1-1-2008

Web multimedia files characterization

Mojgan Soraya
Ryerson University

WEB MULTIMEDIA FILES CHARACTERIZATION

by

Mojgan Soraya

B.Sc. (Isfahan University, Isfahan, Iran) 1992

A thesis

presented to Ryerson University

in partial fulfillment of the

requirements for the degree of

Master of Applied Science

in the Program of

Electrical and Computer Engineering

Toronto, Ontario, Canada, 2008

I hereby declare that I am the sole author of this thesis.

I authorize Ryerson University to lend this thesis to other institution or individuals for the purpose of scholarly research.

I further authorize Ryerson University to reproduce this thesis by photocopying or by other means, in total or in part, at the request of other institutions or individuals for the purpose of scholarly research.

Web Multimedia Files Characterization
Master of Applied Science, 2008
Mojgan Soraya
Electrical and Computer Engineering
Ryerson University

ABSTRACT

This thesis introduces multimedia workload characterization embedded in popular Web pages and serving by YouTube. The findings of the study are used to build a multimedia workload generator. The workload generator can be used in simulations that investigates methods in improving caching performance for serving multimedia files, reducing multimedia network traffic, increasing YouTube scalability, decreasing startup delay when playing audio/video files embedded in a Web page or serving by short video sharing services, and evaluating the performance of multimedia servers.

In this research, first, an analysis on Web pages consisting of multimedia embedded objects is presented. Also, a characterization study of around 250,000 YouTube popular and regular videos is performed. Based on the analysis of popular Web pages and measurement of YouTube traffic, a workload generator is developed. The workload generator generates the files of popular Web pages and YouTube servers and simulates a user session when accessing a server.

ACKNOWLEDGEMENTS

I am deeply indebted to my supervisor, Dr. Abdolreza Abhari, for his guidance and encouragement on my research. Without his support, this work would not be possible.

I would also like to thank my family for their help. I cannot express my full gratitude to my husband who patiently supported me through whole my graduate study.

Table of Contents

Chapter 1 1
Introduction 1
1.1 Thesis Motivation 2
1.2 Thesis Objectives 3
1.3 Thesis Contributions 3
1.4 Thesis Outline 4
Chapter 2 5
Background Information and Related Work 5
2.1 Background Information 5
2.1.1 Web 2.0 5
2.1.2 YouTube 6
2.1.3 Web Multimedia 7
2.1.4 Multimedia Coding Standards 7
2.1.5 Multimedia Compression 8
2.1.6 Multimedia Streaming Methods 9
2.1.7 Statistical definition 9
2.1.8 Web Workload 14
2.2 Related Work 14
2.2.1 Streaming Media Workload Characterization 14
2.3 Chapter Summary 18
Chapter 3 19
Characteristics of Web Multimedia Files Embedded in the Web Pages 19
3.1 Web Files Properties 19
3.2 Web Multimedia Files Data Collection Strategy 20
3.3 Web Object Sizes 20
3.4 Number of Embedded Objects 26
3.5 Size of Embedded Objects 29
3.6 Multimedia Embedded Objects 32
3.7 Chapter Summary 34
Chapter 4 36
Characteristics of YouTube Popular Video Files Traffic 36
4.1 YouTube Video File Properties 36
4.2 YouTube Popular Data Collection Strategy 37
4.3 YouTube Popular Video Files Statistics 37
4.4 Characteristics of YouTube Popular Videos 41
4.4.1 Video Duration 41
4.4.2 File Size 43
4.4.3 Average Rating of Videos 45
4.4.4 Rating Count 47
4.5 Popularity Analysis of Videos 48
4.5.1 Zipf Analysis 48
4.5.2 Duration of Popularity 52
4.5.3 Metadata Analysis 52

4.6 Implications on Caching 53
4.7 Chapter Summary 54
Chapter 5 55
Characteristics of YouTube Regular Video Files Traffic 55
5.1 YouTube Data Collection Strategies 55
5.2 YouTube Regular Videos Statistics 56
5.3 Characteristics of YouTube Regular Videos 58
5.3.1 Video Duration 58
5.3.2 File Size 60
5.3.3 Average Rating of Videos 61
5.3.4 Rating Count 62
5.3.5 Popularity Analysis of Videos 63
5.4 Correlation between Meta-data Attributes 65
5.5 Implications on Caching 66
5.6 Chapter Summary 66
Chapter 6 68
Multimedia Workload Simulator 68
6.1 Methodology of Generating Workload Simulator 68
6.1.1 Workload Characteristics 68
6.1.2 Server Workload Generator 68
6.1.3 Distribution Form 74
6.1.4 Settings Form 75
6.1.5 Client Session Generator 77
6.1.6 Client Session Form 78
6.2 System Requirements 80
6.3 Chapter Summary 80
Chapter 7 81
Conclusions 81
7.1 Thesis Summary 81
7.2 Thesis Results 81
7.3 Future Work 82

LIST OF TABLES

Table 3.1: Statistics of Web Object Sizes (in bytes) 21
Table 3.2: K-S Values for Popular Web Object Sizes 24
Table 3.3: Estimates of α and R^2 for Popular Web Object Sizes 25
Table 3.4: Statistics of Number of Embedded Objects 26
Table 3.5: K-S Statistics for the Number of Embedded Objects 28
Table 3.6: Estimates of α and R^2 for the Number of Embedded Objects 29
Table 3.7: Statistics of Web File Sizes (in bytes) 29
Table 3.8: K-S Values for Popular Embedded Object Sizes 31
Table 3.9: Estimates of α and R^2 for Size of Embedded Objects 31
Table 3.10: Statistics of File Size and Number of Files 33
Table 3.11: Mean Values of Embedded Object Sizes (in bytes) 33
Table 3.12 : Statistics of Multimedia File Sizes (in bytes) 34
Table 3.13: Summary of Web Object Characteristics 35
Table 4.1: Period Interval 38
Table 4.2: Classification of Daily Video Transactions 38
Table 4.3: YouTube Most Popular Videos Statistics 40
Table 4.4: YouTube Popular Correlation Statistics 53
Table 5.1: YouTube Regular Videos Statistics 57
Table 5.2: Regular and Popular Correlation Statistics 65
Table 6.1: Popular Web Page Record ID Structure 70
Table 6.2: YouTube Record ID Structure 72
Table 6.3: YouTube Video Characteristics Distributions 73
Table 6.4: Example of Expression 6.3 78

LIST OF FIGURES

Figure 3.1: CDF plot of Web object sizes 22
Figure 3.2: pdf Plot of Web Object Sizes 22
Figure 3.3: CDF Comparisons for Feb. 2006 Popular Data Set and Candidate Models..... 23
Figure 3.4: CDF Comparisons for Feb. 2005 Popular Data Set and Candidate Models..... 23
Figure 3.5: CDF Comparisons for Feb. 2004 Popular Data Set and Candidate Models..... 24
Figure 3.6 : LLCD Plot of Web Object Sizes 25
Figure 3.7: CDF Plot for the Number of Embedded Objects 27
Figure 3.8: pdf plot of the Number of Embedded Objects 27
Figure 3.9: LLCD Plot of the Number of Embedded Objects..... 28
Figure 3.10: CDF Plot of Web File Sizes 30
Figure 3.11: pdf Plot of Web File Sizes..... 30
Figure 3.12: LLCD Plot of Web File Sizes..... 31
Figure 4.1: Histogram of YouTube Popular Video Duration 42
Figure 4.2: CDF of Daily Popular Video Durations 42
Figure 4.3: CDF of Weekly Popular Video Durations 43
Figure 4.4: CDF of YouTube Daily Popular Video File Sizes 44
Figure 4.5: CDF of YouTube Weekly Popular Video File Sizes 44
Figure 4.6: Histogram of Average Rating of Daily Popular Files 45
Figure 4.7: CDF of Daily Popular Files Average Rating..... 46
Figure 4.8: CDF of Weekly Popular Files Average Rating 46
Figure 4.9: CDF of Daily Popular Files Rating Count 47
Figure 4.10: CDF of Weekly Popular Files Rating Count..... 48
Figure 4.11: Empirical and model Ranked View Count of Popular Files 49
Figure 4.12: Ranked View Count of Daily Popular Files in log-log Scale..... 50
Figure 4.13: Ranked View Count of Weekly Popular Files in log-log Scale 50
Figure 4.14: CDF of Daily Popular Files View Count 51
Figure 4.15: CDF of Weekly Popular Files View Count..... 51
Figure 5.1: Histogram of Regular Video Durations..... 59
Figure 5.2: CDF of Regular Video Durations..... 60
Figure 5.3: CDF of Regular Video File Sizes..... 61
Figure 5.4: Histogram of Regular Files Average Rating 61
Figure 5.5: CDF of Regular Files Average Rating 62
Figure 5.6: CDF of Regular Files Rating Count..... 63
Figure 5.7: Ranked Video View Count of Regular Files in Log-Log Scale..... 64
Figure 5.8: CDF of Regular Files View Count..... 65
Figure 6.1: Popular Web Page Tree Data Structure..... 69
Figure 6.2: YouTube Tree Data Structure 71
Figure 6.3: Distribution Form..... 75
Figure 6.4: Popular Web Page Setting Form 76
Figure 6.5: YouTube Setting Form..... 76
Figure 6.6: Popular Web Page Client Session Form..... 79
Figure 6.7 : YouTube Client Session Form 79

Chapter 1

Introduction

In the inception of the World Wide Web (WWW), text documents and images were existing documents of Web. With the increasing popularity of the entertainment industry, another kind of content named multimedia became the center of interest. With the growth of multimedia contents, new services are introduced to the Internet. One of these new services is streaming multimedia applications. Recent developments of streaming applications show the popularity of media streaming. This popularity requires introducing new technologies that allow users to receive data at a higher bandwidth and better playback quality.

Another factor in popularity of media streaming is the ability of users to create, publish, distribute, and share their content on the Web. The transformation of the traditional web sites named Web 1.0 into a new generation of Web is referred to as Web 2.0 [1]. Published content posted by Web 2.0 users are also referred to as "User Generated Content (UGC)" such as text in Weblogs (blogs) [2] [3], photos on sites like Flickr [4], and videos on sites such as YouTube [5].

Due to the huge popularity of Web 2.0 sharing sites, modeling Web 2.0 streaming multimedia traffic on the Internet has become a new research topic. From the user's point of view, the streaming multimedia traffic is similar to the Web traffic, as it is varied based on date and time. However, the streaming multimedia traffic has some special characteristics which are different from the traditional Internet traffic. For example, the multimedia traffic is sensitive to delay and jitter, but tolerates some data loss, and it prefers a steady data rate rather than a bursty data rate [6]. Measurement and characterization of streaming traffic can address the performance issues of serving multimedia files on the Web.

The remainder of this chapter is structured as follows. The motivation of the thesis is in section 1.1. The goals of this research are specified in Section 1.2 which is

followed by a brief overview of the research contributions in Section 1.3. This chapter concludes by outlining the organization of the remaining chapters in Section 1.4.

1.1 Thesis Motivation

During the past decade, multimedia streaming technology has become very important part of the multimedia industry. The Internet has experienced an increasing amount of streaming traffic because of increasing the number of multimedia applications which use audio and video streaming [7]. This increase is expected to continue due to increasingly growing of Web 2.0 media streaming technologies.

Our motivation of this thesis is gaining an accurate understanding of Web 2.0 workload characteristics and obtaining statistics in order to model Web 2.0 traffic. Such models can be used for the network traffic management. For example, measuring, analyzing, and modeling the characteristics of multimedia files help develop a workload simulator in order to examine the possibility of effective caching methods and understand how servers and networks respond to variation in load over the Internet. Along with workload simulator, new tools which mimic a set of real users accessing a server are required. This thesis presents a characterization study of one of the extremely popular video sharing Web 2.0 site, YouTube. While many studies of traditional Web workloads have been presented, there have been a few studies of Web 2.0 workloads in the literature. Therefore, this thesis can be considered as a supplementary work in Web 2.0 traffic characterization.

Our measurement findings in this research propose mathematical models for media workloads. Therefore, it could be used to simulate the servers and clients loads in streaming multimedia workload environments. That will be helpful examining the impact of variation of multimedia streaming traffic, the performance evaluation, and the designing of new algorithms for implementing short video sharing systems.

We believe more advanced simulators and workload generators could be designed based on the results obtained from this study.

1.2 Thesis Objectives

The goal of this thesis is generating models to be able to measure the performance of short video sharing services such as YouTube under different workload. Proposing a novel multimedia caching model based on the extracted characteristics is another objective in this research. These findings could lead to generate new streaming traffic models and to design simulators and workload generators for multimedia applications to be able to evaluate the impact of our proposed caching strategy. The validity and applicability of these mechanisms are proven through investigations based on assessment, simulation, and prototype implementation which lead to the final results of this thesis.

1.3 Thesis Contributions

The thesis has the following main contributions:

- We perform a characterization study on the multimedia files embedded in the Web pages. We investigate characteristics of top 500 popular Web sites for 2004, 2005, and 2006. We have considered the popular Web pages for this study. This characterization shows the impacts of embedding multimedia files in the distribution models. The result can be used for development of a multimedia workload generator to examine caching strategy for multimedia files embedded in Web pages. Based on this study, we propose prefetching the embedded multimedia objects of a Web page to reduce the delivery time for the initial portion of the multimedia object as well as reducing the total download time of the Web page. Prefetching allows multimedia objects to begin playback earlier than the other embedded objects of a Web page [8].

- We present a measurement study on YouTube top 100 most viewed videos called YouTube popular videos of the day and week to obtain their characteristics. We have crawled YouTube site from September to November 2007 for popular data set and provided 4,300 complete, unique, daily and weekly popular videos from 17,000 observed contents. According to the

analysis, we find new features for YouTube popular videos such as observing a Zipf's law behavior for popular contents. Based on our findings, we propose that caching of YouTube popular videos at proxies close to clients can improve the performance and scalability of Web 2.0 sites such as YouTube [9].

- We also study on YouTube clips consist of YouTube top ranked list and their related videos called YouTube regular videos. On the crawling between February and April 2008, we have gathered 43,544 complete, unique regular videos from 230,000 distinct videos. Using our results, we suggest that prefetching the related videos of YouTube clips in the client's site can reduce the client's access time and start up delay in watching video.

- Finally, all characterization studies, measurements, and modeling of embedded multimedia objects and YouTube videos apply to develop a multimedia workload generation tool to investigate proposed caching methods and simulate server load variation and user access pattern.

1.4 Thesis Outline

The rest of this thesis is organized as follows: Chapter 2 gives an overview of the multimedia concepts and related works in the area of characterizing workload of multimedia files. Investigations on embedded multimedia files in the Web pages are presented in chapter 3. Chapter 4 describes characteristics of YouTube popular files. Chapter 5 shows characteristics of YouTube regular files. Chapter 6 explains the structure of multimedia workload simulator designed in this study. Chapter 7 summarizes the work and concludes the thesis.

Chapter 2

Background Information and Related Work

There have been many studies characterizing the workloads of traditional and Web 2.0 sites in the literature. This chapter outlines the relevant background information and previous work for the thesis. In section 2.1, a definition of Web 2.0, an overview of YouTube, some multimedia terminologies, a review of the statistical concepts used in the thesis and a definition of Web workload are presented as background information. Recent workload characterization studies of conventional and Web 2.0 streaming media sites are summarized in section 2.2 and then conclusion is in section 2.3.

2.1 Background Information

2.1.1 Web 2.0

Contents in traditional Web are created and distributed by a specified number of Web site owners. Recently, the development of the Web technologies resulted in a significant change in the Internet usage. As users have been able to participate and contribute to Web sites, a new generation of Web sites has emerged. This distinct feature has transformed traditional Web sites named Web 1.0 into a new generation of Web referred to as Web 2.0 [1]. The first attempt to define Web2.0 and understand its implications for the next generation of software was performed by Tim O'Rielly [10]. Web 2.0 aims to enhance Web usability by people. Web 2.0 applications are those that facilitate creating, distributing, sharing, collaborating, consuming, and remixing data from different sources for individual users while allow users mixing their own data and services by others and build a network of participation [10]. One of the most popular multimedia Web 2.0 site is YouTube.

2.1.2 YouTube

During the past three years networked video sharing sites have become very popular Web applications. YouTube [5], the most successful short video sharing site was founded in February 2005 and was acquired by Google in November 2006 for $1.65 billion US. Over 100 million videos are accessed and 65,000 videos are uploaded on this web site every day [41]. YouTube streams its videos based on user requests with about 20 million viewers daily [55]. YouTube enables users to easily share video content and build a social network. It has been one of the fastest growing Websites in the Internet representing a service which is different from the traditional Video-On-Demand (VoD) systems. In traditional VoD systems, content is produced by the site provider and then accessed by viewers. The content and quality of VoD applications are controlled by site owners. In contrast, YouTube enables users to participate and contribute to the site. YouTube users can upload their clips and discuss the contents by using interactive features available on the site.

YouTube video content can be uploaded anytime by anyone. As YouTube expanded, meta-data information such as tag, rating, and comment features were added to build social networking among its users. YouTube provides a list of related videos with keywords or phrases added by video uploaders to best describe their content.

YouTube video playback technology is Adobe Flash Player. Clients can upload their own videos in different video formats like MPEG, WMV, MOV, AVI, and MP4 formats. YouTube accepts uploaded videos and converts into FLV (Adobe Flash Video) format [38]. Using FLV played a key role in the success of YouTube. This enables users to watch the videos without downloading any additional browsers. To playback a video, YouTube benefits from Adobe's progressive download technology. Traditional download-and-play requires the full video file to be downloaded before playing back. Adobe's progressive download feature allows the playback to begin without downloading the entire file. This delivery technique is sometimes referred to as pseudo streaming to distinguish it from traditional media streaming.

YouTube video ID is a unique 11 characters including A-Z, a-z, 0-9, -, and _ symbols and video meta-data consists of author, title, upload time, category, video length, view count, rating count, comment count, average rating, and a "related videos" list. The related videos are considered as videos with the similar description, title, keyword, or tags. They are retrieved through a hyper link of a video Web page.

2.1.3 Web Multimedia

Multimedia streaming refers to the transferring of multimedia data as a continuous stream. In streaming, displaying the multimedia data is started before the entire file has been transmitted. In order to provide reliable and good multimedia streaming quality, several problems have to be addressed. The first is the bandwidth variability between multimedia servers and equipment of end-users. However, this problem can be addressed with caching the data. Therefore, the proper cache method should be considered. Other challenging problems considered through this study are multimedia coding standards, multimedia compression, and multimedia streaming services.

2.1.4 Multimedia Coding Standards

The role of multimedia coding standards is bridging the gap between large amounts of visual data and limited network bandwidth for multimedia delivery. During the past two decades, standard organizations have developed several multimedia coding standards as following [15]:

- H.261 was completed in 1990, and it is mainly used for ISDN video conferencing.
- H.263 was completed in 1996 and it is based on the H.261 framework but includes many additional algorithms to increase the coding performance.
- MPEG-1 was completed in 1991. The target application of MPEG-1 is digital storage media, CD-ROM, at bit rates up to 1.5 Mbps.
- MPEG-2, sometimes also referred to as H.262, was completed in 1994. It is an extension of MPEG-1 and allows for greater input format flexibility and higher data rates for both High-definition Television (HDTV) and Standard Definition

Television (SDTV). The US ATSC DTV standard and European DTV standard DVB both use MPEG-2 as the source-coding format. The MPEG-2 is also used for Digital Video Disk (DVD).

- MPEG-4 Part 2 was completed in 2000. It is the first object-based video coding standard and is designed to address the highly interactive multimedia applications. MPEG-4 Part 2 has been used for mobile application and streaming.
- H.264 is also referred to as MPEG-4 Part 10 Advanced Video Coding. H.264 has greatly improved the coding performance over MPEG-2 and MPEG-4 Part 2. The target applications of H.264 are broadcasting television, high definition DVD, digital storage, and mobile applications.

Currently, the most popular multimedia coding standards for multimedia streaming include MPEG-2, MPEG-4 Part 2 and H.264. It should be noted that besides the video coding standards developed by MPEG and VCEG, there are also multimedia coding schemes such as VC-1 developed by Microsoft, and RealVideo developed by Real Networks. Such media formats are extensively used for multimedia streaming over the Internet.

2.1.5 Multimedia Compression

Multimedia compression is performed by exploiting the similarities or redundancies that exists between given pixels in a multimedia file. Neighboring video frames are typically very similar, Therefore much higher compression can be achieved by exploiting the similarity between frames [16].

Two compression methods are:

- Lossless compression
- Lossy compression

In lossless compression the information is recovered without any alteration after the decompression stage. Lossless compression is applied where the accuracy of the information is essential, such as in medical imaging where it's important to retain fine

detail. Lossy compression refers to the case where the decompressed information is different from the original uncompressed information.

2.1.6 Multimedia Streaming Methods

Basic approaches in multimedia streaming are:

- Media Downloading:

  Multimedia downloading delivery method is similar to a file download. Multimedia downloading approach usually requires long download times and large storage spaces. In addition, the entire multimedia content must be downloaded before viewing can begin. The network bandwidth also plays a significant role in the downloading time as well.

- Media Streaming

  In multimedia delivery by streaming method, the end user can start viewing the content almost as soon as it begins downloading with a limited delay. To achieve a noninterrupted playback, the data must be received at a rate that allows the client device to decode and display each frame of the video sequence according to a playback schedule [17].

2.1.7 Statistical definition

Probability is a useful tool when studying network workloads and performance measurement. Several statistical characteristics are used to analyze data in this thesis such as mean, median, standard deviation, and distribution functions including probability density function (PDF), cumulative density function (CDF), and complementary cumulative density function (CCDF). In this section, we present definitions for these statistical metrics.

2.1.7.1 Mean

In statistics, mean is a commonly used characteristic to state the central tendency of a data set. Given a discrete data set of n data $X = (x_1, x_2, ..., x_n)$, the mean value, denoted by $\bar{x}$, is calculated as:

$$\bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i = \frac{1}{n}(x_1 + x_2 + ... + x_n)$$

Although the mean is used to find central tendency, it may not properly interpret the skew of distributions because mean is significantly influenced by outliers data existed in the data set [19].

2.1.7.2 Median

Median is another characteristic of central tendency indicating the middle value of a sorted data list. The median can be extracted from a list by ordering all the data and picking the middle one. If there is an even number of observations, the average of the two middle values is taken to be the median. In general, median is less affected by outliers than mean, and thus median may be a better statistic to describe the skew of distributions. [19].

2.1.7.3 Standard Deviation

A widely used characteristic of dispersion is standard deviation to show the spread of the data list from the mean. It is computed as the square root of the variance. Given a discrete data set of n samples $X = (x_1, x_2, ..., x_n)$, the standard deviation, denoted by S, is calculated as:

$$S = \sqrt{\frac{1}{n-1}\sum_{i=1}^{n}(x_i - \bar{x})^2}$$

Standard deviation measures how widely a data set is spread [19].

