Characterisation of End-to-End Performance for Web Based File Server Repositories

Manoel Eduardo Mascarenhas da Veiga Alves

Thesis submitted for the degree of

MASTER'S OF ENGINEERING SCIENCE

Centre for Telecommunications Information Networking (CTIN)
Department of Electrical and Electronic Engineering
Faculty of Engineering

ADELAIDE UNIVERSITY
AUSTRALIA

JANUARY 2001

Abstract

The Internet has evolved dramatically in the past few years as result of developments in internetworking/telecommunications technologies and increasing market demand for interactive services. While increasing service diversity is noticeable through steady competition for developing and delivering multimedia content, the Internet infrastructure as a whole has evolved in a rapid and almost "organic" fashion, resulting in an enormous mesh of hosts, networks and network peering points – a complex and fault susceptible environment. In addition, the lack of Quality of Service (QoS) standards and the lack of consensus on what is an adequate QoS level has led to an atypical situation where customers are serviced in a best-effort basis, without clear guarantees of effective performance.

In the face of these performance issues and the assortment of application service level requirements, several research initiatives have gained attention in the last few years. The main initiatives currently being debated are:

- Backbone improvements: bandwidth maximisation, development of new resource management/measurement techniques and tools, Mbone (Multicast Backbone), etc;
- Protocol improvements: IPV6, RSVP, Tag Switch, Jumbo-frames, TCP for high performance networks, routing protocols, etc;
- Data compression techniques: development of new compression methods for encapsulating video, audio and images over IP;
- New access technologies: Fibre wavelength re-use via DWDM (Dense Wavelength Division Multiplexing), Digital Subscriber Line (xDSL), Hybrid Fibre Coax (HFC) platforms, Local Multipoint Distribution Service (LMDS) systems, Multichannel Multipoint Distribution Service (MMDS) systems and high bandwidth satellite access;

- Economic drivers: new mechanisms for charging Internet traffic due to the inefficiency of traditional PSTN charging methods for measuring Internet usage and unfairness of peering agreements.

This report investigates the behaviour of TCP bulk file transfer application sessions in a broadband access environment. The focus is on the development of an end-to-end throughput measurement tool for evaluating the effects of *both* internetworking topology *and* application server traffic demand *over* throughput for a broadband user while downloading files from remote Web based file server repositories. In addition, the decision to carry out this research was influenced by three other reasons:

- Lack of standards for diagnosing, evaluating and correcting performance problems in a particular network;
- TCP packets are responsible for 90 to 95 percent of all Internet traffic [15];
- The importance of file downloading has increased together with the development of new compression formats because multimedia content such as compressed music, compressed video and general data can be widely found on the Web. Furthermore, NAPSTER[1] and other collaborative download environments have increased the popularity and demand for downloading files among a community of Internet users [9,10,11]. The traffic impact of NAPSTER has been such that some academic and commercial organisations are now prohibiting its use [12,13].

In terms of Internet analysis modelling, this research introduces some concepts for evaluating network behaviour: a path instability parameter (ε) for analysing different TCP connections; a minimum RTT delay and a minimum typical path for estimating path characteristics between a client and application servers.

The main findings of this research are: a strong correlation was observed between throughput performance and traffic demand in *both* the Application Server *and*

interconnecting networks; some cyclical throughput behaviour suggesting a *high* influence of the US backbone on throughput performance for download sessions originating from Application Servers located outside Australia; throughput performance depends not only on technical drivers *but also* on economic drivers such as the inter-network pricing regime; for some downloading sites, low throughput performance seems to be related to a higher number of satellite links within interconnecting networks; finally, in spite of having a connectionless network layer, the Internet environment has a high path stability, i.e., it is predominantly connection-oriented. The latter confirms research carried our by Paxson, Lebovitz et al & Chinoy, where path changes affect less than 1% of Internet connections [41,45,46].

[1] See http://www.napster.com

Declaration

This work contains no material which has been accepted for the award of any other degree or diploma in any university or other tertiary institution and, to the best of my knowledge and belief, contains no material previously published or written by another person, except where due reference has been made in the text.

I give consent to this copy of my thesis, when deposited in the university library, being available for loan and photocopying.

SIGNED **DATE** 23.01.01

Acknowledgments

Firstly, I praise *God* for giving me health, tranquillity, patience and determination to challenge my limits during this great life experience.

This research project was carried out with the assistance of a Scholarship provided by my company, *Research and Development Centre for Information Processing and Automation* (CPDIA). I wish to express my deep appreciation *not only* for the financial support *but also* for the confidence and responsibility placed in me. Sincere appreciation is due to a number of professionals for making this period in Australia possible*: Prof. Edgard A. Romanato, Prof. Katsuyoshi Kurata, Prof. Osório Chagas Meirelles, Mr. Soji Iura, Mr. Aiser C. Cordeiro, Mr. Oswaldo Ken-Ichi Furuzawa* and *Mr. Edson Hayashi.*

I am grateful to my supervisor, *Prof. Reginald P. Coutts* for helping me improve my research skills, by stimulating my independent thinking and, primarily, for making sure I pursued my ideas.

I would like to express my special gratitude to *Dr. Sergey Nesterov* for his professionalism and guidance during the development of the experimental analysis of this project. Undoubtedly, without his fruitful discussions and his talent in software programming, this project would not have been so challenging and rewarding.

Mr. David Klemitz must be acknowledged for his encouragement, enthusiasm, criticisms, suggestions and mainly for his friendship during this academic experience.

I am also very thankful to all *Centre for Telecommunications Information Networking* (CTIN) staff, for providing a pleasant and welcoming research environment. Some people, however, deserve a special reference: *Mrs. Hilde Crook* for her professional

conduct, kindness and helpfulness; And, *Ms. Collete Snowden*, for her assistance in proofreading the final version of this thesis.

In addition, I am indebted to several professionals from *the University of Adelaide*. My sincere gratitude extends to three key people: *Prof. Ken Sarkies*, who was my first contact at the University and who has always demonstrated a welcoming and supportive attitude towards overcoming barriers faced; *Dr. Nigel Bean* for suggestions to improve the end-to-end delay performance analysis; And, *Prof. Lang White* for his relevant technical contributions for enhancing the ultimate development process.

Finally, the completion of this thesis is a victory for a person who has always been an example of life, dedication, determination and honesty. My extreme admiration goes to my *Mum* for all she has provided to me as a *Mum, friend* and *educator*.

List of Publications

Conference Publication:

Manoel Eduardo Mascarenhas da Veiga Alves, Reginald Paul Coutts & Sergey Nesterov, **"International Conference on Performance and QoS of Next Generation Networking - P&Q Net2000"**, November 27-30, 2000, Nagoya, Japan

List of Abbreviations

Acronym	
AAL5	ATM Adaptation Layer 5
AARNET	Australian Academic Research NETwork
ACK	ACKnowledgment flag
ADSL	Asymmetric Digital Subscriber Line
AS	Autonomous System
ATM	Asynchronous Transfer Mode
BDP	Bandwidth-Delay Product
BER	Bit Error Rate
BGP	Border Gateway Protocol
BSP	Backbone Service Providers
CERN	European Centre for Nuclear Research
CRC	Cyclic Redundancy Check
DARPA	Defense Advanced Research Projects Agency
DF	Don't Fragment bit
DNS	Domain Name System
DWDM	Dense Wavelength Division Multiplexing
EGP	Exterior Gateway Protocol
EndT	End Time
ESP	Encapsulating Security Payload
FDDI	Fibre Distributed Data Interface

FEC	Forward Error Control
FIFO	First In First Out
FIN	FINish flag
FTP	File Transfer Protocol
FTTB	Fibre To The Building
FTTH	Fibre To The Home
GEO	Geo-synchronous satellites
GIF	Graphics Interchange Format
GPS	Global Positioning System
HEC	Header Error Control
HFC	Hybrid Fibre Coax
HIPPI	High Performance Parallel Interface
HTTP	Hyper Text Transfer Protocol
ICMP	Internet Control Message Protocol
ID	Identification
IETF	Internet Engineering Task Force
IGP	Interior Gateway Protocol
IP	Internet Protocol
IPMP	Internet Protocol Measurement Protocol
IRR	Internet Routing Registry
IPv4	Internet Protocol version 4
IPv6	Internet Protocol – version 6
ISP	Internet Service Providers
JPEG	Joint Photograph Experts Group
LEO	Low Earth Orbit satellites
LFN	Long Fat Networks

LMDS	Local Multipoint Distribution Service
MEO	Medium Earth Orbit satellites
MMDS	Multichannel Multipoint Distribution Service
MPEG	Motion Picture Experts Group
MP3	MPEG 1 Layer 3
MSS	Maximum Segment Size
MTU	Maximum Transmission Unit
NFS	Network File System
NIC	Network Interface Cards
NNTP	Network News Transfer Protocol
NPD	Network Probe Daemon
NSP	Network Service Provider
NTP	Network Time Protocol
OSI	Open Systems Interconnection Reference Model
OSPF	Open Shortest Path First
OW	One-Way delay
PAWS	Protection Against Wrapped Sequence numbers
PERL	Practical Extraction and Report Language
PoP	Point of Presence
PMTUD	Path Maximum Transmission Unit Discovery
PSTN	Public Switch Telephone Network
QoS	Quality of Service
RAM	Real Audio Metafile
RSVP	Resource reSerVation Protocol
RST	ReSeT flag
RTT	Round Trip Time

SACK	Selective ACKnowledgment
SAA	Single Administrative Authority
SBF	Standard Benchmark File
SLA	Service Level Agreement
SNMP	Simple Network Management Protocol
SMTP	Simple Mail Transfer Protocol
StartT	Start Time
SYN	SYNchronise sequence numbers flag
TCB	TCP Control Block
TCP	Transmission Control Protocol
TTL	Time To Live
UDP	User Datagram Protocol
URL	Uniform Resource Locator
VCI	Virtual Channel Identifier
VDSL	Very high speed Digital Subscriber Line
VPI	Virtual Path Identifier
xDSL	Digital Subscriber Line
ZIP	A compression format
WAN	Wide Area Network
WWW	World Wide Web

List of Figures

List of Tables

In memory of my Father –
whose enthusiasm, spontaneity and passion
for life will always be alive

Chapter 1
Introduction

1.1 Internet Development: a Summary

The Internet origins date from 1973, when the U.S. Defense Advanced Research Projects Agency (DARPA) was carrying out a project for investigating open-architecture network technologies, which would allow interconnecting packet networks with heterogenous traffic/link characteristics [1]. The investigation, which was entitled the *Internetting* project, resulted in a set of communications protocols known as the Transmission Control Protocol (TCP) / Internet Protocol (IP) Protocol Suite or simply, TCP/IP.

The genesis of the Internet revolution as we know it, started with the World Wide Web (WWW) in 1989. The Web was created at the European Centre for Nuclear Research, CERN, where teams of researchers being geographically dispersed required a common environment for sharing documents, reports, graphics and images. However, the real revolution happened after the launching of the first graphical Interface for web browsing called *Mosaic* in February 1993 [2].

Before *Mosaic*, the Internet was mainly used for research and academic purposes. Because *Mosaic* opened the opportunity for developing a user-friendly information environment, new applications and services delivering rich multimedia content were created. This resulted in popularisation of the Internet among non-academic users and in an increasing per capita usage profile. This rise in traffic demand resulted in an exponential investment in backbone capacity infrastructure. In the mid-1980s backbone capacity was 56 Kbps, in 1987 the capacity increased to 1.5Mbps and to 45Mbps in 1991 [3]. Recently, new developments in optical transmission such as Dense Wavelength Division Multiplexing (DWDM) are increasing backbones to the order of Gigabits.

Moreover, for the access network, this overall increase in traffic demand conflicted with the traditional "3-3-3" telephone networks design rule, which states that the average voice call lasts 3 minutes, the user attempts to call 3 times during the peak time and the call occupies a bi-directional 3 kHz channel. Recent Industry studies reveal that the demand for data delivery is increasing 10 times faster than the demand for voice [4].

Due to the high penetration level of the Internet among high-income customers, most businesses enterprises have become aware of the enormous commercial opportunity in establishing a Point of Presence (PoP) on the Web, not only as a means for increasing commercial revenues but also to reduce operational costs. Kennard [5] remarked that revenues from electronic commerce were around $20 billion in 1996 and are expected to expand to $350 billion by 2002. Hence, companies are investing massively in developing user-friendly sites on the Internet, primarily focusing on capturing customers' attention and attracting a loyal Internet-literate customer base.

This market strategy has been enforced by current multimedia design developments based on proprietary software solutions (such as *Shockwave* from *Macromedia*) and in the development of portable and freely distributable computer languages (such as *JAVA*).

From an engineering point of view, the expansion in multimedia content can be explained as an increase in data to be transported by the network. While the backbone can have its capacity increased[1], the access network has limitations. The Public Switch Telephone Network (PSTN) was not designed to carry data but, rather was built around limited and dedicated voice channels, conforming to the "3-3-3" rule, resulting in low overall performance for Internet dial-up users.

[1] In spite of the increase in backbone capacity, some studies report a decrease in the level of Quality of Service by measuring packet loss, packet delay and router flapping (See [6])

As a consequence, for the last few years there have been intensive developments and investments for improving performance of Internet applications. The main initiatives in the currently being debated are:

- Backbone improvements: bandwidth maximisation, development of new resource management/measurement techniques and tools, Mbone (Multicast Backbone), etc;
- Protocol improvements: IPV6, RSVP, Tag Switch, Jumbo-frames, TCP for high performance networks, routing protocols, etc;
- Data compression techniques: development of new compression methods for encapsulating video, audio and images over IP. Compression formats such as MPEG, MP3, JPEG, GIF, RAM, and ZIP are becoming the usual file extension terminology among Internet users;
- New access technologies: Fibre wavelength re-use via Dense Wavelength Division Multiplexing (DWDM), Digital Subscriber Line (xDSL), Hybrid Fibre Coax (HFC) platforms, Local Multipoint Distribution Service (LMDS) systems, Multichannel Multipoint Distribution Service (MMDS) systems and high bandwidth satellite access;
- Economic drivers: new mechanisms for charging Internet traffic due to the inapplicability of traditional PSTN charging methods for measuring Internet usage and unfairness of peering agreements.

1.2 Overview of Internet Applications & Reasons for Choosing File Download Application

These studies have been driven by diverse types of Internet applications, which deliver a comprehensive spectrum of information within heterogeneous Quality of Service requirements.

In terms of end user perception, information can be classified as either time-based or non-time based. The former carries within it intrinsic time properties. For example, *video* has

information about frame rate display and audio/image synchronisation. Conversely, non-time based information has no essential built-in time property for adequate display (e.g., a written document, a JPEG picture, etc).

Furthermore, based on Quality of Service (QoS) requirements, applications can be classified in real-time streaming, real-time block transfer and non-real time applications [7].

Real-time streaming applications deliver time-based information in real time over the network. For adequate user perception, the network should deliver the time-based information without changing its built-in time properties. Therefore, certain QoS requirements such as delay, jitter and error rates must be taken into account for an adequate provision of these applications. The most well known Internet applications in this group are Web television, Internet telephony and Internet radio.

Real-time block transfer applications deliver either time-based or non-time based information. This group delivers one or more blocks of information within a deadline though, unlike real-time streaming applications, consecutive blocks do not have a time correlation. Internet applications in this group are: Web browsing, client-to-client application sharing, online games, chat and file transfer.

Non-real time applications deliver both time-based and non-time based information without a demanding time delivery deadline. In terms of QoS, the main requirement is delivering error-free information. The most well known application in this group is electronic mail (e-mail).

Given the potential bottlenecks in the access network and taking into account the common Internet applications, file downloading via Hyper Text Transfer Protocol (HTTP) or File Transfer Protocol (FTP) is likely the most "unpleasant" application for dial-up customers today. In addition, it is assumed that it might be the most important real-time block transfer

application for a broadband access platform in the future. An experiment with file transfer will then be a good test for analysing current Internet performance.

Moreover, to some degree file transfer application can be considered as an approximation for data streaming (e.g., video) in a non-congested environment because a Transmission Control Protocol (TCP) session with a reasonable window size has a consistent throughput over time after the slow start phase [8].

The importance of file downloading has increased together with the development of new compression formats because multimedia content such as compressed music, compressed video and general data can be widely found on the Web. Furthermore, NAPSTER[1] and other collaborative download environments increase the popularity and demand for downloading files among a community of Internet users [9,10,11]. The traffic impact of NAPSTER has been such that academic and commercial organisations are now prohibiting its use [12,13].

Furthermore, the FTP program is the most significant application for transferring scientific data over the Internet. The final report from the "Advanced Networking Infrastructure Needs in the Atmospheric and Related Sciences" (ANINARS) workshop in July 1999 states:

"... the workhorse networking application in the atmospheric community is still FTP. Aside from [...], FTP is practically the only networking tool used to construct applications in this scientific discipline. There was also a universal cry for FTP to actually deliver the available network bandwidth to the end-user. The lament was that the bandwidth actually obtained is much lower than the apparently available bandwidth.

[1] See http://www.napster.com

Most participants thought that the need for bulk data transfer would never go away, even if sophisticated data extraction methods could be developed to extract subset portions of data sets. Such mechanisms would simply supplement the FTP function but not replace it."

The same report continues:

"FTP (or FTP-like) bulk data-transfer is the most important networking function used to construct applications in this scientific discipline, yet failure to achieve effective bandwidths equal to apparently available bandwidths is most evident with bulk data-transfer applications. A variety of host-software problems contributes to this failure, and programs should be developed to help solve these problems" [14].

Finally, since TCP packets are responsible for 90 to 95 percent of all Internet traffic [15], a TCP file download measurement tool provides a suitable analysis of bulk transfers over the Internet. In the next Chapter, a discussion about popular measurement techniques to monitor Internet performance is introduced, including strengths and possible areas for improvement.

1.3 Thesis Overview and Objectives

This report investigates the behaviour of TCP bulk file transfer application sessions in a broadband access environment. The focus will be on the development of an 'active' measurement tool[1] that can be applied for Internet access measurement performance. This offers the possibility of calculating the visible throughput to a broadband user while downloading from the Internet. Unlike some studies that analyse the total flow at an aggregated point of the network [16], our study measures the traffic on an individual end user flow basis.

[1] The definition of 'active' measurement tool will be provided in Chapter 2.

As opposed to traditional measurement techniques, the deployed methodology combines both *network layer* analysis based on the automation of the *Traceroute* [17] utility and the investigation of the *application layer* via throughput calculation (See Chapter 2 for measurement methodologies). Therefore, the method can be considered a Service Level Agreeement (SLA) type-monitoring tool and provides an end user based perspective of network performance.

Our method uses a PC as a client, which is connected to the Internet backbone via a Fast Ethernet Local Area Network (LAN). Unlike some performance measurement tools this is inexpensive and easy to implement requiring only a PC and a backbone connection. No specialised infrastructure or negotiating skill is required.

Finally, the method has both "access platform technology" and location independence. Using this method, a client server with broadband access in different geographical locations is able to measure its download traffic conditions in relation to its local Wide Area Network (WAN).

This thesis is organised as follows: in Chapter 2, a comprehensive study of current Internet performance measurement methodologies is provided; in Chapter 3, an overview of potential bottlenecks affecting Internet performance is presented; in Chapter 4, the experimental methodology is described and some analysis assumptions are introduced; in Chapter 5, an investigation of measured data is provided and a number of findings are discussed; in the last chapter, the main findings are highlighted and further developments in our methodology are suggested.

Chapter 2

Internet Performance Measurement: Methodologies & State of the Art

This chapter provides an overview of the state of the art of the Internet performance measurement field. Firstly, the main parameters for measuring performance of networks are presented. Secondly, performance measurement methodologies are classified based on their implementation characteristics; for each type of measurement methodology, a comprehensive discussion of current research is provided. This review of research highlights the main *findings* and *drawbacks* reported in different measurement implementations worldwide.

2.1 Reasons for Deployment of Internet Performance Measurements

The current Internet infrastructure is a mesh of networks that has evolved in a rapid and almost "organic" fashion. The remarkable growth after the WWW development has led to an enormous number of hosts, networks and network peering points, which results in a complex and fault susceptible environment.

For easy understanding, the Internet can be analysed as a collection of Autonomous Systems[1] (AS) or administrative domains, which vary in size, geographical coverage and function (Fig 2.1). AS have different IP traffic characteristics, which are classified as local or transit traffic. When AS IP traffic is originated or terminated locally, it is classified as local traffic. AS IP traffic that is not originated or not terminated locally is categorised as

[1] Autonomous Systems are a group of subnetworks administered by a Single Administrative Authority (SAA) with a set of Interior Gateway Protocols (IGPs). The SAA is normally a Network Service Provider or large organisational network (e.g., campuses and corporate networks).

transit traffic. In terms of AS geographical coverage, AS might be classified into three types:

- A *stub AS* which has local coverage and the IP traffic is local. In the early days of the Internet, *stub* AS would connect to only one *multi homed* AS. Currently, they connect to one or more *multi homed* AS or *backbones*;
- A *multi homed AS*, which has regional or metropolitan coverage. In the early days of the Internet, *multi homed* AS IP traffic had local characteristics and would be connected to a single backbone provider, the U.S. National Science Foundation (NSF) backbone - NSFnet. Currently, *multi homed* AS might have connections to other *multi homed* AS and more than one *backbone* providers. IP traffic type might have local or transit characteristics;
- A *transit AS or backbone* has national or international coverage. Traffic might have local or transit characteristics. *Currently, backbones* exchange traffic in interconnecting points, which are guided by bilateral data traffic exchange interconnection arrangements. These arrangements vary depending on characteristics of the AS like traffic flow, number of customers, content information and geographic distribution. An interesting discussion about Internet connectivity in a commercial environment is found at [18].

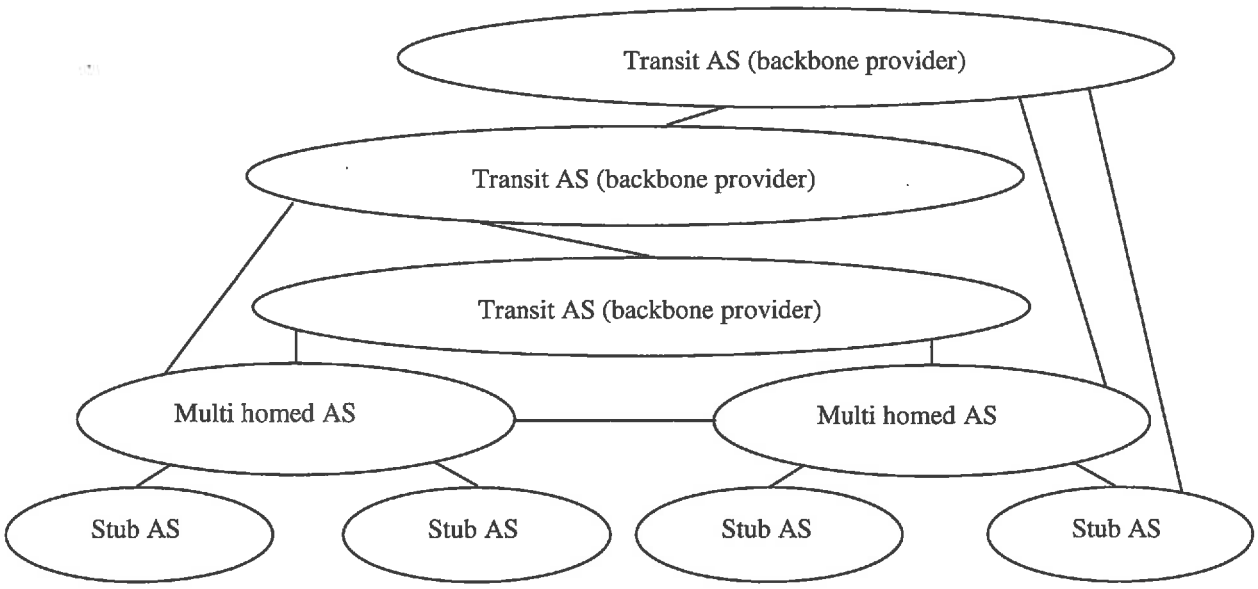


Figure 2.1: Internet Topology in a Commercial Environment
- Stub AS interconnects to Multi Homed AS and Backbone providers;
- Multi homed AS interconnect to other Multi Homed AD and backbone providers;

Historically, the Internet has not been adequately measured [19]. After the Internet transition from the NSFnet backbone stewardship into a competitive backbone provider market in 1995, difficulties in measuring and monitoring the overall Internet performance increased substantially. In this regard, the commercialisation of the Internet has led to difficulties in implementing and developing measurement platforms. For example, Backbone Service Providers (BSPs) do not wish other parties to gain knowledge about the performance of their networks and Internet Service Providers (ISPs) do not undertake research because low profit margins make it prohibitive [20].

Furthermore, there is a lack of standards for identifying, evaluating and correcting performance problems in a particular network. Currently, network performance study is more of an art than a science [21]. This is particularly true for the upper layers (3 and above) at the Open Systems Interconnection (OSI) Reference Model.

Despite this environment, there are some initiatives for measuring Internet performance with different methodologies and techniques. Some of these are run independently by specific groups of researchers, network administrators or *networkaholic* individuals. Others involve consortium research groups working on the deployment of large-scale Internet measurement infrastructures. However, since there is no consensus on what should be measured and because the Internet is a "best attempt" type network – as distinct from the PSTN where there are specific QoS standards-, the development and commercialisation of robust performance measurement tools will await the standardisation process.

There are several reasons for deploying performance measurement tools within a network, for example:

- Useful for diagnosing performance degradation and points of failure within a network;
- Provides a mechanism for evaluating network usage which can be useful for deploying a "packet billing" system;
- Anticipates future upgrades for the network infrastructure;

- Establishes a minimum QoS level for users in terms of network resources and application requirements. The QoS level is normally defined in a SLA.

Measurement tools have become even more important with the introduction of SLAs among some Network Service Providers (NSPs) and customers. These NSPs deploy monitoring tools to report statistics for their clients and therefore guarantee network performance and availability. Providing a certifiable QoS level, these NSPs distinguish themselves from other competitors without SLA monitoring tools. Blacharski [22] also observes that SLA monitoring tools should focus on the *application layer* rather than the *network layer* because the *application layer* provides an end user metric perspective, which is the most appropriate for measuring service level satisfaction.

2.2 Parameters for Measuring Network Performance

There are several parameters that can be measured to estimate network performance: aggregated traffic, packet loss, packet delays, path behaviour and throughput. These are defined as follows:

2.2.1 Aggregated Traffic:

The study of aggregated traffic at a certain point in a network provides an overview of the traffic mix in terms of protocols and application types. Understanding backbone traffic characteristics is useful for modelling traffic tendencies, optimising network bandwidth usage and previewing future network upgrades.

2.2.2 Packet Loss:

Due to the unreliability of some networks and the best-effort characteristics of the IP, packets might get lost while moving from source to destination. Therefore, *packet loss* is useful for evaluating the reliability of an Internet connection and, as a result, estimating application behaviour in such an environment.

2.2.3 Packet Delay:

Packet delay is measured based on the Round Trip Time (RTT) or one-delay measurements. RTT is an important variable for estimating the maximum TCP window size for a TCP session, which is calculated based on both the session's RTT and the session's maximum bandwidth of the least-capable hop. There are three sources of packet delay: *transmission, propagation and queuing* delays. Transmission delay is the time necessary for transmitting a packet to the network via a network interface card. Propagation delay is the time necessary for moving data from source to destination in terms of finite speed of electrons and photons. Queuing delay is the delay due to packet queuing transmissions over the network. The main source for queuing delay is router queuing. Since transmission delay is negligible due to the high speed of network card interfaces, transmission delay will not be an important delay source in new network environments. Hence, Internet end-to-end delays can be considered as a sum of *both* propagation *and* queuing delays.

2.2.4 Path Behaviour:

Path behaviour analysis provides a number of insights about *virtual paths* throughout the Internet, such as path stability and node reachability.

- **Path stability or routing stability:** Path stability is a useful parameter for estimating routing behaviour in a network environment. Path changes might be the result of legitimate connectivity problems, routing policies or routing anomalies. There are a number of origins for routing anomalies in a network: router configuration errors, software bugs and transient physical and data link problems [23]. Low path stability contributes to inadequate end-to-end network performance resulting in increased packet loss, higher delays for network convergence and additional memory/CPU overhead for routers;
- **Reachability:** Reachability is the property of a node to respond to request-replies. A node might become unreachable for network reasons (e.g., link failure) or

intrinsic node reasons (e.g., a computer crash). The evaluation of reachability provides a reliability measurement for estimating the confidence level of a network.

2.2.5 Throughput:

Throughput measurement is a parameter used mainly for evaluating the *application layer* performance of a certain TCP session. The main factors that might influence throughput performance are:

- **Physical layer**: Characteristics of the link *physical layer* such as transmission delay, errors and capacity have a strong influence over throughput. For example, long propagation delays in geostationary satellite links impacts negatively at the performance of TCP sessions both due to the high Bandwidth-Delay-Product (BDP) and the effects of TCP *slow start* and *congestion* control algorithms; moreover, in lossy long-delay links, TCP's data recovery algorithm works inefficiently [24, 25, 26];
- **Data link layer**: Parameters such as Maximum Transmission Unit (MTU) can reduce efficiency of transmissions due to unnecessary overheads while transmitting data;
- **Network layer**: Congestion, path instability and RTT influence the throughput because end-to-end congestion avoidance and congestion control mechanisms guide TCP session performance. Moreover, there is no mechanism for distinguishing delay sensitive and best effort traffic in routers: all traffic is treated equally - First In First Out (FIFO) queuing;
- **Transmission layer**: TCP protocol stack needs improvements for allowing high performance sessions in a multi-application, multi-network environment. For example, TCP protocol was not designed for working in a network environment with high packet loss and large BDP such as a satellite link. Similarly, short-duration transfers on lossy links have a different requirement to long-duration transfers on uncongested high-bandwidth links;

- **Application layer**: The *application layer* design and the interaction of the *application layer* with the TCP stack are very important for an adequate throughput. Designing an application with an end user perspective in mind is important for a successful application performance. A recent report shows that web pages with several objects have low performance compared with homepages with a few objects. Furthermore, applications should report to the TCP stack with their requirements in terms of bandwidth and delay.

In Chapter 3, a further discussion is provided about potential bottlenecks that might influence the performance of TCP applications in the current Internet environment.

2.3 Methodologies for Measuring Performance

Most of the network performance research focuses on measuring several parameters for evaluating network performance. Therefore, for easy reading, in this section an overview of the most important findings is provided. This discussion is organised by research work rather than by a specific parameter.

There are three approaches for measuring network parameters: passive & active measurements and Control monitoring. These techniques will be discussed below.

2.3.1 Passive Measurement

2.3.1.1 Definition & Overview

In a passive measurement methodology (See Fig 2.2) a probe is deployed at a certain point within a network for recording network activity.

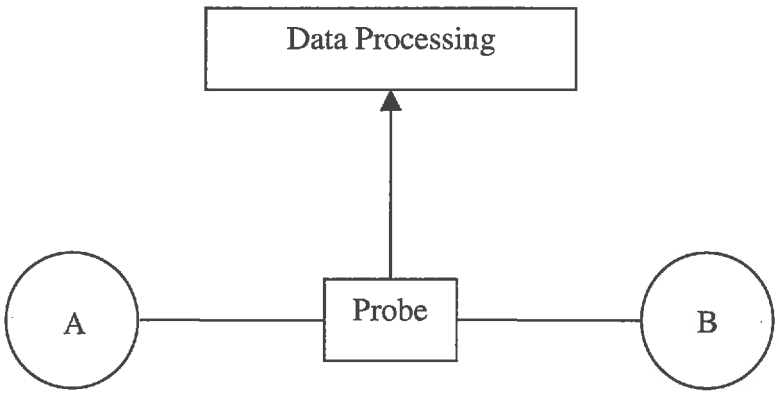


Figure 2.2: Passive Measurement Method

The passive measurement technique has been widely deployed for measuring the traffic conditions at a certain point of a network. It allows a network administrator to understand network behaviour via traffic flow matrices (i.e., the traffic flowing among different points in a certain network) not only in terms of volume (number of bytes and/or packets) but also in terms of type of traffic (SMTP, HTTP, DNS). This tool can provide a map of the network, useful for maximising effective network usage.

2.3.1.2 Related Work

Heimlich [26] and Claffy *et al* [28] were responsible for the first known studies in backbone traffic analysis. Their research evaluated the NSFnet backbone workload characteristics in terms of application traffic such as FTP, e-mail, net-news, telnet, DNS, etc.

One of the first studies to verify the WWW exponential traffic demand increase was based on a passive measurement of a site's wide-area network use. In 1994, Paxson [29] verified that for a site where the number of Internet hosts increased 30 percent per year, the amount of data transferred and number of connections made for a number of TCP protocols grew considerably faster. For example, the WWW traffic growth was found to be 300-fold/year.

Furthermore, while the majority of total traffic involved *.edu* and *.gov* sites, the study suggested a rise in *.com* traffic due to the increasing commercialisation of the web.

Thompson *et al* [18] provided a study of traffic in a backbone provider in the US. Based on the passive measurement technique of directly tapping into the physical medium, several interesting conclusions were obtained:

- TCP was the dominant IP protocol, averaging about 95 percent of the bytes, 85-95 percent of the packets and 75-85 percent of the flows. The User Datagram Protocol (UDP) was the second most important IP protocol accounting for almost all-remaining traffic. Internet Control Message Protocol (ICMP) traffic averages less than 1 percent of all packets;
- TCP traffic characteristics on the International links were similar to domestic links and the average packet size was 300 bytes;
- Average packet size also varies over time and International traffic links have packet size-asymmetry. For example, larger packets flow from the US to Europe while in the reverse path packets are characteristically smaller. This fact suggests that Web clients in Europe request content from US Web servers which replies with larger (content) packets;
- Web traffic responded for 65-80 percent of the bytes, 55-75 percent of the packets and 65-75 percent of the flows;
- Other TCP applications such as FTP-data and Network News Transfer Protocol (NNTP) have a higher percentile contribution over night hours than during day-hours. However, their maximum contribution is seldom 10 percent of total traffic;
- Traffic in WANs followed a 24-hour pattern;
- 50 percent of all packets were smaller than 45 bytes and almost 100 percent were 1500 bytes or smaller;
- Traffic load is highly dependent on time of the day. Traffic also decreases when comparing weekdays and weekend days;
- *Realplayer* traffic follows a traffic pattern that shows the highest number of packets and bytes relayed during business hours.

The main weaknesses of this study were having spanned a relatively short experimental time and not being able to analyse the traffic on an individual flow basis.

