg, Germany, June 10th, 2011]

[10] [M. Zink, K. Suh, Kyoungwon, Y. Gu and J. Kurose, "Watch Global, Cache Local: YouTube Network Traffic at a Campus Network - Measurements and Implications", Computer Science Department Faculty Publication Series. Paper 177, 2008]

[11] [P. Svoboda, W. Karner and M. Rupp, “Traffic Analysis and Modeling for World of Warcraft”, Communications, 2007. ICC '07. IEEE International Conference on, June 28th, 2007]

[12] [M. Kihl, A. Aurelius and C. Lagerstedt, “Analysis of World of Warcraft Traffic patterns and User behaviour”, International Congress on Ultra Modern Telecommunications and Control Systems, Moscow, Russia, October 20th, 2010]

[13] [F. Benevenuto, T. Rodrigues, M. Cha and V. Almeiday, “Characterizing User Behaviour in Online Social Networks”, 9th ACM SIGCOMM conference on Internet measurement, 2009]

[14] [K.N. Hampton, L.S. Goulet, C. Marlow and L. Rainie, “Why most Facebook users get more than they give”, Pew Research Center's Internet & American Life Project, 2012]

[15] [K.N. Hampton, L.S. Goulet, L. Rainie, and K. Purcell, “Social Networking Sites and Our Lives: How People's Trust, Personal Relationships, and Civic and Political Involvement are Connected to Their Use of Social Networking Sites and Other Technologies”, Pew Internet and American Life Project, 2011 ]

[16] [Wikipedia, “http://en.wikipedia.org/wiki/Hypertext_Transfer_Protocol”, retrieved June 30th, 2012]

[17] [Wikipedia, “http://en.wikipedia.org/wiki/Pcap”, retrieved June 30th, 2012]

[18] [Wireshark, “http://www.wireshark.org”, retrieved March 20th, 2012]

[19] [Cqcounter, “http://www.cqcounter.com/whois”, retrieved March 24th, 2012]

[20] [Procera Networks, “http://www.proceranetworks.com”, retrieved July 2nd, 2012]

[21] [Python, “http://www.python.org/about”, retrieved July 2nd, 2012]

[22] [TShark, “http://www.wireshark.org/docs/man-pages/tshark.html”, retrieved July 4th, 2012]

[23] [Businessdictionary, “http://www.businessdictionary.com/definition/Microsoft-Excel.html”, retrieved July 4th, 2012]

[24] [UnixTimeStamp, “http://www.unixtimestamp.com”, retrieved July 4th, 2012]

[25] [MostVisitedWebsites, “http://mostpopularwebsites.net”, retrieved July 5th, 2012]

[26] [J. Ugander, B. Karrer, L. Backstrom, C. Marlow, “The Anathomy of the Facebook Social Graph”, The Economist: The sun never sets, May 19th, 2012]

# Annex 1

## A.1 Pictures classification by Facebook server and extension

Here is presented a detailed classification of the different kinds of FB downloaded pictures during a FB session.

Profile pictures: stored in "profile.ak.fbcdn.net"

- **profilesmall_q:** "_q.jpg" → Profile picture in small size, it is always followed by the name of the user. It is downloaded automatically without clicking on it.

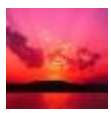

*Figure A.1. Example of a Small Profile Picture*

- **profilebig**_n: "_n.jpg" → Profile picture in big size. It is downloaded automatically when a FB user loads a FB profile.

*Figure A.2. Example of a Big Profile Picture*

Pictures posted by users: stored in "sphotos.ak.fbcdn.net" or in a server which name contains ".ak.fbcdn.net" and "photos-":

- **usermedium_n:** "_n.jpg" and "/s320x320/" or "/s480x480/" in the link → When a user posts a picture and appears in everybody's news's wall. It's like a preview, because to see the picture in big size (*userbig_n* or *userbig2_o*) you have to click on it.

- **userbig_n:** "_n.jpg" → Picture downloaded in big size, it is necessary to click on it. To download a userbig_n it is necessary to click on a *profilebig_n, usermedium_n, fivelast_s* or *fbalbumpre_a.*

- **userbig2_o:** "_o.jpg" → Pictures in the original size it was posted. Picture size can be the same seen in the screen, but this one has higher resolution.