2.1.7.4 Coefficient of Variation (CV)

Coefficient of variation is a relative characteristic of variation among the data. It is the ratio of the standard deviation, S, to the mean, $\bar{x}$ [19]:

$$CV = S / \bar{x}$$

Low-variance distributions are recognized by CV < 1 and high-variance distributions are defined with CV > 1 [56].

2.1.7.5 Correlation

In order to investigate the correlation between data items, the correlation factors between them are calculated. The correlation coefficient, symbolized by the letter ρ is used to find out if there is any possible link between data attributes.

The correlation coefficient $\rho_{X,Y}$ between two random variables X and Y with mean values E(X) and E(Y) and standard deviations σ_X and σ_Y with coefficient of variation (CV) of X and Y variables is defined as [56]:

$$\rho_{X,Y} = \frac{COV(X,Y)}{\sigma_X . \sigma_Y} = \frac{E(XY) - E(X)E(Y)}{\sqrt{(E(X^2) - E^2(X))}\sqrt{(E(Y^2) - E^2(Y)}}$$

Correlations can be strong or weak and it means the relation between two sets of values is significant or slight. The closer value to 1.0 shows the stronger and the closer value to zero shows the weaker correlation between two sets of values.

2.1.7.6 Probability Distributions

Mean, median, and standard deviation statistical measurements show trends of the data set. Those measurements are not adequate for finding the peaks and the tail behavior of the data set. Therefore, we utilize the empirical distribution functions to obtain more insights. Empirical distributions are distributions that are produced by experiment or observation on sample data set.

2.1.7.7 Probability density function (PDF)

The probability density function (PDF) is a function to evaluate the density of a continuous random variable. Continuous random variables are variables that can take any range of values. The PDF of X shows the probability that a random variable X can take on a determined value x. The PDF is denoted as P(x) = P[X = x]. The PDF is graphed by placing the values of random variable X along the x axis and the number of times that x repeats divided by the total number of data sample on the y axis (e.g., Figure 3.2). In the other word, a general shape of a distribution can be depicted by a PDF graph.

2.1.7.8 Cumulative distribution function (CDF)

The cumulative distribution function (CDF) or distribution function is used to find the cumulative probability of a random variable X under a determined value x and is denoted as F(x) = P[X < x]. In a histogram of a PDF, some detailed information is not obvious while CDF provides a better view of changes in a sample data set. CDF is plotted with P[X < x] along the y axis and the values of x along the x axis (e.g., Figure 3.1).

2.1.7.9 Heavy-tailed Distribution

An important feature of some of the distributions is that they exhibit heavy tails. Heavy tail concept is related to the decline of the distribution tail. A heavy-tailed distribution is a distribution which upper tail declines subexponentially like a power law [20]

$$F_c(x) \sim x^{-\alpha} \quad as \quad x \to \infty, \; 0 < \alpha \leq 2$$

Where α is a shape parameter.

Random variables whose distributions are heavy tail exhibit very high variability. It means their variance is infinite, and if $\alpha \leq 1$, their mean is also infinite [46].

2.1.7.10 Zipf Law

Zipf's law is a famous statistical law observed in the behavior of many complex systems with different nature. It states a relationship between the frequency of occurrence of an event and its rank, if the events are ranked based on the frequency of occurrence. The law named for Harvard linguistic professor George Kingsley Zipf (1902-1950) was originally applied to the relationship between words in a text and their frequency of use [60]. If we consider a long text, Zipf's law ranks all words based on the frequency of their occurrences in a descending order. It means the frequency of any word is inversely proportional to its rank in the frequency order. Thus, the most popular word has rank of one; the second most popular word gives rank of two and so on. Therefore the frequency of occurrence (F) is related to the rank of the word (R) according to the following formula:

$$F \sim R^{-\beta}$$

where the constant ß is close to one[45].

Zipf distribution is a distribution characterized by Zipf's law with a set of values or samples that follow the above Zipf's law.

Managing Web cache efficiently is based on studying the Web access characteristics. Many studies have demonstrated that Zipf's law represents many features of the Web characteristics and can be used to describe the popularity of the Web objects. Therefore, Zipf's law describes features of more effective design and use of Web cache resources [60].

2.1.8 Web Workload

Web Workload generator is a tool to investigate responses of servers and network on load variations. This means is exploited in order to model design, performance evaluation, and capacity planning based on current and future usage on the Web. The goal of developing a Web workload generator is to be able to vary the load on servers and networks in a way which mimic the Web. In particular, the effects of high load variability on the Web components like servers and networks is studied [46]. The Web workload characteristics should correspond to those features generated by Web users and its output must correspond to the statistical properties such as file size, file length, and popularity.

2.2 Related Work

2.2.1 Streaming Media Workload Characterization

Workload characterization is an important part of model design and performance evaluation. A number of researches had been done to characterize the commercial products workload, such as [21] and [6].

The research [21] of Veloso *et al.* characterize live streaming multimedia workload from one of the top ten content service provider in Brazil on over 3.5 million requests and a 28-day period of data collection. The authors provide a hierarchical characterization of the client, session, and transfer layers. They conclude that in client layer, client requests follow Zipf-like distribution. In session layer, session ON time and OFF time can be modeled by a Lognormal and exponential distributions but the number of transfers can be fit by Pareto distribution. In transfer layer, Lognormal distribution exhibits the transfer length model. They also build a simulation in GISMO based on the results. This model can be considered as a higher-level traffic characterization of streaming media.

In the Li *et al.* research [6], the study is done on client side characterization of the two streaming multimedia products, RealPlayer and MediaPlayer, which involved the application and network layers traffic characterization. In the application level, packets size and packet interarrival time are analyzed. In network level, traffic characterizations

such as the TCP/UDP/IP packets size and interarrival time are investigated. This could be considered as a lower level streaming media traffic characterization model. The presented results could be useful to generate streaming media simulators.

Other workload characterization studies of media access patterns have considerable insight into the use of streaming media content [22][23][24]. These studies report that the popularity of media files follow a Zipf-like distribution.

Almeida *et al.* [22] study on the client workloads in terms of user session for educational servers. They intend to provide data for building synthetic workloads in order to evaluate caching algorithms. Their results fits curve of the file access frequencies with a concatenation of two Zipf distributions

Chesire *et al.* [23] apply a method to measure streaming media workload in their university network. In their case the RTSP protocol is monitored to collect traces. These traces are then analyzed to obtain characteristics of the streaming media workload. In addition, the trace data is used as input to simulate a caching system in order to study the benefits of caching approach. Basher et al. [26] study how user-interactivity impacts the choice of a suitable caching strategy and cache replacement policy.

Some workload characteristics depend on the media type (e.g. educational versus entertainment). For example, session arrivals for the educational workload could be presented by either a Weibull or a Lognormal distribution, whereas a heavy-tailed Pareto distribution could be considered to model the client session arrivals at the entertainment workloads [24].

There has also been a few research studies related to our work concerned with the analysis and characterization of Web 2.0 streaming media in the Internet. A YouTube traffic characterization is presented by Gill *et al.* [12]. They examine popularity and referencing characteristics, usage patterns, file properties, and transfer behaviors of YouTube videos in a campus network. The authors also analyze social networking aspect of YouTube from the network edge perspective. The edge network considered for their measurement is the University of Calgary campus network which consists of approximately 28,000 students and 5,300 faculty and staff. At an edge network, the number of users is lower than the number of global users and the edge network is physically closer to the clients requesting the content. They conclude using caching

method could improve the end user experience and reduce network bandwidth for accessing YouTube site.

Huang *et al.* [27] analyze the potential benefits of peer-assisted video-on-demand service using the nine-month MSN Video trace and then conduct a trace-driven simulation. In peer-assisted VoD, participating peers that can be other video viewers assist the server in redistributing the downloaded content. Authors show peer-assistance and prefetching can reduce server bandwidth and potentially decrease content providers cost by lowering dependency and the load on the servers. Yu *et al.* [28] conduct another analysis of access patterns and user behaviors in a VoD system in China. They propose a revised Poisson distribution to model the user access rate and Zip's distribution for file popularity. Their results can help to design new strategy in resource allocation and approaches for performance optimization.

Halvey *et al.* [29] provide an analysis of the social interactions on YouTube to understand community behavior. Their results show that many users do not form social network in the online Web sharing sites like YouTube. A few users, who utilize site facilities, create social networks within the site. They find that the distribution of views is not a Zipf-type distribution but the distribution of number of uploads and the number of favorite videos is a Zipf-like distribution.

Mislove *et al.* [32] study four online social networking sites, including Flickr, YouTube, LiveJournal [33], and Orkut [34], and confirm the power-law, small-world and scale-free properties of online social network.

Paolillo [36] investigates the social structure of YouTube, addressing friend relations and their correlation with tags assigned to uploaded videos. Arlitt and Williamson [11] conduct a comprehensive workload characterization study of Internet web servers by analyzing access logs from six different web sites. The authors identify 10 invariants workload characteristics common to all the sites that are likely to persist over time. The authors revisit these results 10 years later at the three academic sites [30]. They find that despite a 30-fold increase in traffic the 10 original invariants still hold.

Another measurement study is presented by Li *et al.* [25] where a media crawler is used to retrieve audio and video contents through the web. The authors find that the distribution of media duration maybe long tailed caused by the Internet traffic associated

with streaming applications. It is important to note that the Li's study was done prior to advent of YouTube.

Zink *et al.* [14] analyze the content distribution in YouTube and then drive a measurement study of YouTube traffic in a campus network. They demonstrate the implications of three content delivery infrastructures: client-based local caching, P2P-based distribution, and proxy caching. The simulation results show that local caching approach improve the overall system performance. P2P-based caching shows a marginal or worse performance than the client-based caching architecture. Simulation results of proxy caching strategy exhibit an effective low-cost solution. The results of their overall simulation show that caching has more potential to decrease network traffic and reduce video access time compared to the other types of content delivery method for YouTube.

Cha *et al.* [31] consider file referencing behavior of user generated content in more details. They sample Daum UCC [35], the most popular UGC service in Korea, and YouTube repository by using web crawler to study file referencing pattern. They focus on the popularity distribution by performing simulations and empirical validation. They observe Zipf-like behavior in the body of the popularity distribution of YouTube contents. Based on a trace-driven simulation, authors conclude caching the most popular videos can reduce server traffic. However, they do not make any assumptions about the exact location of the caches. Their target is investigation of the global cache performance from the server's point-of-view. Cha *et al.* also provide insights into P2P distribution system and find out the small number of files can benefit from P2P.

The similar work to our approach is Cheng's study reported in [13]. They use traces crawled in a three-month period and then analyze the traces to obtain characteristics of YouTube workload. In addition, the trace data is used to investigate a caching or peer-to-peer system in order to study the benefits of such approaches. The authors suggest prefix caching [56] for YouTube videos. In this approach, proxy caches a 5 second beginning part (approximately 200 KB) of the most popular videos for each video. Finally, they conclude utilizing a delivery method based on peer-to-peer architecture for YouTube could be challenging and make the situation even worse. Although, their social network findings drives them to suggest a new approach employed by peer-to-peer technique. They propose a novel peer-to-peer system based on social

network, in which peers are re-distributing the videos that they have cached. A prefetching strategy is used to prefetch the beginning part of the next video in the related social network in each peer in order to decrease the start up delay of P2P overlay. Social networks existing among YouTube videos are made by groups of related videos, if one video has been watched, another video within the group will be more likely to be accessed. Their design is based on the following principle: peers are responsible for re-distributing the videos that they have cached.

Our research is complementary to the previous ones as we compare our results with aforementioned findings. A fact that distinguishes our work from prior studies is our goals. Based on our analysis and the previous works results, we intend to model YouTube characteristics to develop a synthetic workload generator to be able to evaluate the performance of caching or P2P content delivery architectures for YouTube.

2.3 Chapter Summary

In this chapter, we briefly explained Web 2.0 concepts and details of YouTube. An overview of the multimedia terms with an emphasis on the streaming methods, relevant statistical tools and system workload definition were presented. In addition, more relevant related works to this thesis were discussed. Specifically, we reviewed previous studies on YouTube traffic characterization.

Chapter 3

Characteristics of Web Multimedia Files Embedded in the Web Pages

The objective of this chapter is presenting characterizations of popular Web pages to understand the Web traffic issues. We provide insights into the size and number of embedded objects to model characteristics of multimedia files embedded in the Web objects. The remainder of this chapter is organized as follows. Section 3.1 describes properties of embedded multimedia files in Web. Section 3.2 explains Web files data gathering method. Summary statistics of Web object sizes are presented in section 3.3. Section 3.4 describes characteristics observed in number of embedded objects. Section 3.5 studies properties of size of embedded objects. Section 3.6 discusses multimedia embedded objects characteristics. The summaries are given in section 3.7.

3.1 Web Files Properties

Throughout this thesis, we use the term *Web object* to refer to a Web page and a collection of files corresponding to the embedded objects which must be transferred to display the Web page. Popular Web documents also imply to the sites reported as being the most popular pages among the Web users. Focusing on popular Web documents, three workload parameters Web object sizes, number of embedded objects, and Web file sizes are characterized. The html files belong to the front pages of popular Web sites are examined. As front pages of popular Web sites are mostly the more accessible pages in Web sites. When accessing a site, the front page is downloaded automatically but downloading the other pages depends on the user choice.

3.2 Web Multimedia Files Data Collection Strategy

We studied a number of popular Web sites reported in the sites [52][53], naming "Web popular data sets". For this study three data sets were used. A program was written to characterize Web objects. This program generates statistics such as the size of a Web object, and also the type and size of all of its embedded objects. The first step in this program is recognition of the embedded objects that belong to the Web page source file. The program parses the Web page source file and finds the appropriate tags that are related to the unique embedded files such as images, frames, applets, embedded Audio/Video files and embedded dynamic files. Finally, the program reports the size, type and number of embedded objects belonging to each Web object. The result is the characterization of top 500 popular Web sites that fall into three different data sets: Feb. 2006, Feb. 2005, and Feb. 2004. We have considered the popular Web pages for this study because they are more likely to be efficiently designed and they have significant impact on network traffic.

3.3 Web Object Sizes

In this work, a repeated embedded object is considered only once in the measurement of Web object size. The statistics of Web object sizes in popular Web documents are shown in Table 3.1. Based on our results and the previous work [46], Weibull and Lognormal distributions are used for modeling the Web object characteristics in this study.

Table 3.1: Statistics of Web Object Sizes (in bytes)

Characteristic	Feb 2006	Feb 2005	Feb 2004
Mean	179,931	132,335	129,573
Median	110,313	66,887	72,601
Variance	48,722.07	49,589.98	35,716.74
Maximum	1,705.03	2,800.99	1,976.35
Sample size	441	392	351
CV	1.22	1.68	1.45
Model	Weibull a = 0.88 b = 180.02	Weibull a = 0.85 b = 116.79	Weibull a = 0.83 b = 118.37

(a is a shape and b is a scale parameter)

Visual observation of plots (*i.e.,* CDF plot in Figure 3.1 and pdf plot in Figure 3.2), using of goodness of fit methods as suggested in [43] and running ExpertFit [44] simulation software show the Weibull distribution provides a better representation for popular Web object sizes.

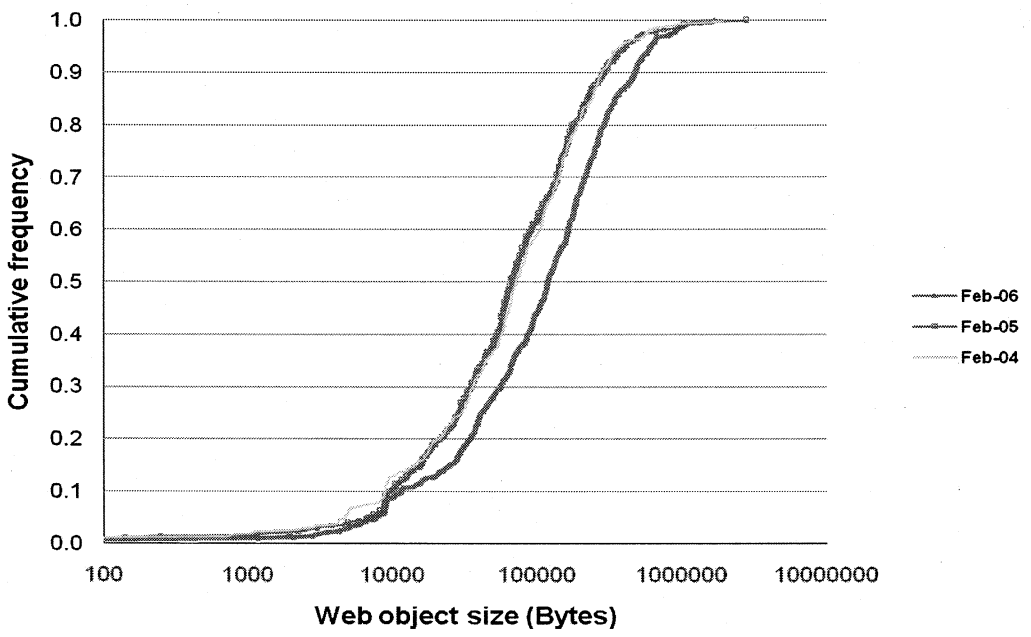

Figure 3.1: CDF plot of Web object sizes

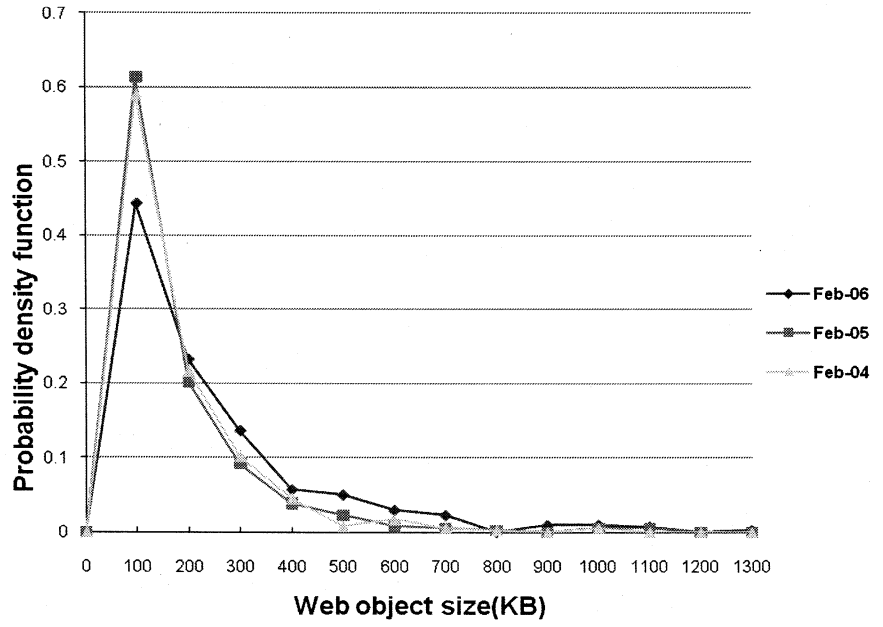

Figure 3.2: pdf Plot of Web Object Sizes

Visual comparison of the CDFs of the models and the CDFs of the data sets are presented in Figure 3.3, Figure 3.4, and Figure 3.5. Visual observation and comparison of

these figures show that the Weibull distribution provides a better representation for the popular Web object sizes.

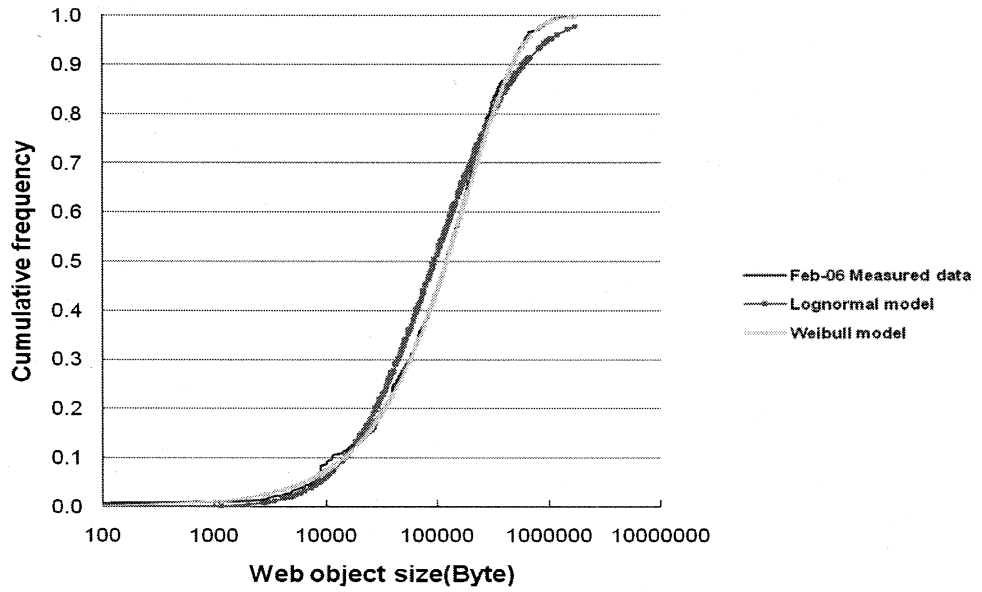

Figure 3.3: CDF Comparisons for Feb. 2006 Popular Data Set and Candidate Models.

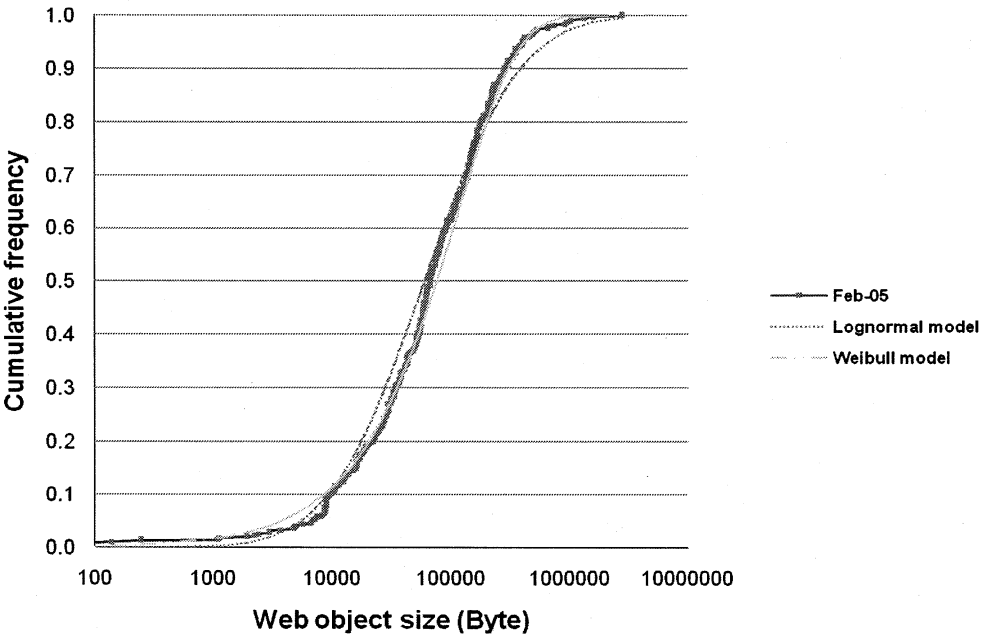

Figure 3.4: CDF Comparisons for Feb. 2005 Popular Data Set and Candidate Models.