A Recent study from McCreary *et al* [30] presents measurements spanning more than 10 months of observations in a Californian Internet Exchange. The study was based on the analysis of IP and *transport-layer* headers and on assumptions related to protocol port number usage. Their main findings were:

- TCP was responsible for 85% of traffic and bulk transfer applications (FTP or HTTP) were responsible for a large amount of this traffic;
- TCP traffic had characteristically three packet sizes:
    - *40 byte packets* result of TCP acknowledgments;
    - *552/576 byte packets* for TCP implementations without path MTU discovery;
    - *1500 byte packets* for TCP implementations with path MTU discovery (the maximum size of Ethernet frames);
- No relevant changes in TCP and UDP traffic balance were observed;
- While the number of fragmented IP datagrams increased (mainly UDP packets), TCP packets were almost never fragmented. This was probably due to the increasing implementation of path MTU discovery;
- In the last two months of the experiment, Napster traffic increased its percentile contribution by over 50 percent;
- Changes in application usage for a short period of time might suggest seasonal behaviour. For example, Simple Mail Transfer Protocol (SMTP) traffic increased significantly before Christmas and dropped at the end of December. This traffic behaviour might indicate e-commerce transactions associated with the Christmas period;
- While some applications such as online games are on the rise, others like RealAudio and FTP have decreased their overall traffic fraction. For FTP applications, this might

be the result of two factors: a move from active to passive mode FTP in addition to an increase in packet filtering implementations on the Internet; or an increase use of other file transfer protocols. Interestingly, no increase in HTTP traffic was reported.

Some problems were identified in using such a strategy for classifying application traffic:

- For some protocols, there is a range of port numbers that might be used by particular applications. In some cases, different applications might have similar port numbers that results in inadequate application workload analysis. For improving analysis in such cases, they are investigating packet distribution and packet arrival patterns for each application[1];
- Some applications might have port numbers that are not well known and thus a significant TCP and UDP traffic cannot be identified;
- IPSEC might constitute a problem for workload analysis because the Encapsulating Security Payload (ESP) protocol encrypts source and destination ports, which increases difficulties for identifying application traffic types.

A recent study from Martin *et al* [32] suggest a methodology for classifying HTTP traffic in terms of physical, queuing and host processing delays. Based on passive data collection, TCP sessions between two hosts are identified and packets are classified as data, ACK, SYN and FIN packets. This classification allows for the estimating of delays by associating each part of an HTTP session to a different source of delay. While the methodology allows for the estimating of delays related to packet loss and TCP processing, the report does not provide analysis of these types of delay. One of the main findings was that even for high loaded trans-oceanic links between New Zealand and the US, host-processing delays contribute significantly to end-to-end performance.

[1] This has become an interesting research topic. A recent work from Mena & Heidemann [31] indicates that for *RealAudio* traffic several parameters might be used for distinguishing it from other applications' traffic.

Cleary *et al* [33] show that some rules should be followed in order to obtain adequate measurements:

- The best sites for deploying a probe are near nodes of heavy aggregated traffic. Therefore, collecting and processing data on these high-speed links requires specialised measurement hardware;
- There are two ways of *snooping* into high-speed links: a router or a switch duplicating the traffic to a measurement output port or directly tapping into the physical wire. Both strategies have disadvantages. While the former can distort the measured signal due to the overloading of internal communication channels the later deployment requires a link disruption. The later strategy can also result in a signal level deviation;
- A hardware measurement tool is more adequate than a software utility as hardware tools are reliable and maintain stability under heavy load traffic.

2.3.2 Active Measurement

2.3.2.1 Definition & Overview

Active measurement (See Fig 2.3) consists of injecting data into the network and observing the network response. This method can be used for evaluating packet loss, RTT, reachability, path stability and throughput. The main advantage in comparison to passive measurement is that it provides an overview of the end-to-end performance of the network while the passive approach only investigates *in loci* characteristics of the network. The disadvantage is that an inadequate insertion of data in the network can disrupt its behaviour, changing the load environment. In the particular case of throughput measurement, the approach should be adopted with more caution because measuring these network characteristics involves transmitting a reasonable amount of data [34].

Since the transmission of data sometimes involves sending packets over different ASs, the data analysis is a complex and challenging process. For example, if the measurement is

based on sending several ICMP echo requests (like the *Traceroute* and *Ping* utilities) from a client to a remote server for measuring the RTT, the researcher should be careful not to associate the measured interval as the real RTT. Due to the connectionless behaviour of the IP, path asymmetries are common, as the upstream path might be different from the downstream path. Similarly, because AS administrators can configure routers to either give lower priority or restrict ICMP packets, an active tool based on ICMP packets will not show an adequate radiography of the network. Further discussion about difficulties in measuring parameters is described in Section 2.3.2.3.

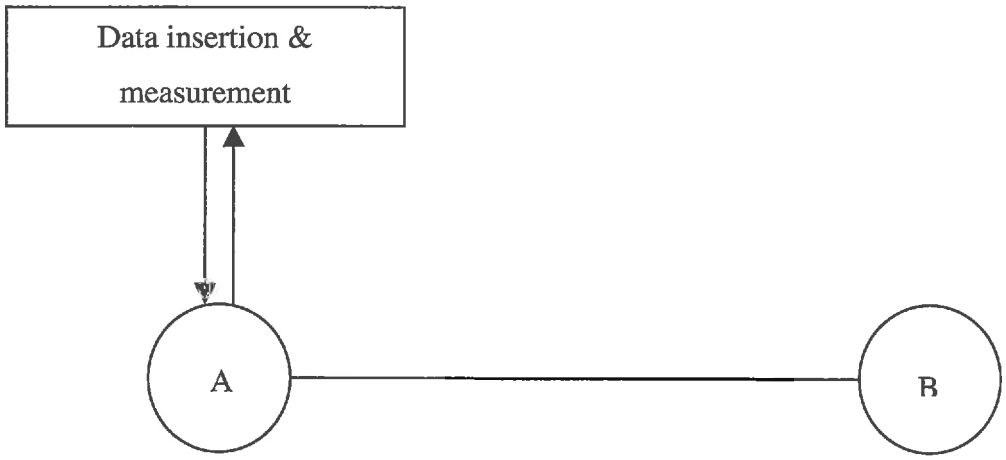


Figure 2.3: Active Measurement Method: Data is injected in A for verifying reachability of B, RTT or throughput

2.3.2.2 Active Infrastructures: an Overview

Active measurements are probably the most ubiquitously deployed approach for measuring network performance. The active measurement approach has evolved significantly during the last few years and has led to the deployment of complex measurement infrastructures. NIMI [19, 35], CAIDA [20], Felix[1] [36], AMP [37], PingER [38] and Surveyor [39] are the most well-known active based measurement infrastructures. These projects are based on

[1] While Felix was cited in some works, its implementation details and results were not found.

measurement infrastructures where several monitoring platforms are placed in different locations over the Internet. Their main differences are related to measured parameters, deployment strategies and measurement goals.

NIMI [19, 35] is based on a scalable measurement infrastructure composed of diversely administered hosts. Its basic concept is to provide a transparent infrastructure for deploying measurement tools developed by third party researchers. Having particular software modules, measurement platforms are able to exchange test traffic among themselves, probing properties of Internet paths and clouds.

CAIDA [20] is an infrastructure that measures forward IP path and RTT from a source to general web and content servers on the web. By simultaneously analysing this data with routing tables and geographical data, CAIDA provides an insight into macroscopic Internet performance and topology dynamics.

AMP [37] provides measurements among cooperative monitoring sites in terms of packet loss, topology, RTT, throughput and topology. Their measurements are publicly available through a prototype web interface [40].

PingER [38] is carried out among a group of collaborative monitoring sites and it provides a study based on the analysis of five metrics: packet loss, RTT, unreachability, quiescence and unpredictability. While packet loss, RTT and unreachability are parameters commonly used for evaluating network performance, unpredictability and quiescence are statistic developments created by PingER researchers to analyse the measured data. Quiescence and unpredictability are described as follows:

- **Quiescence:** If all request-replies in one measurement sample are successful, the network is considered quiescent or non-busy. Mathews & Cottrell [38] report a link that had low performance during business hours and minor packet loss at other periods. A later upgrade of the link improved connectivity to an extent that the link became quiescent;

- **Unpredictability:** This parameter[1] is obtained from a RTT and packet loss variability analysis. The parameter provides an estimation of both link stability and consistency. For a link with low RTT and low packet loss, the parameter will rank it as good. Similarly, a link with steady high packet losses and long RTT will be ranked as good. Poor unpredictable links are only found when packet loss and RTT vary considerably over time.

Surveyor [39] provides one-way measurements (delay, packet loss and routing) among collaborative participant hosts and its measurements are based on a Global Positioning System (GPS) for synchronising measurement platforms.

2.3.2.3 Related Work

One of the first studies for measuring Internet routing behaviour without analysing routing protocols was proposed by Paxson [41]. His report provided extensive results in end-to-end measurements carried out among different Internet sites. These measurements were based on the repeated use of the *Traceroute* utility, which gives an overview of the upstream path. Measurements were scheduled so that consecutive measurements for a certain site would have independent and exponentially distributed time intervals. By following this schedule approach, measurement times respect a Poisson process and the sampling is unbiased: all instantaneous signal values have equal probability, ie, it is an *additive random sampling* process type [42]. Paxson [41] identified some routing pathologies, which are summarised as follows (A, B, C, D, E, F):

A. **Erroneous routing:** One measurement sample had packets erroneously routed to London while the packets should have been sent to Israel. In spite of being a rare case, this illustrates the best-effort characteristics of IP routing algorithms.

[1] $u = \frac{1}{\sqrt{2}}\sqrt{(1-r)^2 + (1-s)^2}$ where s is the ratio of the average and maximum ping success; r is the ratio of average and maximum ping rate; u is the unpredictability.

B. Fluttering: Fluttering is the phenomenon of rapidly oscillating routing and it is normally employed in border routers as a way of balancing load in a network. While fluttering can provide better network usage, it can also result in poor performance for some applications. The main problems related to fluttering are:

B.1 Routing (path) asymmetry: When fluttering happens in just one direction of an application session, a path has routing asymmetry. For network performance analysis, path asymmetry increases the complexity of network troubleshooting.

B.2 Different propagation times: Spurious TCP *fast retransmissions* [43] can generate duplicate acknowledgments due to different packet propagation times, resulting in unnecessary bandwidth usage. Different packet propagation times are common in fluttering implementations due to packets being forwarded to different network paths.

Paxson [41] remarked that fluttering effects could be lessened either when fluttering is implemented in a TCP session basis instead of a packet basis or when fluttering is implemented in a lower layer than the *network layer*.

Moreover, he also suggested that for consecutive *Traceroute* path samples with single-node differences, the node change would not be relevant for stability concerns. These routers would be *tightly coupled* machines with similar traffic loads and possibly co-located.

C. Forward loops: This router pathology is observed when packets forwarded by a router return to the router. Routing algorithms are designed for avoiding *forward loops* taking into account that routers within a network have a consistent view of connectivity. Networks experience forward loops when there is a change in connectivity within the network and routers do not immediately propagate these changes [44]. Since the Border Gateway Protocol (BGP) tags all routing

information with the AS path over which it has traversed, *forward loops* never happen between AS.

D. **Unreachable due to too many hops:** *Traceroute* has a default probe limit of 30 nodes for an Internet path. Paths longer than 30 nodes are considered unreachable due to too many nodes. While no paths were found to have more than 30 nodes during the first experimental period, some paths were unreachable in the second period. Another finding is that comparing the mean path length of the two experimental periods, a slight increase in the number of nodes was found: 15.6 hops for the first period; 16.2 hops for the second one. Therefore, these facts suggested an increase in the number of nodes for Internet paths.

E. **Temporary outages:** *Traceroute* packets might get lost in some network circumstances such as temporary loss of network connectivity or network congestions lasting for 10's of seconds. For the two experimental periods, Paxson [41] found that 55 percent (43 percent) of the samples had no losses; 44 percent (55 percent) had between 1 and 5 losses, which might be attributed to congestion loss; 0.96 percent (2.2 percent) had six or more losses, which might be a temporary loss of network connectivity. Results also show that outages longer than 30 seconds (more than 6 lost packets) have a distribution of lost packets similar to a geometric distribution.

F. **Infrastructure Failures**: A *host unreachable* message originating from a router well inside a network suggests an infrastructure failure. Since a router distant from an individual host has increasingly aggregated routing information for reaching other hosts and networks, a *destination unreachable* message indicates that the loss of connectivity should have affected more destinations than just the host and its local network. Assuming that Paxson's results are statistically significant, his findings suggest an overall availability of 99.5 to 99.8 percent for the Internet infrastructure.

Paxson [41] found that some routing pathologies (temporary outages and infrastructure failures) were associated with time-of-day patterns. The highest number of pathologies happened around 15:00 to 16:00, which is normally a network traffic peak time. His results confirmed earlier research carried out by Labovitz *et al* [45], where fluttering was correlated with network load.

Paxson [41] also introduced a classification for stability: *prevalence*, which is the overall probability that a certain path might be encountered and, *persistence*, which is the probability that a path remains stable for a long period of time. He found that Internet paths are strongly dominated by a single path, ie, they are highly prevalent.

For estimating path stability, data related to fluttering and tightly coupled routers were not used for statistic analysis. After observing changes of path for some hosts, he found that the probability of finding a route change in a ten-minute time might not be negligible.

Paxson [41] found that two thirds of Internet routes persisted for days or weeks and for *long lived* routes, 90 percent of the routes persisted for at least one week. These results confirm [45] and [46], which have found that Internet paths are quite stable over time. Differently from Paxson, their analysis was based on the study of routing update messages for estimating path stability.

A recent report from CAIDA researchers, Huffaker *et al* [47], provides a similar analysis of path stability. Considering two stability granularities, the sequence of IP numbers in a path and the sequence of AS traversed by measurement packets, between 60 percent and 70 percent are IP stable while 90 percent are AS stable throughout the days.

Huffaker *et al* [47] also indicates that normally there is little correlation between the number of hops and the underlying transit infrastructure, e.g., two paths might have a similar number of nodes though completely different physical length. In terms of the number of nodes, *Traceroute* paths originating from sources located in similar regions might have a completely different number of nodes to the same set of destinations. However, in terms of median RTT, these *Traceroute* paths might have similar delays.

Their work also suggests the US Internet backbone as the major intermediary transit backbone for countries in Asia (eg, China-Hong Kong: 90.3 percent; Taiwan: 83.5 percent; Korea: 61.6 percent), Oceania (eg, Australia: 77.8 percent; New Zealand: 79.6 percent) and Latin America (eg, Peru and Chile: 97.8 percent; Mexico: 100 percent).

In terms of packet loss, Mattews & Cottrell [38] observe that due to the TCP algorithm, a packet loss higher than 3 percent considerably affects performance of TCP sessions. Depending on the type of application, packet loss might affect the user perception in a higher or lower scale. For real-time streaming applications in a lossy bandwidth-limited environment, unnecessary video (or audio) interruptions will occur due to TCP retransmissions. Similarly, if the UDP is used for transmitting a real-time streaming application, the lossy environment results in low QoS user perception due to loss of information. Lossy networks do not affect non-real time applications such as e-mail.

Moreover, if a reachable node has a link with high packet loss level, measurements might report it as unreachable due to statistical fluctuation. Mattews & Cottrell [38] observed this situation when links approach packet loss levels of over 90 percent. For more accurate analysis, they suggest analysing the behaviour of other nodes because network problems tend to affect a number of nodes. They also report that in high performance research networks unreachability is typically less than 1 percent, which is a reeasonable QoS value for unreachability.

Another way of analysing QoS of a network is the analysis of the RTT between two hosts. The RTT parameter provides aggregated delay (transmission, propagation and queuing) measurement estimation for direct and reverse paths between two sites on the Internet. For an uncongested path, the minimum observed delay is the propagation delay and there is low variation of delay among measurement packets - which results in measured delays approximately equal to the propagation delay. For a congested path, Kalidindi & Zekauskas [39] found that at least one of the measurement packets is expected not to experience congestion. They also observed that step variations in the minimum delay often suggest routing changes. Finally, their work suggests that healthy Internet paths should work at packet losses below 1%.

Since many Internet paths are asymmetric, one-way measurements allow better measurement estimation for both delay and packet loss in each direction. Still if direct and reverse paths are symmetric, there might be differences in load in both directions - which result in dissimilar performance. This situation is particularly important for trans-continental links where there are dissimilarities of traffic in both directions.

Asymmetries might be a result of link problems, link costs or "hot potato" routing, which is a common routing practice among BSP [48]. For example, suppose that a client from a *BSP A* requests a homepage that is stored in a *BSP B*. The *BSP B* identifies that the request was originated from a client outside its network and, therefore, it forwards the homepage data traffic at the closest peering point between BSP A and BSP B. "Hot potato" is the result of a lack of standard mechanisms for reimbursing Internet providers for carrying third parties' data.

Paxson [41] verified that 49 percent of paths were asymmetric in terms of packets visiting a different city and 30 percent were asymmetric in terms of packets visiting different AS. He also observed that there was a high variation of asymmetries depending on the host analysed. For example, while for one host AS asymmetry was around 84 percent, for another it was 7.5 percent.

Claffy *et al* [49] were among the first researchers to demonstrate that measurement of RTT was insufficient and sometimes a misleading method for estimating one-way delays. Their work provided a study of *variations* in one-way delay between the United States, Europe and Japan. While they pointed to difficulties in measuring absolute differences in asymmetric paths due to a lack of synchronised clocks, *variations* in delay could be obtained because this variable does not demand clock synchronisation.

Kalidindi & Zekauskas [39] measured uni-directional properties of a path such as one-way packet loss and one-way delay. Their study reported higher congestion for US trans-Pacific paths than US trans-Atlantic links and higher congestion for traffic departing the US to overseas. This latter result is justified due to the US content centric nature of the Internet resulting in higher traffic demand for links departing US.

The usefulness of uni-directional measurements was also shown in some reports. Having deployed a Surveyor [39] machine at the Singapore Advanced Research and Education Network (SingAREN), Cheng at all [50] detected a drop in performance in SingAREN's overseas link going to Japan while the reverse link was working properly.

2.3.2.4 Deployment Difficulties

While active measurement tools provide a good overview of end-to-end network performance, some deployment problems were identified, for example:

A. Protocol inefficiencies:

- Ping echo requests might result in more than one echo response with the possible consequence of incorrect measurements due to incorrect association of echo requests and echo responses. Matthews & Cottrell [38] observed duplicate responses in long delay links (up to 94s);
- ICMP echo requests (e.g., P*ing* & T*raceroute* packets) might be used in certain kinds of security attacks and therefore network administrators might block or give low priority to such traffic. Matthews & Cottrell [38] reported deterioration in performance from North American sites to Scandinavian sites due to the installation of *Smurf filters* in the connection link. These filters give low priority to ICMP echo requests however, they do not affect TCP and UDP traffic. Therefore, a measurement based on ICMP packets might suggest a poorer network performance than it actually has;
- IP addresses are not easily mapped[1] into other useful analysis parameters such as: AS, countries, router equipment (multiple IP addresses for the same router) or geographic location coordinates [20].

[1] A recent report from Moore *et al* [51] has introduced a Web tool – *Netgeo* - that maps IP addresses, domain servers and AS numbers to geographic locations. The main facilities of Netgeo are: city granularity lookup, wide range of Whois server sources and recent query result caching for better server response.

B. Implementation Difficulties:

- Paxson [19] reported a number of situations that create difficulties in administering a complex measurement infrastructure, for example: system crashed and failed to restart; daemon configuration accidentally overwritten; updates required for Network Probe Daemon (NPD) or measurement software; packet filter misconfigured; disk space exhausted; firewall re-configurations. In the same report, he suggests some ways of minimising the maintenance problems: make the elements of the measurement infrastructure as homogeneous as possible; built-in self diagnosis at each NPD; remote NPD access for managing the system; develop measurement procedures and analysis for dealing with inevitable outages;
- Impossibility of setting an intermediate measurement point between two measurement points: because IP datagrams traverse different AS, sometimes cooperation with an intermediate network provider is desirable to identify performance problems. However, in most cases an agreement for setting an intermediate probe is not feasible in a short-term period [39]. Paxson [41] reported difficulties while measuring the end-to-end stability of a path with *Traceroute*. While end-to-end measurements can detect problems in a network, identifying the reasons for such problems demands contact with those running the network. For larger measurement infrastructures, such an approach is not adequate due to the excessive number of problems;
- Implementing one-way measurements with ICMP demands the installation of a GPS at each measurement machine [39] or a hybrid GPS-Network Time Protocol (NTP) [52] for time synchronisation. While the GPS approach is cost prohibitive [50], the hybrid approach might incur in errors due to RTT delay propagation among clients and primary timing servers [52];
- For small geographical regions, processing delays relative to TCP measurement machines can make a high contribution to the results [50]. For minimising the processing delay, TCP protocols might be implemented direct in an Ethernet card with sufficient processing capacity. While such implementation might significantly reduce the processing delay in not having TCP packets flowing through the PCI bus in the

measurement machine, a large-scale implementation of this measurement device is currently cost prohibitive. As suggested by Cheng *et al* [50], a TCP measurement delay with compensation for processing delay might be a solution. However, such an approach is adequate only when both client and server machines are accessible for time stamping Ethernet frames. If the focus of research is to analyse commercial servers on the Internet, researchers are not able to measure processing delays in the remote server;

- Cleary *et al* [33] provide a comparison of RTTs based on a single measurement machine approach –the machine runs the PING and also *snoops* the RTT echoes from the network- and on separate machines. The results show that the packet generation process can interfere with the measurements and the collected data might differ by up to 30 ms;
- While some infrastructure projects have an adequate interface for visualising the results [20, 37,39], others are still developing it [35].

In order to improve measurements, some research groups are engaged in designing new analysis methodologies for active ICMP based measurements. Ohta *et al* [53] suggest a mechanism for using ICMP messages associated with Internet hierarchical addressing architecture to identify the location of faults in a network. By aggregating similar ICMP messages and gaining Internet Routing Registry (IRR) [54] information, network administrators might have better value added information for taking decisions in routing policy and filtering.

Other research groups are involved in developing a protocol for active measurements. The IP Measurement Protocol (IPMP) has been developed for eliminating the measurement limitations imposed by ICMP. As described in [55], IPMP allows Bit Error Rate (BER) measurements, forward and reverse path measurements of a single packet and precise RTT measurements. Moreover, IPMP lessens measurement overhead on the network.

Finally, some researchers prefer active measurement tools rather than passive ones because the latter raises privacy and security issues [19].

2.3.3 Control Monitoring

The control monitoring approach is based on the analysis of routing and/or management information of the network. While the focus of the routing approach focus is on the analysis of BGP data, the management approach centres on the monitoring of the network via Simple Network Management Protocol (SNMP). The following sections describe both approaches.

2.3.3.1 Routing Approach

2.3.3.1.1 Overview

The Internet is based on several AS which exchange reachability information on their boundaries via peer border routers running BGP (See Figs. 2.1 & 2.4). BGP is an incremental protocol that only forwards routing information updates in the presence of network topology changes or routing policy changes. Policies are configured manually on peer routers as a result of economic, political and security considerations.

BGP updates can be classified into two types: *announcements* or *withdrawals*. While announcements happen when a router finds out about a new network topology or has new routing policies implemented, withdrawals occur when a router identifies a network that is no longer reachable. BGP updates are distributed to a router's peer or neighbour routers and each update might have multiple announcements and withdrawals. For testing link availability, BGP sends *keepalive* messages periodically –normally 30s - to its neighbour routers.

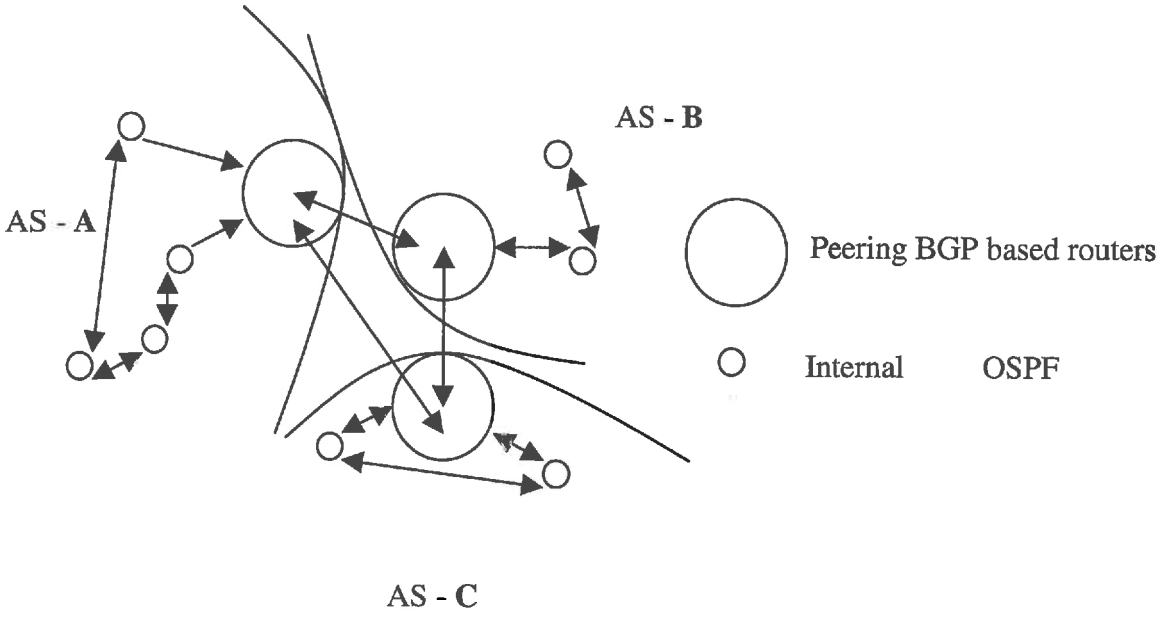


Figure 2.4: BGP Strategy: Peering routers from adjacent AS exchange routing information upon changes in routing policies or network topology

While BGP is an incremental protocol, intra-domain protocols such as the Open Shortest Path First (OSPF) gateway protocol periodically multicasts LINK STATE UPDATE messages to their adjacent routers.

The routing measurement approach is based on the analysis of BGP routing table updates at peering points of AS, which provide relevant data about Internet inter-domain routing behaviour.

Basically, there are three types of inter-domain routing updates:

- *Legitimate instability* which is due to legitimate changes in network topology;
- *Routing policy* which is the result of traffic policies due to strategic considerations;
- *Pathological* which is the result of redundant BGP information.

Most of the research about routing stability has been developed based on BGP routing updates. As discussed in item 1.2.4, high levels of network instability result in poor end-to-end network performance due to packet loss, increased time to network convergence and CPU/memory overhead in routers – which results in higher queuing delays. In a higher scale, network instability might result in loss of connectivity of wide area networks due to "route flap storms".

A route flap storm is a pathological oscillation that may occur in overloaded networks. In such a situation, overloaded routers do not reply to BGP keepalive messages, resulting in unreachable paths. Peer routers will choose alternative routes and will forward routing updates to their peers. When overloaded routers recover from their overloading status, they will retry to establish BGP sessions with their peering. This situation will lead to an increase in network load, resulting in a routing instability avalanche over larger portions of the network. Newer router versions provide higher priority to BGP traffic, which maintains router reachability in such circumstances of instability.

2.3.3.1.2 Related Work

The first well-known study about general Internet routing stability was carried by Chinoy [46]. Through the investigation of Exterior Gateway Protocol (EGP) updates in the NSFnet backbone in 1992, he analysed reachability of AS. While his data collection did not last for a long period of time (12 hours), he provided some interesting conclusions:

- 90 percent of EGP updates contained almost no new information, which results in the waste of routing processing/memory and link bandwidth;
- By analysing the number of networks affected by the problem, he could estimate the hierarchy (stub, multi homed or transit AS) of a network problem. Most of the problems were located in stubs, indicating high overall Internet stability;
- In terms of network time unreachability, disruptions would vary from a few seconds to a number of hours. While short periods suggest self-correcting problems such as temporary physical connectivity problems or routing inefficiencies due to excessive

network loads, large intervals are the result of more complex problems – which normally require human intervention;

- He identified problems with Intra-AS routing protocols because routing update delays used to take three orders of magnitude greater than the average packet delay for crossing the entire NSFnet backbone.

Govindan & Reddy [56] provided the first well-known study about inter-domain (inter-AS) topology and its growth. Route stability was also analysed. The work was carried out based on traces of routing updates collected at a large ISP and at a popular Internet exchange point. Data collection ran during three different 21-day segments (*snapshots*). The 21-day segment was found to be the optimum collection time for not having significant increase in the number of domains, links and prefixes – which would skew the samples. Moreover, because it is generally accepted that reachability information at backbone routers covers a significant fraction of the Internet (around 95 percent), they assumed that reachability to a new Internet prefix would be found at a backbone router.

They focused their analysis in three inter-AS characteristics: degree distribution, diameter and connectivity.

- *Degree Distribution* is the number of data traffic exchange agreements for one AS and it provides an approximate measure of its size. Based on this parameter, they found that there are 4 different domain sizes, which are summarised in Table 2.1;

Table 2.1: Domain Sizes Classified by Degree Range

Class	Degree Range	Fraction of Domains in Class	Types of Domains
C1	>= 28	0.9 %	National or International backbones
C2	10-27	3.1 %	Large US regional providers and European national networks
C3	4-9	9 %	Smaller regional providers and large metropolitan are providers
C4	1-3	87 %	Smaller metropolitan area providers and multi-campus corporate or academic networks

They found that 75 percent of the AS have a degree of 1 or 2. Comparing two different snapshots, the overall fraction of domains having a certain degree has remained approximately the same, which suggests that Internet has been growing laterally, ie, new domains are added to each class without changing the overall degree distribution.

- *Diameter* is the inter-domain topology distance, which is the maximum number of hops from one domain to any other domain. Their research found that Internet diameter had remained constant for the different snapshot periods. However, since the collected data does not show information about router-level diameter (ie, the number of nodes a packet has to traverse for moving from one domain to another domain), they could not provide information about it;
- *Connectivity* between domain classes provides information about AS interconnection hierarchy, ie, information for evaluating if domains are connected to domains in classes immediately above and below their own class. The main findings are:

    - Connectivity between domains is non-hierarchical;
    - Nearly 25 percent of links are between C1 and C4;

- In spite of the increasing connectivity between providers, there was no greater redundancy of connectivity to destinations. This is probably due to routing policies set in bilateral transit agreements between providers.

In terms of route stability, Govindan & Reddy [56] analysed the reachability in terms of prefix *availability* and prefix *steadiness* for two different snapshots. The former is the time a prefix is reachable and the latter the average time a prefix is continuously reachable. The results found are:

- Prefix availability is high, where 90 percent of prefixes are available for more than 95 percent of the time;
- There was degradation in availability between the two snapshots. While in the first snapshot, 90 percent of prefixes are available for more than 99 percent of time, the second snapshot had availability dropping to 97 percent. The authors argue that this might be for two reasons: an increase in the number of routers and links to prefixes resulting in higher probability of link or router failure; increase in errors due to routing misconfiguration;
- 80 percent of prefixes have steadiness for more than 1 day. However, a decrease in prefix steadiness due to Internet growth is observed.

Labovitz *et al* [45] analysed BGP information exchanged between backbone service providers at the major US public Internet exchange points during a period of nine months. Their main findings were:

- Most of the BGP updates (99 percent) consist of pathological updates;
- All pathological updates were a result of problems in small service providers due to misconfigured routers and faulty new hardware/software;
- Part of the pathological updates were due to an implementation decision of a certain router vendor. When receiving topology updates, the router would transmit announcements or withdrawals to all BGP peers despite having previously sent the

peer an update for the route. In spite of this pathological behaviour, this vendor implementation was compliant with the current IETF BGP standard [57];

- Instability and redundant updates follow a regular pattern with a majority of updates having a periodicity of 30 and 60 s. This fact suggests problems with routing software timers, self synchronisation and routing loops;
- Instability and updates follow a network usage pattern, exhibiting daily and weekly cycle trends;
- No single AS was responsible for the majority of instability statistics;
- Instability is not correlated to the size of the AS;
- Instability is distributed over the Internet. No paths or prefixes dominate instability statistics;
- Without considering pathological updates, 80 percent of Internet routes would exhibit a relatively high level of stability.

Finally, it must be observed that while the routing approach can provide interesting data for studying path stability, the approach demands the difficult task of negotiating with ISPs for access to their backbone data. In the increasingly commercial environment of the Internet, the availability of strategic information, such as router information might limit the usefulness of the approach.

2.3.3.2 Management Approach

2.3.3.2.1 Overview

The management approach uses the SNMP for monitoring network performance. SNMP was developed as a set of rules and parameters for monitoring and managing a computer network and devices that might be connected to it (routers, bridges, printers, etc). Management stations which are computers running management utilities determine the management definitions. The managed nodes are capable of communicating their status information to the management stations via a management process called SNMP agent. An

agent is also responsible for informing its management stations about unusual events such as a node crash or reboot, a line congestion or interruption, etc.

The management process is done in a Master-Slave mode where management nodes carry the intelligence of the system, issuing commands and getting responses from managed nodes. Fig 2.5 shows a network with a SNMP based system.

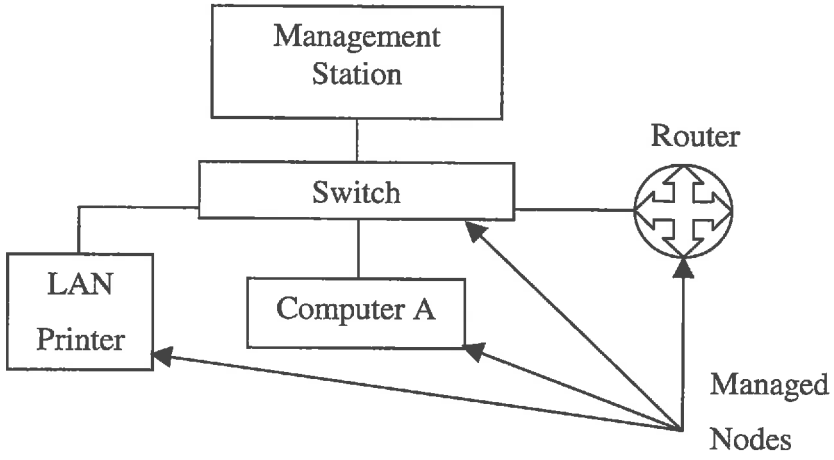


Figure 2.5: SNMP Model: A management station controls several managed nodes

2.3.3.2.2 Related Work

Very little research has been carried out based on SNMP. A recent paper from Kumar *et al* [58] described an SNMP based monitoring utility for obtaining traffic statistics and TCP admission control for QoS purposes in a WAN access link. The results show that adequate traffic control can be obtained within a WAN if an appropriate monitoring tool is developed and a TCP admission control based strategy is deployed.