-

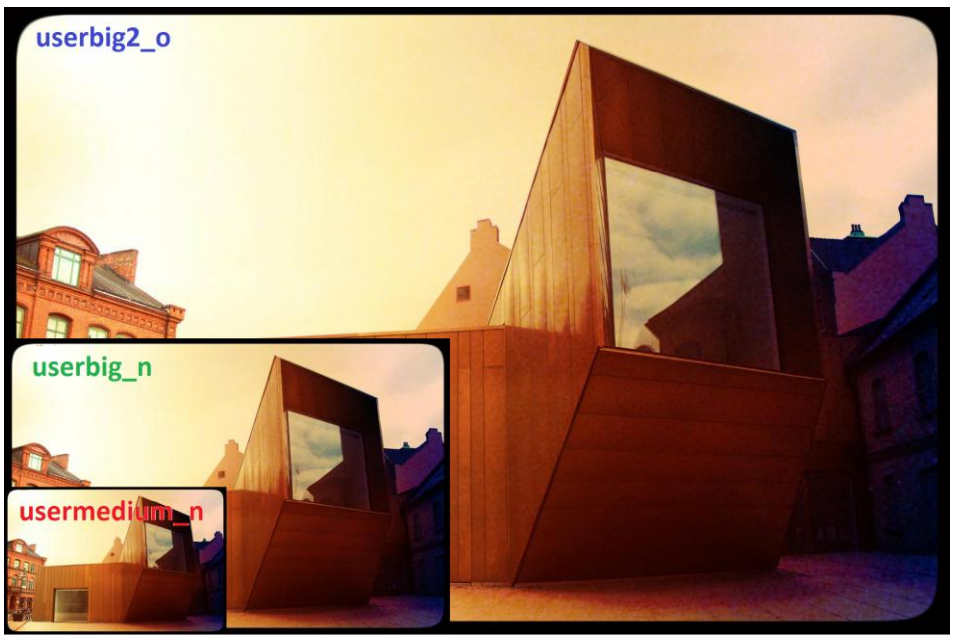


*Figure A.3. Comparison between usermedium_n, userbig_n and userbig2_n.*

- **fivelast_s:** "_s.jpg" → Small preview with the last 5 pictures where the user is tagged. Only in FB profiles without the Time-Line configuration in Facebook.

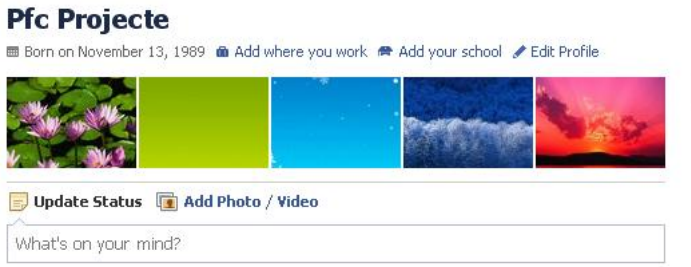


*Figure A.4. Example of the last 5 pictures tagged, in a FB user profile.*

- **fbalbumpre_a: "**_a.jpg" → When a FB user clicks on one album, all photos are shown in small size (album preview).

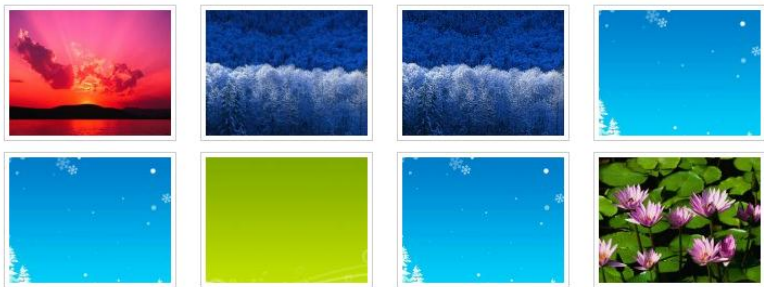

*Figure A.5. Example of a FB album preview.*

No user Pictures:

- **adv:** stored in "creative.ak.fbcdn.net" → Advertisements pictures posted by Facebook.


*Figure A.6. Example of FB advertisements*

- **fbicons** : Facebook icons pictures:
  - Stored in "static.ak.fbcdn.net" → undetermined size facebook icons
  - "http://www.facebook.com/images/icons" → small Facebook icons
  - "http://www.facebook.com/images/profile/" → small profile Facebook icons


*Figure A.7. FB icons*

- **external:** External pictures: stored in "external.ak.fbcdn.net" → These kinds of pictures usually are small previews that redirect to an external webpage. They can be pictures from other web pages, or a frame for a video preview, for instance, from Youtube.


*Figure A.8. Example of an external picture*

Pictures of videos: stored in "vthumb.ak.fbcdn.net"

**fbvideo_t**: "_t.jpg" → preview small size pictures of the videos (a video frame)

*Figure A.9. Example a preview picture of a video*