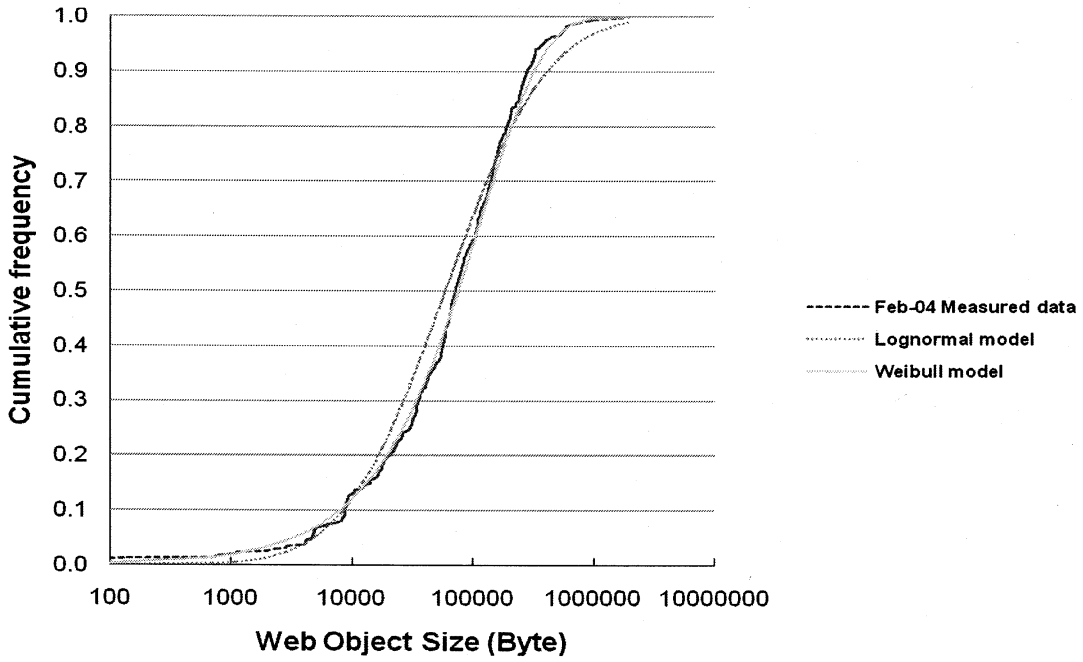

Figure 3.5: CDF Comparisons for Feb. 2004 Popular Data Set and Candidate Models.

As model comparison by visual observation is not accurate enough, the Kolmogorov-Smirnov (K-S) goodness of fit method [43] was utilized in addition to visual comparison of CDF plots. Values (*i.e.,* K-S statistics) for Weibull and Lognormal models are presented in Table 3.2. The values demonstrate Weibull is a more appropriate model for Web object sizes.

Table 3.2: K-S Values for Popular Web Object Sizes

Model	Feb. 2006	Feb. 2005	Feb. 2004
Weibull	0.51	1.03	0.60
Lognormal	1.76	1.54	1.92

The fact that the Weibull distribution is not considered a heavy-tailed distribution suggests that the distribution of Web object sizes for popular data sets is not heavy tail. In order to determine whether or not the data set has a heavy-tailed distribution, LLCD (log-

log complementary distribution) plot as shown in Figure 3.6 and the least-square regression fit as suggested in [47] were both used for the Web object sizes greater than 600 KB. The estimated α values (*i.e.*, slopes) and R^2 values (*i.e.*, coefficient of determination) that assess the goodness of fit for linear regression are presented in Table 3.3.

Table 3.3: Estimates of α and R^2 for Popular Web Object Sizes

Item	Feb. 2006	Feb. 2005	Feb. 2004
α	3.18	2.39	2.18
R^2	0.93	0.94	0.96

Note that R^2 value is closer to 1 indicates a stronger fit, which means more linearity. Since slope values are greater than 2, there is not enough evidence to claim that popular Web object sizes have heavy-tailed distribution.

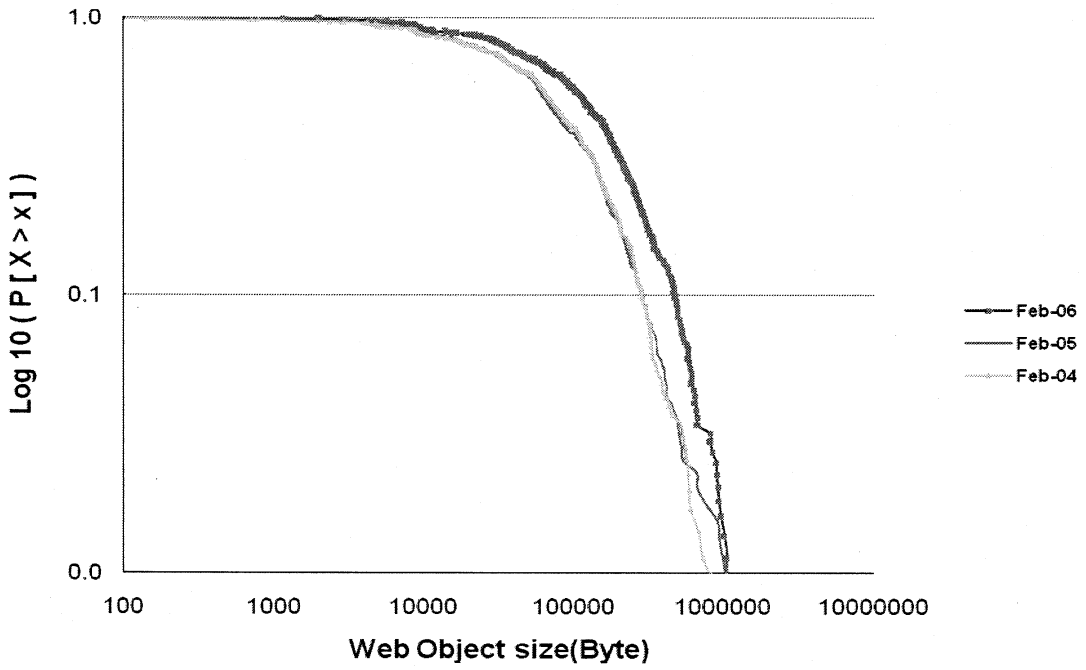

Figure 3.6 : LLCD Plot of Web Object Sizes

3.4 Number of Embedded Objects

The number of embedded objects in a Web page is the number of unique files that must be transferred to display the page at the client site. The statistics of the number of embedded objects for the popular data sets are illustrated in Table 3.4.

Table 3.4: Statistics of Number of Embedded Objects

Characteristic	Feb 2006	Feb 2005	Feb 2004
Mean	33.59	27.54	28.25
Median	25	20	22
Variance	1,001.35	705.18	677.65
Maximum	199	138	125
Sample size	441	392	351
CV	0.942	0.964	0.921
Model	Weibull a = 0.97 b = 33.82	Weibull a = 0.977 b = 27.53	Weibull a = 1.038 b = 28.67

To determine the suitable distribution model, we prepared similar graphs (*i.e.*, CDF plot in Figure 3.7, pdf plot in Figure 3.8, and LLCD plot in Figure 3.9) and tests (*i.e.*, K-S statistics in Table 3.5) as Web object sizes.

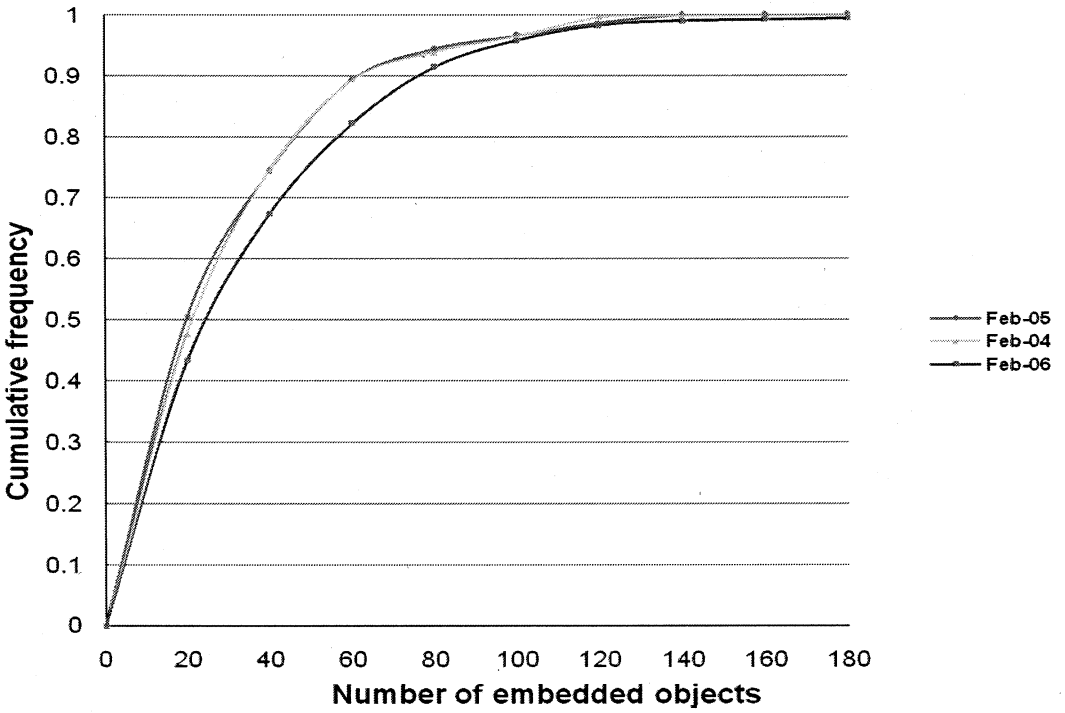

Figure 3.7: CDF Plot for the Number of Embedded Objects

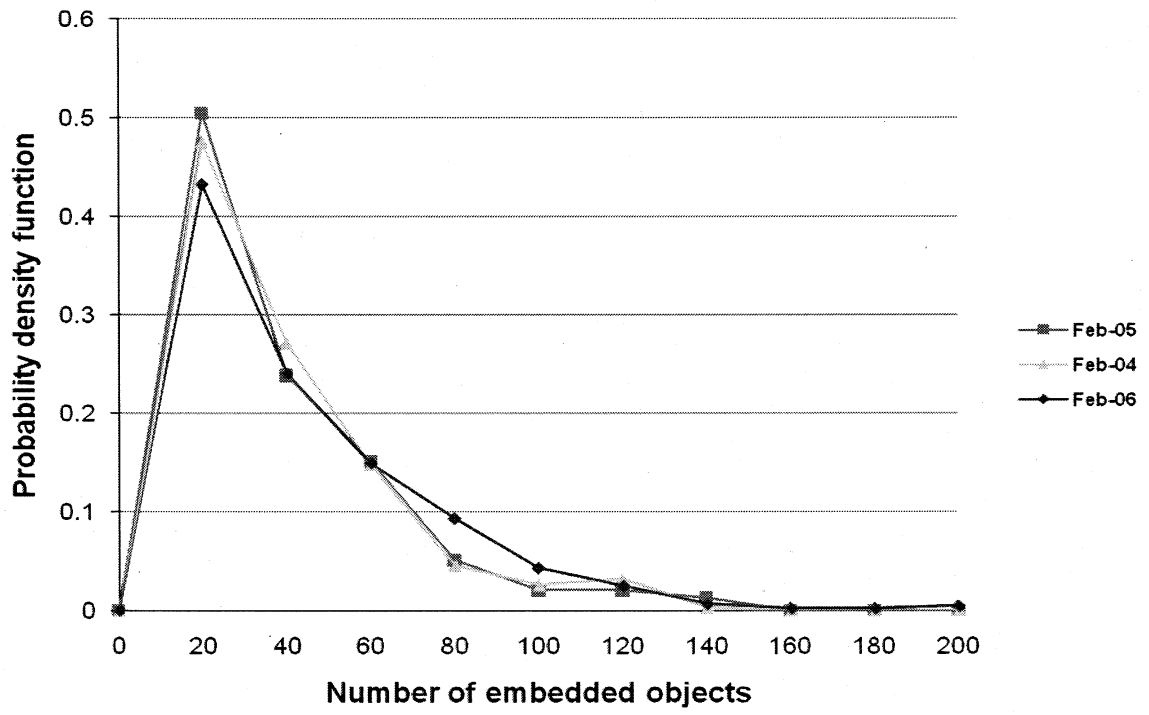

Figure 3.8: pdf plot of the Number of Embedded Objects

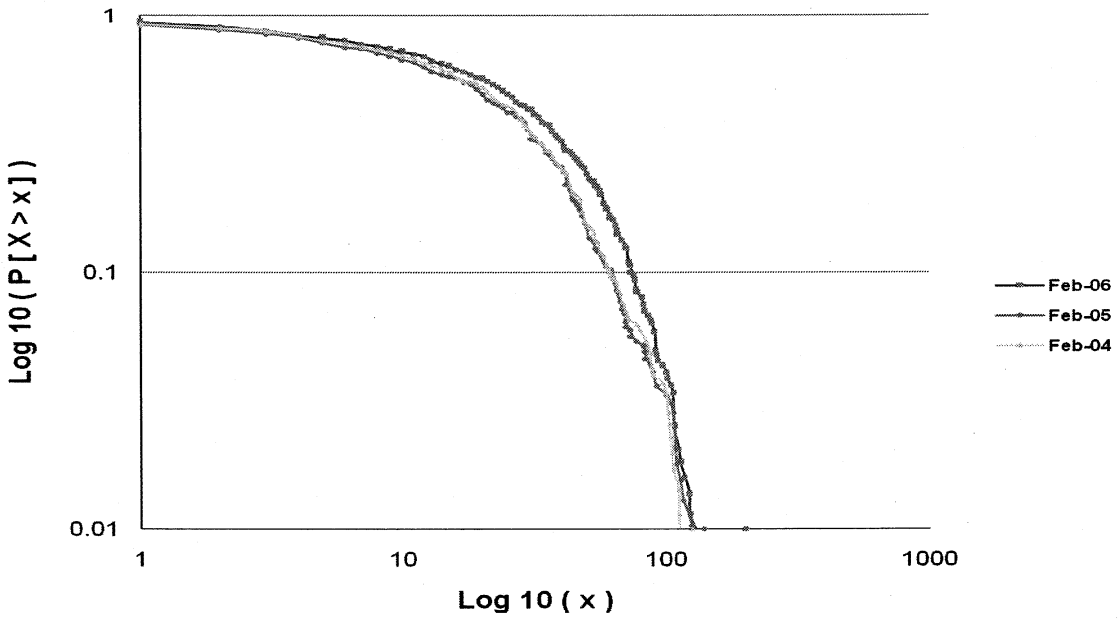

Figure 3.9: LLCD Plot of the Number of Embedded Objects

Table 3.5: K-S Statistics for the Number of Embedded Objects

Model	Feb 2006	Feb 2005	Feb 2004
Weibull	0.98	0.95	0.91
Lognormal	1.76	1.54	1.92

According to the results, Weibull is a better model for number of embedded objects. To examine the linearity and also to estimate the slopes, least-square regression fit was used for the number of embedded object greater than 60 (*i.e.,* k=60). The estimated α and R^2 are listed in Table 3.6.

Table 3.6: Estimates of α and R^2 for the Number of Embedded Objects

Item	Feb 2006	Feb 2005	Feb 2004
α	3.66	3.46	3.72
R^2	0.98	0.84	0.82

The estimated values for slopes are greater than 2, which suggest the number of embedded objects in popular data sets not following a heavy-tailed distribution. Subsequently, it can be concluded that the Weibull model is a better fit for the number of embedded objects and does not follow the heavy-tailed distribution.

3.5 Size of Embedded Objects

The statistics of Web file (*i.e.,* embedded objects) size for the popular data sets are presented in Table 3.7.

Table 3.7: Statistics of Web File Sizes (in bytes)

Characteristic	Feb 2006	Feb 2005	Feb 2004
Mean	5,345.94	4,818.88	4,344.73
Median	1,883.00	1,029.00	971.00
Variance	2.7E+5	2.1E+5	1.7E+5
Maximum	1,153,040	297,773	297,773
Sample size	14,938	10,747	9,578
CV	3.10	3.03	3.06
Model	Lognormal a = 1.74 b = 7.30	Lognormal a = 1.90 b = 6.79	Lognormal a = 1.89 b = 6.75

The same procedure as the Web object sizes and number of embedded objects are performed for size of embedded objects and then the graphs (*i.e.,* CDF plot in Figure 3.10, pdf plot in Figure 3.11, and LLCD plot in Figure 3.12) for measured data are plotted. Using evaluating tests (*i.e.,* K-S statistics in Table 3.8) suggest the Lognormal is a better fit for the Web file sizes distribution.

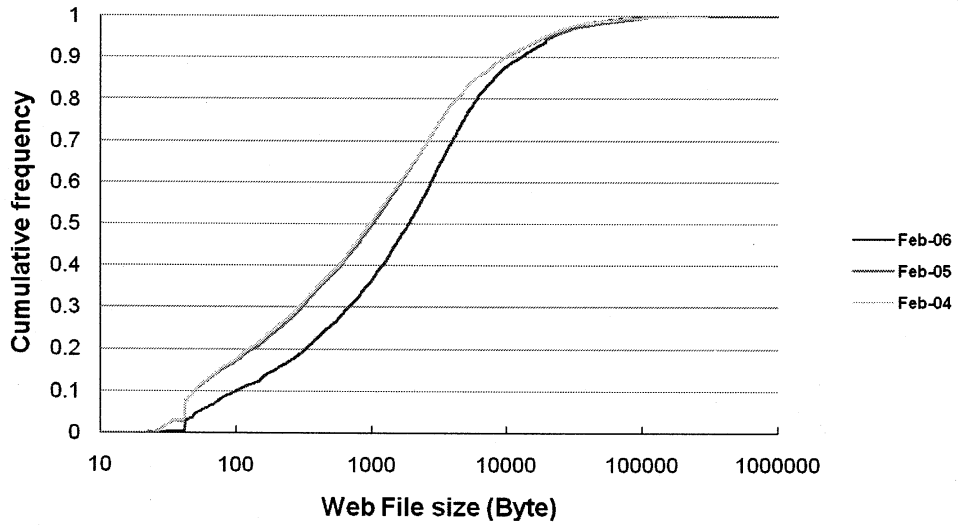

Figure 3.10: CDF Plot of Web File Sizes

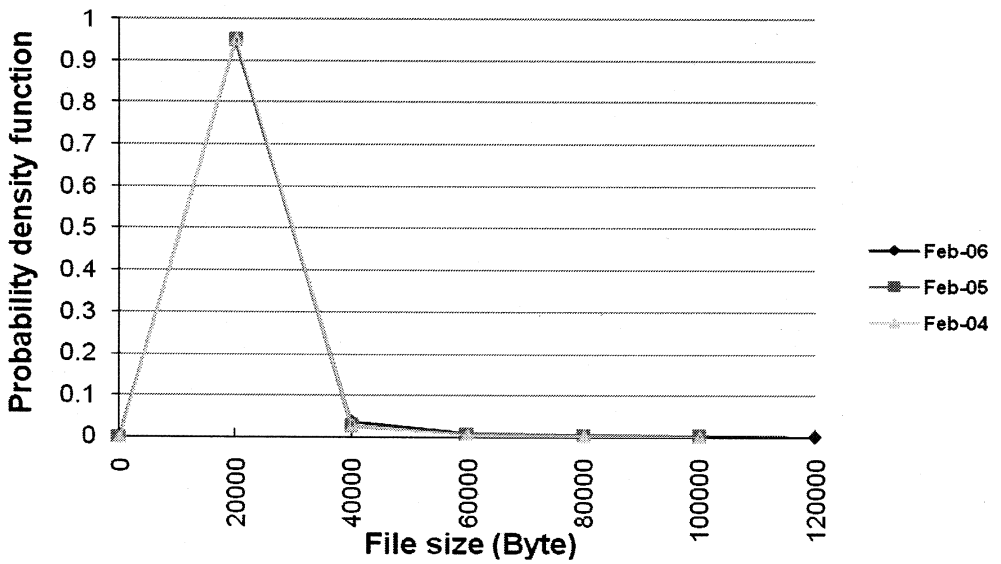

Figure 3.11: pdf Plot of Web File Sizes

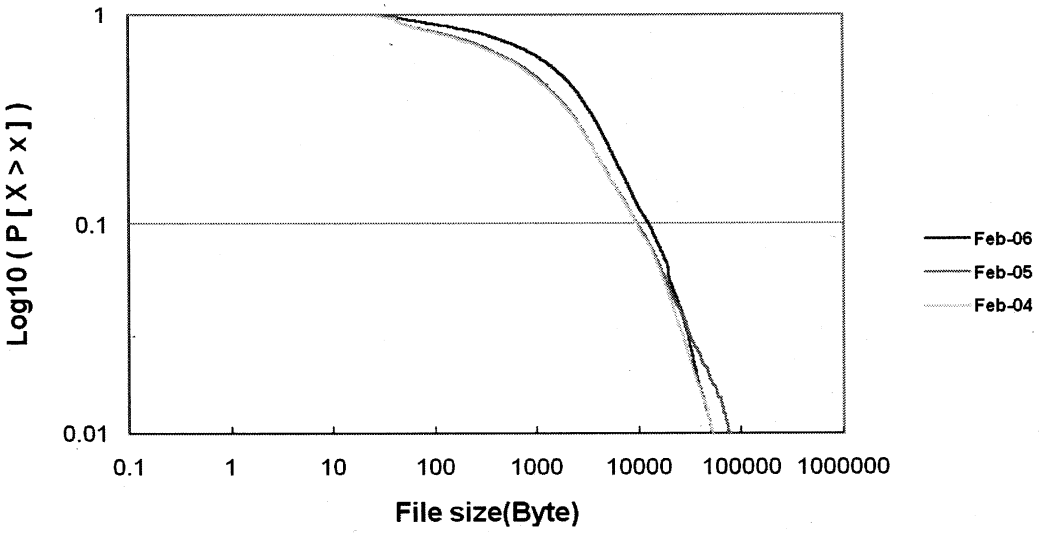

Figure 3.12: LLCD Plot of Web File Sizes

Table 3.8: K-S Values for Popular Embedded Object Sizes

Model	Feb 2006	Feb 2005	Feb 2004
Weibull	0.038	0.066	0.068
Lognormal	0.055	0.049	0.052

The least-square regression was performed to estimate the slops for embedded objects having sizes greater than 14,000 bytes (*i.e.*, k=14,000). The estimated α and R^2 are listed in Table 3.9.