2.4 Summary

In this chapter, a comprehensive discussion about Internet performance measurement research was provided. Firstly, parameters for estimating performance within a network

were described: aggregated traffic, packet loss, packet delay, path behaviour and throughput. Secondly, an overview of the most well known measurement methodologies was presented. Methodologies were classified based on implementation characteristics *as passive, active and control monitoring*. For each type of methodology, the main findings and implementation drawbacks were shown. Therefore, this chapter provides a summary of the state of the art of the Internet performance measurement field.

Chapter 3
Potential Bottlenecks For Internet Content Delivery

3.1 Objectives & Overview

This chapter provides an overview of potential bottlenecks for delivering Internet content in the current internetworking environment. The discussion of factors that affect performance is relevant for understanding the QoS limitations faced by Internet applications. Since the Internet provides a range of services with different QoS requirements and because the Internet is characteristically heterogenous in terms of transport mediums and protocol implementations, performance varies significantly over the network. QoS depends *not only* on a portion of the network *but* in the network as a whole. Therefore, when an end-to-end TCP/IP session is established between an Internet user and an Application Server, application performance will be highly dependent on the behaviour of the interconnecting networks. That is: even if the Application Server is capable of managing a broadband session and its network is congestion free *and* the end user is connected through a broadband access, not necessarily the end user will experience an adequate performance.

Limitations for a suitable content delivery will be discussed based on the TCP/IP reference model, which is similar to the OSI model except it *does not* have the Presentation and Session Layers (See Fig. 3.1). It is assumed that the reader has an adequate knowledge of the TCP/IP reference model and its layer characteristics.

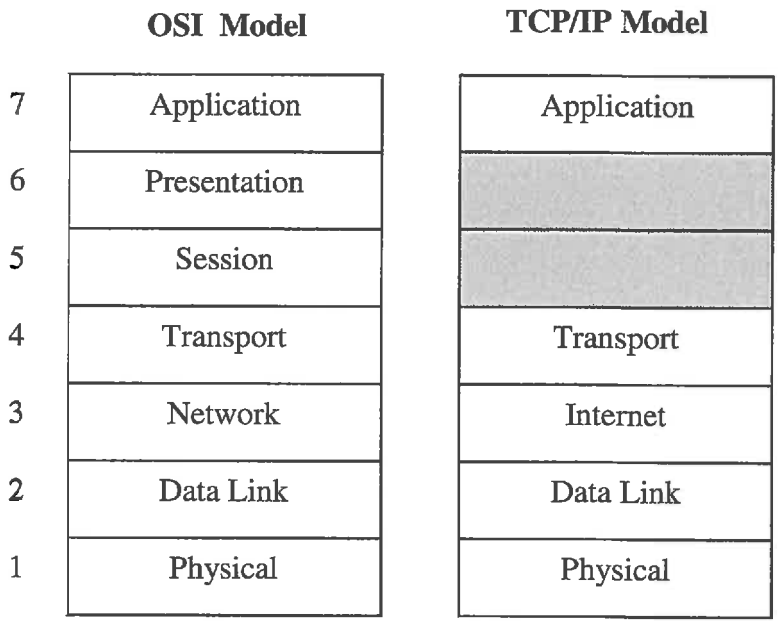


Figure 3.1: OSI x TCP/IP Models

The discussion will be based on a layer-by-layer basis, starting at the *physical layer* of the TCP/IP model and progressively covering the upper layers. This type of analysis provides a better understanding of all layer contributions on overall performance, allowing for the evaluation of particular problems within a layer and how these problems might affect higher layers.

It is important to note that this chapter provides a general discussion *of potential limitations* rather than a complete description of the field. For example, no details are given of how systems based on different technologies such as xDSL, HFC, LMDS and MMDS might affect the performance of multimedia applications. A complete description of those systems and their drawbacks would require further research.

3.2 Physical Layer

The *physical layer* provides the transmission medium for delivering information between two different sites. In a digital environment, the *physical layer* is responsible for transmitting a sequence of bits over a communication channel.

Physical layer characteristics might vary based on technical, economic and strategic decisions. For example, in geographic areas with low population density, satellite is normally the best solution for delivering digital content. This decision is supported by technical factors in addition to cost-benefit studies.

It is possible to distinguish the *physical layer* in terms of its network location: the access or the backbone network. The access network is the network covering the "last mile" between the Internet user and the AS which provides the Internet access (ISP). The most common access mediums currently in use are either twisted copper wire or coaxial cable. The backbone network has national or international coverage. The most common transmission mediums for backbone networks are microwave links, fibre optics & satellites.

Twisted copper wire is by far the most used access medium. It is the basis medium for the telephone system and it is also used for computer internetworking. The main problems related to copper are high attenuation & electromagnetic interference. Interestingly, it is common for non-telecommunications-literate people to associate the copper pairs as the main factor for the bandwidth limitations in Internet dial-up sessions[1]. Obviously the limitations are due to the bandwidth-limited voice channels, which were designed to carry voice instead of data[2]. New technologies such as ADSL (Asymmetric Digital Subscriber Line) & VDSL (Very high speed Digital Subscriber Line) provide throughput in the order of Mbps over copper wire while bypassing the voice channels [59,60].

[1] This is based on my personal observations while working for a Telecommunications provider.
[2] The human voice has characteristically a spectrum up to 3.4 KHz. Voice channel filters were designed to carry 4 KHz signals. For reconstructing a signal, it is necessary a sampling rate of twice the maximum frequency of this signal (Nyquist's sampling theorem). Therefore, for reconstructing a voice signal, it is necessary a sampling rate of 8 KHz. If the signal is encoded at 8 bits per sample, the maximum theoretical rate of a voice channel is 64 Kbps.

Coaxial cable is used as the "last mile" medium in Hybrid Fibre-Coax networks, which are implemented to provide cable television services. The coaxial shielding and its construction characteristics provide high bandwidth and noise immunity. A rate of 1 to 2 Gbps is realistic in cables up to 1-Km long [61].

Before the development of fibre technology, microwave was the main transmission medium for interconnecting networks in different geographic regions. The main drawback of microwave links is related to the high directivity of waves at 100 MHz or above, which demands line of sight between repeaters. While high towers diminish the need for signal repeaters, a careful cost-benefit study should be deployed for evaluating tower heights and number of repeaters. Moreover, microwaves are also affected by multi-path fading resulting in different wave propagation times within a link, which represents a serious problem for some systems. Multi-path fading is also dependent on weather conditions and frequency of operation.

Microwave technology has improved in result of high demand of data services. New developments also allow the use of frequencies in the order of GHz. These higher frequencies are used for delivering services such as MMDS & LMDS. Despite technological breakthroughs, limitations related to rain absorption and foliage attenuation are quite severe at these frequencies [62,63].

Fibre is used as the main backbone transmission medium. It has many advantages such as high transmission capacity, low attenuation and immunity to electromagnetic interference. The main drawbacks are its high deployment costs. This is probably the main factor for not popularising its deployment within the "last mile" such as Fibre To The Home (FTTH) and Fibre To The Building (FTTB), which would be the most appropriate access environment for multimedia delivery. It is also important to note that new technologies for multiplexing wavelengths within a fibre link (DWDM and ultra-DWDM) are increasing the bandwidth capacity of a single fibre up to Terabits per second.

Satellites are the most cost-effective medium for transmitting data to isolated regions where low population density does not justify the deployment of fibre or microwave links. In terms of satellite orbits, there are three different types of implementation strategies: Geo-synchronous (GEO), Low Earth Orbit (LEO) and Medium Earth Orbit (MEO) satellites [64]. LEO and MEO satellites are non-geostationary satellites and are designed for delivering mobile digital services. GEO constellations are the most commonly deployed, delivering a wide-range of services such as telephony, GPS, Internet, video broadcasting, etc. The main drawbacks of GEO satellite links are their high bit error rates and long propagation delays. These problems are extremely relevant for Internet delivery due to the flow control mechanisms of TCP, which incorrectly attribute the packet losses and delays as a signal of network congestion. This will be further discussed in Section 3.5.

3.3 Data Link Layer

The *data link layer* is responsible for providing an adequate service interface to *the network layer*, grouping bits within the *physical layer* into frames, controlling flow rate of frames and correcting transmission errors.

The basis of the *data link layer* operation is to link the *network layer* of two successive machines which are physically interconnected. In an Internet environment, the *data link layer* organises IP packets into frames, which are transmitted through one of the physical mediums described in Section 3.2.

The MTU of frames varies in size depending on the networking technology. For example, Fibre Distributed Data Interface (FDDI) has an MTU of 4500 bytes; ATM Adaptation Layer 5 (AAL5) over Asynchronous Transfer Mode (ATM) has an MTU of 9000 bytes; Fibre Channel and the High Performance Parallel Interface (HIPPI) has a typical MTU of 65280 bytes; Ethernet, Fast Ethernet and Gigabit Ethernet has an MTU of 1500 bytes [65].

The MTU contributes decisively to a system's performance and its impact is observed both on server and network throughputs. These performance issues are even more relevant when

the traffic behaviour is intense and steady; *or* when the system has bulk transfer applications running.

In terms of server performance, four problems are identified [65]:

- Processing time: Large frames[1] are processed more efficiently than small ones. Since the time for processing frames is similar for any frame size within a particular server, large frames allow the processing of a higher amount of data in comparison to small frames;
- Transmitting frames: The processing time for building a frame header is similar for a large or a small payload. Therefore, the processing time involved in transmitting frames is independent of the frame payload size;
- Receiving frames: In most server implementations, Network Interface Cards (NIC) interrupt the server processes to inform on the frame arrival and to process the frame. The higher the number of interruptions, the higher the CPU utilisation. A recent study has showed a reduction in server CPU utilisation of 50 percent when using 9018 byte-sized Ethernet packets as opposed to 1518 byte frames. In the same study, throughput increased by almost 50 percent;
- Copying data to/from host memory: Sending and receiving operations are more efficient with large frames due to the "paging" structure of memory allocation. Normally memory pages are 4Kbytes long and sometimes 8 or 16 Kbytes long.

In terms of network performance, it is verified that larger frames allow larger IP packets and therefore, higher efficiency for routers and switches. Because IP packet headers do not vary, larger payloads mean more data is transmitted within a routing/switching operation. Moreover, it is important to note that in terms of bandwidth, larger IP packets improve network performance due to headers consuming proportionally less network bandwidth.
Some recent work has supported the idea that Gigabit Ethernet (and future Ethernet generations) should support frames larger than 1500 bytes, mainly frames of 9000 bytes, i.e., the *Jumbo* frames [66].

The reason for *Jumbo* frames having 9000 bytes is justified for three reasons. The first one is related to the 32-bit Cyclic Redundancy Check (CRC-32) algorithm, which is used for detecting errors within Ethernet frames. The CRC-32 works with the same accuracy for frames between 376 and 11455 bytes. In addition, for efficient memory management, frame sizes should be a multiple of 4Kbytes, which is commonly used by the majority of commercial systems. Lastly, an adequate frame size should maximise efficiency of most common Internet applications. Since Network File System (NFS) is a common file-share protocol in UNIX environments and NFS datagrams are 8400 bytes long, frames should accommodate this type of datagrams.

A recent theoretical study has also showed that larger frames also improve performance of WAN systems. Mathis *et al* [67] has calculated the estimated throughput in a TCP session based on the RTT, packet loss and Maximum Segment Size (MSS), as follows:

$$\text{Throughput} < \cong \frac{0.7 \times \text{MSS}}{\left(\text{RTT} \times \sqrt{\text{Packet Loss}}\right)}$$

Since RTT and packet loss in open Internet environments are parameters that are not easily controlled because normally TCP sessions depend on third party interconnecting networks, larger frames allow better performance. Obviously a large MSS within a TCP session depends on interconnecting networks having an adequate MTU - otherwise frame fragmentation will occur. While frame fragmentation is an important strategy for allowing different networks to coexist in an Internet environment, in some studies fragmentation was considered harmful to network performance [68]. Some of the problems are described as follows:

- Inefficient use of resources: an additional number of headers increases redundant information, wasting bandwidth; a higher number of packets increase routing complexity within nodes, which results in consumption of computational resources;

[1] For ease of discussing, frames and IP packets are considered as synonyms in this Section.

- Degraded performance in case of fragment reassembly problems: higher layer protocols (i.e., TCP) require an entire packet retransmission when a single fragment is lost in the packet reassembly process.

Another important issue while discussing the *data link layer* is the implementation of Forward Error Control (FEC) in satellite networks. Because satellite networks are prone to low signal-to-noise ratios, FEC improves the BER of satellite channels. While FEC is not able to guarantee information integrity in certain noisy environments (eg, rain fade noise, military jamming, etc), FEC is necessary to prevent TCP from interpreting data loss as a possible network congestion [69]. Therefore, the data link layer protocol deployed within a particular satellite network has a large influence on performance of higher layer protocols.

3.4 Internet Layer (Network Layer)

The *network layer* is concerned with transferring packets between two points within an internetwork. While the *data link layer* is responsible for transferring data (frames) between point-to-point connections, the *network layer* transfers data (packets) through a number of hops - i.e., it provides end-to-end connectivity. An adequate *network layer* provision should manage problems related to selecting the best interconnecting paths (i.e., avoiding congested paths and maximising load distribution along different links) and interfacing networks with different network characteristics.

For a long time, the *network layer* implementation has been a point of debate in the telecommunications industry. While some groups support a connection-oriented layer, others defend the adoption of a connectionless layer.

The connection-oriented approach is based on a connection set-up between *network layer* processes in the beginning of each connection request; the connection setup defines QoS settings, billing specifications and flow control characteristics of the connection. In terms of content delivery, the *network layer* delivers packets in sequence and performs error

check[1]. Therefore, in the connection-oriented approach, the *network layer* carries most of the complexity of the service delivery, guaranteeing as much reliability as possible. Due to the QoS negotiation during the connection set-up process, the connection-oriented approach is adequate for delivering real-time streaming content such as video and audio.

The connectionless approach is based on a non-connection set-up implementation. This implementation strategy assumes the network is an unreliable environment and therefore it transfers the responsibility of flow control, error check and packet sequencing to the *transport layer* within the end-points of the connection. The packet delivery is not sequential and thus packets might take different routes, arrive out-of-order or get delayed/lost before arriving at the destination point.

The Internet has a *network layer*, which is characteristically connectionless, the IP layer. The IP is essentially a best-effort protocol and it does not give any guarantees that a packet is delivered correctly. While this limitation does not affect the end-to-end reliability of the Internet because the TCP layer provides control mechanisms for adequate content delivery, other *network layer* factors contribute to low efficiency of the Internet as a whole. These factors are as follows:

- Routing complexity: Routers perform complex computations for delivering packets. For every IP packet, routers have to check the integrity of the header, always performing a checksum. The header checksum is recomputed in every hop because at least one field has changed in the IP packet header -the Time To Live (TTL) field. In addition, routers have to check the IP destination address and find the best route to the packet;
- Routing pathologies: As discussed in Chapter 2, routing pathologies [41] might affect end-to-end performance;
- Routers ignore the *type of service* field within the IP packet header: Therefore, current Internet routers treat all packets equally, independently of the payload content.

[1] In ATM networks error check is not performed by the ATM layer but by higher layers.

The new version of the IP protocol, the IPv6, addresses some of the limitations of the current IP protocol, the IPv4. Among its improvements, it is possible to point to:

- Simplification of the protocol, which improves routing procedures and reduces the routing tables;
- Identification of the type of traffic, prioritising real-time data in relation to other traffic types.

In addition, the IPv6 provides larger address space for future growth of the Internet, increased security features, IP roaming facilities and multicast deployment aids. For a further discussion about IPv6, reading [70] is suggested.

While it is not an objective of this report to further debate about connection-oriented network layers (since the Internet *network layer* is characteristically connectionless), one implementation has become widely deployed and demands a brief discussion: the ATM layer in ATM networks. Some studies have reported an increasing use of ATM as the backbone technology for different networks types; it is estimated that 60 percent of IP traffic is sent via ATM backbones [71]. Therefore, the performance of the ATM layer will have an important effect over Internet performance as a whole.

Since ATM networks were designed to operate over fibre optic links –i.e., high reliability links-, the ATM layer *does not* provide any error control mechanism for the payload[1] -the error check is provided by upper layers. The ATM layer is connection-oriented both in terms of the service it delivers and its operation characteristics. During connection set-up, a virtual circuit (a virtual channel) is established between the end-points and a class of service is defined for the connection. An aggregated collection of virtual channels forms a virtual path. Each packet (cell) flowing within an ATM network is 53-byte long and has a 5-byte header, which is comprised primarily of a Virtual Path Identifier (VPI) and a Virtual

[1] The ATM Header Error Control (HEC) mechanism allows a single bit error in the header to be corrected or multiple errors to be detected.

Channel Identifier (VCI). These identifiers have solely local significance within a network and are used for switching purposes. Cells are sent sequentially and re-ordering is not permitted. The main advantages and drawbacks of the ATM implementation strategy are:

- ATM cell characteristics - advantages: The fixed-length of ATM cells provide increased routing and switching performance in comparison to variable sized packets (e.g., IP packets). In addition, the local significance of ATM cell headers diminishes routing/switching complexity because routing/switching elements require lower knowledge of the network-interconnecting infrastructure. This results in more cost-effective pieces of networking equipment;
- ATM cell characteristics – disadvantages: Headers respond to approximately 10 percent of overhead within an ATM network;
- Connection set-up – advantages: The connection set-up reserves resources within a network, which results in adequate QoS provision;
- Connection set-up – disadvantages: The connection set-up negotiating phase consumes resources and delays data delivery. This drawback is non-existent in permanent virtual circuits – which are commonly used in backbone connections.

It is also important to note that since ATM networks have functionalities that are redundant with pure TCP/IP networks, an IP over ATM implementation has an inherent level of inefficiency. For further discussion about the efficiency of systems running IP over ATM, read [72].

3.5 Transport Layer

The main objectives of the *transport layer* are to provide reliable, efficient and cost-effective transport services to the *application layer*. This means implementing error control, sequencing and flow management for data transmission between the communicating parties.

There are several ways of implementing a *transport layer* within a communications system: in operating systems kernels, in network interface cards, in detached application processes or in libraries within the network applications.

The Internet has two major *transport layer* protocols: a connection-oriented (TCP) and a connectionless (UDP). Since it was discussed in Chapter 2 that most Internet traffic is characteristically TCP traffic and because our research was carried out based on TCP bulk file transfer applications, limitations related to UDP traffic or short-lived network-flows are not discussed in this section.

Interestingly, most of the TCP limitations are related to the recent year's development in the telecommunications/networking industry. Although TCP works correctly in networks with high transmission rates, TCP control algorithms limit performance of certain applications in a broadband environment. The main problems faced by TCP are discussed in the following sections.

3.5.1 Path Maximum Transmission Unit Discovery

Path MTU Discovery (PMTUD) is a mechanism used for determining the maximum packet size that an end-to-end connection can handle without requiring IP fragmentation. As already discussed in Section 3.3, fragmentation is inadequate due to the inefficient use of resources and poor TCP performance. In addition to avoiding fragmentation, PMTUD allows a better channel utilisation. While TCP congestion control algorithms manage the number of segments to be transmitted by a host, these algorithms do not define the

segments' size. As a result, PTMTUD allows TCP to send larger packets, which results in higher channel efficiency. The PMTUD works as follows:

- When a connection is established, TCP identifies the starting segment size as the minimum value found in either the MTU of the local network card or the MSS of the remote host. If the remote host does not specify an MSS, TCP defaults the starting segment to 536 bytes;
- After the starting segment selection, TCP sends IP datagrams with the "Don't Fragment" (DF) bit set in the IP header. In case one router within the network path has a network interface with a MTU smaller than the starting segment size, the router discards the IP packet and sends an ICMP "Can't Fragment" error;
- When the ICMP error is received, TCP decreases the segment size and retransmits it. In case, new ICMP "Can't fragment" errors are received, TCP reduces the segment sizes and resends them. This process runs until the adequate segment size for the connection is found.

Some problems were identified with the PMTUD mechanism. Among the best known, it is possible to highlight the following:

- Inadequate starting segment size in cases where the remote end does not specify its MSS: As discussed in Section 3.3, current network technologies have MTU values varying between 1500 bytes and 64280 bytes. When a MSS is not specified, TCP defaults the starting segment to 536 bytes - a short segment size that might have a negative effect on end-to-end performance;
- The black hole problem: Some routers do not work properly while sending the ICMP "Can't Fragment" error messages [73]. This is result of kernel bugs or router misconfiguration. As discussed in [74], PMTUD fails to identify the adequate segment size for the connection and the TCP keeps sending large packets. Since the sender does not receive any ICMP error messages, TCP never discovers that segment size should be smaller and therefore packets are lost in a

"black hole"[1]. Generally, this type of connection is terminated after 15 minutes due to a timeout;

- MSS advertisement based on PMTU: some systems use PMTUD determined values to advertise the MSS at the start of the connection instead of the MTU of the interfaces of the local and remote machines. This situation results in a MSS that is smaller than the real MTU of the machines. In long-running connections where path changes might occur, the artificially low segment size makes it impossible to probe later for a larger PMTU.

The main drawback in implementing a PMTUD mechanism is the delay in determining the maximum packet size that can be transmitted in a TCP connection without fragmentation. This problem is even worse in GEO satellite links because of their long propagation delays.

It is important to note that MTU caching implementations can be deployed to bypass the PMTUD process - therefore minimising protocol delays. However, such caching implementations are still under development.

3.5.2 Bandwidth-Delay Product and Long Fat Pipes

One of the most important unsettled problems of TCP is the implementation of automatic mechanisms for determining the Bandwidth-Delay Product (BDP) of a connection, i.e., the looping capacity[2] of the link between the sender and the receiver (in bps). The BDP is used for specifying the maximum TCP window size within a connection.

The *simplified* BDP is obtained by multiplying the RTT of an end-to-end connection by the maximum transfer rate of the least-capable hop of all links within this connection. While the RTT is obtained via *Ping* utility, there is no automatic procedure for obtaining the transfer rate of the least-capable hop. This information depends on users having previous

[1] For further information about the "black hole" problem, join the tcp-impl mailing list
[2] Capacity (bits)= Bandwidth (bits/sec) x RTT (sec)

knowledge of the interconnecting topology between the communicating hosts, which is a non-standard situation.

Some research has been carried out to develop methods for automatic BDP discovery. A recent work [75] suggests modelling a mechanism for identifying the bandwidth of the least capable hop. This mechanism could be a new type of ICMP echo-request/echo-reply pair or a new IP option class/number combination within the existing echo-request/echo-reply pair. Routers would process the BDP request/reply messages similarly to the existent ICMP echo-request/echo-reply messages except for intercepting messages and updating the next-hop-least-bandwidth fields with information about the next hop's maximum possible bandwidth. This process would provide bandwidth information of the least-capable hop for both the upstream and downstream directions. The information would be stored in a new kernel table indexed by destination IP addresses and it would be refreshed in adequate intervals of time.

Another method for determining the BDP uses the TCP congestion feedback for dynamically tuning the maximum window size of a TCP session. This method is called auto tuning and more information can be found at [76].

In addition to the research for developing automatic BDP discovery methods, another field of work has been the study of performance limitations of TCP in networks with large Bandwidth-Delay Product (BDP). Networks with large BDP are called Long Fat Networks (LFNs) and numerous performance issues are found within these networks, for example [77, 78]:

- An environment with high packet loss (e.g., satellite links) has a negative impact on throughput because the Fast Retransmit and Fast Recovery mechanisms of TCP [79] are not adequate for impeding the link to drain when multiple packets are lost from one window of data. Some researchers suggest simultaneous use of a Selective Acknowledgment (SACK) mechanism and a selective repeat retransmission policy for overcoming packet loss negative effects [80]. In the

original cumulative acknowledgment implementation, the sender only learns about a single lost packet after a RTT, which results in TCP loosing its ACK-based clock and therefore a drop in throughput performance. In the SACK implementation, the receiver informs the sender about all delivered segments and therefore the sender only re-transmits the segments that have *really* gotten lost. It is important to observe that for a non-LFN environment, SACK introduces higher TCP operation complexities *but* it does not improve system performance;

- The TCP window size, which is defined in the 16-bit field in the TCP header, is limited to 65535 bytes. While for short-duration data transfers the TCP window size does not limit performance [82], TCP window size has been identified as a limiting factor which impacts performance of various data transfers [83, 76]. In LFNs, because of the large BDPs, the sender stays idle for a long period of time waiting for acknowledgments – which results in low link occupancy. For a better link utilisation and increased throughput performance, such systems require a TCP window scale option, which increases the TCP windows to 32 bits. However, it is important to note that a larger window size increases the probability of more than one packet per window being lost, which results in low throughput performance due to inefficiencies of Fast Retransmit and Fast Recovery algorithms;
- Many TCP implementations only measure one RTT per transmission window. Since TCP requests the retransmission of segments that are not acknowledged within a Retransmission Timeout (RTO) interval, an improved systems operation demands a more dynamic RTO estimation. A method called "Timestamps" is introduced in RFC 1323 for increasing the number of RTT measurements within a TCP connection. This method allows almost all segments to be timed during a TCP session without extra computational complexity;
- The TTL field in the IP header, which is used for limiting the packet lifetime to 255 seconds or 255 hops, might be a problem in LFN: Since the TCP sequence number field is 32 bits long, the same sequence number is reused after 4,294,967,296 bytes. In Gigabit networks the sequence counter restarts after 34 seconds, i.e., before the TTL expiration deadline for the packets within the last sequence. In case a segment from the old sequence gets delayed and reappears later while the TCP connection is

still alive, this might result in improper operation of the TCP algorithm. One possible solution to the problem would be to increase the size of the sequence number field. Some researchers have introduced different solution like the Protection Against Wrapped Sequence numbers (PAWS), which works based on the "Timestamps" method [78].

Another important improvement of the TCP stack, which could substantially improve performance, would be the development of an auto-configure TCP stack based on the running application, buffer capacity and network conditions. Such an implementation would provide a more efficient TCP stack, providing a range of transport services in result of diverse QoS requirements. Semke *et al* [76] provides a study where TCP automatically adjusts its characteristics to buffer capacity and network conditions.

It is also important to observe that TCP performance is also affected by long-delay links as in satellite networks (not necessarily a large BDP link). In long delay-links, the slow start and congestion control mechanisms can result in low efficiency of available channel bandwidth due to senders being idle for long periods of time waiting for acknowledgments [81,24]. For a further discussion about current TCP algorithm developments over satellites read [26]. In addition, some organisations have been developing new satellite implementations where the TCP layer is substituted by other type of protocols within satellite links. For example, the SkyX protocol [84] has been used in commercial satellite implementations between terrestrial gateways and satellites. The SkyX protocol provides a streamlined handshake mechanism, selective retransmission algorithm for acknowledging received data, unlimited window sizes and full-link-adaptive-transmission-rates; these features result in faster connection set-up, low dependency on the BDP and increased end-to-end throughput.

3.6 Application Layer

The *application layer* provides the ultimate interface between users and multimedia services. While the lower layers are responsible for delivering digital information in a reliable fashion, the *application layer* introduces the service environment in a user's point of view. From a software design perspective, the *application layer* should provide a user-friendly interface, allowing users to control applications without demanding special knowledge of their working functionality. A good example of a successful *application layer* design is the WWW browsers: the Internet popularisation coincides with the development of pieces of software (browsers) for improving the user experience while searching for digital content. While the WWW creation was important for introducing the concept of web-linked pages, browsers were the key-strategy-element for popularising the WWW.

However, it is important to note that browser implementation strategy might vary and hence the end-user perception of the WWW. For example, [85] reports an experiment where a dial-up client was configured with a particular browser for accessing a homepage and downloading some documents within the homepage. The results collected suggest maximum performance is obtained when browsers are configured to open four TCP connections simultaneously. However, while a multiple-connection implementation might appear a good solution for improving end-user performance, it also has drawbacks. The same report observes that such an implementation increases the probability of overflowing the server's incomplete connection queue, resulting in higher response delays.

The adequate implementation of a server's application layer also affects decisively the throughput and response performance. For example, a system with *proxy caching* decreases latency because normally clients are located closer to the *proxy* servers than to the content providers. Moreover, *proxy* caching usually minimises bandwidth usage because clients retrieve information from a closer site - without needing to traverse several links of the public Internet. Interestingly, a *proxy* will not always provide performance improvements. A recent study from Feldmann *et al* [86] shows that a bandwidth mismatch

between the client and the server and, the proxy and the Internet, can result in traffic increase from content providers to the ISP. This situation occurs when low speed end-users (e.g., dial-up users) abort a document request and by the time the proxy learns of the transfer abortion it will have already downloaded a large amount of the document from the content provider. The same study shows a mean improvement of 47.5 percent and 24 percent in latency for broadband users and dial-up users, respectively, when data caching and connection caching[1] is used. For a further discussion about the state of the art of *proxy* caching, it is suggested reading [87].

The throughput performance of WWW servers has been the center of discussion *both* in the economics of delivering Web content *and* in the technical evaluation of Internet servers. A recent paper from [88] has found that slow WWW download speeds have a high impact on the performance and productivity of Internet transactions. The research suggests the volume of data within an Internet homepage - i.e., the visible Internet throughput for an end-user – has a high correlation to users aborting the homepage transfer. The study estimates a loss of US$ 362 million per month in e-commerce transactions in the US due to inadequate throughput performance of Web applications.

In terms of performance of WWW servers, the main discussion has been associated to the analysis of memory overhead on Servers due to the TCP TIME-WAIT state. While it was possible to discuss this limitation as a *transport layer* bottleneck, the TCP TIME-WAIT state is a consequence of *application layer* protocols (such as HTTP and FTP) requesting the end of an Internet transaction by closing the transport connection.

TCP TIME-WAIT state results from TCP's method for isolating old connections from new ones. The method establishes that the endpoint closing a connection should isolate the host/port addresses used for the old connection for a period equivalent to twice the Maximum Segment Life (MSL) –i.e., the longest period a packet can be undeliverable within a network. This mechanism prevents packets in transit from an old connection to

[1] Connection caching is a proxy implementation that has persistent connections between the proxy server and the content providers. These connections are re-used for obtaining documents for multiple clients.

arrive in a host/port pair that is already in use by a new connection between the same hosts – what could result in TCP malfunction. In order to block the host/port pair, the endpoint closing the connection keeps a copy of a TCP Control Block (TCB), what indicates a recent connection termination. While keeping the TCB, the connection is in TIME-WAIT state [89].

For HTTP applications, the server is normally the endpoint entering the TIME-WAIT state [90]. In the case of FTP, FTP clients understand a connection being closed gracefully as a signal of a successful file transfer [91] – i.e., the server closes the connection.

A recent study of Faber *et al* [92] provides an extensive discussion of TIME-WAIT state and its effect on heavy loaded servers. Their study reports that in particular cases TCB load in the server decreases throughput performance by as much as 50 percent. Three possible protocol modifications were proposed for transferring the TIME-WAIT state to the clients, which would improve performance on busy servers. These alterations are described as follows:

- A new TCP option, TW-Negotiate: During the three-way handshake while establishing a TCP connection, both hosts would indicate which endpoint would enter the TIME-WAIT state after closing the connection. In the case of one of the hosts not supporting TW-Negotiate, the TCP handshake would be without the TW-Negotiate option. The main advantage of such an implementation is in isolating the application layer from the implementation details of the transport layer; the main drawback is the difficulty in changing the TCP stack, which would require significant experimental analysis before becoming an accepted standard;
- TCP solution based on the exchange of the TIME-WAIT state after a connection closure: This strategy changes the TCP implementation within the client. The client's modified TCP would send an <RST> packet to the server after a successful passive close[1] and put the client in TIME-WAIT state. The <RST> packet would be

[1] The endpoint that closes the connection performs an active close; the endpoint receiving the closure request performs the passive close.

used for removing the TIME-WAIT state from the server. The main advantage of such an implementation is the compatibility with current protocol specifications, which would facilitate widespread deployment in a short period of time; the main drawback is changing the meaning of the <RST> packet, which is normally used for aborting a connection *or* for signalling that a segment has arrived and does not appear to be correct for the connection;

- Modification to the HTTP: While the early versions of HTTP would understand a closing of a TCP connection as an end of a transaction, new versions of HTTP [90] decoupled the end-of-connection and end-of-transaction indications as a way of supporting multiple transactions over a TCP connection – i.e., persistent connections. The suggested alteration to the HTTP is based on this transaction/connection decoupling. The client would notify the server that the connection has been closed through a CLIENT_CLOSE request, which would be a new extension to the HTTP protocol. The main advantage of the HTTP approach is requiring modifications exclusively on the client side; the drawback is that TIME-WAIT loading resulting of other application protocols will not be covered by these modifications. While HTTP is currently the main source of Internet traffic, this situation might change due to the Internet dynamics.

Another field of research within the *application layer* has been the development of new compression formats. In the past few years new compression formats like *ZIP, MPEG, WAV* and others have become common acronyms among the Internet community. More recently, a new software extension entitled *DivX* has become the focus of attention because it entitles video files to be compressed as much as 85 percent [93]. *DivX* video files can be transmitted over the Internet as files rather than streaming using a bandwidth of 750 Kbps - a feasible speed for clients connected with broadband technologies such as ADSL, cable modems and others. Probably, the unique pre-requisite for guaranteeing file (i.e., video) download at those speeds would be the implementation of *mirror servers*[1] closer to the clients, which would by-pass potential bottlenecks of the public Internet.

[1] Mirror servers are replicas of content servers located in strategic points of a network (e.g., the Internet) for improving the reliability and performance of file requests.

The design and implementation of *mirror servers* have also become an important area of research. A recent study from Myers *et al* [60] has reported a large variation in performance among some mirror servers; on occasion throughput was found to vary two orders of magnitude when comparing the best and worst servers. In addition, their study has shown that only a few servers can provide near-optimal performance to a client - and therefore clients should only consider a few servers out of a group of servers while retrieving files.

In addition to the improvement in performance, an adequate mirroring implementation has other advantages: content providers can personalise the information to be provided to a group of clients – a key-element for e-commerce success within an increasing globalised market. Finally, mirroring is also a new market niche for delivering time-sensitive interactive applications such as stock trading [94].

3.7 Summary

In this chapter an *overview* of potential bottlenecks for Internet content delivery was provided. The discussion was based on the TCP/IP model and it was organised in a layer-by-layer basis. For each layer, a summary of working characteristics was shown and performance limitations were described. In addition, some operation characteristics of a particular layer that might influence performance of other layers were described (e.g., satellite links affecting end-to-end performance due to TCP slow start and congestion control algorithms).

