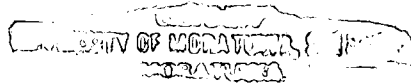


ON-THE-FLY INTER-PROXY DATA COMPRESSION FOR WEB ACCESS

THESIS PRESENTED BY
P.S.K. GURUSINGHE

SUPERVISED BY
P.G.V. DIAS



This thesis was submitted to the Department of Computer Science and Engineering of
the University of Moratuwa – Sri Lanka

In partial fulfillment of the requirements for the
Degree of Master of Science

004 '05
004 (043)



Department of Computer Science and Engineering
University of Moratuwa

University of Moratuwa

Sri Lanka

December 2005



85978

85978

85978

The work presented in this dissertation has not been submitted
for the fulfillment of any other degree



University of Moratuwa, Sri Lanka.
Electronic Theses & Dissertations
www.lib.mrt.ac.lk

Handwritten signature of P.S.K. Gurusinghe in black ink, written over a dotted line.

P.S.K. Gurusinghe
(Candidate)

Handwritten signature of P.G.V. Dias in black ink, written over a dotted line.

P.G.V. Dias
(Supervisor)

Dedicated to My Parents...



University of Moratuwa, Sri Lanka.
Electronic Theses & Dissertations
www.lib.mrt.ac.lk

Abstract

Obtaining a reasonable speed for web access is a problem in many areas, especially for small organizations such as schools, dial-up, and mobile users, due to the low bandwidth of the available links. One solution, which is supported by HTTP 1.1, is the compression of web pages, but this needs support by both the web server and web client. As most web servers have not enabled such support, this feature is in limited use.

An alternative option is to implement compression between two proxy servers located at each end of the bandwidth-limited link. This dissertation describes the implementation of such a system.

A compression scheme was implemented which is transparent to both client and server. Data is compressed at the upstream proxy server of the bandwidth-limited link, and de-compressed at the downstream proxy server of the link. Different types of content are identified based on the content-type HTTP header and different compressors are used on each content-type.

HTTP headers and text content-types (html, css, txt etc.) are highly compressible. A number of text compression schemes were evaluated, and, gzip was selected as the compressor for such content. A unique feature of our system is the use of a pre-set dictionary for HTTP header compression, which enabled us to get very good compression ratios.

Although jpeg images are already in a compressed format, they can generally be further compressed without excessively degrading the perceived image quality. We do so whenever feasible.

Persistent connections over the limited bandwidth links were introduced to eliminate the delay caused by TCP connection establishment.

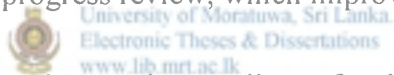
The performance of the system under different workloads was analyzed, which showed that the system provides a significant improvement in response time over a low-speed connection.



Acknowledgement

I am taking this opportunity to thank Prof. (Mrs.) Rathnayake, the Director of Postgraduate studies for selecting and nominating me for the Asian Development Bank scholarship. And also, I wish to acknowledge the financial support received by Asian Development Bank's personnel development fund of Ministry of Science & Technology, Sri Lanka.

I am grateful to the following: project supervisor, Dr. Gihan. V. Dias for his direction and supervision of the project and particularly for his instruction and frequent scrutiny of results, recommendation on methods, analysis and presentation of results. Without such guidance, I would not have been able to implement such a complex system easily; project co-supervisor, Ms. Vishaka Nanayakakara for her assistance, friendliness and guidance throughout the research; Dr. Mark Adler and Dr. Jean Loup-Gailly who are the owners of the gzip for their help. Dr. Sanath Jayasena and Mr. Shantha Fernando for their untiring urge to offer assistance and guidance where needed. My special thanks go to Dr. Ajith Pasqual, for his comments, corrections and guidance given at each progress review, which improved my works.



I am grateful to staff members and my colleges for their frank comments and helping me with my daily work.

Finally my deep gratitude for my spouse and parents for their encouragements and helps right through out this period.

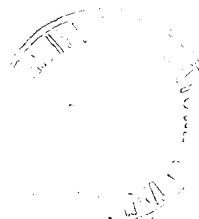


Table of Contents

Table of Contents.....	vi
Table of Figures	viii
1. Introduction	1
1.1. Motivation.....	1
1.2. Problem Definition	2
1.3. Features of Web Traffic.....	2
1.4. System Introduction.....	4
1.4.1. System Key Features	5
1.4.2. Key Benefits	6
1.5. Organization of the thesis.....	6
2. Technology Brief.....	8
2.1. Compression Algorithms.....	8
2.1.1. Lossless Compression.....	8
2.1.2. Lossy Compression.....	18
2.2. TCP/IP, HTTP	28
2.2.1. HTTP	28
2.2.2. TCP/IP	29
2.3. Berkeley Socket.....	30
2.4. Other Related Work.....	32
2.4.1. Layer 2 Payload Compression.....	32
2.4.2. Web Compression.....	32
3. Web Traffic Analysis.....	33
3.1. Data Formats on the Web	33
3.2. Analyzing Network Traffic.....	35
4. System Architecture	37
4.1. System Overview.....	37
4.2. System Requirements	37
4.3. Permission for Upstream Proxy Placement.....	37
4.4. System Modules	38
4.4.1. Upstream Proxy	38
4.4.2. Downstream Proxy	40
4.5. Algorithm of the System.....	41
5. System Implementation.....	46
5.1. HTTP Header Compression.....	46
5.1.1. Building the Dictionary at the Upstream Proxy	48
5.1.2. Using the Dictionary at the Upstream for Compression.....	49
5.1.3. Building the Dictionary at the Downstream Proxy.....	50
5.1.4. Decompressing Data at the Downstream with the Dictionary.....	52
5.2. Text Compression.....	53
5.2.1. Selecting the Best Algorithm for Text.....	53
5.2.2. Compressing Text Data at the Upstream	56