Table 3.9: Estimates of α and R^2 for Size of Embedded Objects

Item	Feb 2006	Feb 2005	Feb 2004
α	2.00	1.38	1.58
R^2	0.80	0.92	0.98

The estimated values for slopes are less or equal to 2 that suggest the size of embedded objects in popular data sets have a heavy-tailed distribution. This fact indicates existence of large embedded objects in our data. Multimedia files usually have large sizes. To be able to determine whether or not the multimedia files existed in embedded objects cause a heavy-tailed file size distribution, we performed the final analysis on the type of embedded objects presented in the next section. In this study, we have also used Aest software developed by Crovella and Taqqu [49] to measure the tail weight (α). The result of Aest software on the data agrees with our observation from using LLCD to measure the tail weight.

3.6 Multimedia Embedded Objects

Embedded objects are mostly image files with a BMP, GIF, JPEG, and PNG format. In the popular data set recorded in Feb. 2006, from 14,747 observed embedded objects, 14.64% of the embedded objects are text files (HTML, JAVASCRIPT, PLAIN, and XML), 84.37% of embedded objects are picture files, and 0.90% of the embedded objects are animation files (FLASH) that can be considered embedded multimedia files with SWF format.

Comparing the sizes of text and image files with the file size of multimedia files shown in Table 3.10 indicates that the size of text and image files of the popular Web pages are smaller than the size of multimedia files embedded in the Web pages. Considering size and number of multimedia files shown in Table 3.10 for Feb 2006 data set suggests that the existence of multimedia files have caused heavy-tailed file size distribution for 2006 data set. The similar results were found for 2005 and 2004 data sets.

Table 3.10: Statistics of File Size and Number of Files

Type	Feb 2006		Feb 2005		Feb 2004	
	size%	number%	size%	number%	size%	number%
Multimedia	5.87	0.91	6.76	1.16	8.78	1.57
Text	28.09	14.64	30.56	8.33	29.14	8.01
Image	72.27	84.37	61.56	90.32	60.81	90.28

Table 3.11 presents the mean values for different objects embedded in the popular Web pages.

Table 3.11: Mean Values of Embedded Object Sizes (in bytes)

Type	Feb 2006	Feb 2005	Feb 2004
Multimedia	33,745.55	23,910.98	24,349.83
Text	10,881	15,913	16,205
Image	4,449.87	2,900	2,926.43

As indicated in Table 3.11, the mean values of multimedia files are larger than the other types of embedded objects. We studied the stored animation files from popular Web files for the three data sets. Statistics on their sizes are presented in Table 3.12.

Table 3.12 : Statistics of Multimedia File Sizes (in bytes)

Characteristic	Feb 2006	Feb 2005	Feb 2004
Mean	33,745.55	23,910.98	24,349.83
Median	16,997.50	16,692.00	17,143.00
Variance	3.4E+9	6.7E+8	9.3E+8
Maximum	451,965	185,198	200,799
Sample size	136	125	150
CV	1.743	1.09	1.26
Model	Lognormal a = 1.46 b = 9.53	Lognormal a = 1.03 b = 9.62	Lognormal a = 0.97 b = 9.63

Table 3.12 shows multimedia files have large sizes and large variances. Also observing the small number of multimedia files shown in Table 3.10 indicates that a few large multimedia files are included among popular embedded objects. All of these facts support our hypothetical theory that the main reason of following the heavy-tailed file size distribution resulted from the existence of multimedia embedded objects in popular Web pages. This conclusion has direct impact on caching methods that are used for proxy caching responsible for caching popular Web pages.

3.7 Chapter Summary

A summary of the important characteristics of Web objects for popular Web documents is presented in Table 3.13.

Table 3.13: Summary of Web Object Characteristics

Characteristic of Web objects	Popular Web documents
Distribution of Web object sizes	Weibull distribution Not heavy-tailed distribution
Distribution of number of embedded objects	Weibull distribution Not heavy-tailed distribution
Distribution of embedded file size	Lognormal distribution heavy-tailed distribution

We observed that the distribution of the Web object sizes is a Weibull distribution and it is not a heavy-tailed distribution. We found that the distribution of the number of embedded objects is a Weibull distribution and also it is not heavy-tailed, which agrees with the results reported by other researches on popular Web pages in [11][49][50][51]. The distribution of popular Web file sizes is a Lognormal distribution and it is heavy-tailed. We believe the existence of embedded multimedia files causes this heavy-tailed distribution. Thus the caching methods that are used for proxies should take this fact into consideration to provide better cache performance.

Chapter 4

Characteristics of YouTube Popular Video Files Traffic

Our motivation to study YouTube popular files traffic is to understand its workload characteristics and obtain statistics to examine the possibility of effective caching methods. To perceive a clear view of YouTube traffic, we take into account a number of characterization metrics. The metrics include view count, average rating, video length, file size, and rating count. The remainder of this section details the characterization metrics considered in this study. Section 4.1 defines YouTube video properties. Section 4.2 describes YouTube data collection strategy for popular video files in this work. Summaries of the statistical data of popular videos collected in this work are presented in section 4.3. Section 4.4 describes characteristics observed in YouTube popular videos over the measurement period. In section 4.5 popularity analysis of YouTube popular videos are mainly considered. Implications on caching are specified in section 4.6. Finally, conclusion is described in section 4.7.

4.1 YouTube Video File Properties

YouTube is a web-based service allowing the sharing of videos on the Internet. The video playback technology of YouTube is according to Adobe Flash Player. Clients can upload their own videos in different video formats like MPEG, WMV, MOV, AVI, and MP4 formats. YouTube accepts uploaded videos and converts into FLV (Adobe Flash Video) format [38]. Using FLV played a key role in the success of YouTube.

YouTube video ID is a unique 11 characters including A-Z, a-z, 0-9, -, and _ symbols generated by YouTube provider. YouTube video meta-data consists of author, title, upload time, category, video length (duration), view count, rating count, comment count, average rating, and a "related videos" list. The related videos are considered as

videos with the similar description, title, keyword, or tags. They are retrieved through a hyper link of a video Web page.

4.2 YouTube Popular Data Collection Strategy

In the first phase of our data gathering, popular data collection, we focused on the top 100 most viewed videos of the day and week to draw insights into the YouTube popular attributes. We developed software in java language to retrieve the top 100 most viewed videos of YouTube. The software establishes a connection to YouTube Web site and utilizes an API function provided by YouTube for developers to identify the top 100 videos of a day and a week [39]. The result of the YouTube API function is an XML file included some statistics such as the video ID, video length, average rating, rating count, view count, upload time and access time. In order to provide the video file sizes, each video ID is extracted from the XML file and sent to a server [40] which is one of the YouTube contents providers. This routine is repeated every day and every week during the data collection period to gather daily and weekly data sets respectively. In this way, we are able to collect the attributes of the popular video files which we referred to as "popular data set" based on daily or weekly duration, for further analyses.

During the crawling from September to November 2007 for popular data set, we provided 4,300 complete, unique, daily and weekly popular videos from 17,000 observed contents.

4.3 YouTube Popular Video Files Statistics

We observed the top 100 most viewed videos in a day and in a week during 54 consecutive days, starting from September 13, 2007 to the end of November 5, 2007. The summary statistics of YouTube popular traffic such as the total number of daily and weekly video files and the total number of unique daily and weekly video files are listed in Table 4.1. The prevalence of the transactions including completed, gapped, and interrupted categories are also counted as tabulated in Table 4.2.

Table 4.1: Period Interval

Item	Information
Start Date	13/09/07
End Date	05/11/07
Daily Video Files	5,400
Weekly Video Files	900

Total Video Files	6,300
Unique Daily Video Files	3,605
Unique Weekly Video Files	700

Total Unique Video Files	4,305

Table 4.2: Classification of Daily Video Transactions

Category	Transactions	% of Total
Completed	3,435	56.32
Gapped	2000	32.78
Interrupted	665	10.90
Total	6,100	100

A transaction is referred to a replied message including video file information requested by the data collector software. Each transaction belongs to one of the following three categories [12]:

1. "Completed", which means the transaction was successfully obtained.

2. "Interrupted", which shows the transaction that its connection to the server was reset before the transaction was complete.

3. "Gapped", which indicates the transaction that the data collector missed its data, and thus software was unable to parse the remainder of the transaction.

Our next analysis is based on completed and interrupted transactions accounted for 4100 video files. Interrupted transactions are also considered in analysis because before resetting server some characteristics have been extracted which can be used. Table 4.3 summarizes statistics of YouTube popular videos including number of unique video files based on the highest view count, video duration, file size, average rating, rating count and view count of 4100 video files which were collected on daily basis. We also repeated the same procedure to gather data for YouTube weekly popular videos that is shown in Table 4.3. The results in Table 4.3 derived by tools such as Excel and ExpertFit softwares.

Table 4.3: YouTube Most Popular Videos Statistics

Time Frame	Daily	Weekly
Unique IDs	3,605	700
Video Duration (Minutes)		
Mean	4.42	3.89
Median	3.41	3.11
CV	0.93	1.08
Model	Weibull a =1.13 b = 278.03	Weibull a = 1.13 b = 246.07
File Sizes (KB)		
Mean	10,631.41	9,269.17
Median	8,330.83	7,297.01
CV	0.93	1.11
Model	Weibull a = 1.144 b = 11,193	Weibull a = 1.133 b = 11,212
Average Rating		
Mean	4.25	3.96
Median	4.60	4.32
CV	0.21	0.23
Model	Weibull a = 3.63 b = 4.44	Weibull a = 4.73 b = 4.25
Rating Count		
Mean	228	802
Median	69	357
CV	2.23	1.47
Model	Weibull a = 0.62 b = 148	Weibull a = 0.75 b = 668
View Count		
Mean	52,759.18	271,090.60
Median	27,382.50	190,827.50
CV	1.74	1.01
Model	Log-Logistic a = 2.32 b = 30,149	Log-Logistic a = 2.05 b = 195,492

(a is the shape parameter and b is the scale parameter.)

In popular data set, 100% of gathered data have rating count and view count meta-data value.

4.4 Characteristics of YouTube Popular Videos

In this section, we study the characteristics of video duration, file size, average rating, rating count, and view count. A comprehension of YouTube file properties implies many implications for Web 2.0 designers. For example, file size attribute has implications for selecting cache strategy.

4.4.1 Video Duration

Figure 4.1 illustrates a histogram plot of YouTube popular video durations in YouTube popular daily and weekly data set. YouTube imposes a limitation of 10 minutes on video duration. In our studies, we recognized a few videos longer than 10 minutes limit. The mean value of YouTube daily popular video duration is 4.42 minutes with a median of 3.41 minutes. The CV is close to 1. We also found that 56% of the daily videos and 64% of the weekly videos are between 1 and 5 minutes long.

Our analysis indicates that duration of YouTube popular video in our daily and weekly data sets are longer than YouTube popular video found by Gill [12]. Gill's study [12] had found that the median value of the duration of YouTube popular videos was around 3 minutes for daily popular videos and 2.2 minutes for weekly popular videos. The difference that we observe in our data sets is likely to be a result of different time of measuring YouTube characteristics.

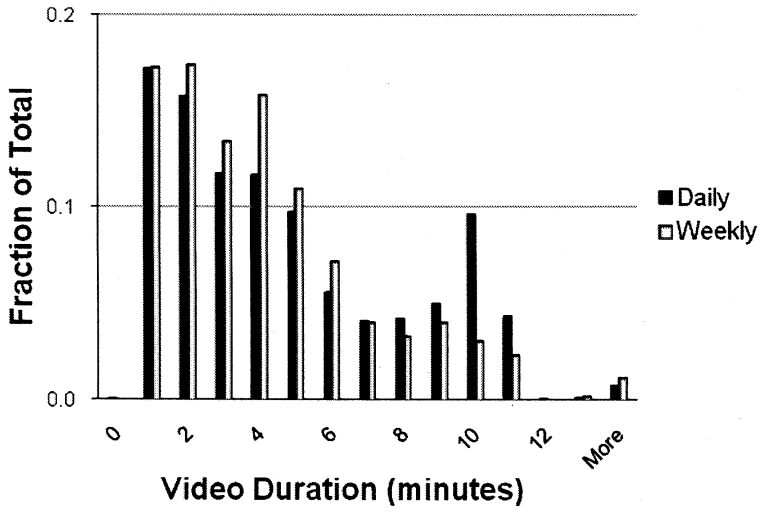

Figure 4.1: Histogram of YouTube Popular Video Duration

Figure 4.2 plots CDF graph of YouTube popular videos durations in daily basis. The daily popular video duration is fitted by a Weibull distribution, whose parameters are shown in Table 4.3.

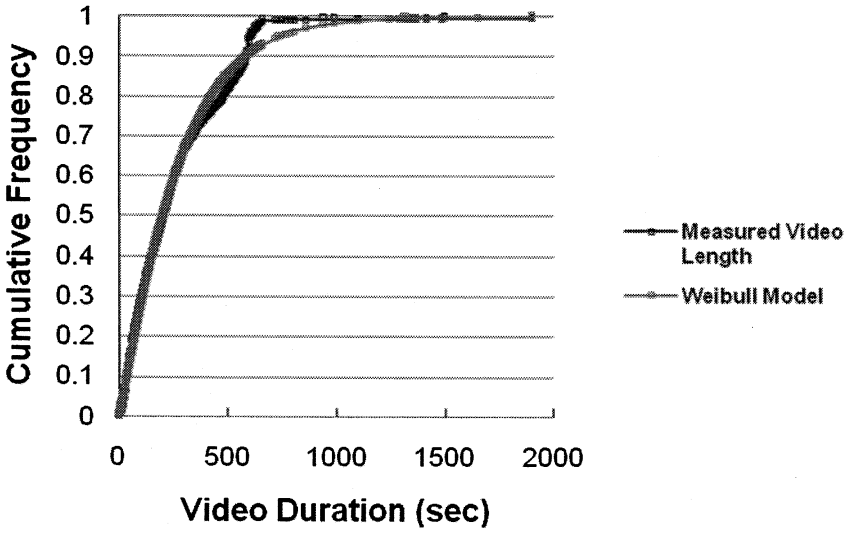

Figure 4.2: CDF of Daily Popular Video Durations

Figure 4.3 plots CDF graph of YouTube popular videos durations in weekly basis. The weekly popular video duration is fitted by a Weibull distribution as well. The corresponding parameters are shown in Table 4.3.

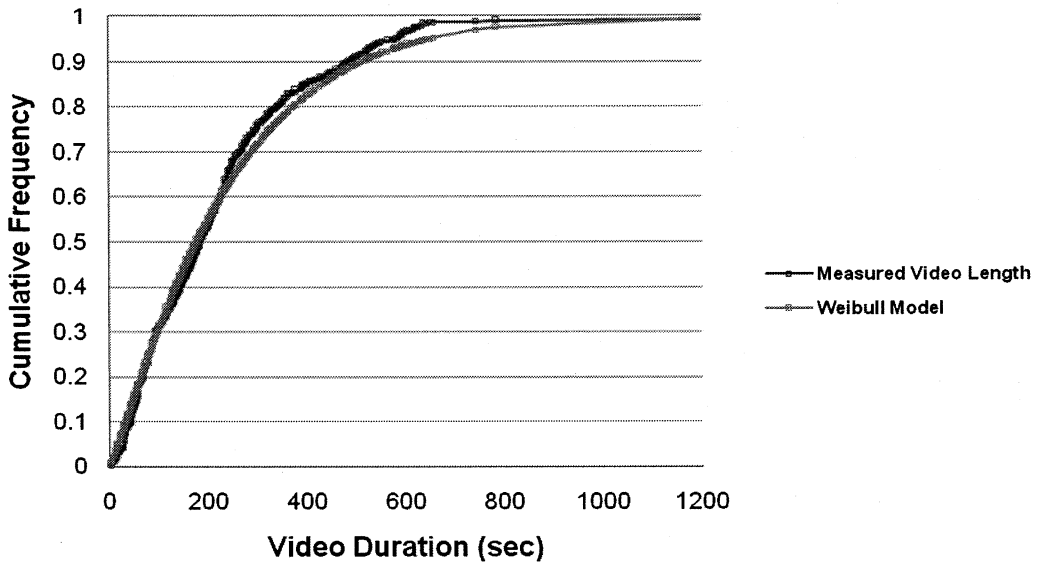

Figure 4.3: CDF of Weekly Popular Video Durations

4.4.2 File Size

The CDF plot of the file sizes for daily popular video files is shown in Figure 4.4. YouTube allocates a limited space of 100 MB for video files [42]. A small number of the videos, approximately 0.1% in our popular data set, are larger than 100 MB that shows the restriction of file size is not applied properly. In addition, according to our investigations it was found out that 90% of the popular videos requested by users having sizes less than 23.5 MB. It concludes that there are not significant numbers of large size video files posted or accessed by users. Gill's analysis [12] showed that YouTube popular video file sizes are slightly less than what we concluded. In Gill's study 90% of the requested popular videos are less than 21.9 MB. These facts and the low coefficient of variation (CV) of file sizes suggest using a disk-based caching can be effective if a proxy caching system employed for YouTube popular video files.

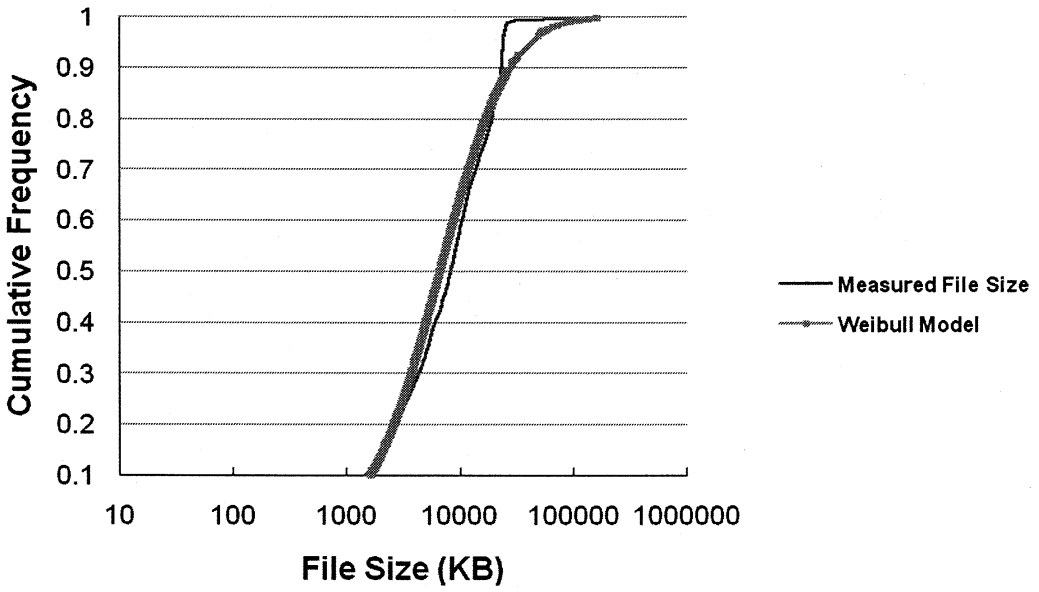

Figure 4.4: CDF of YouTube Daily Popular Video File Sizes

The CDF plot of the weekly popular video file sizes is shown in Figure 4.5.

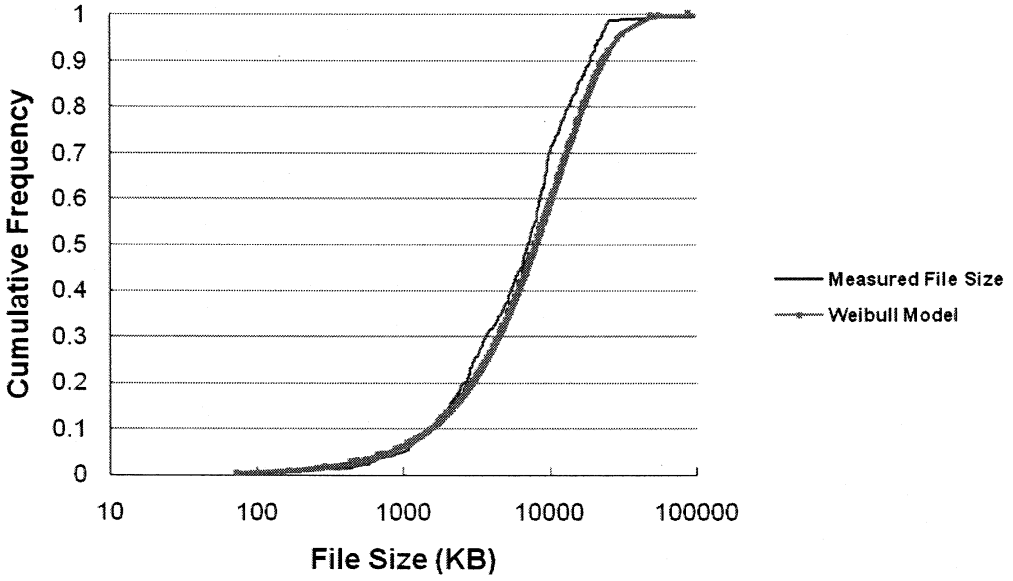

Figure 4.5: CDF of YouTube Weekly Popular Video File Sizes

Based on our measurement and analyses, Weibull and Lognormal distributions can be used for modeling the popular video file size characteristics in this study. To be

able to assess how well lognormal and Weibull models fit the distributions of the popular data set, the parameters of lognormal and Weibull were measured. Also visual observation of CDF plot using goodness of fit methods as recommended by [43] and running ExpertFit [44] simulation software show the Weibull distribution provides a better representation for YouTube popular video file sizes.

4.4.3 Average Rating of Videos

Average rating is the interactive feature provided by UGC sites. The average rating of a video provides a criterion of how much YouTube users interested in a video clip. This user interaction attribute is utilized to rate videos in a scale of 0 to 5 red stars (0 showing low and 5 indicating high). In Figure 4.6 present a histogram of average ratings for YouTube popular video files is presented.