Chapter 4
Methodology

This chapter describes the methodology used for collecting and analysing the measured data. Based on a client-server Internet connectivity model and some experimental assumptions (See Sections 4.1 & 4.3, respectively), software utilities were developed for automating download sessions, calculating throughput performance and measuring virtual Internet path[1] characteristics such as RTT, DNS and IP addresses.

Throughput analysis provides an *application layer* measurement perspective, which is useful for previewing end-user interaction. The analysis of RTT, DNS and IP addresses for a particular virtual path provides a better understanding of how Internet path dynamics and different AS contribute to overall path performance. While it was difficult in some cases to match network and application layers results –due to the number of uncontrolled variables within an Internet network environment and their effect over TCP working behaviour-, such an approach provides an extra analysis facility. Some findings are provided in Chapter 5.

Three performance models were developed based on the analysis of measured data and the client-server Internet connectivity model:

- Throughput Model: provides a model for estimating throughput performance behaviour, which depends on interconnecting networks within a virtual path and remote server load conditions;
- Path Instability Model: introduces a variable for estimating path instability, based on changes of IP/DNS node addresses within a virtual path;

[1] Virtual Internet path can be understood as an application session that is established between two hosts connected to the Internet. In this study, a virtual Internet path (or *virtual path*, for ease reading) will always refer to a virtual client-server Internet path.

- Minimum RTT delay Model: provides a study for estimating path characteristics based on the minimum RTT delays of interconnecting networks along a virtual path.

This chapter has five sections. In the first section a client-server connectivity model is introduced, which is the basis for analysing the collected data. Path instability, throughput and minimum RTT delay models are introduced in the second section. In the third section, data collection procedures are shown, including a description of the software utilities that were developed for collecting data. The fourth section provides a statistical methodology based on the Internet connectivity model and some analysis assumptions. In the last section, a summary of the main highlights of this chapter is provided.

4.1 Client-Server Internet Connectivity Model

For ease of analysis, the Internet environment can be summarised as a collection of AS which vary in size, geographical coverage and function (See Chapter 2). A client-server model based on AS connections provides an overview of virtual paths through the Internet and this model can be used for better understanding the influence of different AS on overall path performance. A client-server Internet connectivity model is shown in Fig. 4.1 where its main elements are described below:

A. **Client Machine.** It is a computer running a standard Operating System and having dial-up or dedicated line access to the Internet. For our study, clients ran NT 4.0 Operating System, had similar hardware configurations (Pentium type systems) and were connected to the University of Adelaide Internet gateway via a 100 BaseT Local Area Network (LAN).

B. **Local Network**. It is a stub AS (See Chapter 2, Section 2.1) that provides Internet access to the client machine. Normally for a dial-up user such a network is a traditional ISP; for a dedicated line user a broadband ISP, research/commercial organisation or

BSP provides the connectivity. For our study the local access is via the University of Adelaide Internet gateway.

C. **Local/Remote Regional Network.** It is a multi homed AS, which provides connectivity to local networks. Depending on the local network geographic location (serving the Client or the Application Server), the regional network can be classified either as *local* or *remote*.

D. **Backbone.** A backbone is a transit AS, providing *regional, nation*al or *international* coverage. For our study, three types of backbone interconnections are defined: the Australian backbone, the US backbone and the remote server backbone.

E. **Remote LAN.** This network provides Internet connectivity to an Application Server.

F. **Application Server.** The Application Server is used for storing and delivering digital content such as Web pages and program files. File application servers can have mirror sites closer to users, resulting in better response time and higher throughput [95].

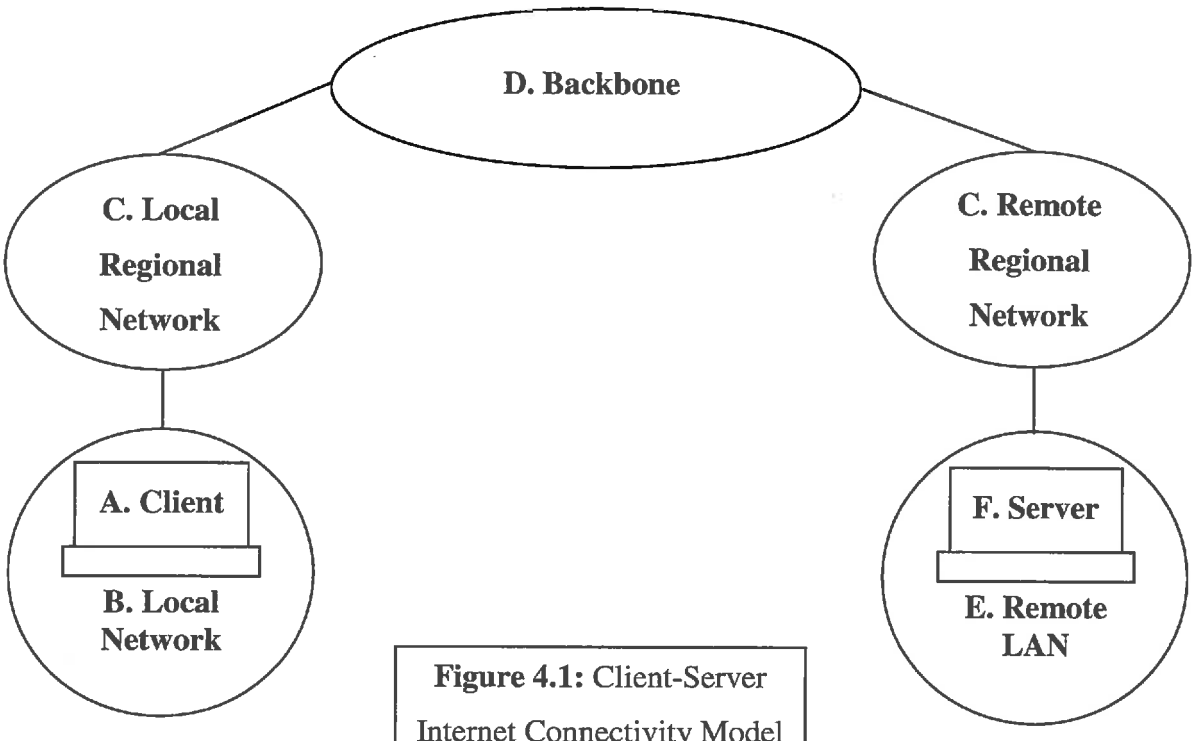


Figure 4.1: Client-Server Internet Connectivity Model

4.2 Throughput, Path Instability and Minimum RTT Delay Models

4.2.1 Throughput Model:

The Throughput rate for our model follows a function of six parameters:

$$\textbf{Throughput} = f\left(\alpha(t), \beta(t), \phi(t), \chi(t), \pi(t), \delta(t)\right). \quad (4.1)$$

Each element of the Equation (4.1) describes the network capacity of the defined areas of our connectivity model. Equation elements will vary over time (t) and are related to the QoS settings in each network. It is postulated that the throughput in each network element is inversely proportional to the traffic demand in that network element (See Section 4.1). The named elements of the equation then reflect our model and are

α : Local Network,

β : Local Regional Network,

ϕ : Backbone,

χ : Remote Regional Network,

π : Remote LAN,

δ : Application Server.

Backbone traffic conditions will vary being influenced by national and international backbone characteristics, both at the Client and Application Server sides. In Chapter 5, our results will also show that backbone traffic conditions are highly influenced by backbone traffic conditions in the USA.

For our study, it was assumed that bottlenecks will result from regional network, backbone or Application Server limitations. Therefore, Equation (4.1) reduces to

$$\textbf{Throughput} = f(\beta(t), \phi(t), \chi(t), \delta(t)) . \quad (4.2)$$

For the case where Application Servers are close to the Client and there are neither backbone nor regional network bottlenecks, Equation (4.2) reduces to:

$$\textbf{Throughput} = f(\delta(t)) . \quad (4.3)$$

4.2.2 Path Instability Model:

The analysis of IP and DNS characteristics over time for all nodes in a virtual path provides a good overview of path stability. This section suggests a new variable for estimating the stability of a virtual path. If N is a number of *Traceroute*[1] (TRACERT) samples collected and each sample has a variable number H of nodes per sample, then ϑ is the average number of nodes,

$$\vartheta = \frac{\sum_{i=1}^{N} H(i)}{N} . \quad (4.4)$$

Moreover, if there are W changes in path configuration during a period of time T, the path instability variable results in

$$\varepsilon = \frac{W}{\vartheta T}, \quad (4.5)$$

where ε is the number of changes of node configuration for a certain period of time T. The dimension of ε is $[T]^{-1}$. The software *Utility* 5, which is described in Section 4.3.8.2, calculates this variable. In our study, T was not considered for the calculation because the experimental analysis period was common for all paths.

[1] For ease reading, *Traceroute* is called TRACERT

Path instability (ε) is useful for comparing the stability of *different* virtual paths. Furthermore, through the analysis of changes in path configuration, it is possible to classify path instability as a result of *local, remote, regional* or *backbone* path changes.

4.2.3 Minimum RTT Delay Model:

A RTT delay equation for a certain virtual path is defined as

$$RTT_{Total} = RTT_{Propagation} + RTT_{Switching} + RTT_{Queuing}\ , \qquad (4.6)$$

where RTT_{Total} is the RTT delay from a Client to an Application Server, $RTT_{Propagation}$ is the RTT delay due to propagation of electrons and photons on a particular medium, $RTT_{Switching}$ [1] is the RTT delay for transmitting/switching a packet over a network interface and $RTT_{Queuing}$ is the RTT delay due to packets being delayed by congestion in routers.

A minimum RTT delay model provides an estimate of a best end-to-end RTT performance along a virtual path. Packets experience a minimum RTT delay when there *is no queuing delay* on routers along a virtual path, i.e., there is *no network congestion*. In this case, Equation (4.6) reduces to

$$RTT_{\text{Minimum}} = RTT_{\text{Propagation}} + RTT_{Switching}\ . \qquad (4.7)$$

Similarly to the minimum RTT delay model, it is possible to define a minimum One-Way (OW) delay model for end-to-end analysis of a virtual path. In this case, it is said that

$$OW_{Minimum} = OW_{\text{Propagation}} + OW_{Switching}\ , \qquad (4.8)$$

where

$$OW_{\text{Propagation}} = \sum_{\text{First link}}^{\text{Last link}} OW_{\text{Inter-city propagation delay}} \quad , \tag{4.9}$$

$$\text{OW}_{\text{Switching}} = \text{Typical Switching Delay} \times \frac{(2\text{N}-1)}{2} \quad . \tag{4.10}$$

N is the number of nodes within a client-server virtual path. For Equation (4.9), the One-Way inter-city propagation delay is calculated based on what transmission medium is used for interconnecting two cities. Fibre or GEO satellite links are the typical transmission mediums for backbones and regional networks. While for fibre the propagation speed is 200.000 Km/s, propagation delays on GEO links vary between 250 to 300 ms [21]. In our study, a One-Way GEO satellite propagation delay between two satellite-connected cities is asserted as 250 ms, independently of where geographically the connection is originated and terminated. For a fibre link between two cities, the propagation delay is calculated as

$$OW_{\text{Inter-city propagation delay}} = \frac{\text{Distance between cities (Km)}}{200} \quad , \tag{4.11}$$

where $OW_{\text{Inter-city propagation delay}}$ is expressed in *ms*.

The *Typical Switching Delay* (shown in Equation 4.10) is associated to a minimum typical path definition, which is introduced in Section 4.3.3. The Typical Switching delay is defined in Section 4.3.4.2.

Moreover, for *symmetric* upstream and downstream paths it can be assumed that

$$RTT_{\text{Minimum}} \approx 2 \times OW_{\text{Minimum}} \quad . \tag{4.12}$$

[1] $RTT_{Switching}$ is also known in the literature as $RTT_{Transmission}$

Furthermore, in the case where Client and Application Server are located at the same geographic region, Equation (4.8) reduces to

$$OW_{Minimum} = OW_{Switching} \quad . \tag{4.13}$$

4.3 Data Collection Procedures & Assumptions

Data was collected to preview the effects of network interconnections, Application Server traffic demand and time zone differences over file transfer sessions. As discussed, the parameters measured were throughput, RTT and path stability. A number of assumptions were made, which are discussed below:

4.3.1 The file transfer Application Servers (See Section 4.1, Fig 4.1: F) are common file repositories used by Internet users. They have a standard WWW interface and are located in several geographic locations worldwide. The popular TUCOWS[1] Web site was selected, which has a number of mirror servers on all continents. The file transfer connections used HTTP, which is the standard WWW transfer method. The selected mirror site regions are presented in Table 4.1:

Table 4.1: Location for TUCOWS Mirror Sites

Region A	Region B	Region C
South Australia, Australia	USA W. Coast	Argentina
Victoria, Australia	USA E. Coast	Brazil
Hong Kong	Germany	South Africa
Israel	England	Zimbabwe

[1] Tucows servers are configured based on a suggested hardware/OS configuration. During this study the suggested characteristics were: Pentium 133 or greater, UNIX OS, 32Mb of RAM or greater, 8Gig hard disk space, T-1 or greater bandwidth.

Regional classification was based on expected throughput performance for different geographic regions visible from Adelaide, Australia. Time zones were also an influence on the locations selected, as these impact performance because load on *both* servers *and* interconnecting networks depend on local traffic conditions in each region [95].

Three computers were used as clients to run the throughput experiment as shown in Fig. 4.2. In addition, Client 1 ran a software utility collecting paths to each application file server shown in Table 4.1 using the TRACERT[1] program. This utility ran in parallel to all sites and sampled respective paths every five minutes. The throughput data collection was carried out from Nov 3 until Nov 22 1999 for the first two clients while for the third client the period was extended until Nov 30 1999. The decision of extending the experimental period for the third client was due to the small number of throughput samples collected during the first experimental period. The path collection ran from Nov 3 until Nov 30 1999.

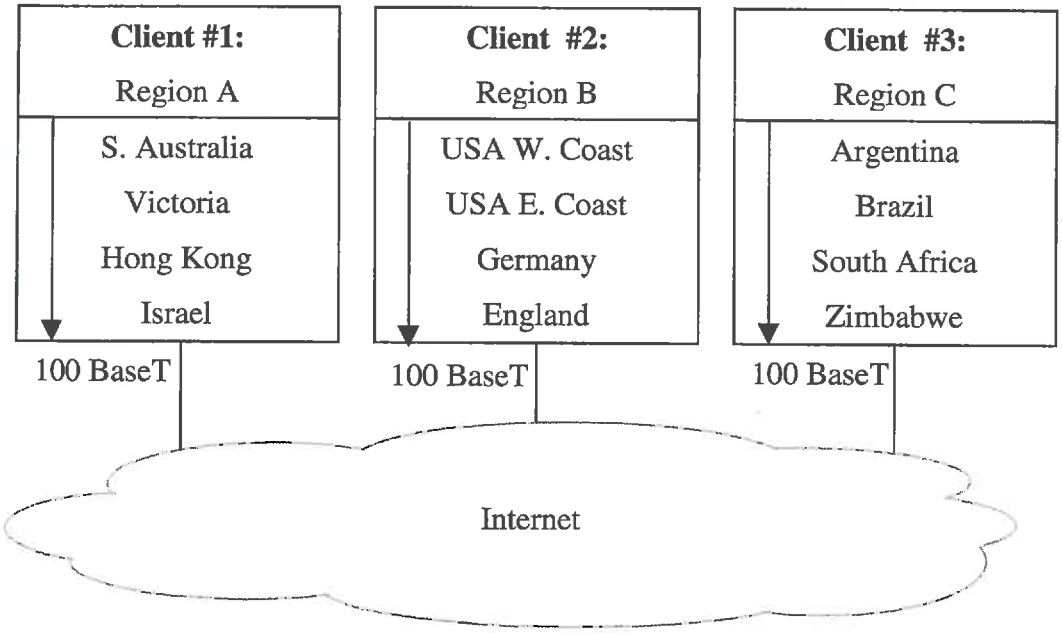


Figure 4.2: Clients and Target Application Servers

[1] For more information about TRACERT operation characteristics, read [43].

Client 1 was responsible for servers in three countries, Australia, Hong Kong and Israel, each country having a well-developed national network infrastructure and represented three time zones.

Client 2 was responsible for servers in two European countries, Germany and England, as well as the USA. The servers are located in three different time zones.

Client 3 was responsible for servers in countries with less developed national network infrastructure and with two time zone groups (Brazil and Argentina; South Africa and Zimbabwe).

4.3.2 TRACERT provides performance information (RTT, packet loss and IP/DNS addresses) for every node within a virtual path. Each node within the path is probed with three echo-request datagrams. For each echo-reply received, TRACERT calculates the estimated RTT and registers it. A TRACERT sample is shown below:

#3/11/99 0:04:58
Tracing route to ***www.ozbytes.net.au [203.152.224.5]***
over a maximum of 30 hops:

1 <10 ms <10 ms <10 ms vlan0180.atm2-0.pancho.net.adelaide.edu.au [129.127.180.253]
2 <10 ms <10 ms <10 ms lis255.atm1-0.central.saard.net [203.21.37.2]
3 <10 ms 10 ms <10 ms fa0-0-108.boomerang.internode.on.net [203.16.212.13]
4 <10 ms 11 ms <10 ms 198.32.240.100
*5 10 ms 20 ms * ser-1-0-bigpipe-grote-adl.bna.com.au [203.34.35.82]*
6 10 ms <10 ms 10 ms nostromo.senet.com.au [203.56.239.98]
7 <10 ms 10 ms 10 ms www.ozbytes.net.au [203.152.224.5]

The sample above provides information about the local time when the TRACERT is activated and the destination IP/DNS address that TRACERT is probing. For each node within a path, it is possible to obtain estimated RTTs and IP/DNS address details. In case echo-request datagrams are lost for a particular node, a star (*) is registered to indicate packet loss (e.g., see node 5 in the TRACERT sample).

Our observation shows that normally the total number of TRACERT echo-replies (RTT replies) for each node within a virtual path has a similar occurrence distribution. As an example, if there are one thousand RTT replies for node 1 during an experimental period of time *t*, it can be expected that all nodes within the same virtual path will also have approximately 1000 RTT replies.

By simultaneously analysing the number of RTT replies for each node in a particular client-server path and the node's IP and DNS addresses, *typical virtual paths* could be identified for each Application Server in relation to our Client. It was only considered typical virtual paths as the virtual paths that were persistent [41] and not short-lived.

While the total number of RTT replies for each node within a virtual path should be approximately the same, there are some routing topologies that should be discussed for a better understanding of this analysis: *tightly coupled machines & fluttering*. These routing topologies, which were briefly discussed in Chapter 2, were identified by Paxson [41] and they are as follows:

- **Tightly Coupled Machines**

Figure 4.3 shows a tightly coupled router network topology within a typical virtual path. When packets are flowing between nodes C and D, they might take one of the alternative *IP addresses IP1...IPN,* which are possibly collocated within the same premises. In terms of RTT replies, the *total number* of RTT replies from nodes *IP1...IPN* will be approximately the same of RTT replies within nodes C or D. This routing topology normally does not provide any drop in performance for end-to-end TCP sessions because nodes *IP1...IPN* have equivalent performance. Observe that a tightly coupled router topology involves only one intermediary node between node C & D.

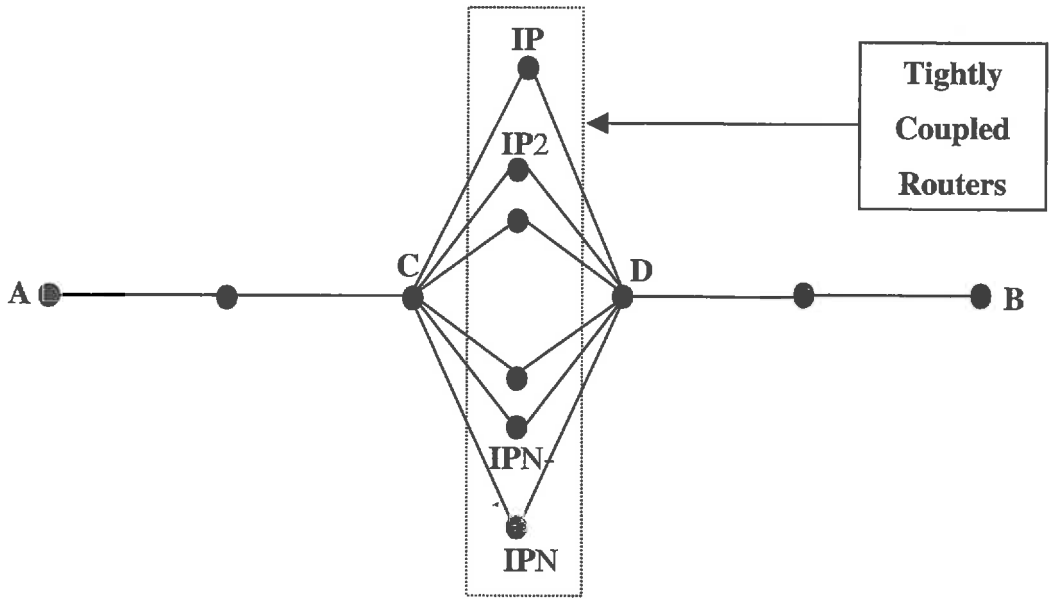


Figure 4.3: Tightly Coupled Router Topology

- **Fluttering**

Figure 4.4 shows the fluttering topology, where a router forwards packets to different networks[1] (*A...N*) in a way of balancing load in a network. As discussed by Paxson [41], fluttering might result in performance problems due to routing asymmetry and different packet propagation times. In terms of number of RTT replies, it is found a similar number of RTT replies for each node within a fluttered path. If one node is selected from each of the fluttered paths, the sum of the number of RTT replies from these nodes is expected to be similar to the total number of RTT replies within any node of the non-fluttered path (i.e., nodes A, B, C or D). Observe that a fluttering topology involves paths with more than one node between nodes C & D.

[1] In this study, these different networks are called fluttered paths.

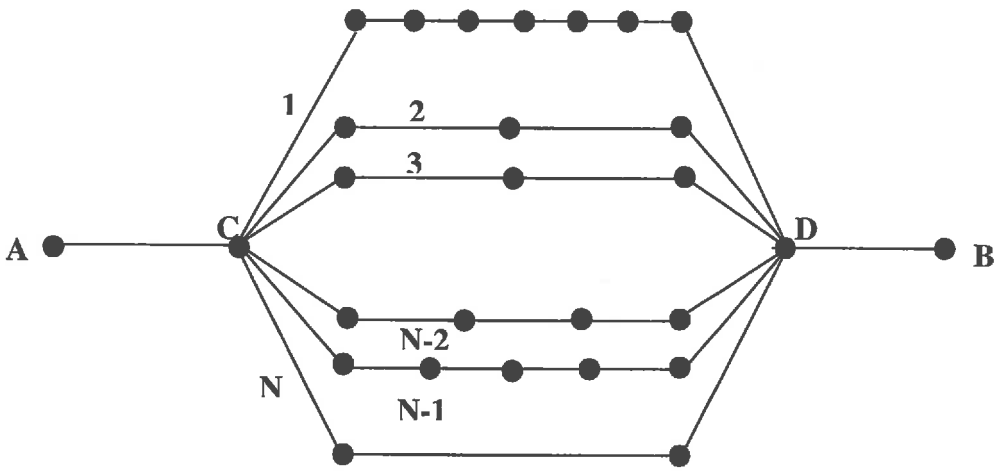


Figure 4.4: Fluttering Topology

Further details in relation to our typical virtual paths are provided in Chapter 5.

The main value of characterising a typical virtual path is that such an approach allows a better classification of nodes in accordance to the client-server Internet connectivity model introduced in Section 4.1. This provides a better understanding of each AS contribution on overall performance. Since AS are administered by different organisations with both dissimilar geographical reach and QoS settings, RTT, packet loss and other network parameters might vary considerably across an Internet virtual path.

For ease of analysis, nodes within a typical path are classified based on the client-server Internet connectivity model described in Section 4.1 and with the following assumptions:

- The first and last nodes are classified as Local Network nodes. This is done because the first node is expected to be an Internet gateway router either for the Client or the Application Server;
- The second and second last nodes are classified as Local/Remote Regional Network nodes. This is a hierarchical assumption based on the fact that these networks provide connectivity to Local Networks.

- Intermediary nodes are classified as one of the following types of backbone nodes: Regional, remote server, Australian, or US. For this classification, it was observed DNS information and RTT delays between nodes. In some cases, RTT delays indicate a satellite link interconnecting two nodes and therefore, it is possible to have a better idea of a node's geographical location.

In this study, the classification of typical virtual paths is used for introducing the *minimum RTT delay model* (See Sections 4.2.3 & 4.3.4) and for estimating the level of instability of AS within a typical virtual path (See Sections 4.1 & 4.2.2). Since this model is based on the calculation of end-to-end RTT delays for a congestion-free path, it does not focus on specific AS contributions within a virtual path on overall performance.

4.3.3 A *minimum typical path* for a client-server connection is defined as a typical path with the lowest end-to-end RTT delay measured during the experimental period. Kalidindi & Zekauskas [39] observed that at least one measurement packet is expected not to experience congestion within a congested Internet virtual path. Because their probing sample rate (2 packets per second) was much higher than our TRACERT implementation sample rate (1 TRACERT sample for all Application Servers, every 5 minutes), it is assumed that at least one packet is going to experience either *no* congestion *or* a low level of congestion during our experimental period.

4.3.4 For a minimum typical path, some assumptions were also made for estimating the minimum One-Way delay for a particular client-server connection. The minimum One-Way delay calculation is based on Equation (4.8).

4.3.4.1 It is assumed that DNS information obtained via the TRACERT sample inspection (See Section 4.3.2) of a minimum typical path provides a good estimate of geographic location[1] for each node.

[1] It was observed from TRACERT sample analysis that normally DNS addresses have information about airport codes *or* city initials.

For estimating the One-Way propagation delay for a minimum typical path, it is assumed that Internet transmission links between continental cities are fibre links while links between intercontinental cities might be satellite or fibre links. This assumption is used for calculating propagation delays between different cities within a minimum typical path. The calculation is based on Equation (4.9) and by considering two facts:

- GEO satellites links have a typical One-Way propagation delay of 250ms;
- Light propagation speed within fibre is 200.000 Km/s (Equation 4.11).

4.3.4.2 A *typical switching delay* is defined as the average time for routers in a *minimum typical path* for processing probing packets. For our research, it is used the fact that the South Australian Tucows server is in the same region of our Client for calculating the typical switching delay via Equation (4.13). It is also assumed that for this client-server connection there is symmetry between upstream and downstream paths (See Equation 4.12). Therefore, combining Equations (4.12) & (4.13), it is found that

$$OW_{Switching} \approx \frac{RTT_{Total}}{2}. \qquad \textbf{(4.14)}$$

Based on our assumption in Section 4.3.3 that at least one packet will experience no delay or a low delay level within a minimum typical path, it was found that the minimum measured RTT delay for the South Australian server is 10 ms. Taking into account that the minimum typical path for the South Australian Application Server has seven nodes and using Equations (4.10) & (4.12), the *Typical Switching Delay* was calculated as

$$OW_{Switching} = 10/2 = 5ms \overset{7\ \text{nodes}}{\Rightarrow} \text{Typical Switching Delay} = \frac{5}{6.5} = 0.76\,ms. \qquad \textbf{(4.15)}$$

It is assumed that this typical switching delay is going to be a typical value for router switching in *any* node within *any* virtual path. This value is used in Equation (4.10) for estimating the $OW_{Switching}$ within *any* minimum typical path.

4.3.5 A Standard Benchmark File (SBF) was used to measure throughput performance from TUCOWS sites and a large size file was selected in order to avoid the initial low transmission throughput due to the TCP slow start mechanism [8]. The Netscape Communicator distribution setup file[1] was the selected SBF.

4.3.6 The file download session was scheduled as described in Section 4.3.1. For each Client computer, the SBF was sequentially downloaded from the TUCOWS servers as shown in Fig. 4.2. For all download sessions, the *Start time* (StartT) and *End Time* (EndT) were registered in a log file. These values are used for calculating the average connection throughput, which is calculated as

$$\textbf{Throughput (Kbps)} = \frac{\textbf{SBF size (bits)}}{\textbf{(EndT - StartT)}} \times \frac{\textbf{1}}{\textbf{1000}} \quad \textbf{(4.16)}$$

In our case it was not considered any bottleneck that would affect the transfer rate in either the Client or Local Network, as there is a 100 BaseT connection to the Internet backbone gateway from the Client and this gateway does not have congestion problems (See Section 4.1, Fig 4.1: A & B). In addition, recent data suggests Application Servers are responsible for 33% of congestion while 42% are a result of core network problems [9]. The impact of these assumptions on our results is discussed in Chapter 5.

4.3.7 Further to our discussion in 3.6 on the position of any potential bottleneck in our analysis approach, assumptions are made about the influence of interconnecting networks and Application Servers with respect to throughput for any particular Internet virtual path.

It is assumed that due to the popularity of the TUCOWS file server among Internet users, the influence of traffic demand within the Application Server on throughput will *always be high*. Further, it is postulated that if the Client is geographically close to the Application Server, the influence of interconnecting networks *is negligible* (See Section 4.2.1, Equation 4.3). Conversely, if the Client is geographically distant from the Application Server, it is

[1] **Version Number:** 4.7, **byte size:** 18.1 MB – (18,968,232 bytes)

asserted that influence of interconnecting networks will exist (See Section 4.2.1, Equation 4.2).

During the course of the experiment a wide variation of transfer rates was observed. Since a dial-up session under the best circumstance (64 Kbps) would require approximately[1] 40 minutes to download the SBF, it was decided to set this as the time limit for the download sessions. To maintain simulating a broadband access environment any transfers taking longer than this were discarded. Given this broadband access definition, later it is discussed its statistical significance, as it may have an impact on the median value of throughput transfer rate.

4.3.8 Software Supporting the Experiment

Software utilities for analysing the web sites' throughput and path were developed to run the experiment as follows:

4.3.8.1 Throughput Analysis Utility (See Fig 4.5)

Utility 1 - Scheduling the downloading sessions. Activates the web browser and directs it sequentially to the web sites described in Fig. 4.2.

Utility 2 - Timing the downloading sessions. Registers the variables *Start Time* and *End Time* in log files, which are used for calculating the duration of the downloading session. *Utility 3* does this calculation.

Utility 1 was developed using the Automate[2] software, version 4.3e, which is a software platform for constructing and automating tasks in a step by step basis. *Utility 2* is a script developed with Visual Basic. *Both* utilities were developed as part of this research project.

[1] In reality, 64 Kbps is obtained after 39 minutes 31 seconds
[2] http://www.unisyn.com

The logical operation sequence (*Steps*) for *utilities 1* and *2* is as follows:

A. *Utility 1* opens a browser window and directs it to a SBF download link at one of the TUCOWS servers shown in Fig. 4.2. A popular browser was chosen for implementing our experiment: the Microsoft Internet Explorer browser, (version 5.0.2314.1003). This decision is due to our interest in making the experiment as close as possible to a real browsing experience as seen by Internet users worldwide. This implementation differs from recent research carried out by Myers *et al* who used Practical Extraction and Report Language (PERL) script to run fetches [96]. While their approach might be simpler, implementing a Uniform Resource Locator (URL) fetch code does not capture the characteristics of actual browsers.

B. *Utility 1* waits for 25 seconds for a browser "file download" window. This window waiting time value is chosen based on our observation that the Microsoft Internet Explorer Browser tries to connect to a certain address for approximately 45s, after which a failure window entitled "Cannot find server" is displayed to the user. It was decided that a value closer to 45 seconds would be not adequate due to processing delay variations among our Client machines. Also, values below 25 seconds would not be suitable because servers with large RTT in relation to our clients would have high occurrence of "Cannot find server" failures. In the case where the "file download" window does not appear, the utility registers the occurrence of a failed response of the server in a log file. In the event of a successful "file download" window, the utility sends two keystrokes: the first is used for confirming the file download session and the second is used for saving the SBF to a default directory in the Client computer.

C. After the second keystroke in the case of a successful "file download" window, *Utility 1* starts Utility *2* which registers the *StartT* for the download session in a log file.

D. A successful download session ends with a "Download complete" window. At this moment, *Utility 1* starts *Utility 2* for registering the *EndT* in the log file.

E. If a "Download complete" window does not appear before a 40-minute download session, the download session is "killed" by *Utility* 1 and as such is registered in the log file. *Utility* 1 "kills" the 40 minute-download session by sending a keystroke. This 40-minute deadline period also protects the data collection in case a failure occurs during the download period (eg, connection to the server gets timed out). In case a "timed out" situation happens, *Utility* 1 identifies the "timed out" window and it starts "*Utility 2*" which registers the EndT and a "timed out" message.

F. After *Utility* 2 registers the EndT in the log file, *Utility 1* sends a keystroke for closing the "Download complete" window and also deletes the SBF from the default download directory. After the SBF deletion, *Utility 1* starts a new download session (*Step A*) to the next TUCOWS server as shown in Fig. 4.2. For clarity in later discussion, Utilities 1&2 are referred collectively as THROUGHP.

Utility 3 - Calculating the throughput. Calculates the throughput based on the output from *Utility* 2 and follows Equation (4.16).

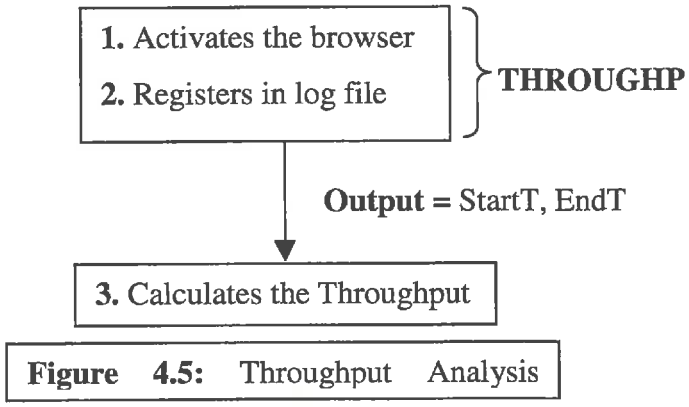


Figure 4.5: Throughput Analysis

4.3.8.2 Path Analysis Utility (See Fig 4.6)

Utility 4 - Tracing the route to all TUCOWS Web sites every five minutes in parallel. Activates the TRACERT network tool every five minutes generating a log file for the path analysis. This software utility was developed based in Visual Basic code and will be called TRACER. While a sampling interval of 5 minutes might loose short-lived network

phenomena [41], this was found a better granularity because the path stability analysis is focussed on longer routing changes.

The log file parameters that were used to analyse the paths are:

- Node number;
- RTT;
- IP address for a particular node;
- DNS for a particular node.