5.2.3.	Decompressing Text Data at the Downstream	58
5.2.4.	A Possibility of a Pre-set Dictionary for Text	59
5.2.5.	Failures: Using only one Zlib Stream per Proxy	60
5.3.	Image Compression - Image/jpeg	61
5.3.1.	Further Compression on image/jpeg	61
5.3.2.	Deciding the Current Compression Level	61
5.3.3.	Deciding the Compression Ratio to be applied	63
5.3.4.	Jpeg Decompression	65
5.3.5.	Jpeg Compression	67
5.3.6.	Compressed Data Handling	70
5.3.7.	I/O Suspension	73
5.3.8.	Putting it all together	74
5.4.	Other Improvements - Persistent Connections	78
5.4.1.	Overview	78
5.4.2.	Implementing Persistent Connections	79
6.	Results	81
6.1.	Bandwidth Usage	82
6.2.	Round-trip Time	82
6.3.	CPU and Memory Usage	84
7.	Conclusion and Future Work	86
7.1.	Merits of the system	87
7.2.	Future Enhancements	87
References		89



University of Moratuwa, Sri Lanka
 Electronic Theses & Dissertations
www.lib.mrt.ac.lk

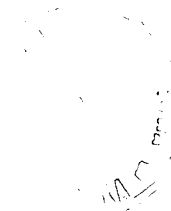


Table of Figures

Figure 1.1: Expensive and limited bandwidth international links	2
Figure 2.1 General Adaptive Compressions	10
Figure 2.2 General Adaptive Decompressions	10
Figure 2.3 (a) Sliding Window Concept with LZ77	15
Figure 2.4 (b) Sliding Window Concept with LZ77	15
Figure 2.5 Dictionary structure built with LZ78	17
Figure 2.6 Two matrices representing the DCT input and output blocks from a gray-scale image	24
Figure 2.7 A Sample Quantization Matrix with a quality factor of two	25
Figure 2.8 DCT Matrix before Quantization	26
Figure 2.9 DCT Matrix after Quantization	26
Figure 2.10 The DCT block are reordered in a zig-zag sequence and the direction of the path is shown by the arrow head	27
Figure 2.11 The TCP/IP client-server communication scenario	31
Figure 3.1 Examples for popular content-types	34
Figure 3.2 The Distribution of Content-types with Byte Count Based on LEARN Network Traffic	36
Figure 4.1 Overview of the system	37
Figure 4.2 Main components of the upstream proxy	38
Figure 4.3 A Node on the List	39
Figure 4.4 Main components of the downstream proxy	40
Figure 4.5 An Element of the queue	41
Figure 4.6 Communication between web browser and web server	41
Figure 4.7 Flow chart of the Downstream Proxy	44
Figure 4.8 Flow Chart of the Upstream Proxy	45
Figure 5.1 An example of an HTTP response header	46
Figure 5.2 HTTP Header compression statistics	47
Figure 5.3 Generating the compression engine at the upstream	49
Figure 5.4 Compressing HTTP Header using the dictionary	49
Figure 5.5 Building the Decompression Dictionary at the downstream	51
Figure 5.6 Compression Ratio Vs File Size	53
Figure 5.7 Compression Ratios for text files (File size < 4,000 Bytes)	54
Figure 5.8 Compression Ratios for text files (4KB < File size < 20 KB)	54
Figure 5.9 Compression Ratios for text files (20KB < File size)	55
Figure 5.10 Compression Time Vs File Size	56
Figure 5.11 Decompression process at the downstream proxy	58
Figure 5.12 Out of Order Data Arrival at the downstream proxy	60
Figure 5.13 JPEG Image Compressions	62
Figure 5.14 JPEG compression performances	62
Figure 5.15 JPEG compression performances – compression time	63
Figure 5.16 JPEG Image samples with different quality levels	64
Figure 5.17(a) JPEG Image Compression at the upstream proxy	75
Figure 5.17 (b) JPEG Image Compression at the upstream proxy	76
Figure 5.18 Compressor and Decompressor - working buffer	76
Figure 5.19 TCP Connection Queue at the downstream proxy	79

Figure 6.1 Bandwidth usages while retrieving a web page with and without involvement of inter-proxy data compression system	82
Figure 6.2 Round-trip time data on a 32Kbps link	83
Figure 6.3 Round-trip times to retrieve a web page with and without compression ...	83
Figure 6.4 Memory and CPU Usage of the system with compression	84
Figure 6.5 Memory and CPU Usage of the system is negligible compared to other processes	85



University of Moratuwa, Sri Lanka.
Electronic Theses & Dissertations
www.lib.mrt.ac.lk