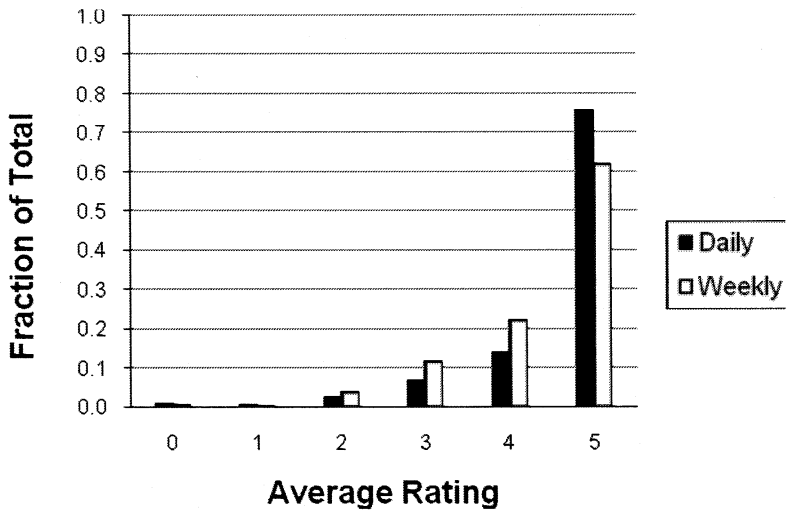

Figure 4.6: Histogram of Average Rating of Daily Popular Files

The CDF model of the average rating for daily popular video is shown in Figure 4.7.

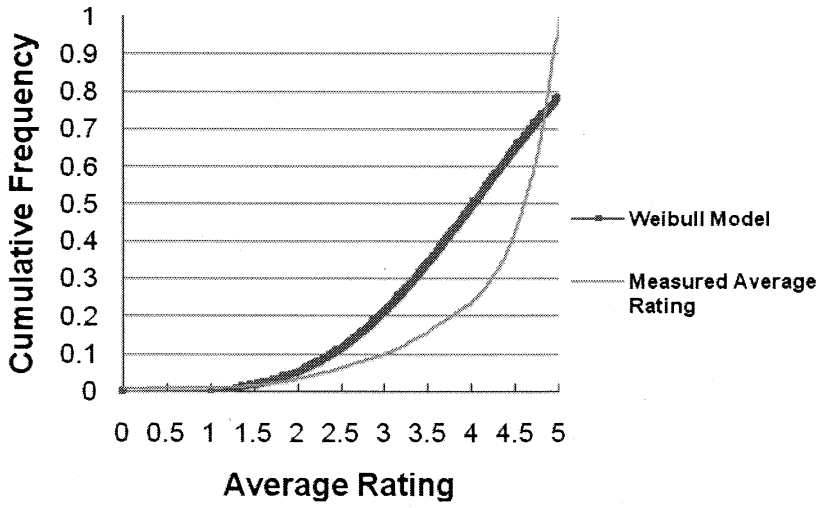

Figure 4.7: CDF of Daily Popular Files Average Rating

The CDF model of the average rating for weekly popular video is shown in Figure 4.8.

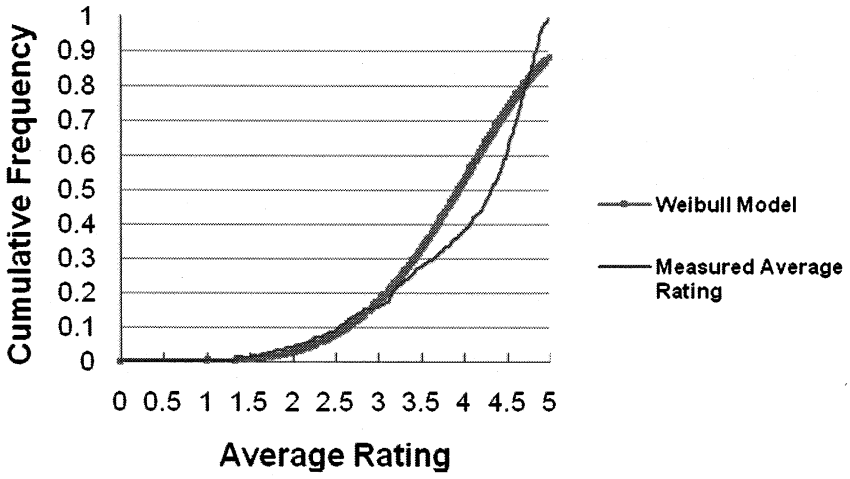

Figure 4.8: CDF of Weekly Popular Files Average Rating

According to our modeling findings, Weibull distribution fits to our daily and weekly popular data set. For all sets of popular videos, we also observed over 90% of the ratings have the average of 3 or higher which is close to Gill's results [12]. According to information listed in Table 4.3. the mean values of average rating of daily and weekly

popular videos are 4.25 and 3.96 with very little coefficient of variation (CV) of 0.21 and 0.23 in daily and weekly data sets, respectively. There is a very similar observation by Gill's analysis where the mean rating values are 4.20 and 3.93 and the coefficient of variation (CV) is 0.24 and 0.23 for daily and weekly popular videos, respectively. YouTube contains a high volume of video files such that searching a content of interest for a viewer is difficult. Consequently, YouTube rating system helps viewers to find the high rating video among the group of their desirable YouTube videos.

4.4.4 Rating Count

Rating count meta-data for each video indicates number of users who rated that video. Figure 4.9 shows the CDF graphs of rating count feature for daily popular contents in log-log scale.

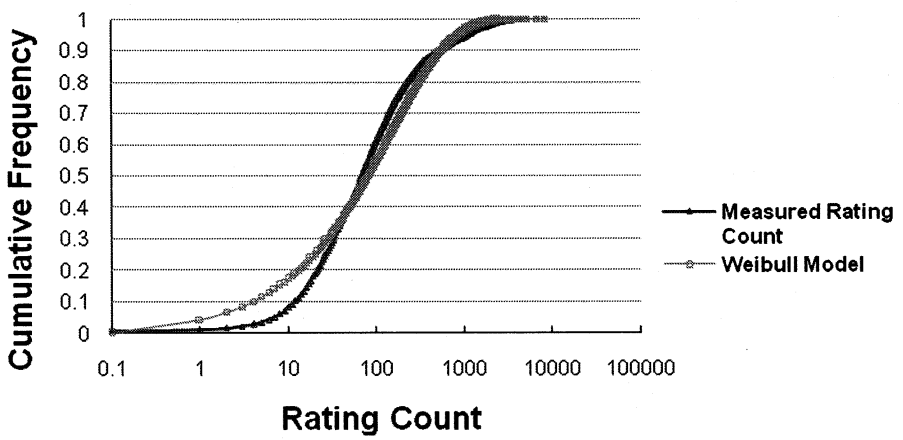

Figure 4.9: CDF of Daily Popular Files Rating Count

Figure 4.10 shows the CDF graphs of rating count feature for weekly popular contents in log-log scale as well.

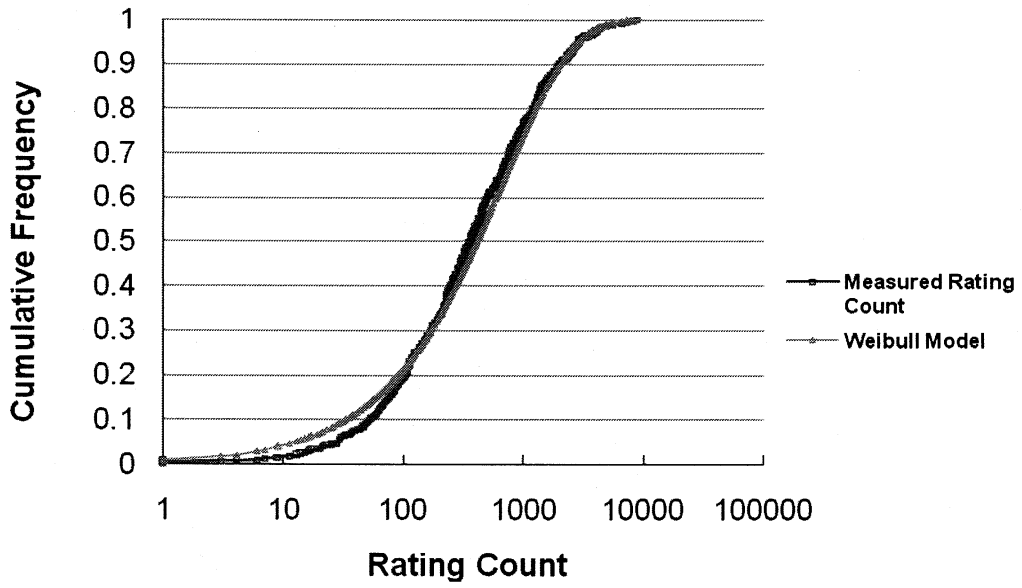

Figure 4.10: CDF of Weekly Popular Files Rating Count

The best candidate to model rating count characteristic in our daily and weekly popular data is Weibull distribution.

4.5 Popularity Analysis of Videos

File popularity is one focus of interest for designing video caching systems. In this section Zipf analysis and popularity analysis are considered to analyze file popularity and referencing behavior of YouTube users in our popular data set.

4.5.1 Zipf Analysis

A Zipf-like graph is plotted based on the rank ordered list of objects versus the frequency of the access to that object on a log-log scale and the existing of a straight line is an indication of Zipf's distribution.

Figure 4.11 and Figure 4.12 shows that the popularity distribution for our daily popular data set follows a Zipf-like distribution on a log-log scale. This distribution of

references among video files means that some files are extremely popular, while most files have relatively few references.

Based on a regression analysis on the daily popular data set, the exponent ß = 0.80 and goodness of fit R^2 value is 0.98. This ß value is more than the values reported by Gill [12]. Note that higher ß value means more similar Zipf-like behavior. Observing the ß value of 0.80, we expect to have an effective proxy caching when proxy only caches YouTube popular files.

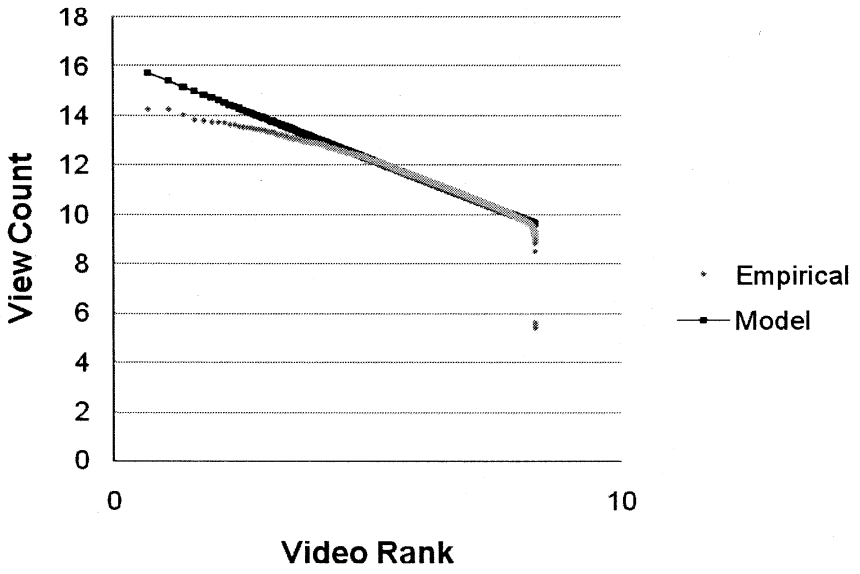

Figure 4.11: Empirical and model Ranked View Count of Popular Files

As illustrated in Figure 4.11, there is a linear graph at the beginning of the plot and a tail at the high rank position. The existence of the tail shows that there are less regular videos at the high rank position of our popular data set than the regular data set observation.

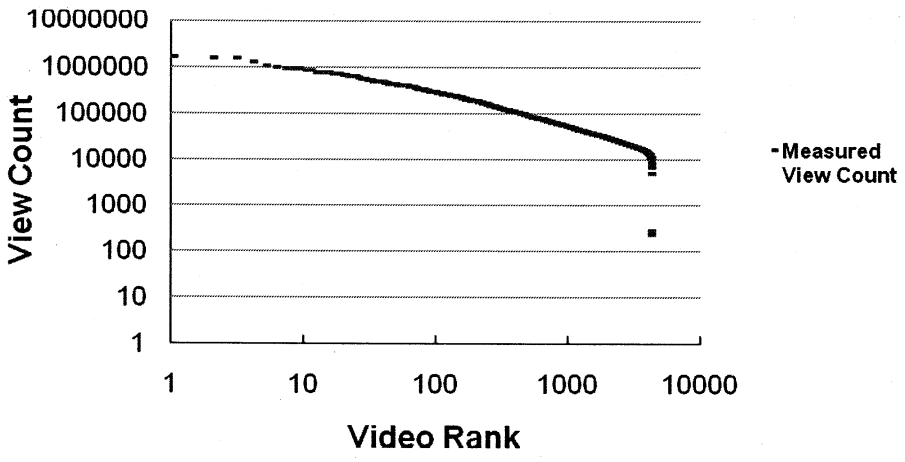

Figure 4.12: Ranked View Count of Daily Popular Files in log-log Scale

Figure 4.13 shows a Zipf-like distribution for our weekly popular data set. Based on a regression analysis on the weekly popular data set, the exponent ß = 0.79 and goodness of fit R^2 value is 0.71.

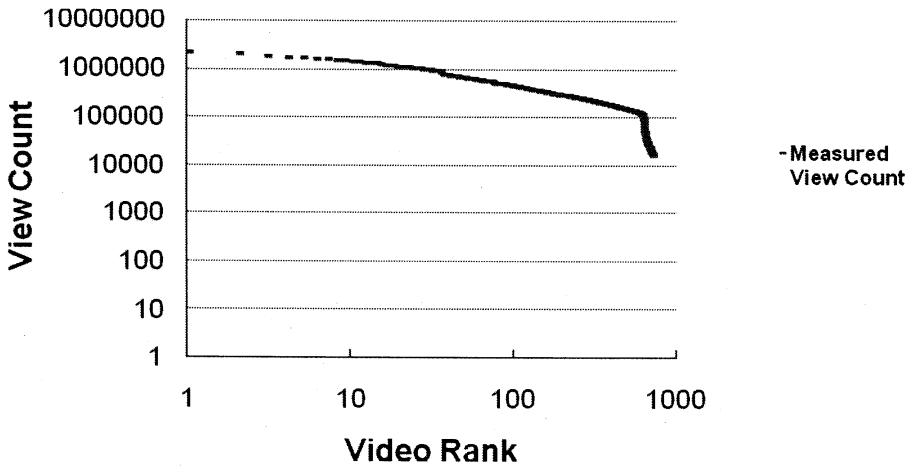

Figure 4.13: Ranked View Count of Weekly Popular Files in log-log Scale

Figure 4.14 shows CDF graph of YouTube daily popular view count. The daily popular view count is modeled by a log-logistic distribution displayed in Table 4.3.

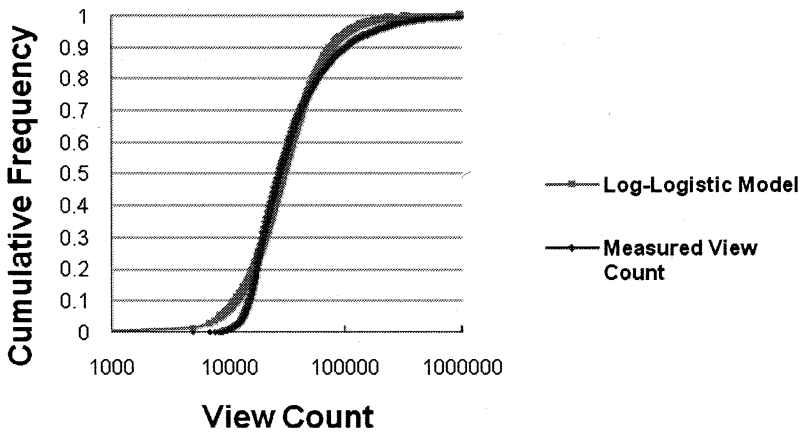

Figure 4.14: CDF of Daily Popular Files View Count

Figure 4.15 shows CDF graph of YouTube weekly popular view count. The weekly popular view count is modeled by a log-logistic distribution displayed in Table 4.3.

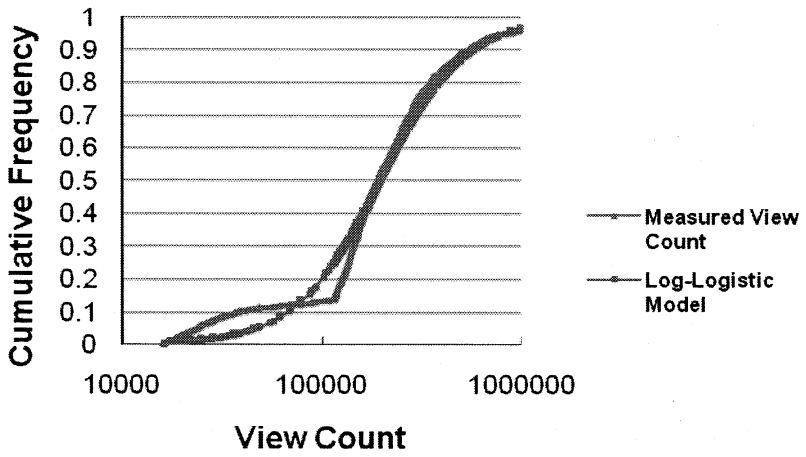

Figure 4.15: CDF of Weekly Popular Files View Count

As YouTube infrastructure does not allow downloading of videos, a considerable number of video clips are requested more than once resulting in increasing the number of views. When considering the reference behavior of users, we observe more Zipf-like

distribution for popular files. This characteristic indicates that caching the popular contents in proxy has the potential to decrease network traffic and bandwidth requirements for serving YouTube popular video files. We will discuss the implications on caching of our results in more detail in section 4.6.

4.5.2 Duration of Popularity

The objective of popularity duration is to determine the amount of time that a video file remains in the most popular video list. In order to investigate this feature of YouTube popular video files, we observed YouTube traffic for 5 consecutive days, hour by hour, from the evening of Dec.13, 2007 to Dec.17, 2007. In total we considered 10,200 most viewed videos in which the number of unique videos was 766.

We found that the mean value of popularity duration for most viewed videos in our data set is 12.22 hours with a median of 9 hours. The CV is less than 1 (CV = 0.91) suggesting videos on the most popular list are not highly variable in their popularity durations. The minimum duration that a popular content stays in the popular list is 1 hour and the maximum time is 41 hours. Our analysis on duration of popularity suggests that the optimum time value for the cache refreshment of popular videos should be approximately every 12 hours.

4.5.3 Metadata Analysis

In order to investigate the correlation between meta-data items, we calculate the correlation factors between them. For example, we examine the correlation of video length with video view count. Also we intend to see if there is any correlation between video length and video average rating, or possible correlation between video view count and video average rating. Therefore, we investigate correlation coefficient to find out if there is any possible link between mentioned meta-data attributes. Table 4.4 lists the possible relations of different properties of YouTube popular video files.

Table 4.4: YouTube Popular Correlation Statistics

	Weekly Popular Videos		Daily Popular Videos	
Characteristics	Correlation Coefficient	Relation	Correlation Coefficient	Relation
Video duration & Average Rating	0.21	Weak	0.2	Weak
View Count & Average Rating	0.17	Weak	0.18	Weak
Video duration & View Count	0.12	Weak	0.08	Weak

According to Table 4.4, we conclude that weak correlations exist between tabulated attributes. For example, the correlation value between video duration and view count (the third row) and the relation of video duration and average rating (the first row) reflect duration of a video has a weak correlation with its popularity. This fact proposes a weak correlation among cited features. We expect that high rating videos to be more popular than low rating videos, but the low coefficient correlation of view count and average rating (the second row) does not suggest this fact.

We assume some of the meta-data attributes such as view count can be used to design the cache replacement policy for popular video file caching. It means a policy which removes a video from the cache with the least view count may outperform other cache replacement policies for video caching.

4.6 Implications on Caching

An efficient approach to save bandwidth and prevent user latency is caching the most used data at proxies close to clients. Existence the high number of videos with smaller size in YouTube than traditional videos such as popular Web sites studied in chapter 3 and observing a Zipf's law for popular contents, both have implications on web caching. Observation of Zipf-like behavior in popular data set suggests that proxy caching of YouTube popular videos can significantly increase the scalability of the server and decrease the network traffic. Caching YouTube most popular contents by cache refreshing time of every 12 hours is probably the effective choice.

4.7 Chapter Summary

In this chapter, we studied YouTube traffic over the most popular videos to observe and explore their characteristics. After the analysis of the YouTube workloads, we found that popular video file sizes have low coefficient of variation (CV) with the mean size of 10 MB. Also plotted graphs, visualized observations, and regression analysis indicated Zipf-like behavior for popular video files. The observations of characteristics suggest that caching of popular videos in a proxy can improve the performance and scalability of Web 2.0 sites such as YouTube. The huge growth of Web 2.0 creates scalability problem for its centralized resources and requires decentralized approaches such as caching. The increased availability of meta-data in Web 2.0, for example view count in YouTube is an appropriate metric that can be exploited to make such proxy caching techniques more effective. In addition, we also showed that another important aspect of cache efficiency can be cache refreshing time.

Chapter 5

Characteristics of YouTube Regular Video Files Traffic

This chapter introduces a workload characterization study of YouTube regular videos. The observations obtained in this chapter provide insights into the properties of Web 2.0 traffic. Section 5.1 describes the YouTube data collection strategies in this work. Summary statistics of the data collected in this work are presented in section 5.2. Section 5.3 describes characteristics observed in YouTube popular videos over a specified measurement period. In section 5.4, correlation between meta-data attributes is considered. Implications on caching are explained in section 5.5 that followed by conclusion in section 5.6.

5.1 YouTube Data Collection Strategies

In order to search YouTube regular clips that we name it "regular data set" we modified the crawler. Our regular data set comprises of a beginning set of top ranked list consists of "Most Discussed", "Most Viewed", "Recently Featured", and "Top Rated" video files, for the time range of "Today", "This Week", "This Month" and also "All Time". The next data sets are related videos of the previous ones. The related videos are defined as the contents linked to other videos with a similar description, title, keyword, or tag named by uploaders.

We ran the crawler between two to three times per week such that in every crawl the beginning set initials with the top ranked videos comprising of 100 to 200 videos. We collected a number of YouTube videos in a graph-like structure, where each video is a node and the top ranked videos are the first level in the graph. When video b is seen in the list of related videos of video a, then we can assume there is an edge from video a to video b. Thus, video b can be retrieved from the list of related videos of node a. A recursive method by our crawler is used to identify all related videos up to four levels.