Our path analysis utility output presents only *one way traces*, i.e, upstream paths from Client to Servers. It does not guarantee that downloaded packets from the TUCOWS servers (downstream) will follow the same path. However, this utility does provide an extra facility for estimating packet flow behaviour between two Internet destinations. Furthermore, as already discussed in Chapter 2, TRACERT might occasionally provide wrong conclusions due to low priority of ICMP packets in relation to other packet types.

TRACER also provides information for better understanding Throughput results because it enables the analysis of interconnecting paths and their influence over download sessions. In addition, this client software utility also provides information for estimating path instability and the end-to-end minimum RTT to a particular Application Server destination.

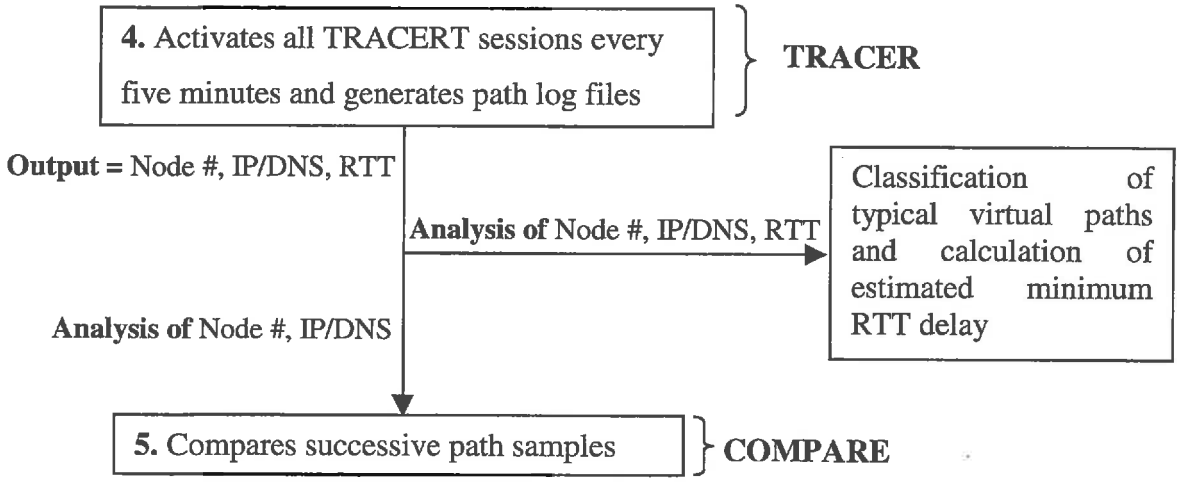


Figure 4.6: Path Analysis Utility

4.3.8.2.1 Path Instability Analysis

Node number and IP/DNS addresses (See Fig 4.6) are used for calculating the path instability variable ε (See Section 4.2.2, Equation 4.5). This calculation is based on the *Software Utility 5*.

Utility 5 - Comparing the sampled paths for a particular TUCOWS web site. Compares sequential samples from a particular TUCOWS log file. This utility is called COMPARE.

Below it is described the manner in which COMPARE measures the difference in path giving successive TRACERT samples to a particular Application Server:

Node changes: If a change in an IP address node occurs in sequential samples, COMPARE registers the sample number and the new node configuration.

Pseudo-node changes for IP addresses in Class C format: IP addresses are classified as described in Table 4.2. The last 8-bit number field of the IP address Class C format describes the number of hosts for a certain network address ID. It was not considered a

change in path for cases where two IP Class C nodes in sequential samples have common network IDs. In most cases this means a balancing policy has been implemented for routers in the same network and are possibly located physically in the same premises.

Therefore, if a certain node in two sequential samples has an IP address in the Class C range, COMPARE verifies a match in the first three fields of the IP address. If a match occurs, COMPARE does not analyse the fourth field. A change in host ID does not change the QoS settings for this network significantly. For Class A and B, COMPARE looked at all fields of the IP address.

Table 4.2: IP Class Types

IP Class Types	Range of Host Addresses	Maximum # of Networks	Maximum # of Hosts per Network
A	1.0.0.0 to 127.255.255.255	128	16777216
B	128.0.0.0 to 191.255.255.255	16384	65536
C	192.0.0.0 to 223.255.255.255	2097152	256

In some cases TRACER may generate "abnormal" messages, for example, *timed out* or *destination unreachable* messages. How these are incorporated into our *Software Utility* is discussed below.

Timed Out: If a "Timed out" situation happens in a certain node from a sample, COMPARE verifies the subsequent node matches the IP address of the same node from the previous sample. If this situation happens, the path has not changed between the sequential samples and COMPARE verifies the successive nodes in the samples until the destination host is reached. If however the IP address does not match the previous sample node IP address, COMPARE considers this a change in the path.

Destination Unreachable: If the message "host or network – unreachable" happens in a certain node, COMPARE compares this path with the previous sample path until the "unreachable" node is processed.

4.3.8.2.2 End-to-End Minimum RTT Analysis

All four parameters obtained through *Utility* 4 were used for defining typical virtual paths to a server (See Section 4.3.2). Virtual paths were classified based on aggregation of RTT occurrences for each node within a virtual path and this analysis procedure was done running queries on Microsoft Access 97.

By identifying the *minimum measured RTT delay* for each client-server virtual path, it was possible to distinguish *a minimum typical path* for each Application Server (See Section 4.3.3). Based on the geographical information obtained via node inspection from the minimum typical paths and by assuming symmetrical paths, it was calculated possible minimum RTT for the minimum typical path based on the combination of different propagation mediums between cities. The calculation is done as described in Section 4.3.4.

By comparing the *measured* minimum RTT delay and the *calculated* minimum RTT delays, it was possible to identify the best physical medium combination for matching the measured RTT. This procedure is shown in detail in Chapter 5. Deviations from measured and calculated minimum RTTs are discussed and they might result from at least one of the following reasons:

- Inaccurate DNS geographic information in TRACERT samples. As already discussed in Chapter 2, Section 2.3.2.3, IP addresses are not easily mapped in geographic coordinates;
- The minimum measured RTT delay might not be effectively the minimum RTT delay for the client-server connection. This fact might happen due to three reasons:

- Lower priority of ICMP echo-request packets than other type of traffic (See Chapter 2, Section 2.3.2.3), resulting in higher switching delay for ICMP packets;
- Packet generation process can interfere in the measurements and the collected data might differ by up to 30 ms [33];
- Path asymmetries.

- For small geographical regions, processing delays relative to TCP measurement machines can have a high contribution over the results [50]. Therefore, the Typical Switching delay, which was calculated based on a small geographical region, might have errors. Moreover, since the Internet is an heterogeneous environment, Switching delays might also vary due to routers having different switching responses based on their location within a network (e.g., backbone routers are faster than Local Network routers) and their manufacturer routing strategy implementation.

4.4 Data Analysis Methodology

All data was measured with the time reference based on the client time zone. In order to have a better understanding on how time zone differences might affect throughput and RTT delays for a certain Internet virtual path, it was decided to shift the data time stamp from the local client time to the remote Application Server time (See Table 4.3). All statistical analysis will be based on this time-shift. For the purpose of having a more accurate geographical location of each server, a shareware network analysis tool called NEOTRACE[1] was used for obtaining the server's latitude/longitude coordinates. By checking the coordinates on a map, it was possible to verify the veracity of the 'NEOTRACE coordinate lookup' and the most appropriate city location for each server. For easy implementation, a Web map viewer was used, which is publicly available[2]. For more information about this tool, please read [97]. When the coordinate lookup would generate a mismatch in relation to the Web map viewer, the TUCOWS server administrator

[1] http://www.neoworx.com/
[2] http://pubweb.parc.xerox.com/map/

would be contacted for confirming its geographical location. While this procedure proved to be adequate in most cases (South Australia, Victoria, USA East Coast and England), in some it was not successful (Zimbabwe). For the TUCOWS server in Zimbabwe, the coordinate lookup resulted in Cape Town City, which is in South Africa, and the server administrator gave no feedback about its actual location. For our experiment, it is assumed that this server is located in Harare, Zimbabwe, and not in South Africa. In fact, this is probably true because the DNS address for the Application Server in Zimbabwe suggests this server is located in Harare (*tucows.harare.iafrica.com*). One conclusion for such 'coordinate lookup' approach is that in some cases it might result in wrong conclusions when there is no research cooperation from the Application Server administrator.

Table 4.3: Client x Servers Time-Shift: Time shift for Application Servers in relation to Adelaide based Clients.

Sites	USW	USE	AR	BR	ENG	GER	ISR	ZB	SF	HK	SA	VIC
Time Shift	-18:30	-15:30	-13:30	-12:30	-10:30	-9:30	-8:30	-8:30	-8:30	-2:30	0	+0:30

USW: USA West Coast; *USE*: USA East Coast; *AR*: Argentina; *BR*: Brazil; *ENG*: England; *GER*: Germany; *ISR*: Israel; *ZB*: Zimbabwe; *SF*: South Africa; *HK*: Hong Kong; *SA*: Adelaide, South Australia; *VIC*: Victoria

4.4.1 Throughput Statistics:

For statistics comparison between different virtual paths, the measured throughput was classified in terms of expected time-of-day patterns. This decision was also supported by other studies that showed a high correlation of data traffic characteristics with time-of-day patterns [18, 41,45].

Each day was partitioned into three zones that correspond to known profiles of Internet usage throughout the day and this is illustrated in Table 4.4. Table 4.4 also shows the

expected throughput for a download session within a virtual path where the Client and the Application Server are located in the same geographical region and there are *no* network bottlenecks between them (See Section 4.2.1, Equation 4.3).

Table 4.4: Time-of-Day Patterns

In-Day Zone	Usage Profile Description	Expected Throughput
00:00-09:00	Off-peak time	High
09:00-18:00	Business usage	Low
18:00-24:00	Family usage	Medium

Based on our time-of-day traffic pattern assumptions and for ease of comparison between different virtual paths, graphs of 'median value of transfer rate' against 'day of the week' were plotted. These graphs represent a median throughput during a typical week for each Application Server. In some cases where there are a great many changes on a server's typical Internet virtual path (See Section 4.3.2), there will be more than one graph for a server.

Deviations from the expected time-of-day patterns (See Table 4.4) are found in virtual paths where influence of regional networks, backbones and server traffic demand is substantial. For ease of understanding, in some cases (USA West Coast & Germany) graphs of 'median value throughput' against 'hours' for a typical weekday were plotted. This approach allows a better analysis of throughput behaviour for a typical weekday. By matching this information with RTT analysis, it is possible in some cases to infer the causes of a non-standard time-of-day pattern (ie, a pattern that does not follow Table 4.4).

4.5 Summary

This chapter describes the experimental methodology used for developing this research project. Firstly, a *client-server Internet connectivity model* was introduced based on the AS definition shown in Chapter 2. The main elements of the model were described and these were relevant for developing the *throughput, path instability* and *minimum RTT delay* models and for analysing the collected data. Subsequently, data collection procedures and experimental assumptions were discussed. These were used for developing the measurement utilities and for introducing analytical procedures required in the result discussion (Chapter 5). The software utilities were described and the main operation characteristics were highlighted. In addition, for the path analysis utility, an overview of potential sources of deviation between *measured* and *calculated minimum RTTs* was provided. Lastly, some data analysis procedures were stated for improving data investigation and increasing analysis accuracy.

Chapter 5
Results & Discussion

This Chapter provides a summary of the main findings in our research. Based on our initial assumptions as described in Chapter 4, this Chapter presents an analysis of end-to-end performance from a client in Australia in relation to Application Servers located in different geographic locations worldwide. The main performance parameters in consideration are *throughput, path instability and minimum RTT*. As discussed in Chapter 4, Section 4.4, *all* measured data was shifted to the respective time zones of the Application Servers.

In terms of throughput, variations that happen to each Application server are analysed and deviations from the expected throughput profile shown in Table 4.4, Chapter 4, are investigated. By analysing throughput measurements and the respective typical virtual paths, it was possible to estimate the level of influence of each part of the network on each virtual path. This approach also helped identifying changes in throughput performance and hypothesising the reasons for throughput variations.

While analysing path instability, the main objective was to verify the overall end-to-end instability of virtual paths. The *simultaneous analysis* of node variations between sequential samples within a virtual path *and* DNS information from changing nodes provides a method for better evaluating the instability level of a client-server connection. Moreover, the comparison of the path instability variable ε (See Chapter 4, Section 4.2.2) among different virtual paths provides an approach for comparing end-to-end path instability for different virtual paths along the Internet.

In terms of minimum RTT, our analysis is based on the comparison of *measured minimum RTT* and *calculated minimum RTT* delays within a virtual path for evaluating

interconnecting link characteristics. The former is obtained experimentally based on the identification of the *minimum typical path* for a client-server connection (See Section 4.3.3, Chapter 4); the latter relies on the calculation of *potential* minimum RTT delays for a minimum typical path when different interconnecting link characteristics are hypothesised (See Chapter 4, Sections 4.2.3 & 4.3.4.2). While this comparison is proven to be quite effective for estimating path characteristics within a minimum typical path, some deviations might be observed based on our methodology assumptions. Possible reasons for these deviations are hypothesised.

This Chapter is organised in four sections, which will be discussed in a server-by-server basis, following the order: South Australia, Victoria, USA West Coast, USA East Coast, Hong Kong, Israel, Germany, England, Argentina, Brazil, South Africa & Zimbabwe. The decision of choosing such a server order that departs from our initial Client organisation (See Chapter 4, Section 4.3.1, Figure 4.2) is result of the high influence of the USA backbone on overall performance of all Application Servers outside Australia. Therefore, a better understanding of the performance of Application Servers within the US helps us discuss the other Application servers. The four sections in this Chapter are arranged as follows:

1. Typical virtual paths for all client-server connections are described. The analysis is based on Section 4.3.2, Chapter 4, and remarks the most important findings and characteristics of measured paths.
2. Throughput results are analysed and some conclusions are drawn based on the investigation of typical virtual path characteristics.
3. A path instability study from our client to the Application Servers is presented. Based on the path instability parameter, instability is estimated for all client-server connections.
4. Minimum typical paths are defined for each client-server connection and for each minimum typical path, interconnecting link characteristics are hypothesised (i.e., fibre or satellite). The combination of different link mediums is used for calculating

potential minimum RTT delays for a client-server connection (See Sections 4.2.3 & 4.3.4.2, Chapter 4). A comparison of the *measured minimum RTT* and the *calculated minimum RTT* delays is provided, which allows in most cases the inference of interconnecting link characteristics.

5.1 Typical Virtual Paths: Classification & Discussion

Based on Section 4.3.2 from Chapter 4, typical virtual paths were found to each Application Server. Typical virtual paths are useful for understanding throughput behaviour, analysing path instability and estimating path characteristics.

Since throughput data collection happened between Nov 3 and Nov 22 1999 for Clients 1 & 2 and between Nov 3 and Nov 30 1999 for Client 3 (See Fig. 4.2, Section 4.3, Chapter 4), typical paths for each Application Server were defined in accordance to these throughput collection periods.

In this section the most interesting findings in our methodology analysis are highlighted. For ease reading, *bold formatting* is applied in selected parts of tables while discussing about these findings. *All* tables are grouped in Appendix A.

5.1.1 South Australia

The typical virtual path to the South Australian server has 7 nodes as shown in Table 1. In terms of node location, all intermediary nodes are classified as Australian backbone nodes with a *regional* coverage. The *regional* characteristic of the backbone is clear when this typical virtual path is compared to other Application Server paths. The South Australian server typical path is unique in having *203.16.212.12/fa0-0-108.boomerang.internode.on.net* as IP/DNS addresses for node 3.

Another interesting point to be observed is that node 5 is a typical case of *tightly coupled router topology*, with similar traffic loads and possibly collocated in the same physical area (See Section 4.3.2, Chapter 4). This load balance implementation can be easily identified by comparing the total number of RTT replies from the tightly coupled router node with the number of RTT replies from the next node and the node before. In this case, node 5 has 5570 RTT replies (1400+1390+1383+1397=5570), which is comparable to RTT replies in node 4 (5581) and RTT replies in node 6 (5579).

The South Australian server typical path is used in Chapter 4, Section 4.3.4.2, for calculating the typical switching delay for any router within any virtual path. This value is used in Section 5.4 for calculating the *minimum RTT* for each client-server connection.

A *Forward Loop* routing pathology, which was identified in a research carried out by Paxson, has also been observed in one TRACERT sample [41]. This router pathology is found when packets forwarded by a router return to the router. This pathology is observed when there is a change in connectivity within the network and routers do not immediately propagate these changes [44].

5.1.2 Victoria, Australia

The typical path to the Victorian server has 6 nodes as shown in Table 2. This typical path has a node 3 with IP/DNS addresses as 192.65.88.225/atm2-0-4.mb1.optus.net.au, which are different from the local South Australian server and the International servers. This node is assumed to be a gateway for the national backbone infrastructure in South Australia. Therefore, in terms of node location, the Victorian server has intermediary nodes that are backbone nodes with a *National* coverage.

A routing pathology shown in a research carried out by Paxson [41] was also observed. An erroneous routing happened in one of the TRACERT samples and the probing packets were forwarded to the US instead of being forwarded to the Victorian Application server.

This was a surprise because despite the last few year improvements in routing algorithms, such a problem still happens.

5.1.3 USA West Coast

There are three typical paths for this server and these are shown in Tables 3,4 & 5. The paths have a persistent behaviour in three different periods: 03/11/99 until 11/11/99, 11/11/99 until 13/11/99 & 13/11/99 until 22/11/99. This persistent characteristic led us think that these paths are result from peering agreements and not path re-configurations due to Internet congestion.

The first important observation about these paths is that the node number 5, which has IP addresses in Class C format *(192.65.89.*)*, is probably[1] the standard US gateway for all AARNET[2] traffic originated from US, Europe, Latin America, Middle East & Africa. This Class C address IP is found in *all* typical paths to Application Servers except Hong Kong, South Australia & Victoria. Possibly this also might be the gateway for traffic originated from Asia though it is hard to affirm because of path asymmetries and the limitations of TRACERT (See Section 5.1.5 for a better discussion about this hypothesis). Furthermore, another interesting remark is that node 5[3] has a load balance – i.e, it is a typical case of *tightly coupled router* topology (See Chapter 4, Section 4.3.2).

Similarly to node 5, node 4 -which has IP addresses in Class C format *(202.139.1.*)*- is the standard gateway within Australia for forwarding traffic originated within AARNET to overseas destinations in the US, Europe, Latin America & Africa. This gateway is *not* used for sending traffic to Hong Kong because node 4 for a Hong Kong typical path has a different Class C format: 202.139.7.* (See Section 5.1.5). It is also important to note that node 4 does not provide any DNS information and therefore, our assumption that this node

[1] It is said "probably" because our work is based on a *sample* of nine Web sites.
[2] AARNET stands for Australian Academic Research NETwork, which is the organisation that provides Internet backbone connections to Academic institutions in Australia.
[3] For not being repetitive, this node behaviour is not discussed for other Application Servers.

is within the Australian backbone is based on the analysis of the minimum RTT. While comparing minimum RTTs, node 4 has a RTT value closer to node 3 (Australian backbone) than to node 5 (US backbone).

Another important remark is that node 6 is a default node for typical paths between 11/11 and 22/11. This node is found both in Tables 4 and 5 and the number of RTT replies for this node (3295) is approximately equal to the total number of RTT replies of node 7 in Tables 4 & 5 (460+2796= 3256).

Node 10 in Table 5 behaves as a *fluttering node*, where a border router forwards packets to different networks in a way of balancing load in a network (See Section 4.3.2, Chapter 4).

An interesting fact is illustrated in Table 3: despite *both* the US backbone gateway *and* the Application Server being located in the West Coast, packets are forwarded to the East Coast before looping back to the West Coast. This fact increases our hypothesis of peering agreements happening in the re-configuration of the paths, because as a rule Internet packets flowing in such an incorrect fashion for a long period of time would not be expected.

5.1.4 USA East Coast

Two different typical paths were identified for this server (Table 6 & 7). Similarly to the USA West Coast server, each path is persistent during a distinct period of time, what suggests paths are result of peering agreements. An interesting point for these typical paths is the inability of TRACERT to verify nodes further than the node with IP address *144.228.60.9* (See Table 6, node 13 & Table 7, node 15*)*. This is the typical case where router administrators give lower priority (or block) to ICMP echo requests in relation to other type of traffic. This phenomena was shown by Matthews & Cottrell [38] which reported deterioration in performance from North American sites to Scandinavian sites due to installation of *Smurf filters* in the connection link. It is important to note that in

spite of having the ICMP echo-requests blocked, the file download sessions (HTTP traffic) ran during the experimental period without any problems.

Tightly Coupled Routers are found *not only* in the US gateway (node number 5) *but also* in node 7 of the first typical path (See Table 6).

Fluttering is observed in Table 7, starting at node 10 and ending at node 14. One could expect a node 15 with approximately 3228 RTT replies, which is the number of RTT replies in node 9. However, since node 15 has 2204 RTT replies, it is argued that ICMP echo-request packets within some of the fluttered paths might have been blocked and were not able to reach node 15.

5.1.5 Hong Kong

This site provides interesting observations (Table 8). The first important finding for the Hong Kong virtual path was observed while investigating a drop in throughput performance from the Application Server (See Section 5.2.5 for further discussion). It was found that for a Client located within AARNET, traffic to Asia-Pacific destinations is routed via direct links to Asian peers *but* downloaded traffic to such a Client is via U.S. peers. Therefore, this virtual path is asymmetric and TRACERT will provide accurate information only for the upstream direction.

This path asymmetry is quite relevant for our study because *not only* it shows the inefficiencies of RTT measurements based on ICMP echo-requests *but* it *also* highlights the importance of analysing both the *application layer* (e.g., Throughput) and *network layer* (e.g., TRACERT sample). Our finding also confirms a research from Huffaker *et al* [47], which suggests the US Internet backbone as the major intermediary transit backbone for Australia and other countries.

The path asymmetry finding is also relevant when estimating the *minimum RTT delay* for this path. In Section 5.4.5, the minimum RTT is *calculated* based on this triangle interaction (Australia-Hong Kong-USA-Australia) and results that quite successfully match the measured minimum RTT are presented.

Another interesting fact to be observed in Table 8 is the low performance behaviour of node 4. One should expect this node to have similar number of RTT replies as other nodes within the path. However, since a high packet loss is observed in this node, it is asserted that there is a traffic policy for this node, probably giving lower priority to ICMP echo-requests in relation to other traffic types.

A load balance in node 6 is also noted, which suggests a Coupled Switching topology.

5.1.6 Israel

This server had 3 typical paths. The first path, which is shown in Table 9, was the dominant path from 3/11 until 11/11. Between 11/11 and 22/11, this path appeared only 8 times, what suggests this was a stand-by path for this period. Conversely, paths 2 & 3 (See Tables 10 & 11) are more frequent during the second period and are rarer during the first period. In addition, while comparing paths 2 & 3, a higher number of occurrences for path 2 is observed, what indicates this as the default path between 11/11 and 22/11.

All paths have node 4 as a *fluttering node* and node 5 as *tightly coupled routers*. Paths 1 & 2 have also nodes 12 & 14 as *tightly coupled routers*.

Path 2 & 3 are result of a *fluttering topology*. One can observe that for these paths, packets are forwarded to node 6 for either the IP address *205.174.74.165* (Path 2) or *207.124.109.57* (Path 3). At this point, packets flow through different networks until reaching the Application Server.

5.1.7 Germany

The German Application Server has 2 typical paths (Tables 12 & 13) and these are persistent during two different periods of time (3/11 until 12/11; 12/11 until 22/11). Both paths have the same network topology until node 16 and path re-configuration was due to introducing a new German/US gateway (See Table 13, Node 17).

5.1.8 England

The Application Server in England has 2 typical paths, which are persistent during 2 different periods of time: 03/11 until 11/11; 11/11 until 22/11 (See Tables 14 & 15). The analysis of the IP and DNS addresses of both paths shows such a similar topology that the path re-configuration (from node 7 until node 11) could be considered as inexistent and both paths could be classified as a unique path. In addition, the second path has fluttered paths between nodes 10 & 11.

5.1.9 Argentina

This Application Server has 2 typical paths, which are persistent in two different periods of time (See Tables 16 & 17). This Application Server demands a more complex analysis due to a combination of fluttering and *tightly coupled router topology*.

In both typical paths, *fluttering* starts in node 7 and RTT echo-requests are forwarded via 2 *fluttered* paths, the first with initial IP address *146.188.248.110* and the other with initial IP address *146.188.248.98*. This *fluttering topology* persists until node 11 and each *fluttered* path is discussed in separate. Bold formatting is applied to the first *fluttered* path[1] in both Tables 16 & 17.

[1] For ease reading, differently from the other server tables, bold formatting is not applied for the standard US gateway node (node 5).

5.1.9.1 First Fluttered Path:

- *In node 8,* a new *fluttering topology* starts and RTT echo-requests from node 7 with IP address *146.188.248.110* are forwarded through 2 different paths *(152.63.112.178 & 146.188.248.242)*. In node 10, a high *tightly coupled router topology* (*146.188.232.89 & 146.188.232.85*) happens for RTT echo-requests originating from node 9 with IP address 146.188136.49.

5.1.9.2 Second Fluttered Path:

- This fluttered path is less complex than the first one and the unique topology observed was a *tightly coupled router topology* implemented in nodes 8 & 10.

Between node 12 & 15, two different paths were observed - which were used for load balance during all experimental period. It is not clear how this load balance was implemented in terms of load share. One can observe that there is no consistency of number of RTT replies within a path and therefore these are not fluttered paths.

Another interesting point to observe is that node 14 has probably a policy of *blocking* RTT echo-requests because all samples resulted in packet loss. Therefore, similarly to the Hong Kong & USA East Coast servers, this case suggests another example of low priority for ICMP echo-request traffic. While ICMP echo-request traffic had low priority in one of the nodes of the Hong Kong virtual path and it was totally blocked in the US East Coast virtual path, a *partial block* is observed in this case. While node 14 does not reply to ICMP echo-request packets, it still forwards these packets to the next node in the virtual path.

5.1.10 Brazil

There are two typical paths for this Application Server (See Tables 18 & 19). While the first path appears during the whole experimental period the second path is present only

from 11/11 to 27/11. Therefore, the first path seems to be the default path to this server while the second is a stand-by path. The main difference between both paths is a node re-configuration between nodes 7 and 11.

First, the topologies within the path differences are analysed. For the first path, a topology similar to *fluttering* happens between nodes 10 & 11; however, since the number of RTT replies within the paths are not similar, this cannot be considered *fluttering*. The analysis of the number of RTT replies does not provide enough information to fully identify this load share strategy. For the second path, two tightly coupled router implementations are observed in nodes 7 & 10.

For the common parts within the typical paths, some topologies are identified: in node 16, a *tightly coupled router topology* is observed; between nodes 12 & 13, a load balance strategy is implemented, however, it was not possible to classify this as *fluttering* or *tightly coupled router* topologies.

5.1.11 South Africa

There are two typical paths for the South African server (See Tables 20 & 21). Since the first path is present during the whole experimental period, this is assumed as the default path. The second path only happens between 24/11 & 30/11 and therefore this is a stand-by path. Both paths are similar except for the subsection between nodes 13 & 16. For the first path, node 15 has a *tightly coupled router topology;* for the second path, *fluttering* happens between nodes 13 & 14 and a *tightly coupled router topology* is implemented in node 16.

One interesting point to note about these paths is that nodes 11, 12, 13, 15 in path 1 and nodes 11, 12, 13, 14 & 16 in path 2 do not provide DNS information. Therefore, it was not possible to classify these nodes in terms of geographical location.

Another important observation is that one could find strange that in the second typical path node 20 has a higher number of RTT replies than node 19. This is result of TRACERT operation characteristics when a destination node within a virtual path has high packet loss (the analysis of the server's log file shows a second path with high packet loss for node 20 - i.e., the destination node). As discussed in Section 4.3.2, Chapter 4, TRACERT probes each node within a virtual path with a set of three echo-request datagrams. In addition, TRACERT has an extra feature when probing a destination node: TRACERT waits for a response from the *third* datagram within a probing set; if the *third* datagram is received, the trace is completed; in case the *third* datagram is lost, TRACERT sends a new set of probing packets; This process is maintained until TRACERT receives an echo-reply from the third datagram within a probing set *or* the node probing limit[1] is reached.

5.1.12 Zimbabwe

Zimbabwe had the highest number of typical paths among all servers: seven paths (See Tables 22, 23, 24, 25, 26, 27 & 28). While this server was an important example in terms of routing re-configurations, interestingly there were not many routing topology implementations within typical paths. For this reason, it is understood that these are not relevant in our discussion and therefore the main focus is on the reasons for such high level of path re-configurations[2].

The first important observation about this Application Server is that differently from other servers, the Remote Backbone peers with a non-US backbone provider- *Teleglobe*, a Canadian backbone provider. This characteristic is the main reason for such a high number of path variations during the experimental period. While a number of American backbone providers have peering facilities for interconnecting their networks with OPTUS, which is the Australian Backbone provider for AARNET traffic, *Teleglobe* does not seem to have these facilities. Therefore, *Teleglobe* depends on different US backbone providers for

[1] Normally, TRACERT default configuration limits the maximum number of nodes to 30 nodes.
[2] The reader is invited to draw conclusions for the few router topologies found within this Application Server. For ease analysis, similarly to the other tables, bold formatting has been applied.

exchanging traffic with OPTUS. As can be observed in the typical paths, *Teleglobe* exchanges data with several US backbone providers such as *ALTER.NET, CERF.NET, ATT.NET, BBNPLANET.NET & CWIX.NET.* While this high number of exchange points might result in higher path instability, it also provides an increasing overall QoS improvement due to a larger number of alternative routes for bypassing network congestions.

5.2 Throughput: Results & Discussion

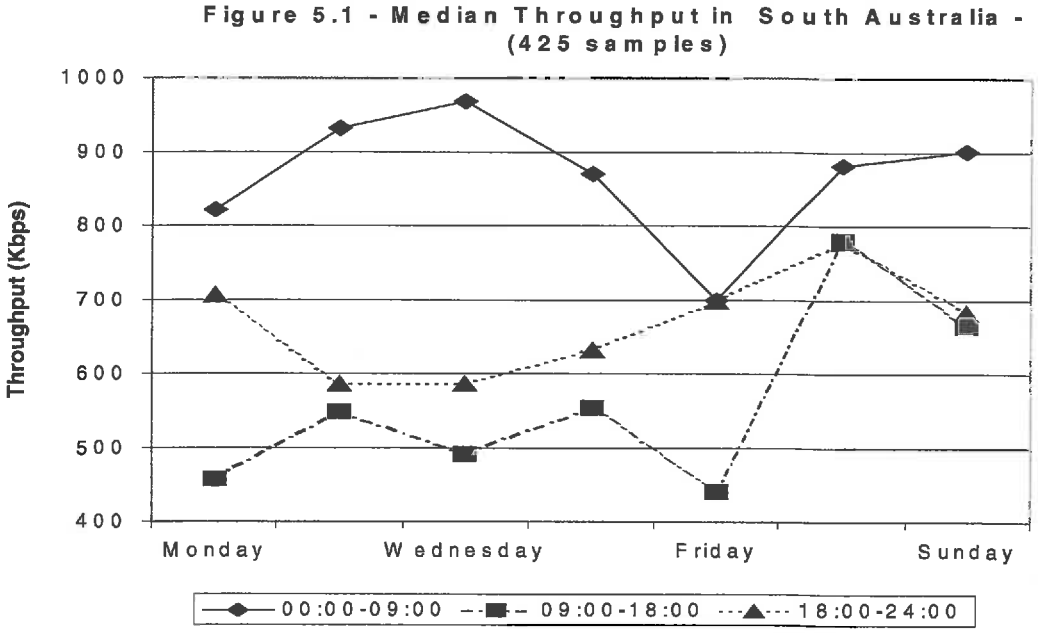
It was assumed that backbone characteristics would remain approximately constant for the whole experimental period while calculating the median throughput curves. While this assumption was adequate for most of the servers, this was not a valid statement for the US West Coast server. As already discussed in Section 5.1.3, there was a major change for this server path on Nov 11 1999.

5.2.1 South Australia

Fig. 5.1 show the throughput from the South Australian TUCOWS server follows Table 4.4 in Chapter 4. This behaviour suggests a high influence of the file server traffic demand and low influence of the backbone and regional networks with respect to throughput (See Chapter 4, Section 4.2.1, Equation 4.3).

It was observed larger variations in throughput rates across divisions of the day on weekdays when compared to the weekend. The maximum throughput for the file server is around 1Mbps.

As expected, on the weekend the Business and Family times exhibit similar behaviour in terms of throughput because there is no Business traffic demand on the server.



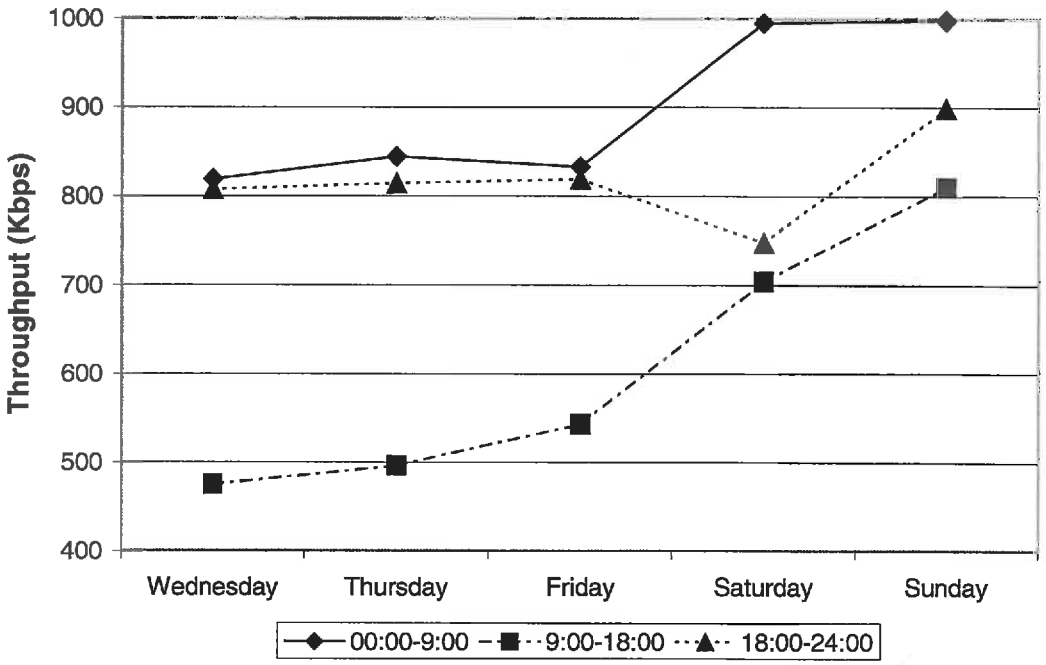
5.2.2 Victoria, Australia

In terms of Internet usage profile, it was found that this site also conforms to our assumptions stated in Table 4.4 in Chapter 4, that is, there is a high influence of the application server with respect to throughput (See Chapter 4, Section 4.2.1, Equation 4.3).