Whenever the crawler finds a video, the related video list is checked and the video is added to the list in case of being a new video. The crawler ignores any repeated videos. For each video in the list, at the first step, the crawler retrieves information from YouTube API including our required meta-data except file size. Then the crawler connects to one of the YouTube servers to extract video file sizes. It also rejects videos with incomplete meta-data information.

On the crawling between February and April 2008 for regular data set, the crawler traversed YouTube to four levels of the graph to obtain totaling 43,544 complete, unique regular videos from 230,000 processed contents.

5.2 YouTube Regular Videos Statistics

We provide an extensive analysis on the characteristics of YouTube regular videos. In our measurements, some characteristics such as YouTube category, length, size, and date added are considered as static attributes. Their values do not change in a period of data collection. Other characteristics like view count, rating count, and comment count that change from time to time are referred to as dynamic features. The dynamic information is assumed to be static in a single crawl. The summary statistics of YouTube including number of unique video files based on the highest view count, view count, average rating, video length, file size, and rating count are listed in Table 5.1 for regular video files.

Table 5.1: YouTube Regular Videos Statistics

Unique IDs	43,544
Video Duration (Minutes)	
Mean	4.55
Median	3.91
CV	1.05
Model	Log-Logistic a = 2.54 b = 226.54
File Size (KB)	
Mean	9,809.52
Median	8,480.33
CV	0.93
Model	Gamma a = 1.80 b = 5,441.93
Average Rating	
Mean	4.56
Median	4.75
CV	0.11
Model	Weibull a = 15.95 b = 4.73
Rating Count	
Mean	1,309
Median	298
CV	3.61
Model	Weibull a = 0.52 b = 629.24
View Count	
Mean	541,564.31
Median	145,119
CV	3.18
Model	Weibull a = 0.55 b = 290.130

(a is a shape and b is a scale parameter.)

Our rating count analysis is based on 42,941 regular rated videos that represent 98.6% of the entire regular data set rated by users. We also noticed that the entire regular data set accounted for 100% of data contains view count meta-data value.

We observed the mean values of average rating, rating count, and view count items in regular data set are higher than the similar values in popular data set (Table 4.3 and Table 5.1). The main reason is that the beginning set of regular data set is a sample of videos from the list of top ranked videos. The next data sets are built by considering related videos of the list of the top ranked videos. Therefore, the high mean values for regular videos is a result of existence of the most popular videos with large values of rating count, average rating, and view count as a starting point of the regular data collection procedure.

5.3 Characteristics of YouTube Regular Videos

In this section, a detailed analysis of Table 5.1 through plotting their distributions models is discussed.

5.3.1 Video Duration

YouTube consists of short clip videos. In our entire data set, the majority of the regular video durations are between 4 and 5 minutes. Figure 5.1 shows the histogram graph of YouTube regular video durations. Cheng [13] analysis depicted three peaks of YouTube video durations. The first peak is within one minute, which identifies YouTube as a site of very short videos. The second peak is between 3 and 4 minutes. This peak is seen due to the existence of a large number of videos in the Music category. The third peak is close to 10 minutes that is the limit of the videos duration. They found that the histogram of video duration can be fit by a normal distribution. Our analysis indicates two peaks. The first one is around 4 minutes and the second one is close to the YouTube limitation of 10 minutes. Our 4 minutes peak may be the result of the huge volume of music videos and 10 minutes peak is outcome of the YouTube video duration limit. We also observed 3.4% of videos in our data set are longer than 10 minutes, as some

authorized users have permission to upload videos longer than 10 minutes. Also since the length limitation was applied from March 2006 [58], it is possible the large videos have been uploaded before that date.

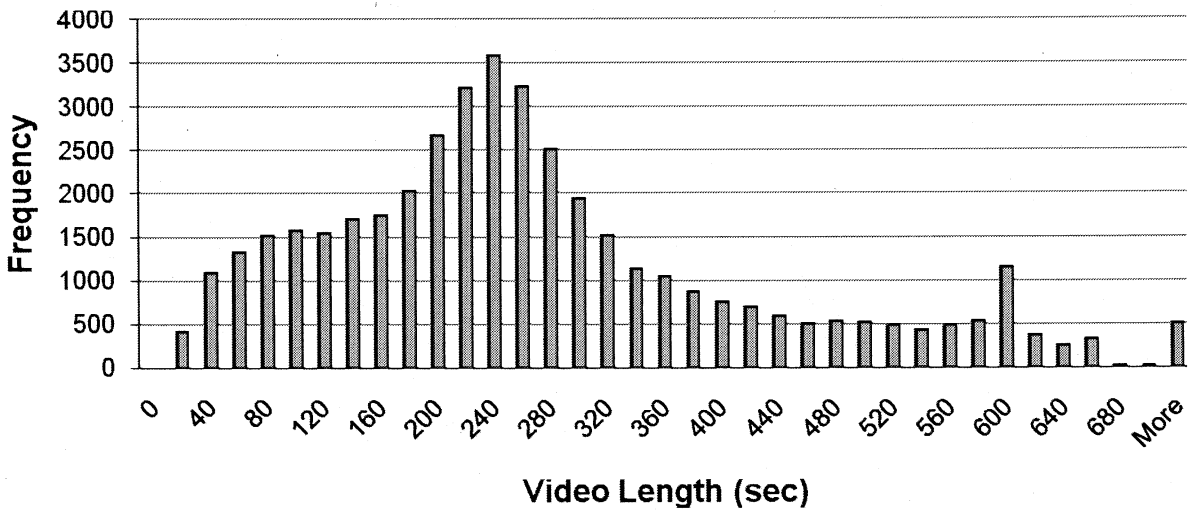

Figure 5.1: Histogram of Regular Video Durations

Figure 5.2 exhibits CDF graph of video durations of YouTube regular videos. The CDF graph shows that the regular video durations can be modeled by a log-logistic distribution, whose parameters shown in Table 5.1.

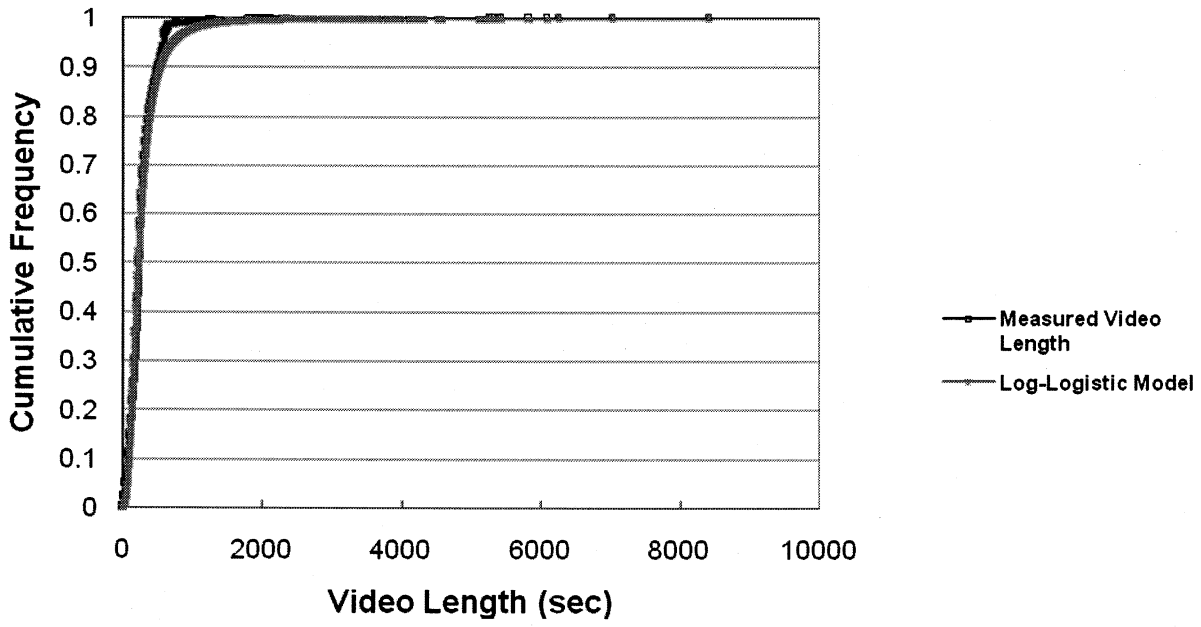

Figure 5.2: CDF of Regular Video Durations

5.3.2 File Size

Using the video IDs from the YouTube crawl, we retrieved 43,544 regular video file sizes. In our data, 90% of the regular videos are less than 19.2 MB and nearly 0.1% of our regular data contains video files larger than 100MB. We calculated that the average regular video file size is about 9.8 MB whereas Cheng [13] found it as 8.4 MB. Considering the mean value that we found for regular video file size and existence of 83.4 million videos on YouTube servers [59], as of April 9, 2008, the required disk space to store all YouTube videos is more than 817 terabytes. For such fast growing site, a high efficient storage management must be considered. We found that the YouTube regular file size distribution can be modeled by a Gamma distribution (shown in Table 5.1). The CDF plot of YouTube regular video file sizes and its model are drawn in Figure 5.3.

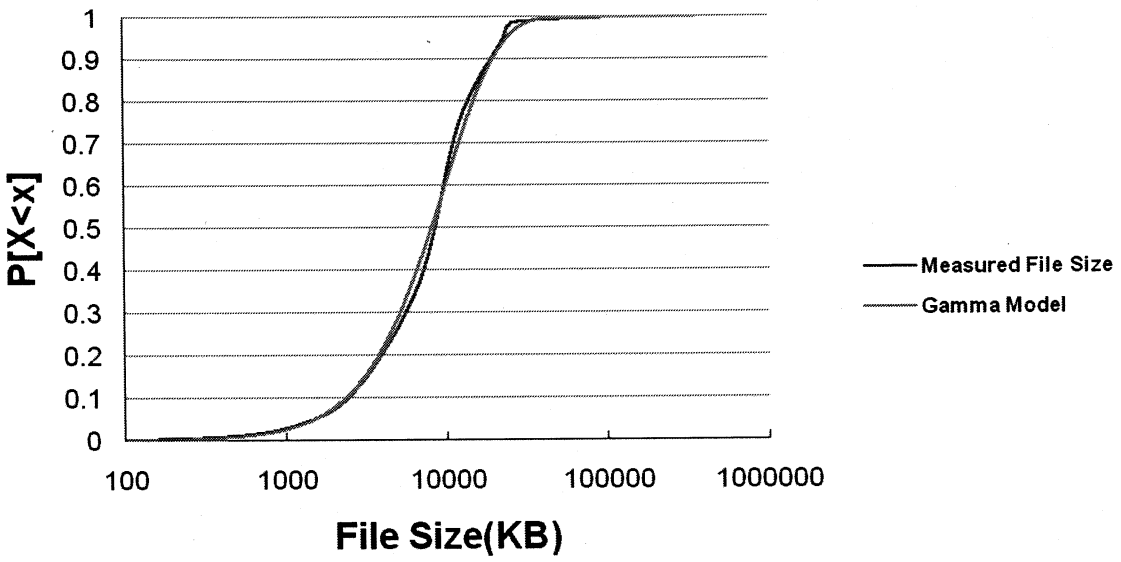

Figure 5.3: CDF of Regular Video File Sizes

5.3.3 Average Rating of Videos

Figure 5.4 plots the histogram of average rating for our regular data set.

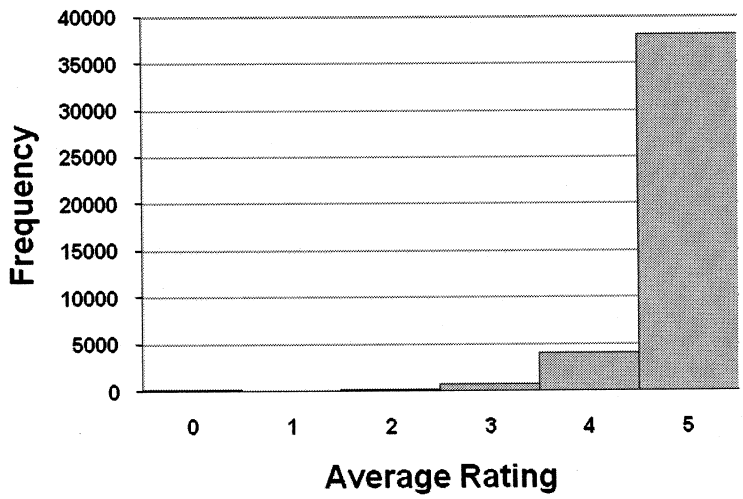

Figure 5.4: Histogram of Regular Files Average Rating

Figure 5.5 shows the CDF plot of average rating of YouTube regular videos. Visual observation of CDF plot and running ExpertFit [44] simulation software show that Weibull distribution fit better than other models for average rating in our regular data set.

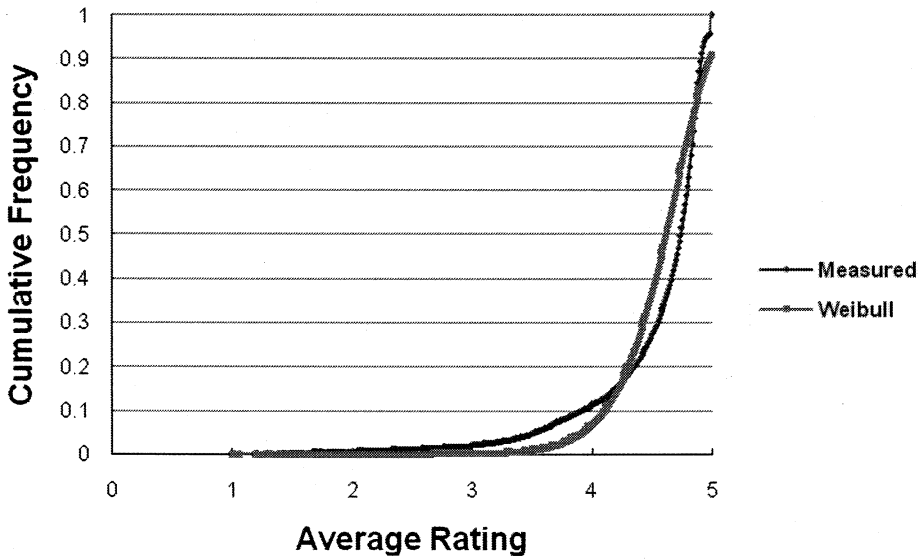

Figure 5.5: CDF of Regular Files Average Rating

The regular data set graphs show over 90% of the average ratings are 4 or more. In Table 5.1, the mean average rating value for regular data is recorded as 4.56 with coefficient of variation (CV) of 0.11. It should be noted that the presence of high rating videos in regular files is the result of considering the top ranked video list as the starting set for collecting regular data set.

5.3.4 Rating Count

Table 5.1 shows the mean and median values of rating count are fewer than view count in our data. This fact is caused as YouTube provides the rating system just for registered users. This point demonstrates that many users do not rate a video. The CDF graphs of rating count property for regular videos in log-log scale are modeled in

Figure 5.6. Weibull distribution observed as the best candidate to model rating count characteristic in our regular data.

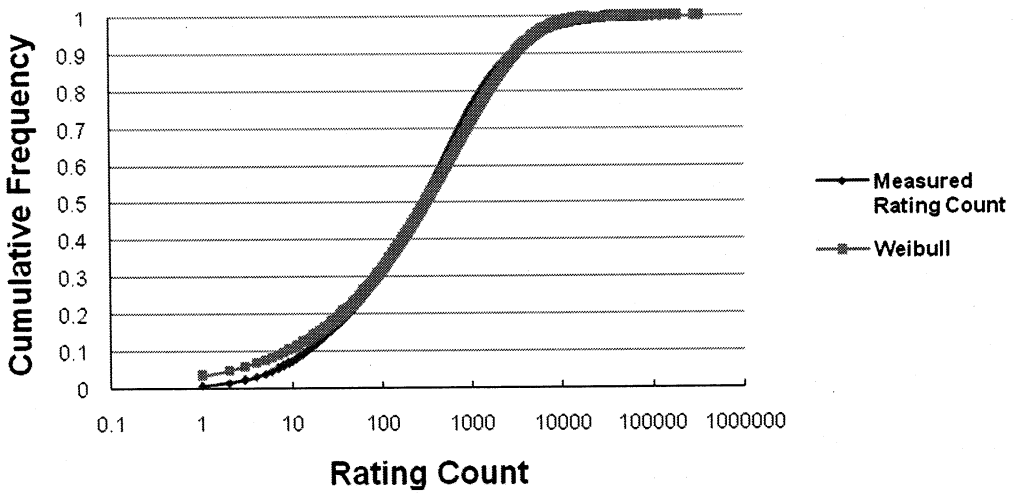

Figure 5.6: CDF of Regular Files Rating Count

5.3.5 Popularity Analysis of Videos

One of the significant characteristic in our data sets is the viewing incidence of a video, as it indicates the access patterns and popularity of the video. Although, this property is a dynamic feature, we considered it to be static in our data set during the data collection period. From the obtained data, distinct videos are extracted based on the largest view count observed for each video. The distribution of popularity has an important effect on the choice of cache behavior, since popular files will tend to stay in caches. Zipf analysis in access pattern data is a method to evaluate cache efficiency. The observance of more Zipf-like behavior is an indication of better cache performance.

Figure 5.7 shows the view count of regular files as a function of the video rank. The plot does not indicate a Zipf distribution as it has a long tail on a log-log graph. In addition, regression analysis with the exponent ß = 0.63 and the value of 0.71 for

goodness of fit R^2 does not represent a Zipf distribution in the popularity graph of regular videos. It follows the Weibull distribution.

This result is compatible with the findings of Cheng [13] representing video accesses on a media server which also indicate popularity does not follow Zipf's law with ß = 0.54. The plot shows the head of the graph is linear but the tail sharply drops off. It suggests less popularity of regular contents at the tail which is in contrast with Zipf's law stating the curve is skewed linearly from the beginning to the end.

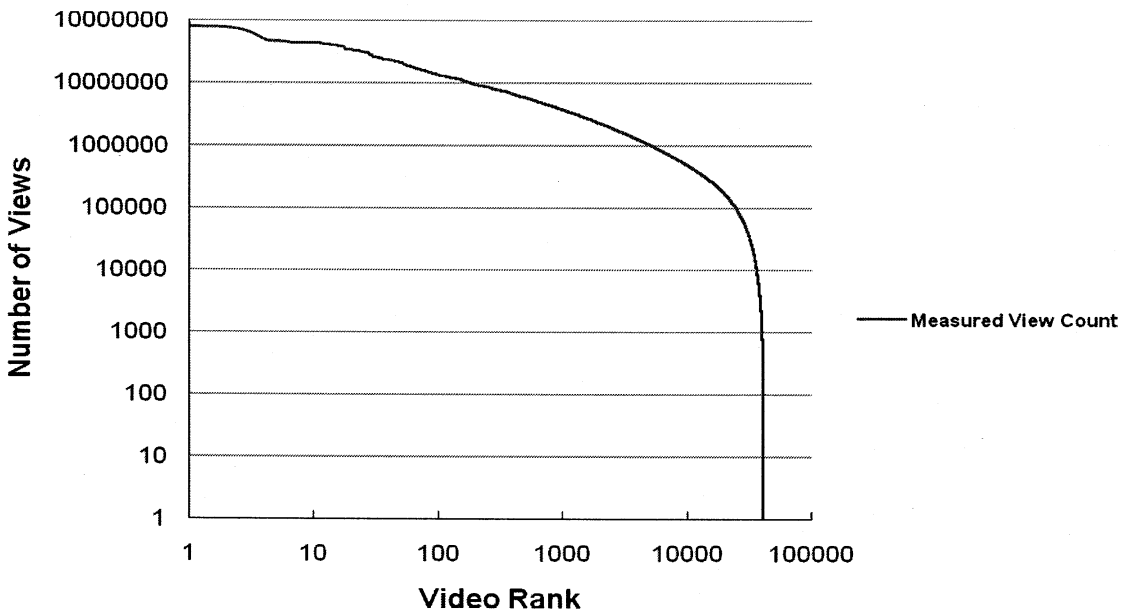

Figure 5.7: Ranked Video View Count of Regular Files in Log-Log Scale

According to our results, the regular files view count graph can be modeled by Weibull distribution better than other models which is shown in Figure 5.8. Considering Figure 5.7 without observing Zipf-like behavior in regular videos popularity pattern, using proxy caching for the regular videos may be an inefficient solution for YouTube.

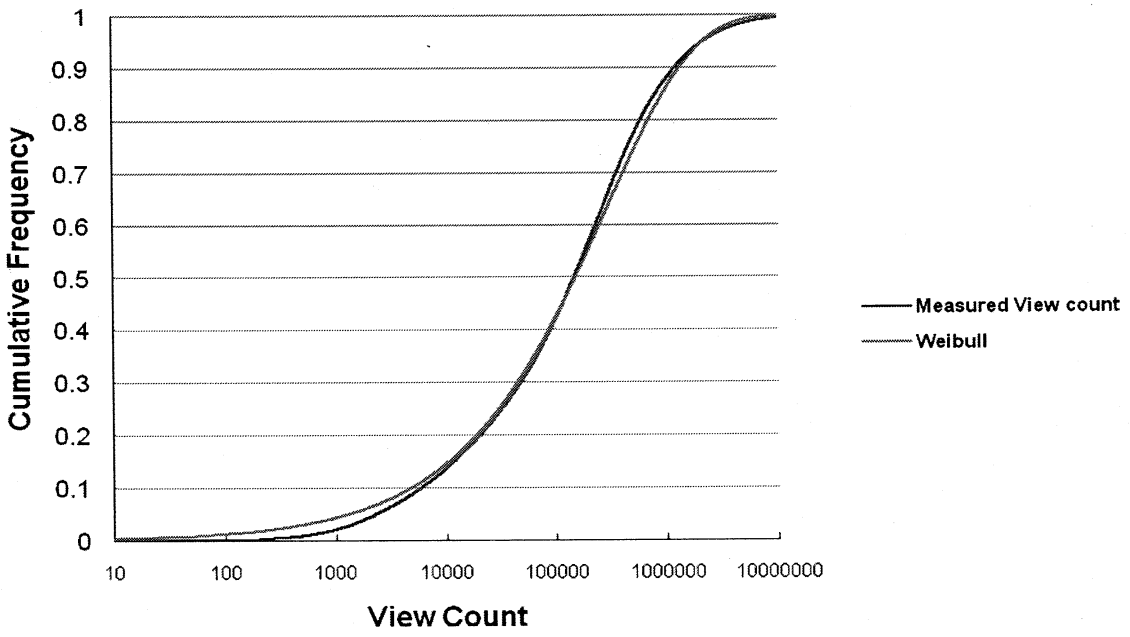

Figure 5.8: CDF of Regular Files View Count

5.4 Correlation between Meta-data Attributes

Table 5.2 lists the possible relations of different properties of YouTube popular video files.

Table 5.2: Regular and Popular Correlation Statistics

	Regular Videos	
Characteristics	Correlation Coefficient	Relation
Video duration & Average Rating	0.002	Weak
View Count & Average Rating	0.048	Weak
Video duration & View Count	0.018	Weak

Based on Table 5.2, weak correlation is between all tabulated attributes.

5.5 Implications on Caching

Not observing a Zipf distribution in view count property of regular files reflects proxy caching may not be an ideal solution for regular files but we believe that YouTube can benefit from other methods to improve serving regular files that is explained in the following paragraphs.

Exploiting the concept of related videos can improve the cache efficiency at the client site for regular videos. In other words, for a group of videos related to each other, it is very possible a viewer watches next video from the related list after viewing the previous one. Therefore, once a video is played back by a user, the related videos of the selected video can be prefetched and then cached in the client's site. Cheng *et al.* [13] have also suggested that this fact can be used to design new peer-to-peer methods for short video sharing.

Full-object caching of related videos is somehow wasting the storage and bandwidth. Since for each video the number of related videos is significant and user may watch none of the related videos. Considering this fact, the optimum number and bytes of the related videos for prefetching in the client's site should be determined. Another vital issue is the cache space limits and cache replacement strategy that can be used to save the cache space. The important impact of pre-fetching is on the startup delay. Prefetching results in eliminating or reducing the startup delay in the client's site and enhancing the playback quality.

5.6 Chapter Summary

We have presented an analysis of the characteristics of one of the most popular short video sharing site, YouTube. Through investigating an extensive amount of data, we have represented YouTube popularity distribution and access pattern. These characteristics introduce new opportunities to improve the performance of short video sharing web sites by proposing a novel caching model such that the related videos of watching video are prefetched in the user's machine.

Based on our findings in this chapter and chapter 4, caching the most popular videos in the proxy along with prefetching the related videos in the client's site can

reduce the client's access time and start up delay in watching video. To estimate and simulate how combining of prefetching and caching method performs when the traffic varies, a workload simulator as a tool is essential. The simulator generates synthetic workload with the same characteristics as real YouTube popular and regular videos. In this chapter and chapter 4, essential elements required for development of a realistic workload generator were presented.

Chapter 6

Multimedia Workload Simulator

In this chapter, the basic elements required to generate a multimedia workload simulator are demonstrated. First, we describe the database architecture used in workload simulator and then the structure of our developed software will be presented. We demonstrate how the developed software applies provided multimedia workload characteristics in this study to create a realistic multimedia workload generator.

6.1 Methodology of Generating Workload Simulator

6.1.1 Workload Characteristics

In order to design a workload generator, first, it is necessary to find out a set of probability distributions to model workload characteristics of video files on server such as file size, video length, and view count which shows popularity of individual files on the server. In previous chapters, we analyzed workload characteristics and derived the important features of related distributions that are required to generate the elements of the workload simulator. The workload generator can be divided into two main parts naming server workload generator and client session generator. The server part generates a collection of files corresponding to the distributional models in order to create server database. The purpose of client session generator is to simulate a user when accessing video files.

6.1.2 Server Workload Generator

Server Workload Generator simulates the files available on servers of popular Web pages and YouTube server.

To generate server files of popular Web pages, we consider a tree data structure as shown in Figure 6.1. The first level is considered as popular Web pages and the second level of nodes are assumed as embedded objects of each popular Web page.

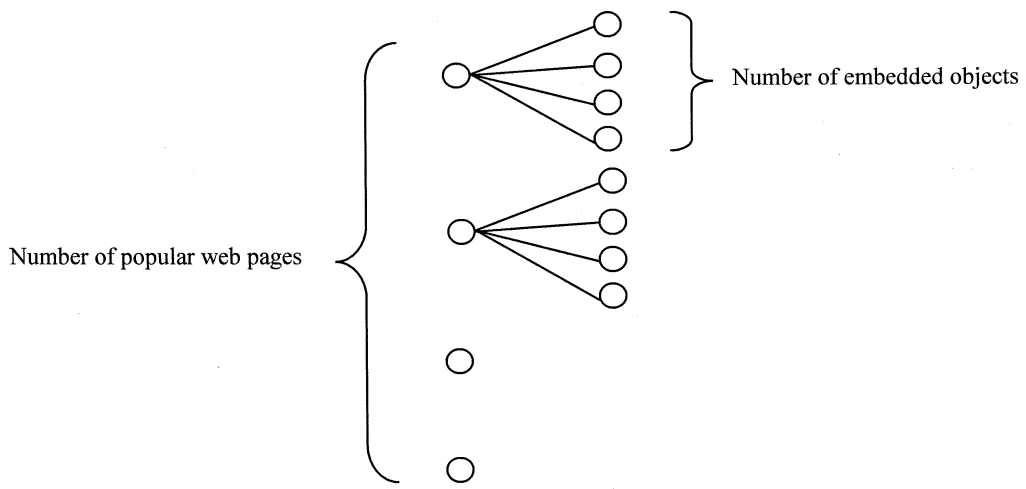

Figure 6.1: Popular Web Page Tree Data Structure

Each node in the popular Web page database is considered as a record with a unique field named "Record ID". Each record has four fields correspond in the following specifications:

1. Record ID (must be unique)
2. Description (which can be used as Object ID)
3. File size
4. Type (which can be as text, image, or multimedia)

Record ID consists of number of digits specifying the location of each node in the tree. Table 6.1 is an example to illustrate how Record ID is structured.

Table 6.1: Popular Web Page Record ID Structure

Object ID	Record ID	Location
A	51	The fifth record of the first level
B	51-3	The third embedded object of A

File size field is generated according to Weibull distribution. Type field has a value of text, image, or multimedia categories based on the weight entered by user in "Setting" form.

Popular Web page database is designed based on algorithm 1.

Algorithm 1 GeneratePage(*n*)

---

read *n* number of popular Web pages
for each popular Web page **do**
    create a node in the first level
    k ← extract a number from distribution of "number of embedded objects"
    for each *k* embedded objects **do**
        create a node at the second level
    end for
end for

After generating popular Web pages in first level, their embedded objects are created at the second level. The number of popular Web pages has already been set in "Setting" form shown in Figure 6.4. The number of embedded objects for each popular Web page is determined based on the corresponding distribution.

To construct server files of YouTube, we are using tree data structure shown in Figure 6.2.

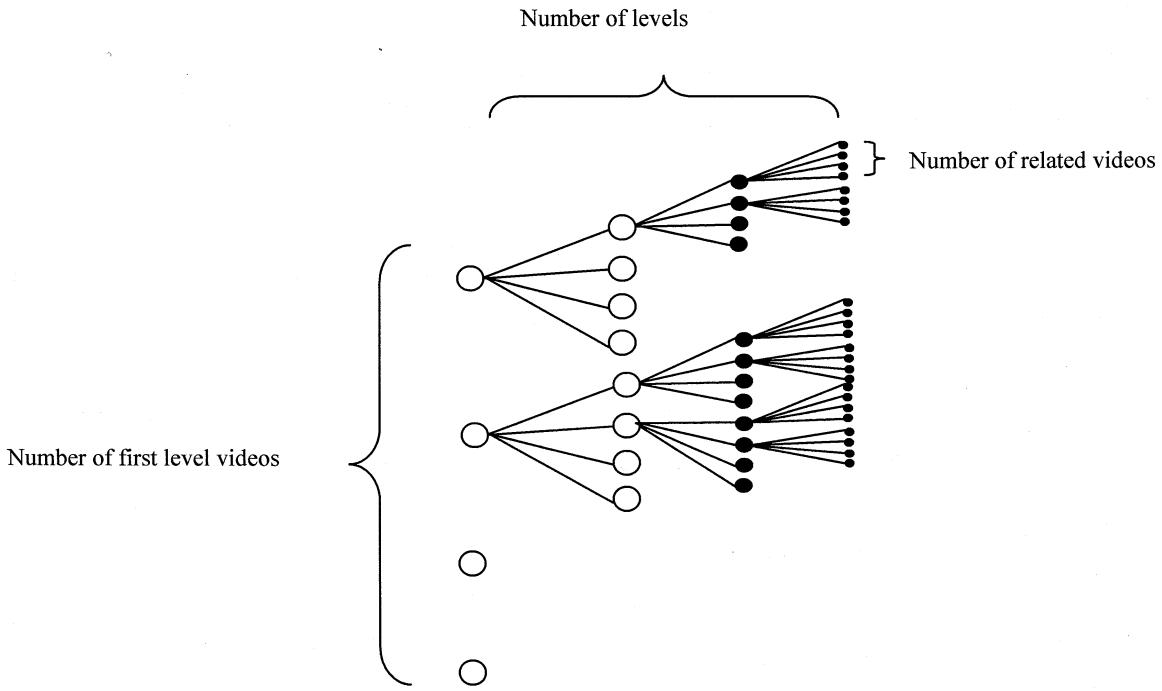

Figure 6.2: YouTube Tree Data Structure

(Each node is a record)

In order to calculate the total number of the created video files in database, the following terms and formulas are applied:

l= number of levels

r= maximum number of related nodes for each node

n= number of nodes in the first level

$$\textit{The total no of data} = n * \sum_{i=0}^{l} r^i \qquad (6.1)$$

For example, if n =9, l =5 and r =6, the total number of YouTube videos in database based on equation 6.1 will be

$$9 * (6^0 + 6^1 + 6^2 + 6^3 + 6^4 + 6^5) = 83{,}979$$

Each node in the tree structure database is considered as a record with a unique field named "Record ID". Each record has seven fields correspond in the following video specifications:

1. Record ID (must be unique)
2. Description (which can be used as YouTube Video ID)
3. File size
4. View Count
5. Duration
6. Average Rating
7. Rating Count

Record ID consists of number of digits specifying the location of each node in the tree. Table 6.2 is an example to illustrate how Record ID is structured.

Table 6.2: YouTube Record ID Structure

Video ID	Record ID	Location
A	5	The fifth record of the first level
B	53	The third related node of A's related list
C	532	The second related node of B's related list
D	5321	The first related node of C's related list

Other fields are generated individually based on a distribution. Table 6.3 summarizes Table 5.1 and Table 4.3 which contain distributions used for each video characteristic.

Table 6.3: YouTube Video Characteristics Distributions

Characteristic	Regular	Popular
File size	G	W
View count	W	LL
Duration	LL	W
Average rating	W	W
Rating count	W	W

(W = Weibull, G = Gamma, LL = Log-Logistic distribution)

YouTube database is designed based on algorithm 2.

Algorithm 2 GenerateVideo(*l, r, n*)

---

read *l* as number of levels

read *r* as maximum number of related nodes

read *n* as number of nodes in the first level

for each *n* nodes in the first level **do**

Create_related_nodes(*levelNumber* ← *1*)

end for

Create_related_nodes(*levelNumber*)

if *levelNumber* ≤ *1* **then**

k ←Generate a random number less than *r*

for each *k* related nodes **do**

Create_related_nodes(*levelNumber* + 1)

end for

end if

After generating the first level videos, their related videos are created. The number of the related videos for each video is determined based on a random number

between 1 and the maximum number of the related videos which has already been set in "Setting" form shown in Figure 6.5. This process is repeated for all videos in the related videos list.

As each distribution has its settings parameters, software provides a simple user interface to facilitate setting distributions and other characteristic parameters. Server workload simulator contains the form "Distributions" to represent distributions parameters for video characteristics and "Settings" form which exhibits parameters used to create server and also client session workload.

6.1.3 Distribution Form

Figure 6.3 is a snapshot of characteristic parameters settings form in order to set parameters for popular Web page features or YouTube video characteristics. It is also possible to save generated data points into an Excel file (*.csv) for characteristics of popular Web pages like sizes of Web objects, number of embedded objects for each popular Web page, and sizes of embedded objects individually. The Excel file also can save generated data point of YouTube characteristics such as file size, view count, duration, average rating, and rating count. The distribution type is also shown for each characteristic to help the user adjust the parameters. It is possible to switch between "YouTube Regular", "YouTube Popular", and "popular Web page" options and the related parameters for each option will be automatically restored accordingly. By clicking on "OK" button, parameters will be saved into a file.

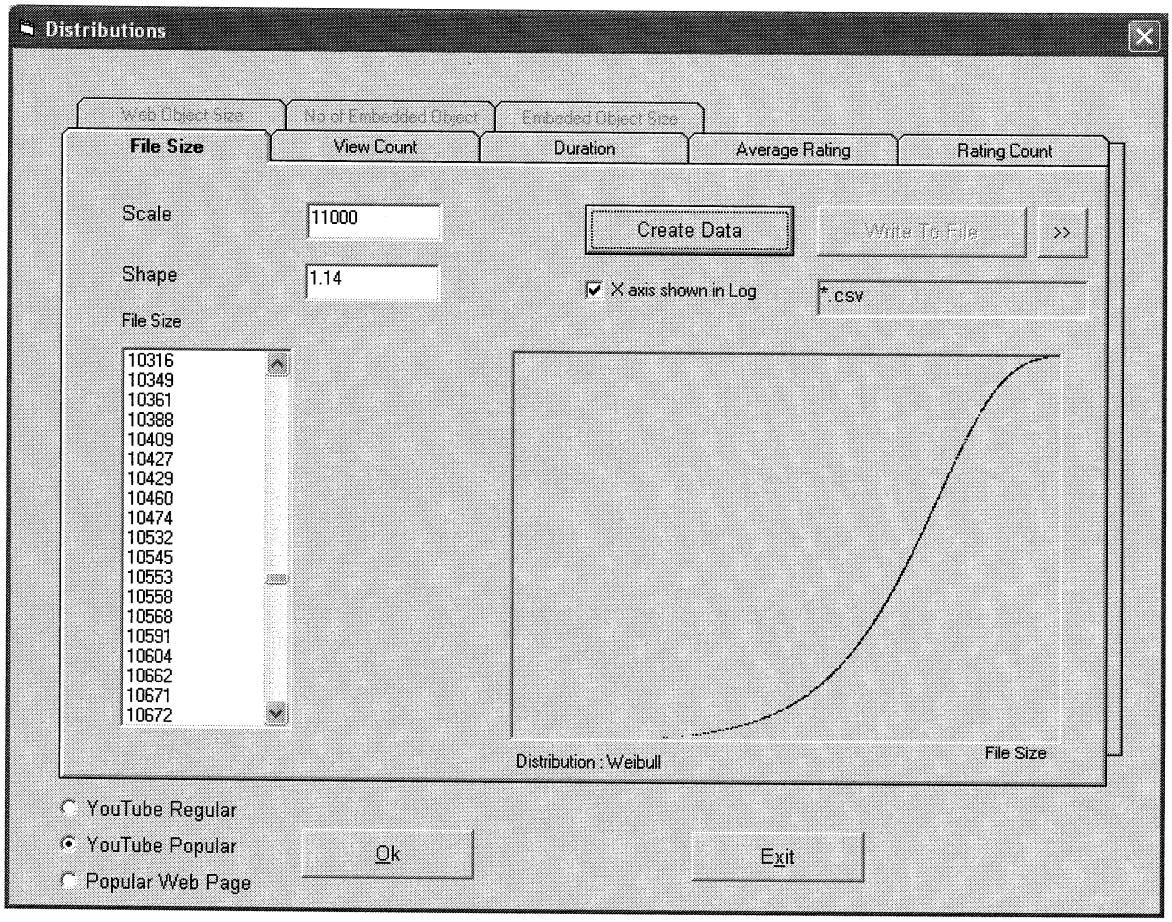

Figure 6.3: Distribution Form

6.1.4 Settings Form

Figure 6.4 and Figure 6.5 are snapshots of the form Settings to determine the "number of popular Web pages" for popular Web page tab and "number of levels", "number of related videos", and "number of videos in first level" parameters in YouTube tab. These parameters are the main factors to generate the database and are saved into files.

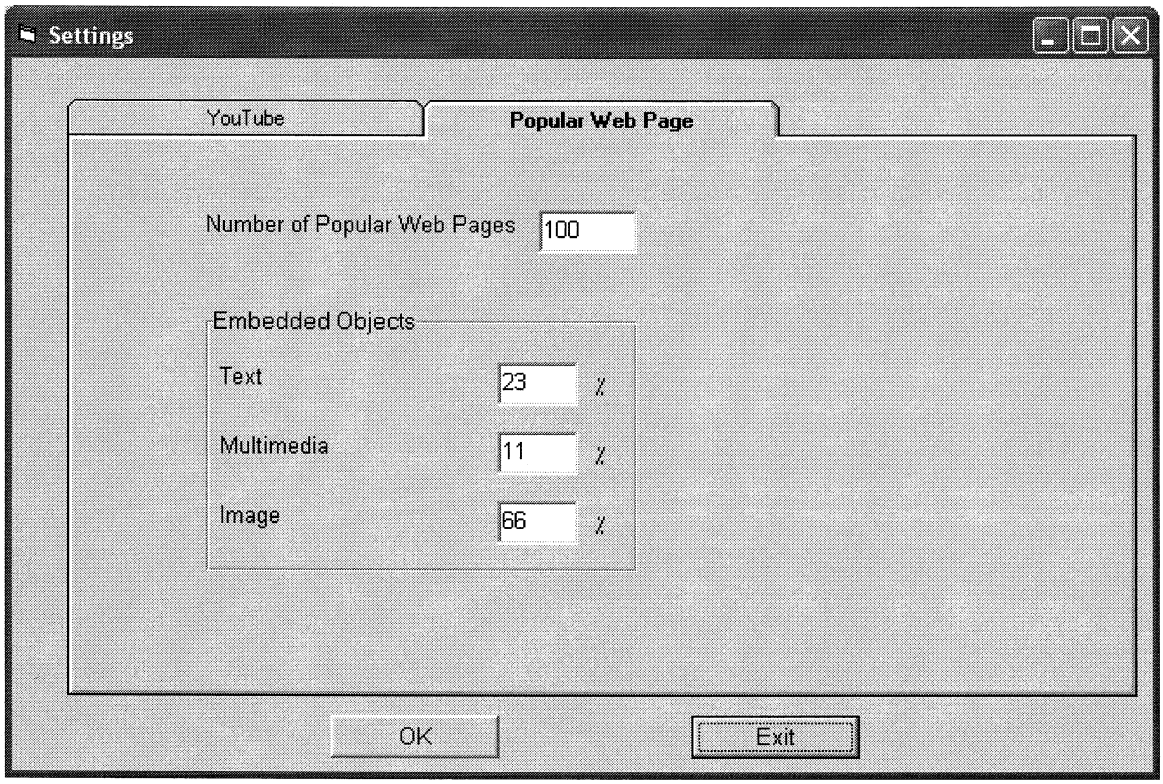

Figure 6.4: Popular Web Page Setting Form

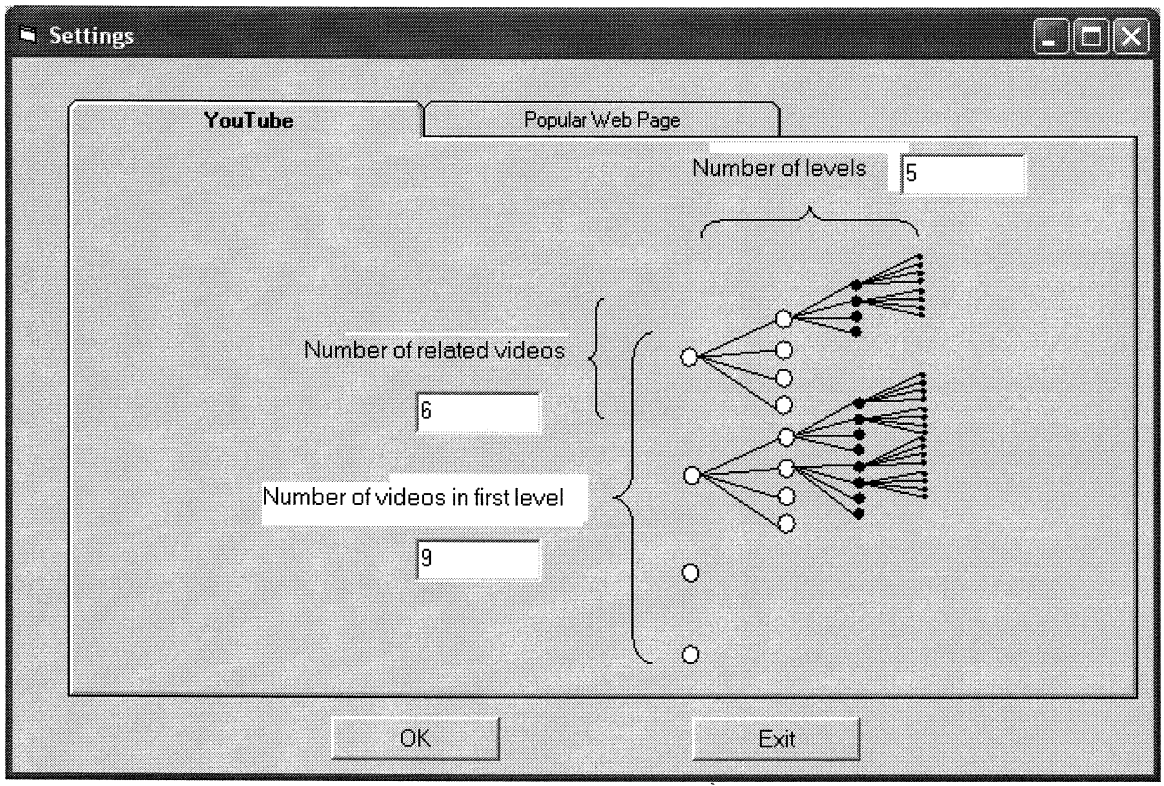

Figure 6.5: YouTube Setting Form

6.1.5 Client Session Generator

Client session generator simulates user accessing the server by selecting data from database and generating an output stream.

To simulate popular Web page client session, selecting data from the first level is performed randomly. Chosen data is considered as a popular Web page which has a number of embedded objects. We assume that if any popular Web page is selected, all of its embedded objects are automatically chosen. Figure 6.6 demonstrates an output stream of a selected popular Web page and a list of its embedded objects.