Unlike most other sites, two median values of throughput calculation were appropriate each equating to separate periods of the experiment (Fig. 5.2 & 5.3). A drop in throughput performance was experienced between these periods and it was suspected this could be result of either bottlenecks in the backbone, regional networks or Application Server (See Chapter 4, Fig. 4.1: C, D & F). In fact the path analysis did not show any changes to the application server at the time the throughput dropped and suggested a bottleneck in the Application Server; this was found to be the case and surprisingly the bottleneck was economically imposed. The ISP hosting the Application Server experienced a change of

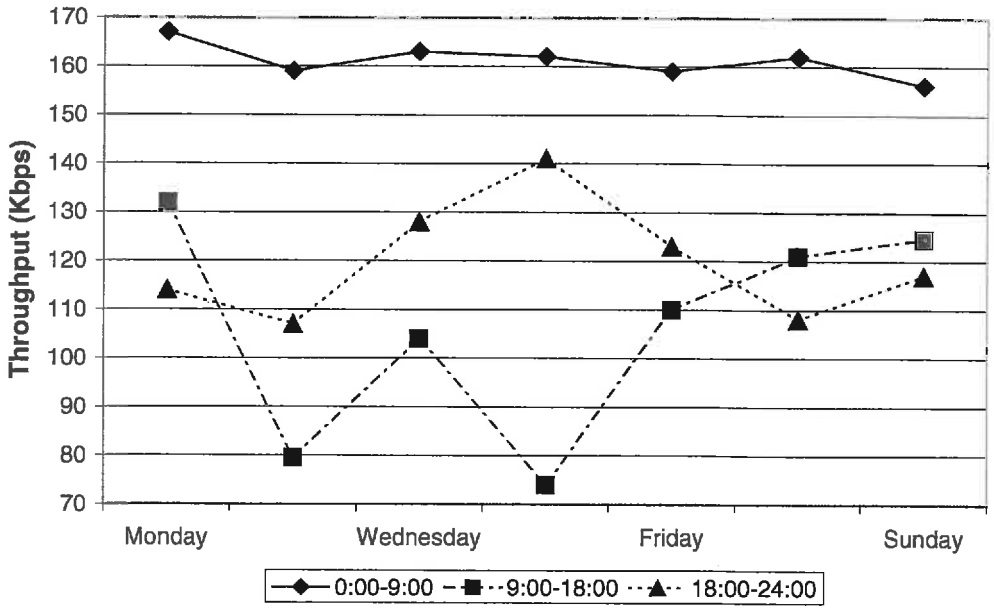
charging structure for bandwidth from the backbone service provider[1]. In an attempt to make its Application Server less attractive in terms of speed for Internet users outside its network, the ISP limited the maximum speed for a download session to 256Kbps.

Figure 5.2 - Median Throughput in Victoria
Period 03/11/99 - 07/11/99



[1] The ISP network administrator informed us the major backbone provider in Australia, *Telstra,* was charging AU$ 80.00 for every 1GB uploaded from its Application Server.

Figure 5.3 - Median Throughput in Victoria
Period 08/11/99 - 22/11/99



5.2.3 USA West Coast

The median download throughput from this application server remained approximately constant throughout the period of the experiment. The throughput does not follow Table 4.4 in Chapter 4, suggesting throughput is also influenced by traffic demand within the backbone and regional networks (See Chapter 4, Section 4.2.1, Equation 4.2).

Based on the typical paths described for this server in Section 5.1.3, a 'typical weekday median throughput' curve was traced for each of its typical paths. Because the second typical path only persisted for two days (See Appendix A, Table 4) and there were not enough throughput samples for plotting a curve, throughput curves were plotted for the first and last typical paths (Figures 5.4 & 5.5). The main difference between these typical paths is that from 3 Nov until 11 Nov 1999 (*first typical path*) the path looped across the North American continent for reaching the Application Server while for the remainder of the experimental period the path went directly to the Application Server.

Figure 5.4 - Median Throughput for a Typical Weekday
USA WEST COAST
Period - 03/11 - 11/11

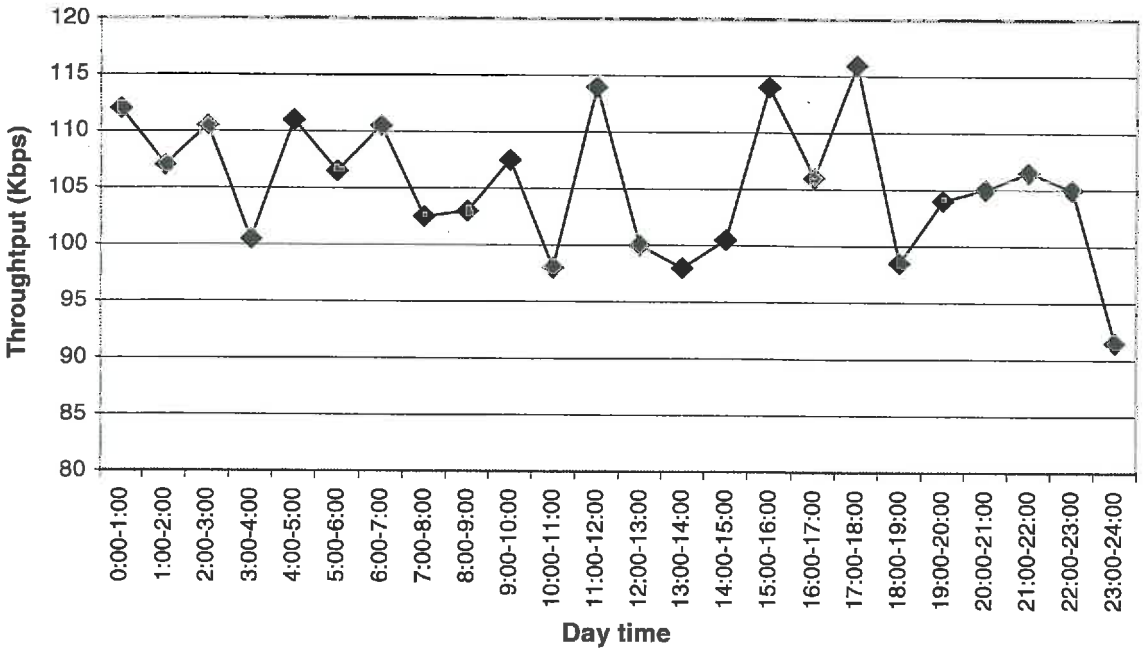
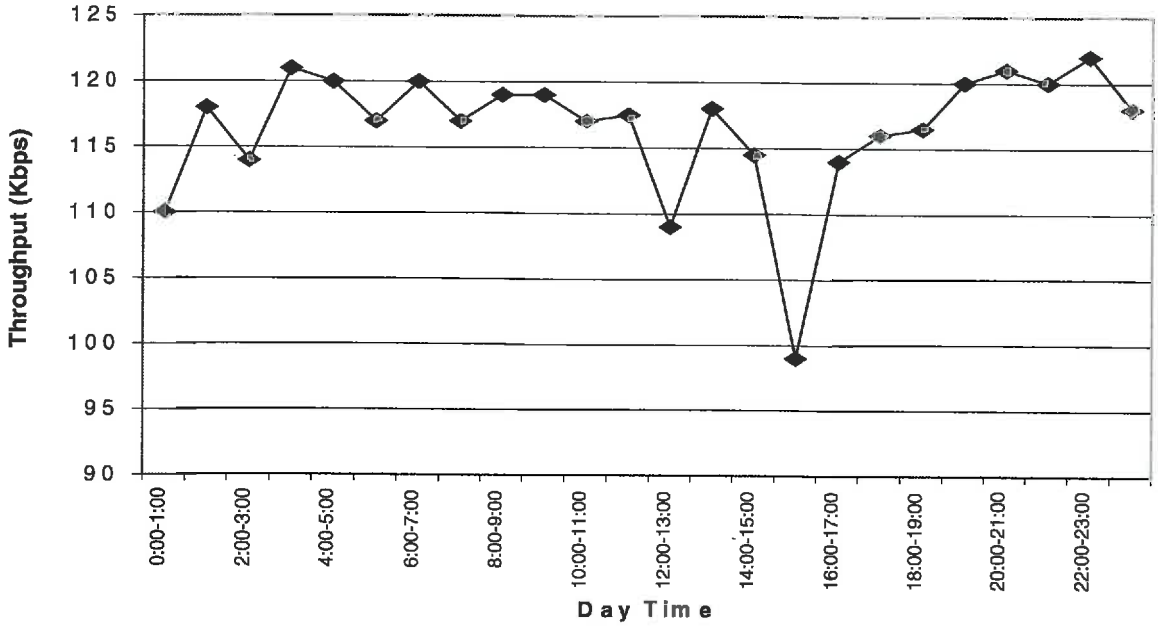


Figure 5.5 - Median Throughput for a Typical Weekday
USA WEST COAST
Period: 13/11 - 22/11



By observing the throughput behaviour for these typical paths, it was found that while median throughput mainly varies between 100 Kbps and 115 Kbps for the first typical path, throughput variation for the third path fluctuates from 110 Kbps and 120 Kbps (See Figs. 5.4 & 5.5, respectively).

If it is assumed there are no significant path asymmetries for this server and that traffic demand on Regional Networks, Australian backbone and Application Server has not changed considerably during this experimental period, it is possible to assert that this variation is due to packets looping back within the US for the first typical path. The drop in throughput performance due to this looping topology might be explained by at least one of the following reasons:

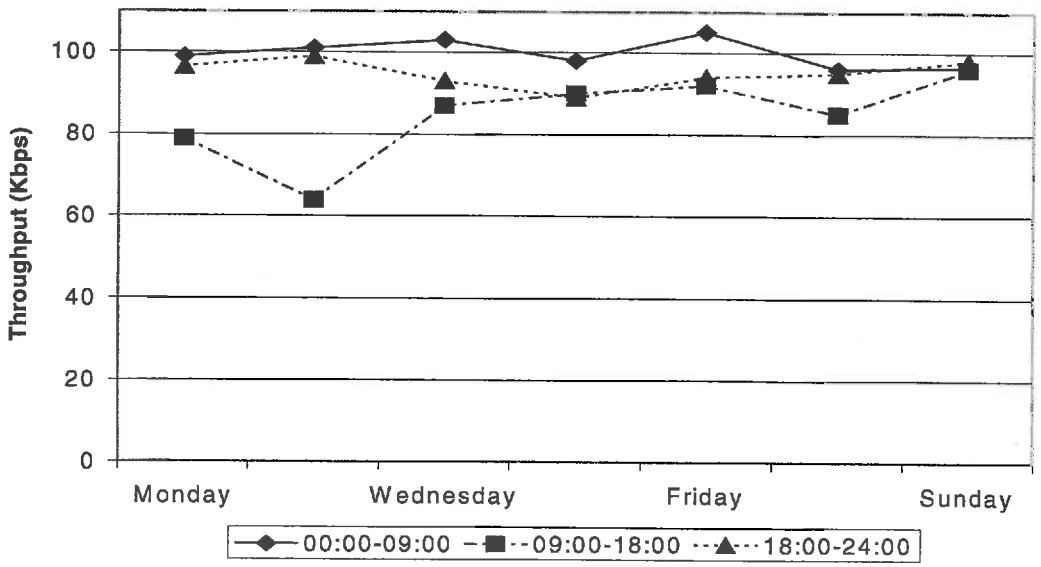
- Increase in RTT what results in longer delays for the receiver to acknowledge data received;
- Higher probability of packets crossing a congested or low bandwidth network.

5.2.4 USA East Coast

This site exhibited a performance characteristic very similar to the US West Coast typical path. Fig 5.6 does not follow completely Table 4.4, Chapter 4. In spite of the Off-peak time having a median throughput value higher than that during the Family time, the Business time throughput sometimes approaches the Family time throughput. This situation leads us to conclude there are contributions from the Application Server, Backbones and Regional Networks to the client throughput.

The median value is estimated at 100Kbps which is similar to the measurements for the US West Coast when packets were looping back in the US (See Figure 5.4). No significant changes in throughput were observed - in spite of this Application Server having two typical paths (See Appendix A, Tables 6 & 7).

Figure 5.6 - Median Throughput in the USA East Coast



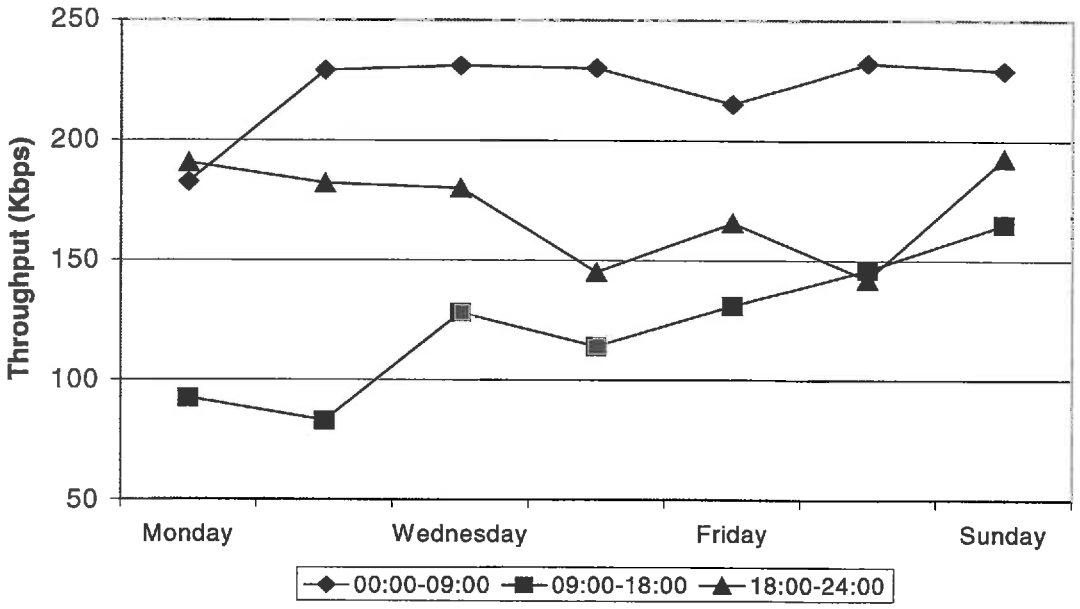
5.2.5 Hong Kong

As with other sites within Client 1, it was expected to collect throughput data from the experiment for a period of 20 days (See Figure 4.2, Chapter 4). However, a drop in throughput performance was experienced after the third day, which limited the data collection to 18 days. To attempt to explain the drop in performance, the same investigation for bottlenecks -as for the Victorian file server- was performed (See Section 5.2.2).

While it was difficult to determine a cause for this throughput decrease, a discussion with the backbone service provider highlighted the importance of peering arrangements that needs to be considered during any discussion of such throughput performance. From this it is possible to say the upstream path is likely to be different from the downstream path for this Application Server unlike other cases considered.

Due to service level requirements between AARNET and C&W Optus, traffic to Asia-Pacific destinations is routed via direct links to Asian peers *but* downloaded traffic to Australia is via U.S. peers. Under this situation it is not possible to say to what extent the backbone influences download throughput. However, Figure 5.7 does in fact exhibit a profile highly influenced by the Application Server traffic demand and conforms to our assumptions in Table 4.4 in Chapter 4.

Figure 5.7 - Median Throughput in Hong Kong

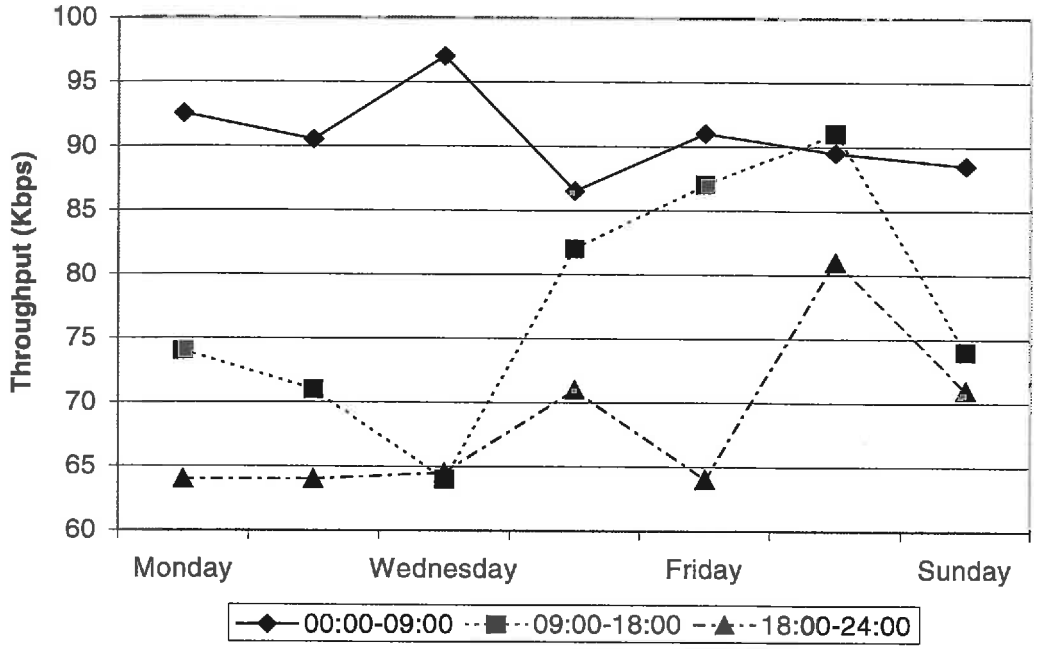


5.2.6 Israel

The Israeli site breaks with our assumptions on Internet usage profiles given in Table 4.4, Chapter 4. This is possibly due to the influence of the US backbone that was verified by path analysis.

Specifically it was found that the *Family time* in Table 4.4, Chapter 4, gives a transfer rate from the Israeli site *lower than* that of the Business time (See Fig. 5.8). Normally one would expect the reverse.

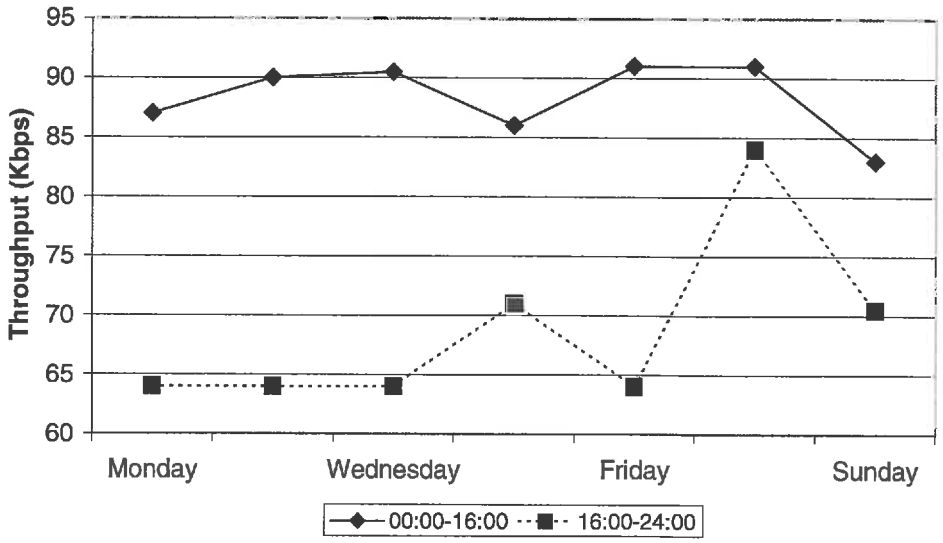
Figure 5.8 - Median Throughput in Israel



Why have a marked departure from our model happened? By assuming the upstream and downstream paths are the same, it is possible to say local traffic within the US backbone that requires local access in its own right might be affecting the throughput from the Israeli Application Server to the Client.

It is argued that at times when the dominant nodes in the New York region (e.g., NewYork.Teleglobe.net) are at their highest demand locally, this coincides with the Application Server usage *Family profile* in Israel (See Table 4.4, Chapter 4). To emphasise the different usage periods from those expected, days are sub-divided into two periods as seen in Fig. 5.9.

Figure 5.9 - Median Throughput in Israel Based on a Two-Period Usage Pattern



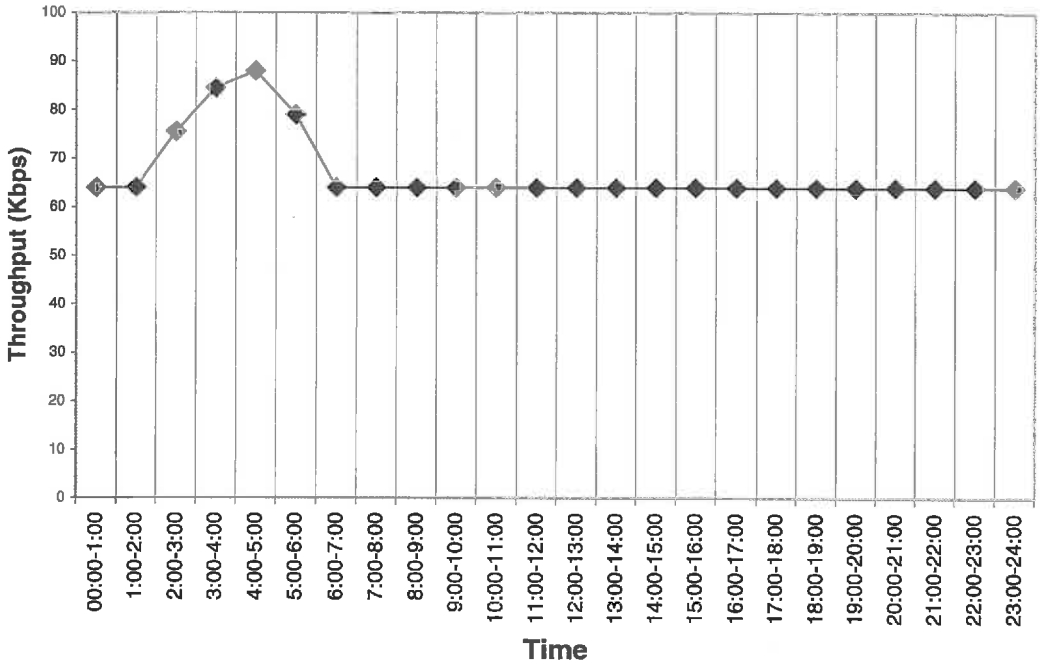
Therefore during the Off-peak time (00:00-16:00), the throughput has a maximum rate, which is the result of low traffic demand in Israel and local demand on the US backbone (specific to the East Coast).

Interestingly, a higher level of packet loss was not observed for the New York regional nodes compared to other regional nodes. A higher packet loss could suggest a higher traffic demand in those nodes. In order to be more definitive in our conclusions, more samples would be required.

5.2.7 Germany

Similar to the Israeli TUCOWS server, the German server did not follow Table 4.4 in Chapter 4. In order to emphasise this, a median value of throughput was chosen, which corresponds to a typical weekday value (See Fig. 5.10). This graph suggests a high influence of Backbone & Regional Networks on the results of this Application Server.

Figure 5.10 - Typical Median Throughput Weekday in Germany

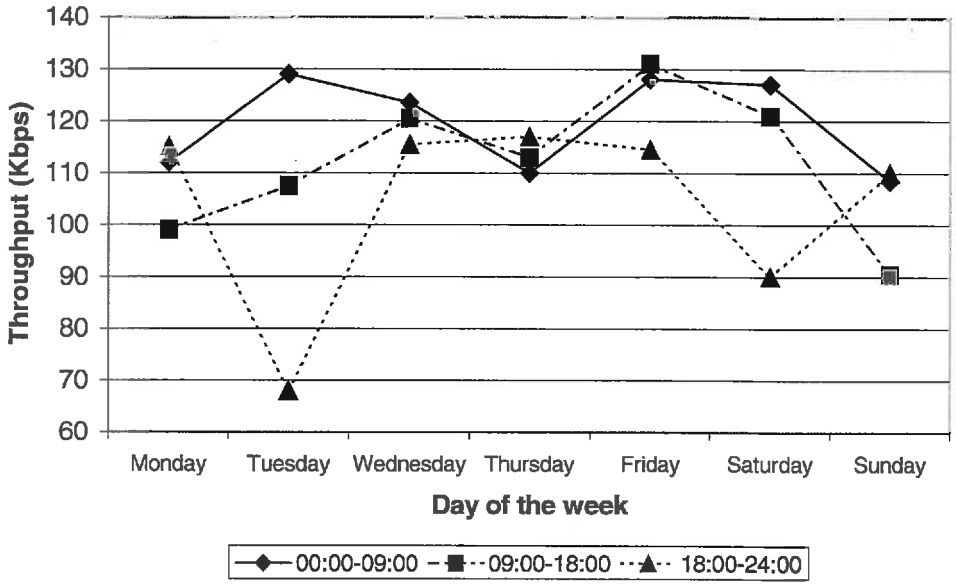


It was found that approximately 76 percent of all samples for this site equated to a transfer rate of 64Kbps or less, further, these samples lay in the period of Family and Business time (See Appendix A, Table 29). Again these results show the high influence of the US backbone to the extent that the only appreciable rise in throughput rate was observed during the Off-peak period as localised to the German Application Server. In order to be more definitive in our conclusions, this Server would require further analysis.

5.2.8 England

Fig. 5.11 suggests that backbone and regional networks influence this Application Server throughput and does not follow Table 4.4, Chapter 4. While it is hard to characterise the throughput behaviour for this server, more samples would be required for a better analysis.

Figure 5.11 - Median Throughput in England



5.2.9 Argentina, Brazil, South Africa & Zimbabwe

It was observed a great many throughput samples for these Application Servers having performance below 64 Kbps. One can also verify a higher number of "Timed Out" & "No server response" messages for these servers (See Appendix A, Table 29). However, it is important to note that this low performance behaviour is different from the German Server case. While the German server has a daily period with higher throughput (See Fig. 5.10), these servers have low performance sessions distributed *equally over all day periods.*

Because paths to most of the servers[1] have similar routes crossing the Australian and the American backbones, it is argued that poor performance for Argentina, Brazil, South Africa and Zimbabwe are caused by remote regional network/backbone performance limitations. Moreover, as it will be discussed in Section 5.4, typical paths for these Application Servers have more satellite links than that of other servers' paths. Because, satellite links might affect performance of TCP algorithms (as discussed in Section 3.5, Chapter 3), it is argued that low performance is also result of these extra satellite-links.

5.3 Path Instability

Based on the path instability analysis as described in Section 4.3.8.2.1 in Chapter 4, all paths have been analysed in terms of our path instability model (See Section 4.2.2, Chapter 4). While the software for path analysis proves to be a useful tool for finding changes in node configuration within a virtual path, it is found that *in some cases* a higher value of node changes has been mistakenly associated with an Application Server path. In some cases, sequential TRACERT samples contain different IP addresses for a particular node number, however these are *pseudo-node* changes. Such pseudo-node changes are described as follows:

- *Variations in node IP number having similar DNS information*: It was observed a number of samples where the IP address of a certain node changes in sequential samples *but* the DNS properties show the same network characteristics/geographical locations. The COMPARE *utility* does not consider changes where two IP Class C nodes in sequential samples have common network IDs but different hosts IDs. Since COMPARE does not analyse DNS information, for other types of IP addresses *pseudo-node* changes might happen (See Chapter 4, Section 4.3.8.2.1, *Utility* 5);

- *Cyclical Changes of IP addresses without DNS information:* It was found a number of samples where the IP address of a certain node changes in a cyclical fashion. Cyclical

[1] Except South Australia, Hong Kong & Victoria (Australia)

changes of Class C format IP addresses with common ID network are not considered by COMPARE *utility* as changes in path. However, for other IP address types, our software does consider cyclical changes as changes in path.

Path re-configurations complexity forced us to "filter" the results obtained from the COMPARE *utility* to achieve a fairest path instability (ε) analysis/calculation.

For the filtering process, it was not considered changes in path where the *varying* node reports loss of reachability. Nodes returning "host unreachable or net unreachable" messages are not adequate for analysing changes in path. Administrators of such nodes might be blocking ICMP echo-requests due to traffic policies or valid network problems might be occurring.

The *filtering* process is suitable for analysing data where there is a small number of pseudo-node changes to a Server. In addition, subjective assessment needs to be made in terms of whether a node change is a pseudo-node change or a valid change.

In the next section an analysis of the 'filtered' results covering the period between Nov 3 and Nov 22 1999 is provided. For ease reading, a general discussion of the main findings is introduced; this discussion is not organised in a server-by-server basis. The Path Instability analysis is summarised in Table 30 in Appendix A.

5.3.1 Path Instability Discussion

The first important finding in our study is the high stability of all client-server connections. One can observe in Table 30, Appendix A, that there are few changes in path in relation to the total number of samples. For all Application server paths, path changes affect less than 1% percent of the samples. This result confirms research carried out by Paxson [41], Labovitz *et al* [45] & Chinoy [46].

In addition, the number of changes in a path is not found relevant for previewing throughput performance. An Application Server might have lower throughput *but* higher number of path changes (or higher path instability) than other servers (eg, Application Server in Brazil).

The South Australian server path has the lowest Path Instability (ε). While it was observed that the COMPARE utility results in a reasonably high number of changes for this site, the path re-configurations do not seem to be result of congestion *but* of a higher level of capillarity within the Regional Backbone. It was observed that 20 of 24 changes for the South Australian path was result of packets being forwarded to a stand-by node 3 IP address (192.65.88.225). It was not observed DNS similarities between this IP address and the default node 3 IP address (See Table 1 in Appendix A: 203.16.212.13). However, since it was observed via Throughput analysis that there is *a low* influence of the Regional backbone/network over our results (See Section 5.2.1), these node changes are not considered for the Path Instability (ε) calculation.

For the USA East Coast server, a great many changes (260 occurrences) were found happening in node 7 between the Nov 3 and Nov 11 1999. While the COMPARE utility classifies these as changes, the analysis of the typical path for this Application Server (See Appendix A, Table 6) shows these are pseudo-changes due to a *load balance* in node 7 (*166.49.11.69 & 204.70.10.9*).

The Server in England had a high number of changes in path (See Appendix A, Table 30). While analysing the output, it was noted that this Application Server path had a high number of *Net/Host unreachable* occurrences within varying nodes. It was observed that 38 path changes happened in nodes reporting loss of reachability.

The Server in Argentina had 240 changes. Similarly with the USA East Coast Server, a great many changes (151) were due to a load balance implementation in node 15 between Nov 3 and Nov 15 1999. The load balance is confirmed by analysing Table 16 in Appendix

A. Moreover, some changes are due to variations in node IP number having similar DNS information (41) and others are not clear due to lack of DNS information (13). The rest of the path changes are due to nodes responding to "net/host unreachable" messages and therefore are not considered for the Path Instability (ε) calculation.

The Server in Israel had 88 changes. The analysis shows that a number of changes (26) are due to nodes responding to "net/host unreachable" messages. The other changes are again *pseudo-node* changes due to variations in node IP number having similar DNS information.

The South African Server had 6 pseudo-changes in result of a load balance implementation and 3 result of variations in node IP number having similar DNS information. The rest of the changes were not considered for the path instability (ε) calculation for having "net/host unreachable" messages within varying nodes.

The Servers in the US West Coast, Germany, Brazil and Zimbabwe had all pseudo-node changes due *to variations* in node IP number having *similar* DNS information.

5.4 Path Characteristics Based on the Minimum RTT Analysis

In this section the minimum typical path characteristics for each Application Server are discussed (See Section 4.3.3, Chapter 4). Based on the measured minimum RTT, the type of propagation mediums that interconnects our Client to the Remote Servers is hypothesised. This is done by calculation as described in Section 4.3.4 in Chapter 4 and by assuming that Internet transmission links between continental cities are *fibre* links while links between intercontinental cities might be *satellite* or *fibre* links. For ease understanding, each calculation is discussed in a server-by-server basis.

5.4.1 South Australia

Since the South Australian server only has one typical path (See Appendix A, Table 1), this path is the *minimum typical path*. As discussed in Chapter 4, Section 4.3.4.2, any propagation delay for this Server can be disregarded and therefore the minimum RTT delay is used for finding a typical switching router delay. This value is used for all virtual paths for calculating the total switching delay within a path.

5.4.2 Victoria, Australia

Since this Server only has one typical path (See Appendix A, Table 2), this path is the *minimum typical path*. The *minimum measured RTT* was 13.33 ms.

This Application Server is approximately 662 Km from our Client; therefore, the two-way propagation delay is calculated as 6.62 ms. By adding the switching delay contributions of all routers within this path, the propagation delay is calculated as 14.98 ms (6.62+ 0.76 **x** 11)[1]. This is found to be a close value to our measured minimum RTT delay. For this case, deviations might be result of overestimating the router switching delays, measurement inaccuracy [33] or path asymmetries.

5.4.3 USA West Coast

It was found that the *minimum typical path* for this server is the path shown in Table 5 in Appendix A. The *minimum measured RTT* was 300.33 ms.

While calculating the minimum typical path, it was found three possible paths -as shown in Table 31 in Appendix A. Since it is assumed that all intra-continental links are fibre links, path differences are due to dissimilar medium considerations for the inter-Pacific links,

[1] For not being repetitive, the switching delay calculation is omitted to other servers. One can easily calculate it by using the Typical Switching Delay (0.76 ms) in Equation (10) from Chapter 3.

which can be all-fibre, all-satellite or hybrid fibre-satellite. By comparing *these minimum calculated RTTs* to the *minimum measured RTT*, it is argued that the link follows Path B in Table 31, which has a hybrid Trans-Pacific link. This hypothesis was confirmed by OPTUS, which is the Trans-Pacific link carrier for traffic requests originated within AARNET network sites. All *upstream traffic* to the US is sent via fibre links while *downstream traffic* flows via either cable or satellite links. The downstream link selection will depend on *both* the type of traffic as router-map filters are applied *and* on the capacity of the links (particularly, OPTUS has *much higher* satellite link capacity *than* fibre link capacity). For example, traffic coming from Asia-Pacific links has higher preference within OPTUS network and thus follows a downstream fibre link (See Section 5.4.5). This hybrid Trans-Pacific link implementation is used for supporting our discussion in the following sections.