In simulating YouTube client session, choosing data is started from the first level and continued until the number of levels is reached. We suppose that the first video is chosen randomly. The first selected video has a list of related videos. We suppose a user only chooses the next video from the related videos. Therefore, the client session simulator chooses a video from the related nodes for the next levels based on the probability that is proportional to the view count field. Therefore, videos with the greater value of view count are more likely to be selected. To simulate choosing next videos, we define following terms and formulas:

r = number of related nodes for each node

VC = View Count of each node

s = total view counts of related videos for each node

$$s = \sum_{i=1}^{r} VC_i \tag{6.2}$$

To select the next video out of the r related nodes, a random number m which would be between 1 and s (equation 6.2), the total number of the related videos view counts, is created. Then the application will find the node j ($1 \leq j \leq r$) with the total view count ($\sum_{i=1}^{j} VC_i$) greater than m if the total view count of the previous node (j-1), $\sum_{i=1}^{j-1} VC_i$, is less than m. It means when m has a value between $\sum_{i=1}^{j-1} VC_i$ and $\sum_{i=1}^{j} VC_i$, the *jth* related video will be the next selected video.

The expression 6.3 explains how related video is chosen and Table 6.4 is an example to illustrate the expression. In this example, a random number m is selected between 1 and 1310 (s=1310). The probability of being m in the range of View Count of each Video ID is shown in the last column. It is clear that the Video ID B has the most probability to be selected because of the biggest view count.

$$\begin{cases} \sum_{i=1}^{j-1} VC_i < m \leq \sum_{i=1}^{j} VC_i \quad , \quad 1 < j \leq r, 0 < m \leq s \\ \\ 1 \leq m \leq VC_j \quad , \quad j = 1 \end{cases} \tag{6.3}$$

Table 6.4: Example of Expression 6.3

Index(j)	Video ID	View Count (VC)	View Count Range $\sum_{i=1}^{j-1} VC_i \sim \sum_{i=1}^{j} VC_i$	Probability of Selected Video ID
1	A	95	1 - 95	0.072
2	B	850	95 - 945	0.649
3	C	12	945 - 957	0.009
4	D	38	957 - 995	0.029
5	E	315	995 - 1310	0.240

Client session simulator continues selecting videos based on the above strategy until the maximum number of levels is reached. Client simulator shows user selection videos in "Client session" form shown in Figure 6.7.

6.1.6 Client Session Form

Figure 6.6 and Figure 6.7 exhibit user options such as "YouTube Regular", "YouTube Popular", and "Popular Web Page". "Simulate Session" button generates output stream into the output window.

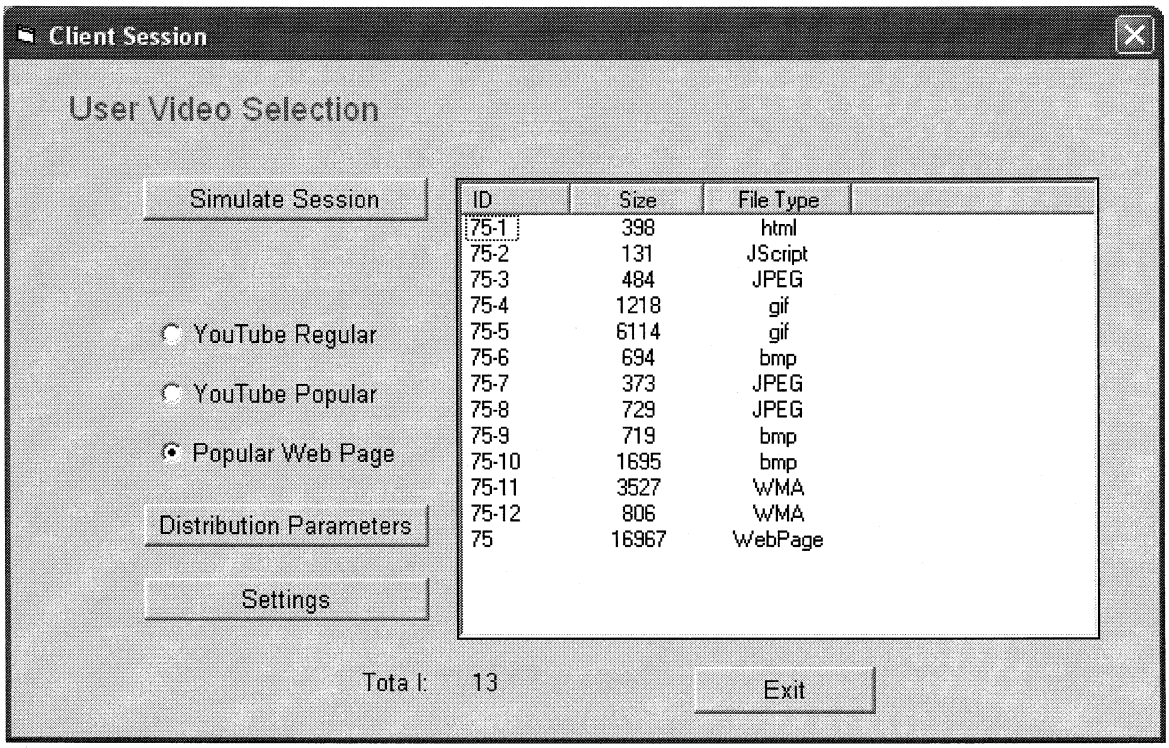

Figure 6.6: Popular Web Page Client Session Form

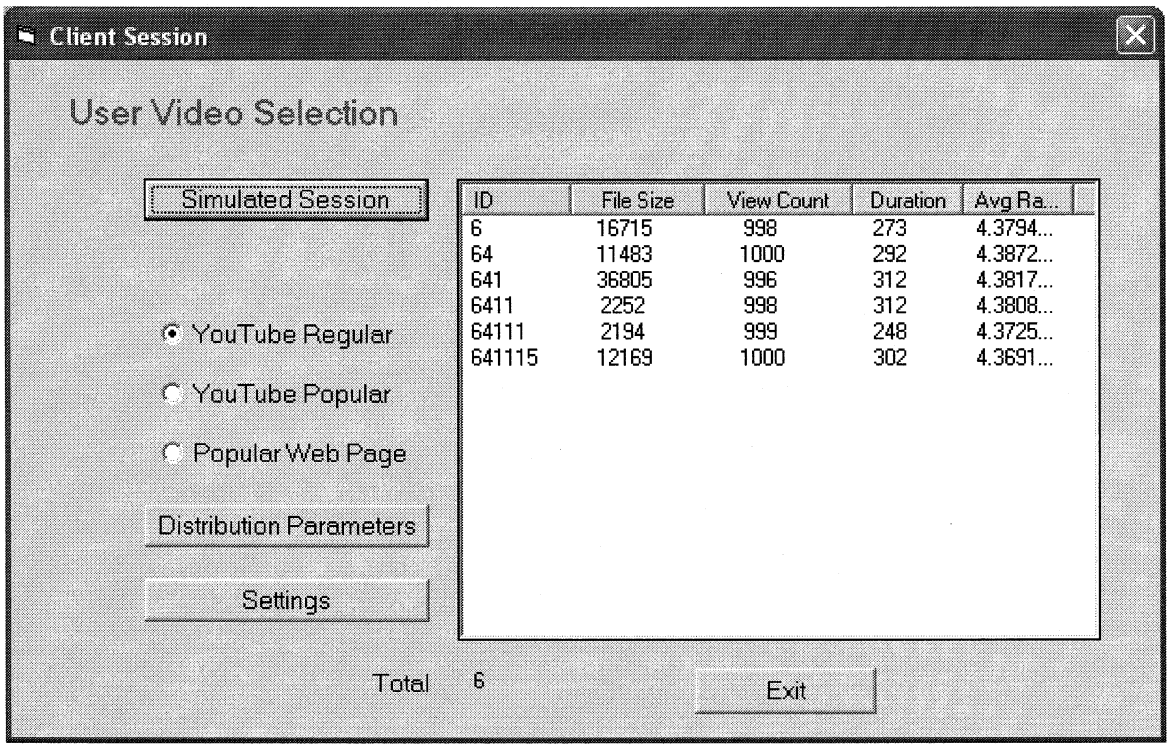

Figure 6.7 : YouTube Client Session Form

6.2 System Requirements

The experimental environment consists of one PC configured with 1GHz Pentium Pro CPUs and 500MB of RAM and at least 50 MB free space on hard disk. Windows XP and later is good to run this application.

6.3 Chapter Summary

We developed a multimedia workload generator into two parts; server workload generator and client session generator. Details of software data structure, data generation method, and data selection strategy were presented. A view of utilized forms and their components were provided.

Chapter 7

Conclusions

In this chapter, we summarize the thesis and its results. The content of the thesis is restated in section 7.1. Conclusions are discussed in section 7.2.

7.1 Thesis Summary

The thesis presented a characterization study on multimedia files embedded in the Web pages as well an analysis of YouTube usage. The study characterized top 500 popular Web sites for three years, 2004, 2005, and 2006. In addition, the research focused on YouTube top 100 most viewed videos of the day and week during 54 consecutive days in 2007. This thesis also considered YouTube regular data set comprising of YouTube top ranked list and their related videos in 2008. The results of the thesis used to build Web multimedia workload simulator to be able to evaluate efficiency of multimedia delivery strategies.

Chapter 2 described an overview on relevant background material. An introduction of Web, YouTube, multimedia concepts and statistical definitions were presented. Also a brief history of related studies relevant to the thesis was discussed. Related studies consist of researches on multimedia workload before and after advent of YouTube. Detailed of the workload characterization study was presented in chapters 3, 4, and 5 and methodology of building a workload simulator was explained in chapter 6. All chapters' findings are summarized in the following section.

7.2 Thesis Results

In chapters 3, 4, and 5, we analyzed significant multimedia characteristics on the Web for generating a Web multimedia workload. A characterization study on multimedia

files embedded in the Web pages was presented in chapter 3. The analysis was performed on top 500 popular Web sites. The results of analysis of embedded multimedia object sizes proposing caching multimedia objects in proxy cache is an effective way to reduce down load time.

A study of YouTube usage on popular and regular video files was presented in chapters 4 and 5. Through this study, an understanding of YouTube distributional models was obtained. The findings aimed on the improvement of caching strategies. As YouTube popular videos have low coefficient of variation in terms of file sizes with the mean size of 10 MB and observing Zipf-like behavior, we concluded caching of popular videos in proxy can improve the performance and scalability of YouTube. In chapter 5, we also considered lists of YouTube related videos for each video clip and analyzed their statistical behavior. Detailed measurement study on the characteristics of related videos suggests caching of related videos can be effective at the client site. Prefetching and caching the related videos of the selected video in the client's site reduce startup delay and improve the quality of playback.

The set of YouTube workload characteristics provided in the mentioned chapters enabled us to design a multimedia workload generator in chapter 6. In this chapter, based on our derived models, software for generating a multimedia workload simulator implemented. The multimedia workload generator was developed as a means to simulate server files and clients sessions when accessing a multimedia server.

7.3 Future Work

Applying the findings obtained in this thesis can lead to more efficient and better playback quality of videos over the Internet. This work performed a measurement study on YouTube workload characteristics except for the parameters such as bit rate to determine streaming rate and network utilization. Estimating optimum bit rate requires lower level network measurement study that can be considered as the future work. Another extended work is analysis of social network made by YouTube related videos to provide opportunities for improving the video delivery system. Social structures in YouTube can be employed by the new delivery schemes such as caching, peer-to-peer

(P2P), and prefetching to improve the playback quality and increase the scalability of the infrastructure of short video sharing Web 2.0 sites such as YouTube.

Thus, using the developed workload generator in this research is necessary to evaluate caching, P2P, and prefetching methods to reduce streaming traffic and support a large number of clients. Performing caching simulations and P2P experiments to reduce the workload of server as well as bandwidth requirements will be possible by using workload generator designed in this thesis. The next direction of this research is to build a framework with more simulator tools to be able to determine the effectiveness of proposed caching and P2P systems for YouTube.

Another future work can be characterization of the other popular Web 2.0 sites such as Facebook, MySpace, and Flicker which contains features similar to YouTube.

References

[1] E. Chang, M. Davis, P. Schmitz, and S. Boll. Panel Discussion: "Web 2.0 and Multimedia: Challenge, Hype, Synergy," *In Proceedings of the 14th annual ACM international conference on Multimedia; Panel Session,* Santa Barbara, USA, Oct. 2006.

[2] Blogspot. http://www.blogspot.com.

[3] Wordpress. http://www.wordpress.com.

[4] Flickr. http://www.flickr.com.

[5] YouTube. http://www.youtube.com.

[6] M. Li, M. Claypool, and R. Kinicki, "MediaPlayer versus RealPlayer - A Comparison of Network Turbulence," *In Proceedings of the 2nd ACM SIGCOMM Workshop on Internet measurement Workshop*, pp. 131-136, Marseille, France, Nov 2002.

[7] S. McCreary and K. Claffy, "Trends in Wide Area IP Traffic Patterns," *In Proceedings of 13th ITC Specialist Seminar on Internet Traffic Measurement and Modeling*, Monterey, CA, USA, September 2000.

[8] A. Abhari, M. Soraya, "Workload Characterization for the Multimedia Files Embedded in the Popular Web Pages," *In Proceedings of International Conference on Internet and Multimedia Systems and Applications (IMSA 2007),* Honolulu, USA, August 2007.

[9] M. Soraya, M Zamani,A. Abhari, "Modeling of Multimedia Files on the Web 2.0," *In Proceedings of the 21th annual IEEE Canadian Conference on Electrical and Computer Engineering 2008(CCECE'08): Symposium on Computer Systems and Applications*, Niagara Falls, Canada, May 2008.

[10] T. O'Reilly, "What Is Web 2.0 - Design Patterns and Business Models for the Next Generation of Software,"" http://www.oreillynet.com/pub/a/oreilly/ tim/news/2005/09/30/what-is-web-20.html, September 2005.

[11] M. Arlitt and C. Williamson, "Internet Web Servers: Workload Characterization and Performance Implications," *IEEE/ACM Transactions on Networking (TON)*, Volume 5, Number 5, pp. 631 – 645, October 1997.

[12] P. Gill, M. Arlitt, Z. Li, and A. Mahanti, "YouTube Traffic Characterization: A View From the Edge," *In Proceedings of the 7th ACM SIGCOMM conference on Internet measurement*, pp. 15-28, San Diego, USA, October 2007.

[13] X. Cheng, C. Dale, and J. Liu, "Understanding the Characteristics of Internet Short Video Sharing: YouTube as a Case Study," Technical Report arXiv: 0707.3670v1 [cs.NI], Cornell University, arXiv e-prints, July 2007.

[14] M. Zink, K. Suh, and J. Kurose, "Watch Global, Cache Local: YouTube Network Traffic at a Campus Network - Measurement and Implications," *In Proceedings of ACM/SPIE MMCN '08 conference*, Volume 6818, pp. 5-13 San Jose, USA, January 2008.

[15] H. Sun, A. Vetro, J. Xin, "An overview of scalable video streaming," Mitusubishi Research Articles, Volume 7 , Number 2, pp. 159 – 172, Cambridge, Massachusetts, February 2007.

[16] G. K. Wallace, "The JPEG Still Picture Compression Standard," *Communications of the ACM*, Volume 34, Number 4, pp. 30-44, April, 1991.

[17] J.G. Apostolopoulos, W. Tan and S.J. Wee, "Video Streaming: Concepts, Algorithms, and Systems," Mobile and Media Systems Laboratory HP Laboratories Palo Alto, HPL-2002-260, Sept. 2002.

[18] Z. Miao and A. Ortega, "Scalable Proxy Caching of Video under Storage Constraints," *IEEE Journal on Selected Areas in Communications*, Volume 20, Number 7, pp. 1315-1327, September 2002.

[19] A. C. Tamhane and D. D. Dunlop, "Statistics and Data Analysis," Prentice-Hall, Inc., NJ, USA, 2000.

[20] M. Crovella and B. Krishnamurthy, "Internet Measurement: Infrastructure, Traffic and Applications," John Wiley and Sons Ltd., 2006.

[21] E. Veloso, V. Almeida, W. Meira, A. Bestavros, S. Jin, "A Hierarchical Characterization of a Live Streaming Media Workload," *IEEE/ACM Transactions on Networking (TON)*, Volume 14, Number 1, pp. 133 – 146, February 2006.

[22] J. M. Almeida, J. Krueger, D. L. Eager, and M. K. Vernon, "Analysis of Educational Media Server Workloads," *In Proceedings of 11th Int'l Workshop on Network and Operating Systems Support for Digital Audio and Video (NOSSDAV),* pp. 21-30, Port Jefferson, New York, USA, June 2001.

[23] M. Chesire, A. Wolman, G. M. Voelkar, and H. M. Levy, "Measurement and Analysis of a Streaming Media Workload," *In Proceedings of the 3rd conference on USENIX Symposium on Internet Technologies and Systems (USITS),* pp. 1-1, San Francisco, USA, March 2001.

[24] C. Costa, I. Cunha, A.Borges, C. Ramos, M. Rocha, J. M. Almeida, and B. Riberio-neto, "Analyzing Client Interactivity in Streaming Media," *In proceedings of 13th International conference on World Wide Web,* pp. 534-543, New York, USA, May 2004.

[25] M. Li, M. Claypool, R. Kinicki, and J. Nichols, "Characteristics of Streaming Media Stored on the Web," *ACM Transactions on Internet Technology (TOIT)*, pp. 601–626, Nov. 2005.

[26] N. Basher, A. Mahanti, "A Simulation Study of Proxy Caching Algorithms and Strategies for Interactive Streaming Media," *In proceedings of International Symposium on Performance Evaluation of Computer and Telecommunication Systems (SPECTS) 2006*, pp. 605-612, Calgary, Alberta, Canada, July 31- August 2, 2006.

[27] C. Huang, J. Li, and K. W. Ross, "Can Internet Video-on-Demand be Profitable?," *ACM SIGCOMM Computer Communication Review*, Volume 37 , Number 4, pp.133-144,October 2007.

[28] H. Yu, D. Zheng, B. Y. Zhao, and W. Zheng, "Understanding User Behavior in Large-Scale Video-on-demand Systems," *ACM SIGOPS Operating Systems Review*, Volume 40, Number 4, October 2006.

[29] M. Halvey and M. Keane, "Exploring Social Dynamics in Online Media Sharing," *In Proceedings of the 16th international conference on World Wide Web,* Banff, Canada, May 2007.

[30] A.Williams, M. Arlitt, C. Williamson, and K. Barker, "Web Workload Characterization: Ten Years Later," *Web Content Delivery*, Volume 2, pp.3-21, Springer US, 2005.

[31] Cha, H. Kwak, P. Rodriguez, Y. Ahn, and S. Moon, "I Tube, You Tube, Everybody Tubes: Analyzing the World's Largest User Generated Content Video System," *In Proceedings of the 7th ACM SIGCOMM conference on Internet Measurement,* pp.1-14, San Diego, USA, October 2007.

[32] A. Mislove, M. Marcon, K. Gummadi, P. Dreschel, B. Bhattacharjee, "Measurement and Analysis of Online Social Networks," *In Proceedings of the 7th ACM SIGCOMM conference on Internet measurement Internet Measurement,* pp. 29 – 42, San Diego, California, USA, 2007.

[33] LiveJournal. http://www.livejournal.com.

[34] Orkut. http://www.orkut.com/Home.aspx.

[35] Daum UCC. http://ucc.daum.net.

[36] J. Paolillo, "Structure and Network in the YouTube Core," *In Proceeding of the 41st annual Hawaii International Conference on System Sciences (HICSS'08)*, pp.156, 2008.

[37] API Documentation (YouTube). http://youtube.com/dev docs.

[38] YouTube: Video Format (from Wikipedia). [Online]. Available: http://en.wikipedia.org/wiki/Youtube#Video format

[39] http://www.youtube.com/api2_rest?method=youtube.videos.list_popular&dev_id=dev_id&time_range

[40] http://cache.googlevideo.com/get_video?video_id&origin=1

[41] USA Today. YouTube Serves up 100 million Videos a Day Online, July 2006. http://www. usatoday.com/tech/news/2006-07-16-youtube-views_x.htm? July 2006

[42] http://www.google.com/support/youtube/bin/answer.py?answer=55743&topic=10527

[43] A. M. Law, W. D. Kenton, "Simulation Modeling and Analysis," Third Edition, pp.292-402, McGraw-Hill, Boston, 2000.

[44] A. M. Law & Associates, "*ExpertFit© Version 6 User's Guide*," February 2004. www.averill-law.com

[45] G. Zapf, "Human Behavior and the Principle of Least Effort," Addison-Wesley (Reading MA), 1949.

[46] P. Barford and M. Crovella, "Generating Representative Web Workloads for Network and server performance Evaluation," *In Proceedings Of ACM SIGMETRICS '98 Conference,* pp.151-160, Madison, Wisconsine, June 1998.

[47] R. Jain, "The Art of Computer Systems Performance Analysis: Techniques for Experimental Design, Measurement, Simulation, and Modeling," Wiley & Sons, Inc., New York, USA, April 1991.

[48] M. Arlitt and C. Williamson, "Web Server Workload Characterization: The Search for Invariants," *In Proceedings Of the ACM SIGMETRICS '96 Conference,* pp.126-137, Philadelphia, USA, April 1996.

[49] M. E. Crovella and M. S. Taqqu, "Estimating the Heavy Tail Index From Scaling Properties," Methodology and Computing in Applied Probability, Volume 1, Number 1, 1999.

[50] A. Abhari, "Web object based policies for managing proxy caches," PhD thesis, Carleton, Ottawa, ON, 2003.

[51] A. Abhari, S. P. Dandamudi and S. Majumdar, "Structural Characterization of popular Web Documents," *International Journal of Computers and Their Applications*, Volume 9, pp 15-24, March 2002.

[52] http://www.alexa.com

[53] http://archive.org

[54] M. Soraya, A. Serbinski and A. Abhari, "A Prefetching Server for Reducing Startup Time of Embedded Multimedia," *IEEE International Workshop on Advanced in Multimedia (AIM-07) held in conjunction with IEEE International Symposium on Multimedia 2007 (ISM2007),* Taichung, Taiwan, December 2007.

[55] L. Gomes, "Will All of Us Get Our 15 Minutes On a YouTube Video?," Wall Street Journal, Aug. 30, 2006.

[56] S. Sen, J. Rexford, D. Towsley, "Proxy Prefix Caching for Multimedia Streams." *In Proceeding of the 18th IEEE Conference on Computer Communications (INFOCOM'99)*, Volume 3, pp. 1310-1319, New York, NY, USA, March 1999.

[57] http://en.wikipedia.org/wiki/Coefficient_of_variation

[58] http://youtube.com/blog

[59] The Wall Street Journal (from Wikipedia). http://en.wikipedia.org/wiki/The_Wall_Street_Journal

[60] Lei Shi, Zhimin Gu, Lin Wei, Yun Shi, "An Applicative Study of Zipf's Law on Web Cache," *International Journal of Information Technology*, Volume 12, Number 4, 2006.