5.4.4 USA East Coast

It was not possible to find a minimum typical path for this server due to ICMP echo-requests being blocked within a router as discussed in Section 5.1.4. Since our definition for a minimum typical path involves measuring an end-to-end RTT delay within a Client-Server connection (See Section 4.3.3 in Chapter 4), it was not viable to have a minimum measured RTT or identify the end-to-end network routing topology - necessary for the minimum RTT calculation.

5.4.5 Hong Kong

Since this Application Server only had a typical path during the experimental period and the upstream and downstream paths are asymmetric (See Section 5.1.5), it is only possible to say that the typical path measured is the minimum *upstream* typical path to the Server. The *minimum measured RTT* was found as 170 ms.

By considering this path asymmetry and assuming the standard US gateway for all AARNET traffic within California region, different path topologies are calculated as shown in Table 32 in Appendix A. For our calculation, the US gateway for all AARNET traffic is considered as being located in Los Angeles[1]. By comparing the calculated values to the minimum delay measured (170 ms), it is possible to say that path A within Table 32 has the best medium configuration choice (all fibre). Path A was confirmed in a discussion with OPTUS, the carrier that provides backbone interconnections for AARNET. Since Asia-Pacific links have higher local preference within OPTUS, traffic coming from these regions via the US will flow via fibre links. Deviations are justified by our discussion in Section 4.3.8.2.2, Chapter 4.

5.4.6 Israel

It was found that the minimum typical path for this server is the path shown in Table 9 in Appendix A. The minimum measured RTT was 471 ms.

While hypothesising about the medium configurations for this minimum typical path, two path configurations were found to have a RTT close to the minimum measured RTT. These are paths B & E shown in Table 33 in Appendix A. While continental links are fibre links (by assumption), path configuration differences between paths B & E are due to a hybrid Trans-Pacific link (Path B) or a hybrid Trans-Atlantic link (Path E). Since most of the Client-Server connections cross a standard US gateway in California (See Section 5.1.3) and the Trans-Pacific backbone provider has confirmed the hybrid fibre-satellite implementation within its link, it is possible to argue that Path B is the most adequate medium combination.

[1] San Francisco could also have been considered the standard US gateway. However, since San Francisco and Los Angeles are close cities, calculation of the propagation delay would not be much affected.

5.4.7 Germany

It was found that the *minimum typical path* for this server is the path shown in Table 13 in Appendix A. The *minimum measured RTT* was 437.33 ms.

Similarly to the Israeli Server, while calculating the minimum typical path, it was noticed two path configurations (See Paths B & E in Table 34, Appendix A) having a RTT close to the minimum measured RTT. Likewise, it is asserted that the best path is the one that has the hybrid Trans-Pacific link (Path B). Deviations might be result of measurement inaccuracies *or* path asymmetries.

5.4.8 England

It was found that the *minimum typical path* for this server is the path shown in Table 15 in Appendix A. The *minimum measured RTT* was 450.66 ms. Similarly to the Servers in Israel and Germany, two possible path configurations were observed (Paths B & E, Table 35). Again, the path was selected based on the hybrid Trans-Pacific link characteristic (Path B).

5.4.9 Argentina

It was found that the *minimum typical path* for this server is the path shown in Table 16 in Appendix A. The *minimum measured RTT* was 788 ms. It was noticed that three different paths have RTT close to this value, Paths C, G & J, Table 36.

Path G shows a hybrid Trans-Atlantic link and a satellite Trans-Pacific link. As already discussed in Sections 5.4.3, 5.4.6, 5.4.7 & 5.4.8, this is not an adequate path configuration because the Trans-Pacific link is a hybrid link.

Path I shows a fibre Trans-Pacific link and a satellite Trans-Atlantic link. Similarly to Path G, this is not suitable because the Trans-Pacific is a hybrid link.

Therefore, the most appropriate path configuration is Path C, which has hybrid configuration for both Trans-Pacific and Trans-Atlantic links. Deviations from the observed value might be due to path asymmetries & measurement inaccuracies.

5.4.10 Brazil

It was found that the *minimum typical path* for this server is the path shown in Table 18 in Appendix A. The *minimum measured RTT* was 691 ms. Similarly to the Server in Argentina, it was noticed three different paths that have RTT close to this value, Paths C, F & J, Table 37.

Path F shows a satellite Trans-Pacific link and a fibre Trans-Atlantic. As discussed before, the Trans-Pacific link was found to be hybrid and this path is not adequate. Similarly, Path I is not suitable because it has a fibre Trans-Pacific link and a satellite Trans-Atlantic link. Therefore, Path C is the best configuration option with two hybrid intercontinental links. Deviations from the minimum measured RTT are result of path asymmetries and measurement inaccuracies.

5.4.11 South Africa

It was found that the *minimum typical path* for this server is the path shown in Table 20 in Appendix A. The *minimum measured RTT* was 604.33 ms. While it was observed that several paths have similar RTT to the measured minimum RTT (See Table 38 in Appendix A), there are only two paths that have hybrid Trans-Pacific links: Paths B & C. The main difference between these paths is that while Path B has a fibre Trans-Atlantic link, Path C has a hybrid Trans-Atlantic link. This provides a difficult task in selecting the correct path

due to inaccuracies in measurement and path asymmetries (See Section 4.3.8.2.2, Chapter 4).

5.4.12 Zimbabwe

It was found that the *minimum typical path* for this server is the path shown in Table 23 in Appendix A. The *minimum measured RTT* was 851.33 ms. It was observed that two paths have similar values to this minimum RTT: Paths D & G. By observing that Path G has a satellite Trans-Pacific link, it is asserted that Path D is the best option because it has a hybrid Trans-Pacific link and a satellite Trans-Atlantic link. Deviations are due to path asymmetries and measurement inaccuracies.

5.5 Summary

This chapter provides a summary of the main highlights of our research. Firstly, *typical virtual paths* were presented for each client-server connection. The classification followed Section 4.3.2, Chapter 4. For each typical virtual path, routing topologies were identified and discussed. *Fluttering* and *tightly coupled routers* were observed in a number of typical virtual paths. In some cases, *routing pathologies* were also noted.

In addition, the analysis of typical paths provided an extra facility for evaluating the influence of interconnecting networks on throughput performance. This was possible because *downstream* and *upstream* paths were considered approximately symmetric. While this was an appropriate assumption in most cases, path *asymmetries* might happen due to peering agreements as observed in the Hong Kong case. When path asymmetries are present, typical paths *do not* provide accurate information about the downstream path and, therefore, it is not possible to evaluate throughput behaviour from TRACERT inspection.

In terms of *throughput*, it was observed that when interconnecting networks have *low* influence on a client-server connection, throughput behaviour follows Table 4.4, Chapter 4. In this case, throughput is *significantly* affected by traffic demand on the Application Server. In cases where throughput is influenced by interconnecting networks, deviations from Table 4.4 are expected. For example, the German and Israeli cases suggested interconnecting networks within the US had a negative effect on throughput behaviour.

In terms of *path instability*, a study of end-to-end path instability was provided to each Application Server. In order to provide a more accurate investigation of measured data, a 'filtering' process was used for eliminating *pseudo-node* changes, i.e., node changes that could have been incorrectly associated by our software to an application server path. The results showed *overall high* path stability, confirming research carried out by Paxson [41], Labovitz [45] and Chinoy [46]. It was also observed that path instability *is not* relevant for estimating throughput performance.

In terms of *minimum RTT analysis*, the main goal was to estimate interconnecting link characteristics between client and servers. Firstly, *minimum typical paths* were identified for each client-server connection. Subsequently, by assuming that Internet transmission links between continental cities were *fibre* links and links between intercontinental cities were *either* satellite *or* fibre links, it was possible to calculate potential values of *minimum RTT* for each client-server connection. For each client-server connection, the comparison of the calculated *minimum RTTs* and the *measured minimum RTT* provided an adequate analysis methodology for hypothesising path characteristics.

Chapter 6
Conclusions & Further Work

6.1 Conclusions

Our research describes a new methodology to assess end-to-end performance analysis of broadband access networks and Internet backbones allowing the user to generate a picture of traffic profiles to chosen Application Servers throughout the world. This methodology is a hybrid application-network layer analysis, providing a better understanding of network factors that influence throughput performance. The first part of our methodology consisted in developing software utilities for collecting TRACERT samples and measuring throughput to different Application Servers. The second part comprised in a performance analysis based on four Sections: Client-Server typical paths, Throughput analysis, Path Instability analysis & minimum typical path analysis.

The analysis of TRACERT samples allows us to identify typical paths to different Application Servers. These paths are useful not only for developing the concept of *minimum typical paths* but also for understanding changes in throughput behaviour during the experimental time.

It was observed that throughput is highly influenced by traffic demand on the Application Server and traffic conditions within interconnecting networks. One of the main findings is the high influence of the US backbone on Application Servers located overseas. All overseas Application Servers had virtual paths crossing the US backbone. In some cases, throughput variations were observed in a cyclical fashion and we assert these were due to traffic demand within the US backbone (See Israel & Germany cases).

It was also noticed that throughput depends not only on technical but also on economic drivers (See Victoria case). The inter-network pricing regime and the way this impacts on available bandwidth are important aspects that were observed in this research but were not analysed. For other factors that impact on total throughput, it is recommended reading [98]. For example, TCP protocol configurations running on different Operating Systems have significant influence on TCP total throughput [99]. This topic was not investigated in our research.

It was observed a low performance throughput for Application Servers in countries with less developed national network infrastructure. We argue that this low performance is due to remote regional network/backbone limitations and due to a higher number of satellite links within these paths – which were observed via minimum typical path analysis.

In terms of Path instability, our Research has confirmed research carried our by Paxson, Labovitz *et al* & Chinoy, where path changes affect less than 1% of Internet connections [41, 45, 46]. In addition, a parameter (ε) was introduced for evaluating path instability within a virtual path. Since this parameter can be used for comparing path instability of distinct virtual paths, it can be considered a "QoS" analysis parameter. This study was not focused in finding an optimum value for (ε) *but* in comparing the instability behaviour of the Client-Application Server paths.

The minimum typical path analysis provides a useful tool for analysing path characteristics within a virtual path. By simulating different inter-city propagation mediums within a minimum typical path and calculating the respective minimum RTTs, it is possible to find the best path topology for matching the measured minimum RTT.

The analysis of the throughput and path parameters result in an "Internet weather forecast". These ideas could possibly be used for monitoring or previewing the performance of an internetworking environment.

The research suggests current Internet networks and heavily loaded Web File Servers provide a range of transfer rates up to 1 Mbps. Thus, new broadband access technologies such as Asymmetric Digital Subscriber Line (ADSL), Cable Modems and Local Multipoint Distribution Service (LMDS) cannot provide downstream throughput to their full capability in the current Internet environment.

6.2 Further Work

While the method holds for the given analysis of path instability and throughput, more definitive conclusions in terms of these parameters could be further obtained if more samples are collected. For improvement in throughput analysis the SBF could be smaller and/or the Client could run in a dedicated mode for a particular Application Server. In the case of path analysis, the sample collection interval could be decreased.

For the path instability analysis, a more automated system for "filtering" the samples could shorten the time necessary for analysing node changes. Such a tool could improve path instability analysis and make the system more robust for a larger number of sites.

While classifying the typical virtual paths, it was observed that the classification process is not an obvious task, demanding the analysis of RTT replies for each node within a virtual path for identifying different paths and routing topologies. Further work should be carried to automate the process, making it more appropriate for a larger number of measurements.

In the general case, it was assumed that upstream and downstream paths are likely the same. Nevertheless, in particular cases such as Hong Kong, the paths are different because of peering agreements between ISPs and their backbone providers. In this case throughput can be affected by different time zones (i.e., USA West Coast) causing inaccurate conclusions. To solve this problem and by the way of further work, the path analyser software could be run at the target Application Server instead of running at the Client

Machine. This would provide a better estimate of the throughput path in the downstream direction.

Another improvement that would result in better analysis is separating the packet generation process from the measurement process. As discussed by Cleary *et al* [33], RTT measurements might differ up to 30 ms when the probing packet generation and data measuring processes are carried out in the same machine.

References

[1] *B.M. Leiner, V. G. Cerf, D. D. Clark, R. E. Kahn, L. Kleinrock, D. C. Lynch, J. Postel, L. Roberts, S. Wolff,* "A Brief History of the Internet", Internet Society, http://www.isoc.org/internet-history/brief.html

[2] *R. J. Vetter, C. Spell & C. Ward,* "Mosaic and the World-Wide Web", IEEE Computer magazine, Vol. 27, No. 10, pp. 49-57, Oct. 1994.

[3]*G. Madden & S. J. Savage,* "Pricing and Residential Internet Traffic", Communications Economics Research Program (CERP), School of Economics and Finance, Curtin University of Technology, June 1998.

[4] *J. Skoro,* "The Future of Multimedia", America's Network magazine, June 1, 1999, http://www.americasnetwork.com/issues/99supplements/990601lmds/990601_future.htm

[5] *W. E. Kennard,* "A Broad(band) vision for America", presented at the Federal Communications Bar Association, USA, June 24, 1998.

[6] National Science Foundation (NSF), "Report on the NSF-Sponsored Workshop on Internet Statistics Measurement and Analysis", February 19-20, 1996, http://www.caida.org/outreach/isma/9602/report/

[7] *T. C. Kwok,* "Residential Broadband Internet Services and Applications Requirements", IEEE Communications magazine, Vol. 35, No. 6, pp. 76-83, June 1997.

[8] *V. Jacobson,* "Congestion Avoidance and Control," Computer Communications Review, Vol. 18, No. 4, pp 314-329, August 1988

[9] *B. S. Arnaud,* "CANARIE," presented at "The AARNET Advanced Internet Workshop", Adelaide, Australia, 10th & 11th March 2000. (Unpublished)

[10] *J. W. Gurley,* "Can Napster be stopped? No! ", April 17, 2000, CNET.com, http://www.news.com/Perspectives/Column/0,176,419,00.html

[11] *K. Reichard,* "Napster on Linux: From a Whisper to a Scream", March 20, 2000, LinuxPlanet, http://www.linuxplanet.com/linuxplanet/reviews/1617/1/

[12] *S. Zeidler,* "Napster Changes its Tune to Beat Ban Internet", The Orange County Register, March 24, 2000.

[13] *C. Oakes,* "Napster not at Home with Cable", April 7, 2000, Wired News, http://www.wired.com/news/print/0%2C1294%2C35523%2C00.html

[14] "Advanced Networking Infrastructure Needs in the Atmospheric and Related Sciences (ANINARS) report", final draft, July 21, 1999, http://www.scd.ucar.edu/nets/projects/NETSprojectplans/1999.complete.projects/nlanr/final.report.doc

[15] *D. Newman & R. Mandeville,* "Corporate-Class Internet? Don't Count on It!", Data.com magazine, November 1998.

[16] *K. Thompson, G. J. Miller & R. Wilder,* "Wide-Area Internet Traffic Patterns and Characteristics (Extended Version)", IEEE Network, Volume 11, No. 6, pp. 10-23, November/December 1997.

[17] *V. Jacobson,* "Traceroute", 1989, Network Research Group, Lawrence Berkeley National Laboratory, http://www-nrg.ee.lbl.gov/

[18] *B. Jew & R. Nicholls,* "Internet Connectivity: Open Competition in the Face of Commercial Expansion", presented at the "Pacific Telecommunications Conference", Honolulu, January 1999, http://gtlaw.com.au/pubs/opencompetition.html

[19] *V. Paxson, J. Mahdavi, A. Adams & M. Mathis,* "An Architecture for Large-Scale Internet Measurement", IEEE Communications magazine, Vol. 36, No. 8, pp. 48-54, Aug. 1998

[20] *K. Claffy & S. McCreary,* "Internet Measurement and Data Analysis: Passive and Active Measurement", Cooperative Association for Internet Data Analysis, http://www.caida.org/outreach/papers/Nae/4hansen.html

[21] *A. Tanenbaum,* "Computer Networks", 3rd edition, Chapter 6, pp. 555-572, Prentice Hall PTR, USA, 1996.

[22] *D. Blacharski,* "The Changing Face of Service Level Agreements", Network magazine, pp. 94-97, January 2000.

[23] "Recommendations for Internet Routing", Internet Performance Measurement and Analysis Project, Merit Network, http://www.merit.edu/ipma/docs/help.html

[24] *H. Kruse,* "Performance of Common Data Communications Protocols Over Long Delay Links – An Experimental Examination", 3rd International Conference on Telecommunication Systems Modeling and Design, pp. 409-415, 1995, http://jarok.cs.ohiou.edu/papers/kruse.ps

[25] *M. Allman, C. Hayes, H. Kruse & S. Ostermann,* "TCP Performance over Satellite Links", in Proceedings 5th International Conference on Telecommunications Systems, 1997, http://roland.grc.nasa.gov/~mallman/papers/nash97.ps

[26] *M. Allman, S. Dawkins, D. Glover, J. Griner, D. Tran, T. Henderson, J. Heidemann, J. Touch, H. Kruse, S. Ostermann, K. Scott & J. Semke,* "Ongoing TCP Research Related to Satellites", RFC2760, IETF, 2000, http://www.ietf.org/internet-drafts/draft-ietf-tcpsat-res-issues-12.txt

[27] *H. Heimlich,* "Traffic Characterisation of the NSFNET Backbone", USENIX Conference Proceedings, Winter 1989, http://www.research.att.com/~jrex/papers/ton98.ps

[28] *K. Claffy, H.W. Braun & G. Polyzos,* "Long Term Traffic Aspects of the NSFNET", Proceedings of INET'93, 1993, http://ftp.sdsc.edu/pub/sdsc/anr/papers/inet93.NSFNETtraffic.ps.Z

[29] *V. Paxson,* "Growth Trends in Wide-Area TCP Connections", IEEE Network, Vol. 8, No. 4, pp. 8-17, July/August 1994.

[30] *S. McCreary & KC Claffy,* "Trends in Wide Area IP Traffic Patterns – A View from Ames Internet Exchange", Cooperative Association for Internet Data Analysis, http://www.caida.org/outreach/papers/AIX0005/

[31] *A. Mena & J. Heidemann,* "An Empirical Study of Real Audio Traffic", in Proceedings IEEE INFOCOM 2000, Vol. 1, pp. 101-110, 2000, http://www.ieee-infocom.org/2000/papers/84.ps

[32] *H. S. Martin, A. J. McGregor & J. G. Cleary,* "Analysis of Internet Delay Times", in Proceedings "Passive & Active Measurement Workshop", Hamilton, New Zealand, April 3 & 4, 2000.

[33] *J. Cleary, S. Donnelly, I. Graham, A. McGregor & M. Pearson,* "Design Principles for Accurate Passive Measurement", in Proceedings "Passive & Active Measurement Workshop", Hamilton, New Zealand, April 3 & 4, 2000.

[34] *T. McGregor, H. W. Braun & J. Brown,* "The NLANR Network Analysis Infrastructure", IEEE Communications magazine, Vol. 38, No. 3, pp. 122-128, May 2000.

[35] *V. Paxson, A. K. Adams & M. Mathis,* "Experiences with NIMI," in Proceedings "Passive & Active Measurement Workshop", Hamilton, New Zealand, April 3 & 4, 2000.

[36] *M. W. Garrett, C. Huitema, J. DesMarais & W. Leland,* "Project Felix: Independent Monitoring for Network Survivability", Telcordia Technologies, http://govt.argreenhouse.com/felix/

[37] "Active Measurement Project", National Laboratory for Applied Network Research, http://amp.nlanr.net

[38] *W. Mattews & L. Cottrell,* "The Pinger Project: Active Internet Performance Monitoring for the HENP Community", IEEE Communications magazine, Vol. 38, No. 3, pp. 130-136, May 2000.

[39] *S. Kalidindi & M. J. Zekauskas,* "Surveyor: An Infrastructure for Internet Performance Measurements", Advanced Network & Services, http://telesto.advanced.org/~kalidindi/papers/INET/inet99.html

[40] AMP Project, "RTT measurements", National Laboratory for Applied Network Research, http://amp.nlanr.net/active/

[41] *V. Paxson,* "End-to-End Routing behaviour in the Internet", IEEE/ACM transactions on Networking, Vol. 5, No. 5, pp. 601-615, Oct. 1997.

[42] *I. Bilinskis & A. Mikelsons,* "Randomized Signal Processing", Prentice Hall International, 1992.

[43] *W.R. Stevens & G. R. Wright,* "TCP/IP Illustrated – Volume 1", 16th printing, Chapter 21, pp. 312-316, Addison-Wesley, 2000.

[44] *C. Huitema,* "Routing in the Internet", Prentice Hall PTR, 1st edition, 1995.

[45] *C. Labovitz, G.R. Malan & F. Jahanian,* "Internet Routing Instability", in Proceedings of ACM SIGCOMM' 97, pp. 115-126, 1997.

[46] *B. Chinoy,* "Dynamics of Routing Information", in Proceedings SIGCOMM' 93, pp. 45-52, 1993.

[47] *B. Huffaker, M. Fomenkov, D. Moore & E. Nemeth,* "Measurements of the Internet topology in the Asia-Pacific Region", in Proceedings Inet 2000, Cooperative Association for Internet Data Analysis, http://www.caida.org/outreach/papers/asia_paper/

[48] *J. Angel,* "Toll Lanes on the Information Superhighway", Network magazine, pp. 42-49, January 2000.

[49] *K. Claffy, G. Polyzos & H-W. Braun,* "Measurement Considerations for Assessing Unidirectional Latencies", Internetworking: Research and Experience, Vol. 4, No. 3, pp. 121-132, September 1993.

[50] *H. S. Cheng, L. H. Ngoh, J. Ong, C. K. Yip, S. M. Ow & S. Lim,* "Measuring IP Network Performance – The SingAREN's Approach", in Proceedings "Passive & Active Measurement Workshop", Hamilton, New Zealand, April 3 & 4, 2000.

[51] *D. Moore, R. Periakaruppan & J. Donohoe,* "Where in the World is netgeo.caida.org?", in Proceedings Inet2000, Japan, 18-21 July 2000.

[52] "Executive Summary – Computer Network Time Synchronization", Electrical & Computer Engineering Department, University of Delaware, http://www.eecis.udel.edu/~ntp/ntp_spool/html/exec.htm

[53] *Ohta K., G. Mansfield, N. Kato & Y. Nemoto* , "Wide Area Fault Detection by Monitoring Aggregated Traffic", in Proceedings "Passive & Active Measurement Workshop", Hamilton, New Zealand, April 3 & 4, 2000.

[54] *C. Alaettinoglu, C. Villamizar, E. Gerich, D. Kessens, D. Meyer, T. Bates, D. Karrenberg & M. Terpstra,* "Routing Policy Specification Language (RPSL)", RFC2622, IETF, 1999, http://www.ietf.org/rfc/rfc2622.txt?number=2622

[55] "IP Measurement Protocol", National Laboratory for Applied Network Research, http://watt.nlanr.net/AMP/IPMP/

[56] *R. Govindan & A. Reddy,* "An Analysis of Internet Inter-Domain Topology and Route Stability", in Proceedings IEEE INFOCOM 1997, Vol. 2, pp. 850-857, 1997.

[57] *K. Lougheed & Y. Rekhter,* "A Border Gateway Protocol (BGP)", RFC1163, IETF, 1990, http://www.ietf.org/rfc/rfc1163.txt?number=1163

[58] *A. Kumar, M. Hegde, S. V. R. Anand, B. N. Bindu, D. Thirumurthy & A. A. Kherani,* "Nonintrusive TCP Connection Admission Control for Bandwidth Management of an Internet Access Link", IEEE Communications magazine, Vol. 38, No. 3, pp. 160-167, May 2000.

[59] "The DSL Source Book – Plain Answers About Digital Subscriber Line Opportunities", Second Edition, http://www.paradyne.com

[60] "VDSL – Frequently Asked Questions", DSL FORUM, http://www.adsl.com/vdsl_faq.html

[61] *A. Tanenbaum,* "Computer Networks", 3rd edition, Chapter 2, pp. 84-85, Prentice Hall PTR, USA, 1996.

[62] *M. Mead,* "Propagation Impairment at 28 GHz", America's Network magazine, LMDS supplement, June 15, 1998, http://www.americasnetwork.com/issues/98supplements/980615lmds/980615_lmdsalcatel.html

[63] *P. B. Papazian, G. A. Hufford, R. J. Achatz & R. Hoffman,* "Study of the Local Multipoint Distribution Service Radio Channel", IEEE Transactions on Broadcasting, Vol. 43, No. 2, pp. 175-184, June 1997.

[64] *P. Clark-Dickon,* "The Internet's Orbital Inclination", Australian Communications magazine, pp. 59-68, November 1998.

[65] "Extended Frame Sizes for Next Generation Ethernets – A White Paper", Alteon Networks, http://www.alteonwebsystems.com/products/whitepapers/jumboframes/

[66] *P. Dykstra,* "Gigabit Ethernet Jumbo Frames – And why you should care", 20 December 1999, WareOnEarth Communications, http://sd.wareonearth.com/~phil/jumbo.html

[67] *M. Mathis, J. Semke & J. Mahdavi,* "The Macroscopic Behaviour of the TCP Congestion Avoidance Algorithm", Computer Communication Review, Vol. 27, No. 3, pp. 67-82, July 1997, http://www.psc.edu/networking/papers/model_ccr97.ps

[68] *C. A. Kent & J. F. Mogul,* "Fragmentation Considered Harmful", Computer Communication Review, Vol. 17, No. 5, pp. 390-401, http://www.research.digital.com/wrl/techreports/abstracts/87.3.html

[69] *M. Allman, D. Glover & L. Sanchez,* "Enhancing TCP Over Satellite Channels using Standard Mechanisms", RFC2488, IETF, 1999, http://www.ietf.org/rfc/rfc2488.txt?number=2488

[70] *R.L. Fink,* "IPv6 – What and Where It Is", The Internet Protocol Journal, Vol. 2, No. 1, pp. 17-29, March 1999.

[71] *T. Parker,* "Wireless ATM Flexes Its Muscle", Telephony magazine, November 17, 1997, http://www1.telecomclick.com/magazinearticle.asp?releaseid=2013&magazinearticleid=7814

[72] *Armitage G. J. & Adams K. M.,* "How Efficient is IP Over ATM Anyway? ", IEEE Network magazine, Vol. 9, pp. 18-26, January/February 1995.

[73] *S. Knowles,* "IESG Advice from Experience with Path MTU Discovery", RFC1435, IETF, 1993, http://www.ietf.org/rfc/rfc1435.txt?number=1435

[74] *J. Mogul,* "Path MTU Discovery", RFC1191, IETF, 1990, http://www.ietf.org/rfc/rfc1191.txt?number=1191

[75] "Automatic Bandwidth Delay Product Discovery", Web100 Project, http://www.web100.org/papers/bdp.discovery.html

[76] *J. Semke, J. Mahdavi & M. Mathis,* "Automatic TCP Buffer Tuning", Computer Communication Review, Vol. 28, No. 4, pp. 315-323, October 1998, http://www.psc.edu/networking/ftp/papers/autotune_sigcomm98.ps

[77] *W.R. Stevens & G. R. Wright,* "TCP/IP Illustrated – Volume 1", 16th printing, Chapter 24, pp. 344-345, Addison-Wesley, 2000.

[78] *V. Jacobson, R. Braden & D. Borman,* "TCP Extensions for High Performance", RFC1323, IETF, 1992, http://www.ietf.org/rfc/rfc1323.txt?number=1323

[79] *V. Jacobson,* "Modified TCP Congestion Avoidance Algorithm", end2end-interest mailing list, April 30, 1990.

[80] *M. Mathis, J. Mahdavi, S. Floyd & A. Romanow,* "TCP Selective Acknowledgment Options", RFC2018, IETF, 1996, http://www.ietf.org/rfc/rfc2018.txt?number=2018

[81] *M. Allman,* "Improving TCP Performance Over Satellite Channels", Master's thesis, Ohio University, June 1997, http://roland.grc.nasa.gov/~mallman/papers/thesis.ps

[82] *M. Allman,* "A Web Server's View of the Transport Layer", ACM Computer Communication Review, Vol. 30, No. 5, October 2000, http://roland.grc.nasa.gov/~mallman/papers/webobs-ccr.ps

[83] *M. Allman & Aaron Falk,* "On the Effective Evaluation of TCP", Computer Communication Review, Vol. 29, No. 5, pp. 59-70, October 1999, http://roland.grc.nasa.gov/~mallman/papers/tcp-evaluation.ps

[84] "SkyX Gateway Technology White Paper", Mentat Inc., http://www.mentat.com/skyx/whitepaper.html

[85] *W.R. Stevens & G. R. Wright,* "TCP/IP Illustrated – Volume 3", 7th printing, Chapter 13, pp. 170-172, Addison-Wesley, 2000.

[86] *A. Feldmann, R. Caceres, F. Douglis, G. Glass & M. Rabinovich,* "Performance of Web Proxy Caching in Heterogeneous Bandwidth Environments", in Proceedings INFOCOM 1999, Vol. 1, pp. 107-116, 1999, http://www.ieee-infocom.org/1999/papers/01d_01.pdf

[87] *G. Barish & K. Obraczka,* "World Wide Web Caching: Trends and Techniques", IEEE Communications magazine, Vol. 38, No. 3, pp. 178-185, May 2000.

[88] "The Economics Impacts of Unacceptable Web-Site Download Speeds", Zona Research Inc., April 1999, http://www.zonaresearch.com/deliverables/white_papers/wp17/index.htm

[89] *J. Postel,* "Transmission Control Protocol", RFC793, IETF, 1981, http://www.ietf.org/rfc/rfc0793.txt?number=793

[90] *R. Fielding, J. Gettys, J. Mogul, H. Frystyk & T. Barners-Lee,* "Hypertext Transport Protocol – HTTP/1.1", RFC2068, IETF, 1997, http://www.ietf.org/rfc/rfc2068.txt?number=2068

[91] *J. Postel & J. K. Reynolds,* "File Transfer Protocol (FTP)", RFC959, IETF, 1985, http://www.ietf.org/rfc/rfc0959.txt?number=959

[92] *T. Faber, J. Touch & W. Yue,* "The TIME-WAIT state in TCP and Its Effect on Busy Servers", IEEE INFOCOM 1999, Vol. 3, pp. 1573-1583, 1999, http://www.ieee-infocom.org/1999/papers/11c_04.pdf

[93] "New Protocol to Download Movies as Files Rather Than Streaming", CANARIE, News List: Canet-3-NEWS@canarie.ca, 18 September 2000.

[94] *A. Dornan,* "Farming Out the Web Servers", Network magazine, pp. 86-90, March 2000.

[95] "Australia's International Bandwidth", Telecommunication Journal of Australia, Vol. 49, No. 1, pp. 3-12, 1999.

[96] *A. Myers, P. Dinda & H. Zhang,* "Performance Characteristics of Mirror Servers on the Internet", in Proceedings *IEEE INFOCOM 1999*, Vol. 1, pp. 304-312, 1999, http://www.ieee-infocom.org/1999/papers/

[97] "Map Viewer Frequently Asked Questions", Xerox Palo Alto Research Center, http://www.parc.xerox.com/istl/projects/mapdocs/mapviewer-faq.html

[98] "Network Analysis Times", Vol. 1, Jan 2000, National Laboratory for Applied Network Research, http://moat.nlanr.net/NATimes

[99] "Enabling high performance data transfers on hosts: notes for user and system administrators", Pittsburgh Supercomputing Center, http://www.psc.edu/networking/perf_tune.html

Appendix A:

Dataset Tables

Table 1: South Australia Typical Path

Node Location	Node	IP Field	DNS Field	RTT Occurrences
Local Network	1	129.127.180.253	vlan0180.atm2-0.pancho.net.adelaide.edu.au	5671
Reg. Network	2	203.21.37.2	lis255.atm1-0.central.saard.net	5685
Reg. Backbone	3	**203.16.212.13**	**fa0-0-108.boomerang.internode.on.net**	**5582**
Reg. Backbone	4	198.32.240.100	n/a	5581
Reg. Backbone	**5**	**203.34.35.85**	**ser-0-2-bigpipe-grote-adl.bna.com.au**	**1400**
Reg. Backbone	**5**	**203.34.35.89**	**ser-0-3-bigpipe-grote-adl.bna.com.au**	**1390**
Reg. Backbone	**5**	**203.34.35.130**	**ser-0-1-bigpipe-grote-adl.bna.com.au**	**1383**
Reg. Backbone	**5**	**203.34.35.82**	**ser-1-0-bigpipe-grote-adl.bna.com.au**	**1397**
Reg. Network	6	203.56.239.98	nostromo.senet.com.au	5579
Remote LAN	7	203.152.224.5	www.ozbytes.net.au	5566

Table 2: Victoria Typical Path

Node Location	Node	IP Field	DNS Field	RTT Occurrences
Local Network	1	129.127.180.253	vlan0180.atm2-0.pancho.net.adelaide.edu.au	5674
Reg. Network	2	203.21.37.2	lis255.atm1-0.central.saard.net	5682
Aus Backbone	**3**	**192.65.88.225**	**atm2-0-4.mb1.optus.net.au**	**5658**
Aus Backbone	4	192.65.89.137	atmsr-1-3.mi1.optus.net.au	5651
Reg. Network	5	202.139.18.18	BluePlanetNet.mi1.optus.net.au	5680
Remote LAN	6	203.26.36.18	home.bluep.com	5679

Table 3: 1st USA West Coast Typical Path - 03/11 ←→11/11

Node Location	Node	IP Field	DNS Field	RTT Occurrences
Local Network	1	129.127.180.253	vlan0180.atm2-0.pancho.net.adelaide.edu.au	5674
Reg Network	2	203.21.37.2	lis255.atm1-0.central.saard.net	5684
Aus Backbone	3	202.139.32.205	atm5-0-0-4.ia4.optus.net.au	5651
Aus Backbone	4	202.139.1.197	n/a	5672
US Backbone	**5**	**192.65.89.226**	**Serial4-1-0.sf1.optus.net.au**	**2419**
US Backbone	**5**	**192.65.89.234**	**hssi11-0-0.sf1.optus.net.au**	**904**
US Backbone	**5**	**192.65.89.230**	**hssi4-0-0.sf1.optus.net.au**	**995**
US Backbone	**5**	**192.65.89.246**	**hssi9-0-0.sf1.optus.net.au**	**1339**
US Backbone	6	207.124.109.57	g-sfd-br-02-f12-0.gn.cwix.net	2377
US Backbone	7	206.142.242.130	sfd-7513-2-f9-0.cwix.net	2370
US Backbone	8	207.124.107.73	nyd-7513-1-a11-0-2.cwix.net	2372
US Backbone	9	206.142.243.3	g-nyd-br-01-f5-0.gn.cwix.net	2374
US Backbone	10	207.124.127.2	g-nyd-br-02-fe9-0.gn.cwix.net	2376
US Backbone	11	157.130.19.25	Serial12-1-1.GW4.NYC4.ALTER.NET	2376
US Backbone	12	146.188.179.174	146.ATM2-0.XR2.NYC4.ALTER.NET	2375
US Backbone	13	146.188.178.105	288.ATM6-0.XR2.NYC1.ALTER.NET	2377
US Backbone	14	146.188.177.149	194.ATM10-0-0.BR1.NYC1.ALTER.NET	2378
US Backbone	15	129.250.9.61	uunet.nyc1.verio.net	2377
US Backbone	16	129.250.3.125	nyc1.phl02.verio.net	2378
US Backbone	17	129.250.3.153	phl02.phl00.verio.net	2377
US Backbone	18	129.250.3.105	phl00.iad3.verio.net	2377
US Backbone	19	129.250.2.209	iad3.dfw2.verio.net	2368
US Backbone	20	129.250.2.241	dfw2.san1.verio.net	2365
US Backbone	21	129.250.16.106	san1.atmnet.verio.net	2369
Reg. Network	22	207.67.247.70	tierranet-gw.sndgca.pacific.verio.net	2368
Remote LAN	23	209.75.4.34	tucows.tierranet.com	2367

Table 4: 2nd USA West Coast Typical Path - 11/11 ←→13/11

Node Location	Node	IP Field	DNS Field	RTT Occurrences
Local Network	1	129.127.180.253	vlan0180.atm2-0.pancho.net.adelaide.edu.au	5674
Reg. Network	2	203.21.37.2	lis255.atm1-0.central.saard.net	5684
Aus Backbone	3	202.139.32.205	atm5-0-0-4.ia4.optus.net.au	5651
Aus Backbone	4	202.139.1.197	n/a	5672
US Backbone	**5**	**192.65.89.226**	**Serial4-1-0.sf1.optus.net.au**	**2419**
US Backbone	**5**	**192.65.89.234**	**hssi11-0-0.sf1.optus.net.au**	**904**
US Backbone	**5**	**192.65.89.230**	**hssi4-0-0.sf1.optus.net.au**	**995**
US Backbone	**5**	**192.65.89.246**	**hssi9-0-0.sf1.optus.net.au**	**1339**
US Backbone	**6**	**205.174.74.165**	**n/a**	**3295**
US Backbone	7	4.1.142.253	s2-0-0.paloalto-cr18.bbnplanet.net	460
US Backbone	8	4.0.3.85	p3-2.paloalto-nbr2.bbnplanet.net	463
US Backbone	9	4.0.5.65	p1-0.paloalto-nbr1.bbnplanet.net	460
US Backbone	10	4.0.6.45	p3-3.paix-bi1.bbnplanet.net	463
US Backbone	11	4.0.6.5	n/a	462
US Backbone	12	129.250.2.129	pao6.pao5.verio.net	463
US Backbone	13	129.250.3.10	pao5.nuq0.verio.net	462
US Backbone	14	129.250.2.198	nuq0.lax00.verio.net	462
US Backbone	15	129.250.2.226	lax00.san0.verio.net	462
US Backbone	16	129.250.16.150	san0.vsca.verio.net	463
Reg. Network	17	207.67.247.70	tierranet-gw.sndgca.pacific.verio.net	463
Remote LAN	18	209.75.4.34	Tucows.tierranet.com	463

Table 5: 3rd USA West Coast Typical path - 13/11 ←→ 22/11

Node Location	Node	IP Field	DNS Field	RTT Occurrences
Local Network	1	129.127.180.253	vlan0180.atm2-0.pancho.net.adelaide.edu.au	5674
Reg. Network	2	203.21.37.2	lis255.atm1-0.central.saard.net	5684
Aus Backbone	3	202.139.32.205	atm5-0-0-4.ia4.optus.net.au	5651
Aus Backbone	4	202.139.1.197	n/a	5672
US Backbone	**5**	**192.65.89.226**	**Serial4-1-0.sf1.optus.net.au**	**2419**
US Backbone	**5**	**192.65.89.234**	**hssi11-0-0.sf1.optus.net.au**	**904**
US Backbone	**5**	**192.65.89.230**	**hssi4-0-0.sf1.optus.net.au**	**995**
US Backbone	**5**	**192.65.89.246**	**hssi9-0-0.sf1.optus.net.au**	**1339**
US Backbone	**6**	**205.174.74.165**	**n/a**	**3295**
US Backbone	7	199.37.127.68	n/a	2796
US Backbone	8	206.132.110.149	s11-1-0.br1.SFO1.gblx.net	2779
US Backbone	9	206.132.110.133	pos2-1-155M.cr1.SFO1.gblx.net	153
US Backbone	9	208.49.160.161	pos2-3-155M.cr2.sfo1.gblx.net	575
US Backbone	9	208.49.161.93	pos2-3-155M.cr1.SFO1.gblx.net	1992
US Backbone	**10**	**206.132.112.86**	**pos0-0-622M.cr2.LAX1.gblx.net**	**2154**
US Backbone	**10**	**206.132.112.82**	**pos0-0-622M.cr1.LAX1.gblx.net**	**612**
US Backbone	11	206.132.112.122	pos1-0-0-155M.br1.LAX1.gblx.net	2160
US Backbone	11	206.132.112.114	pos0-0-0-155M.br1.LAX1.gblx.net	612
US Backbone	12	206.132.112.162	s0-0-0.cr1.SAN.gblx.net	2766
US Backbone	13	206.57.3.62	tierranet.s4-1-1.cr1.SAN.gblx.net	2397
Reg. Network	13	206.57.3.62	tierranet.s4-1-1.cr1.SAN.globalcenter.net	380
Remote LAN	14	209.75.4.34	tucows.tierranet.com	2779

Table 6: 1st USA East Coast Typical Path – 03/11 ←→ 11/11

Node Location	Node	IP Field	DNS Field	RTT Occurrences
Local Network	1	129.127.180.253	vlan0180.atm2-0.pancho.net.adelaide.edu.au	5675
Reg. Network	2	203.21.37.2	lis255.atm1-0.central.saard.net	5685
Aus Backbone	3	202.139.32.205	atm5-0-0-4.ia4.optus.net.au	5653
Aus Backbone	4	202.139.1.197	n/a	5676
US Backbone	**5**	**192.65.89.230**	**hssi4-0-0.sf1.optus.net.au**	**2259**
US Backbone	**5**	**192.65.89.234**	**hssi11-0-0.sf1.optus.net.au**	**2196**
US Backbone	**5**	**192.65.89.226**	**Serial4-1-0.sf1.optus.net.au**	**132**
US Backbone	**5**	**192.65.89.246**	**hssi9-0-0.sf1.optus.net.au**	**1073**
US Backbone	6	207.124.109.57	g-sfd-br-02-f12-0.gn.cwix.net	2433
US Backbone	**7**	**166.49.11.69**	**core7-serial5-1-0.SanFrancisco.cw.net**	**1131**
US Backbone	**7**	**204.70.10.9**	**core7-hssi6-0-0.SanFrancisco.cw.net**	**1296**
US Backbone	8	204.70.4.81	core4.SanFrancisco.cw.net	2429
US Backbone	9	206.157.77.66	sl-stk-1-H9-0-T3.sprintlink.net	2423
US Backbone	10	144.232.4.33	sl-bb11-stk-2-3.sprintlink.net	2430
US Backbone	11	144.232.8.177	sl-bb10-pen-6-0.sprintlink.net	2433
US Backbone	12	144.232.5.14	sl-bb2-pen-0-0-0.sprintlink.net	2434
US Backbone	13	144.228.60.9	sl-gw7-pen-0-0.sprintlink.net	2432

Table 7: 2nd USA East Coast Typical Path – 11/11 ←→22/11

Node Location	Node	IP Field	DNS Field	RTT Occurrences
Local Network	1	129.127.180.253	vlan0180.atm2-0.pancho.net.adelaide.edu.au	5675
Reg. Network	2	203.21.37.2	lis255.atm1-0.central.saard.net	5685
Aus Backbone	3	202.139.32.205	atm5-0-0-4.ia4.optus.net.au	5653
Aus Backbone	4	202.139.1.197	n/a	5676
US Backbone	**5**	**192.65.89.230**	**hssi4-0-0.sf1.optus.net.au**	**2259**
US Backbone	**5**	**192.65.89.234**	**hssi11-0-0.sf1.optus.net.au**	**2196**
US Backbone	**5**	**192.65.89.226**	**Serial4-1-0.sf1.optus.net.au**	**132**
US Backbone	**5**	**192.65.89.246**	**hssi9-0-0.sf1.optus.net.au**	**1073**
US Backbone	6	205.174.74.165	n/a	3239
US Backbone	7	4.1.142.253	s2-0-0.paloalto-cr18.bbnplanet.net	3235
US Backbone	8	4.0.3.85	p3-2.paloalto-nbr2.bbnplanet.net	3235
US Backbone	**9**	**4.0.1.9**	**n/a**	**3228**
US Backbone	10	144.232.3.25	sl-bb10-sj-9-0.sprintlink.net	2201
US Backbone	10	144.232.9.166	sl-bb10-stk-10-0.sprintlink.net	1015
US Backbone	11	144.232.4.106	sl-bb11-stk-8-0.sprintlink.net	1018
US Backbone	11	144.232.9.217	sl-bb10-sj-10-0.sprintlink.net	1026
US Backbone	11	144.232.9.13	sl-bb10-rly-6-0.sprintlink.net	1180
US Backbone	12	144.232.8.177	sl-bb10-pen-6-0.sprintlink.net	1032
US Backbone	12	144.232.0.46	sl-bb11-rly-9-0.sprintlink.net	1180
US Backbone	12	144.232.7.213	sl-bb11-rly-8-0.sprintlink.net	1025
US Backbone	13	144.232.8.154	sl-bb11-pen-7-0.sprintlink.net	2207
US Backbone	13	144.232.5.54	sl-bb7-pen-0-0-0.sprintlink.net	506
US Backbone	13	144.232.5.38	sl-bb5-pen-0-0-0.sprintlink.net	480
US Backbone	14	144.232.5.18	sl-bb2-pen-4-0-0.sprintlink.net	1173
US Backbone	14	144.232.5.58	sl-bb7-pen-4-0-0.sprintlink.net	1025
US Backbone	14	144.228.60.9	sl-gw7-pen-0-0.sprintlink.net	1034
US Backbone	**15**	**144.228.60.9**	**sl-gw7-pen-0-0.sprintlink.net**	**2204**

Table 8: Hong Kong Typical Path

Node Location	Node	IP Field	DNS Field	RTT Occurrences
Local Network	1	129.127.180.253	vlan0180.atm2-0.pancho.net.adelaide.edu.au	5674
Reg. Network	2	203.21.37.2	lis255.atm1-0.central.saard.net	5679
Aus Backbone	3	202.139.32.205	atm5-0-0-4.ia4.optus.net.au	5649
Aus Backbone	**4**	**202.139.7.57**	**atmsr-1-10.ap1.optus.net.au**	**420**
HK Backbone	5	202.84.195.229	s0-0-3.yck05.hkt.net	5670
HK Backbone	**6**	**205.252.130.207**	**f5-0.hk-T3.hkt.net**	**1865**
HK Backbone	**6**	**205.252.130.207**	**f5-0.yckbr01.hkt.net**	**753**
HK Backbone	**6**	**205.252.130.239**	**f5-1.hk-T3.hkt.net**	**2211**
HK Backbone	**6**	**205.252.130.239**	**f5-1.yckbr01.hkt.net**	**839**
Reg. Network	7	202.84.133.114	n/a	5674
Remote LAN	8	202.14.67.35	tucows.pacific.net.hk	5669

Table 9: 1st Israel Typical Path – Mainly between 3/11 ←→ 11/11

Node Location	Node	IP Field	DNS Field	RTT Occurrences
Local Network	1	129.127.180.253	vlan0180.atm2-0.pancho.net.adelaide.edu.au	5674
Reg. Network	2	203.21.37.2	lis255.atm1-0.central.saard.net	5683
Aus.Backbone	3	202.139.32.205	atm5-0-0-4.ia4.optus.net.au	5651
Aus.Backbone	**4**	**202.139.1.205**	**n/a**	**2386**
Aus.Backbone	**4**	**202.139.1.197**	**n/a**	**3283**
US Backbone	**5**	**192.65.89.246**	**hssi9-0-0.sf1.optus.net.au**	**334**
US Backbone	**5**	**192.65.89.242**	**hssi11-0-0.la1.optus.net.au**	**1205**
US Backbone	**5**	**192.65.89.238**	**hssi9-0-0.la1.optus.net.au**	**1160**
US Backbone	**5**	**192.65.89.234**	**hssi11-0-0.sf1.optus.net.au**	**1391**
US Backbone	**5**	**192.65.89.230**	**hssi4-0-0.sf1.optus.net.au**	**1427**
US Backbone	6	157.130.227.181	34.ATM0-0-0.GW1.LAX4.ALTER.NET	2380
US Backbone	7	146.188.248.102	121.ATM2-0.XR2.LAX4.ALTER.NET	2359
US Backbone	8	146.188.249.17	193.ATM10-0-0.GW2.LAX4.ALTER.NET	2355
Timed out				
US Backbone	10	207.45.222.181	if-1-1.core1.LosAngeles.Teleglobe.net	2352
US Backbone	11	207.45.222.25	if-6-2.core1.NewYork.Teleglobe.net	2351
US Backbone	**12**	**207.45.223.153**	**if-4-0-0.bb3.NewYork.Teleglobe.net**	**3835**
US Backbone	**12**	**207.45.221.69**	**if-0-0-0.bb3.NewYork.Teleglobe.net**	**1017**
US Backbone	13	207.45.199.150	ix-10-1-1.bb3.NewYork.Teleglobe.net	4852
Israel Backbone	**14**	**212.117.128.193**	**itele1.goldenlines.net.il**	**1525**
Israel Backbone	**14**	**212.117.128.197**	**itele2.goldenlines.net.il**	**1534**
Israel Backbone	**14**	**212.117.128.201**	**itele3.goldenlines.net.il**	**1782**
Israel Backbone	15	212.117.128.26	cisraserv.goldenlines.net.il	5461
Reg. Network	16	192.114.159.34	kfarsaba-rtr.israsrv.net.il	5460
Remote LAN	17	192.117.192.103	tucows.israsrv.net.il	5454

Table 10: 2nd Israel Typical Path – Mainly between 11/11 ←→ 22/11

Node Location	Node	IP Field	DNS Field	RTT Occurrences
Local Network	1	129.127.180.253	vlan0180.atm2-0.pancho.net.adelaide.edu.au	5674
Reg. Network	2	203.21.37.2	lis255.atm1-0.central.saard.net	5683
Aus Backbone	3	202.139.32.205	atm5-0-0-4.ia4.optus.net.au	5651
AusBackbone	**4**	**202.139.1.205**	**n/a**	**2386**
Aus Backbone	**4**	**202.139.1.197**	**n/a**	**3283**
US Backbone	**5**	**192.65.89.246**	**hssi9-0-0.sf1.optus.net.au**	**334**
US Backbone	**5**	**192.65.89.242**	**hssi11-0-0.la1.optus.net.au**	**1205**
US Backbone	**5**	**192.65.89.238**	**hssi9-0-0.la1.optus.net.au**	**1160**
US Backbone	**5**	**192.65.89.234**	**hssi11-0-0.sf1.optus.net.au**	**1391**
US Backbone	**5**	**192.65.89.230**	**hssi4-0-0.sf1.optus.net.au**	**1427**
US Backbone	**6**	**205.174.74.165**	**n/a**	**2552**
US Backbone	7	134.24.45.5	atm10-0-1.sfo-bb2.cerf.net	2538
US Backbone	8	134.24.29.197	pos0-2-155M.sfo-bb3.cerf.net	2535
US Backbone	9	134.24.32.90	pos4-0-0-155M.sjc-bb3.cerf.net	2537
No details	10			
US Backbone	11	207.45.222.178	if-8-0.core1.NewYork.Teleglobe.net	2498
US Backbone	**12**	**207.45.223.153**	**if-4-0-0.bb3.NewYork.Teleglobe.net**	**3835**
US Backbone	**12**	**207.45.221.69**	**if-0-0-0.bb3.NewYork.Teleglobe.net**	**1017**
US Backbone	13	207.45.199.150	ix-10-1-1.bb3.NewYork.Teleglobe.net	4852
Israel Backbone	**14**	**212.117.128.193**	**itele1.goldenlines.net.il**	**1525**
Israel Backbone	**14**	**212.117.128.197**	**itele2.goldenlines.net.il**	**1534**
Israel Backbone	**14**	**212.117.128.201**	**itele3.goldenlines.net.il**	**1782**
Israel Backbone	15	212.117.128.26	cisraserv.goldenlines.net.il	5461
Reg. Network	16	192.114.159.34	kfarsaba-rtr.israsrv.net.il	5460
Remote LAN	17	192.117.192.103	tucows.israsrv.net.il	5454

Table 11: 3rd Israel Typical Path – Mainly between 11/11 ←→ 22/11

Node Location	Node	IP Field	DNS Field	RTT Occurrences
Local Network	1	129.127.180.253	vlan0180.atm2-0.pancho.net.adelaide.edu.au	5674
Reg. Network	2	203.21.37.2	lis255.atm1-0.central.saard.net	5683
AusBackbone	3	202.139.32.205	atm5-0-0-4.ia4.optus.net.au	5651
AusBackbone	**4**	**202.139.1.205**	**n/a**	**2385**
AusBackbone	**4**	**202.139.1.197**	**n/a**	**3282**
US Backbone	**5**	**192.65.89.246**	**hssi9-0-0.sf1.optus.net.au**	**334**
US Backbone	**5**	**192.65.89.242**	**hssi11-0-0.la1.optus.net.au**	**1205**
US Backbone	**5**	**192.65.89.238**	**hssi9-0-0.la1.optus.net.au**	**1160**
US Backbone	**5**	**192.65.89.234**	**hssi11-0-0.sf1.optus.net.au**	**1391**
US Backbone	**5**	**192.65.89.230**	**hssi4-0-0.sf1.optus.net.au**	**1427**
US Backbone	**6**	**207.124.109.57**	**g-sfd-br-02-f12-0.gn.cwix.net**	**730**
US Backbone	7	206.142.242.130	sfd-7513-2-f9-0.cwix.net	731
US Backbone	8	207.124.107.73	nyd-7513-1-a11-0-2.cwix.net	730
US Backbone	9	206.142.243.3	g-nyd-br-01-f5-0.gn.cwix.net	729
US Backbone	10	207.124.127.2	g-nyd-br-02-fe9-0.gn.cwix.net	729
US Backbone	11	157.130.19.25	Serial12-1-1.GW4.NYC4.ALTER.NET	702
US Backbone	12	146.188.179.162	146.ATM3-0.XR1.NYC4.ALTER.NET	702
US Backbone	13	146.188.178.137	189.ATM9-0-0.GW2.NYC4.ALTER.NET	703
Israel Backbone	14	157.130.15.234	goldenlines1-gw.customer.ALTER.NET	702
Israel Backbone	15	212.117.128.26	cisraserv.goldenlines.net.il	5461
Reg. Network	16	192.114.159.34	kfarsaba-rtr.israsrv.net.il	5460
Remote LAN	17	192.117.192.103	tucows.israsrv.net.il	5454

Table 12: 1st Germany Typical Path 3/11← → 12/11

Node Location	Node	IP Field	DNS Field	RTT Occurrences
Local Network	1	129.127.180.253	vlan0180.atm2-0.pancho.net.adelaide.edu.au	5673
Reg. Network	2	203.21.37.2	lis255.atm1-0.central.saard.net	5684
Aus Backbone	3	202.139.32.205	atm5-0-0-4.ia4.optus.net.au	5653
Aus Backbone	4	202.139.1.197	n/a	5662
Aus Backbone	**5**	**192.65.89.246**	**hssi9-0-0.sf1.optus.net.au**	**1398**
Aus Backbone	**5**	**192.65.89.226**	**Serial4-1-0.sf1.optus.net.au**	**2405**
Aus Backbone	**5**	**192.65.89.230**	**hssi4-0-0.sf1.optus.net.au**	**923**
Aus Backbone	**5**	**192.65.89.234**	**hssi11-0-0.sf1.optus.net.au**	**923**
US Backbone	6	205.174.74.165	n/a	5658
US Backbone	7	4.1.142.253	s2-0-0.paloalto-cr18.bbnplanet.net	5657
US Backbone	8	4.0.3.85	p3-2.paloalto-nbr2.bbnplanet.net	5656
US Backbone	9	4.0.1.2	p4-0.sanjose1-nbr1.bbnplanet.net	5629
US Backbone	10	4.0.5.86	p1-0.sanjose1-nbr2.bbnplanet.net	5630
US Backbone	11	4.24.7.57	p3-0.nycmny1-br2.bbnplanet.net	5625
US Backbone	12	4.24.6.225	p4-0.nycmny1-br1.bbnplanet.net	5631
US Backbone	13	4.0.5.98	p4-0.nyc4-nbr2.bbnplanet.net	5635
US Backbone	14	4.0.5.26	p1-0.nyc4-nbr3.bbnplanet.net	5631
US Backbone	15	4.0.2.85	p1-0-0.nyc4-cr2.bbnplanet.net	5632
US Backbone	16	4.1.70.122	s0.ucs.bbnplanet.net	5635
Germ Backbone	17	212.38.192.129	r001a80065.defra.ecs-ip.net	2513
Reg. Network	18	212.38.194.54	Sontheimer-Network.defra.ecs-ip.net	2515
Remote LAN	19	194.88.180.7	www.tucows.de	2511

Table 13: 2nd Germany Typical Path 12/11 ← → 22/11

Node Location	Node	IP Field	DNS Field	RTT Occurrences
Local Network	1	129.127.180.253	vlan0180.atm2-0.pancho.net.adelaide.edu.au	5673
Reg. Network	2	203.21.37.2	lis255.atm1-0.central.saard.net	5684
Aus Backbone	3	202.139.32.205	atm5-0-0-4.ia4.optus.net.au	5653
Aus Backbone	4	202.139.1.197	n/a	5662
Aus Backbone	**5**	**192.65.89.246**	**hssi9-0-0.sf1.optus.net.au**	**1398**
Aus Backbone	**5**	**192.65.89.226**	**Serial4-1-0.sf1.optus.net.au**	**2405**
Aus Backbone	**5**	**192.65.89.230**	**hssi4-0-0.sf1.optus.net.au**	**923**
Aus Backbone	**5**	**192.65.89.234**	**hssi11-0-0.sf1.optus.net.au**	**923**
Aus Backbone	6	205.174.74.165	n/a	5658
US Backbone	7	4.1.142.253	s2-0-0.paloalto-cr18.bbnplanet.net	5657
US Backbone	8	4.0.3.85	p3-2.paloalto-nbr2.bbnplanet.net	5656
US Backbone	9	4.0.1.2	p4-0.sanjose1-nbr1.bbnplanet.net	5629
US Backbone	10	4.0.5.86	p1-0.sanjose1-nbr2.bbnplanet.net	5630
US Backbone	11	4.24.7.57	p3-0.nycmny1-br2.bbnplanet.net	5625
US Backbone	12	4.24.6.225	p4-0.nycmny1-br1.bbnplanet.net	5631
US Backbone	13	4.0.5.98	p4-0.nyc4-nbr2.bbnplanet.net	5635
US Backbone	14	4.0.5.26	p1-0.nyc4-nbr3.bbnplanet.net	5631
US Backbone	15	4.0.2.85	p1-0-0.nyc4-cr2.bbnplanet.net	5632
US Backbone	16	4.1.70.122	s0.ucs.bbnplanet.net	5635
US Backbone	**17**	**212.38.193.182**	**r003fe810.usnyc.ecs-ip.net**	**3106**
Germ Backbone	18	212.38.193.177	r005p500.defra.ecs-ip.net	3102
Germ Backbone	19	212.38.193.205	r001fe1110.defra.ecs-ip.net	3108
Reg. Network	20	212.38.194.54	Sontheimer-Network.defra.ecs-ip.net	3105
Remote LAN	21	194.88.180.7	www.tucows.de	3084

Table 14: 1st England Typical Path - 03/11← → 11/11

Node Location	Node	IP Field	DNS Field	RTT Occurrences
Local Network	1	129.127.180.253	vlan0180.atm2-0.pancho.net.adelaide.edu.au	5675
Reg. Network	2	203.21.37.2	lis255.atm1-0.central.saard.net	5683
Aus Backbone	3	202.139.32.205	atm5-0-0-4.ia4.optus.net.au	5648
Aus Backbone	4	202.139.1.205	n/a	5585
US Backbone	**5**	**192.65.89.242**	**hssi11-0-0.la1.optus.net.au**	**2844**
US Backbone	**5**	**192.65.89.238**	**hssi9-0-0.la1.optus.net.au**	**2695**
US Backbone	6	157.130.227.181	34.ATM0-0-0.GW1.LAX4.ALTER.NET	5546
US Backbone	7	146.188.248.98	121.ATM3-0.XR1.LAX4.ALTER.NET	2435
US Backbone	8	146.188.249.2	192.ATM2-0.TR2.LAX2.ALTER.NET	2434
US Backbone	9	146.188.138.197	111.ATM7-0.TR2.DCA8.ALTER.NET	2380
US Backbone	10	152.63.32.213	n/a	2362
US Backbone	11	146.188.160.45	192.ATM9-0-0.GW1.TCO1.ALTER.NET	2381
	12	157.130.33.26	Telewest-gw.customer.ALTER.NET	5520
Engl Backbone	13	193.38.108.68	h21-isp1-cro.cableinet.net	5508
Reg. Network	14	194.117.132.26	n/a	5235
Remote LAN	15	194.117.152.71	tucows.cableinet.net	5233

Table 15: 2nd England Typical Path - 11/11 ← → 22/11

Node Location	Node	IP Field	DNS Field	RTT Occurrences
Local Network	1	129.127.180.253	vlan0180.atm2-0.pancho.net.adelaide.edu.au	5675
Reg. Network	2	203.21.37.2	lis255.atm1-0.central.saard.net	5683
Aus Backbone	3	202.139.32.205	atm5-0-0-4.ia4.optus.net.au	5648
Aus Backbone	4	202.139.1.205	n/a	5585
US Backbone	**5**	**192.65.89.242**	**hssi11-0-0.la1.optus.net.au**	**2844**
US Backbone	**5**	**192.65.89.238**	**hssi9-0-0.la1.optus.net.au**	**2695**
US Backbone	6	157.130.227.181	34.ATM0-0-0.GW1.LAX4.ALTER.NET	5546
US Backbone	7	146.188.248.110	121.ATM3-0.XR2.LAX4.ALTER.NET	3147
US Backbone	8	146.188.248.250	293.ATM2-0.TR1.LAX2.ALTER.NET	3148
US Backbone	9	146.188.138.145	111.ATM7-0.TR1.DCA8.ALTER.NET	3149
US Backbone	**10**	**152.63.32.197**	**n/a**	**1475**
US Backbone	**10**	**152.63.32.193**	**n/a**	**1671**
US Backbone	**11**	**146.188.160.41**	**193.ATM9-0-0.GW1.TCO1.ALTER.NET**	**2845**
US Backbone	**11**	**146.188.160.33**	**193.ATM8-0-0.GW1.TCO1.ALTER.NET**	**300**
	12	157.130.33.26	Telewest-gw.customer.ALTER.NET	5520
Engl Backbone	13	193.38.108.68	h21-isp1-cro.cableinet.net	5508
Reg. Network	14	194.117.132.26	n/a	5235
Remote LAN	15	194.117.152.71	tucows.cableinet.net	5233

Table 16: 1st Argentina Typical Path – 03/11 ←→ 16/11

Node Location	Node	IP Field	DNS Field	RTT Occurrences
Local Network	1	129.127.180.253	vlan0180.atm2-0.pancho.net.adelaide.edu.au	7963
Reg. Network	2	203.21.37.2	lis255.atm1-0.central.saard.net	7972
Aus Backbone	3	202.139.32.205	atm5-0-0-4.ia4.optus.net.au	7940
Aus Backbone	4	202.139.1.205	n/a	7899
US Backbone	5	192.65.89.242	hssi11-0-0.la1.optus.net.au	3898
US Backbone	5	192.65.89.238	hssi9-0-0.la1.optus.net.au	3946
US Backbone	6	157.130.227.181	34.ATM0-0-0.GW1.LAX4.ALTER.NET	7848
US Backbone	**7**	**146.188.248.110**	**121.ATM3-0.XR2.LAX4.ALTER.NET**	**5666**
US Backbone	7	146.188.248.98	121.ATM3-0.XR1.LAX4.ALTER.NET	2219
US Backbone	**8**	**152.63.112.178**	**193.at-2-0-0.TR1.LAX9.ALTER.NET**	**1184**
US Backbone	**8**	**146.188.248.242**	**193.ATM2-0.TR1.LAX2.ALTER.NET**	**4481**
US Backbone	8	152.63.112.186	192.at-1-0-0.TR2.LAX9.ALTER.NET	239
US Backbone	8	152.63.112.194	192.at-2-0-0.TR2.LAX9.ALTER.NET	1914
US Backbone	**9**	**152.63.0.109**	**131.at-5-0-0.TR1.ATL5.ALTER.NET**	**1185**
US Backbone	**9**	**146.188.136.49**	**111.ATM7-0.TR1.ATL1.ALTER.NET**	**4482**
US Backbone	9	152.63.0.205	131.at-5-0-0.TR2.ATL5.ALTER.NET	2156
US Backbone	**10**	**152.63.81.17**	**197.ATM6-0.XR1.ATL1.ALTER.NET**	**1152**
US Backbone	**10**	**146.188.232.89**	**299.ATM6-0.XR1.ATL1.ALTER.NET**	**1928**
US Backbone	**10**	**146.188.232.85**	**199.ATM7-0.XR1.ATL1.ALTER.NET**	**2534**
US Backbone	10	152.63.81.33	196.ATM6-0.XR2.ATL1.ALTER.NET	200
US Backbone	10	152.63.81.41	196.ATM7-0.XR2.ATL1.ALTER.NET	1953
US Backbone	**11**	**146.188.232.209**	**195.ATM10-0-0.GW2.ORL1.ALTER.NET**	**1194**
US Backbone	**11**	**146.188.232.217**	**195.ATM9-0-0.GW2.ORL1.ALTER.NET**	**4463**
US Backbone	11	146.188.232.221	194.ATM11-0-0.GW2.ORL1.ALTER.NET	2198
US Backbone	12	157.130.64.202	bs-miami-gw.customer.ALTER.NET	5927
US Backbone	12	157.130.67.46	bs-orlando-gw.customer.ALTER.NET	1922
US Backbone	13	205.152.16.132	n/a	4594
US Backbone	13	205.152.16.196	n/a	3267
No details	14			
Arg Backbone	15	200.41.40.62	n/a	4194
Arg Backbone	15	200.41.25.246	200.41.25.246.impsat.net	3626
Reg. Network	16	200.41.25.238	rcoretlp1.impsat.net.ar	3653
Remote LAN	17	200.31.1.236	tucows.impsat.com.ar	3793

Table 17: 2nd Argentina Typical Path – 16/11 ← → 30/11

Node Location	Node	IP Field	DNS Field	RTT Occurrences
Local Network	1	129.127.180.253	vlan0180.atm2-0.pancho.net.adelaide.edu.au	7963
Reg. Network	2	203.21.37.2	lis255.atm1-0.central.saard.net	7972
Aus Backbone	3	202.139.32.205	atm5-0-0-4.ia4.optus.net.au	7940
Aus Backbone	4	202.139.1.205	n/a	7899
US Backbone	5	192.65.89.242	hssi11-0-0.la1.optus.net.au	3898
US Backbone	5	192.65.89.238	hssi9-0-0.la1.optus.net.au	3946
US Backbone	6	157.130.227.181	34.ATM0-0-0.GW1.LAX4.ALTER.NET	7848
US Backbone	**7**	**146.188.248.110**	**121.ATM3-0.XR2.LAX4.ALTER.NET**	**5666**
US Backbone	7	146.188.248.98	121.ATM3-0.XR1.LAX4.ALTER.NET	2219
US Backbone	**8**	**152.63.112.178**	**193.at-2-0-0.TR1.LAX9.ALTER.NET**	**1184**
US Backbone	**8**	**146.188.248.242**	**193.ATM2-0.TR1.LAX2.ALTER.NET**	**4481**
US Backbone	8	152.63.112.186	192.at-1-0-0.TR2.LAX9.ALTER.NET	239
US Backbone	8	152.63.112.194	192.at-2-0-0.TR2.LAX9.ALTER.NET	1914
US Backbone	**9**	**152.63.0.109**	**131.at-5-0-0.TR1.ATL5.ALTER.NET**	**1185**
US Backbone	**9**	**146.188.136.49**	**111.ATM7-0.TR1.ATL1.ALTER.NET**	**4482**
US Backbone	9	152.63.0.205	131.at-5-0-0.TR2.ATL5.ALTER.NET	2156
US Backbone	**10**	**152.63.81.17**	**197.ATM6-0.XR1.ATL1.ALTER.NET**	**1152**
US Backbone	**10**	**146.188.232.89**	**299.ATM6-0.XR1.ATL1.ALTER.NET**	**1928**
US Backbone	**10**	**146.188.232.85**	**199.ATM7-0.XR1.ATL1.ALTER.NET**	**2534**
US Backbone	10	152.63.81.33	196.ATM6-0.XR2.ATL1.ALTER.NET	200
US Backbone	10	152.63.81.41	196.ATM7-0.XR2.ATL1.ALTER.NET	1953
US Backbone	**11**	**146.188.232.209**	**195.ATM10-0-0.GW2.ORL1.ALTER.NET**	**1194**
US Backbone	**11**	**146.188.232.217**	**195.ATM9-0-0.GW2.ORL1.ALTER.NET**	**4463**
US Backbone	11	146.188.232.221	194.ATM11-0-0.GW2.ORL1.ALTER.NET	2198
US Backbone	12	157.130.64.202	bs-miami-gw.customer.ALTER.NET	5927
US Backbone	12	157.130.67.46	bs-orlando-gw.customer.ALTER.NET	1922
US Backbone	13	205.152.16.132	n/a	4594
US Backbone	13	205.152.16.196	n/a	3267
No details	14			
Arg Network	15	200.41.25.246	200.41.25.246.impsat.net	3626
Reg. Network	15	200.41.40.62	n/a	4194
Remote LAN	16	200.31.1.236	tucows.impsat.com.ar	4159

Table 18: 1st Brazil Typical Path – Whole Period

Node Location	Node	IP Field	DNS Field	RTT Occurrences
Local Network	1	129.127.180.253	vlan0180.atm2-0.pancho.net.adelaide.edu.au	7901
Reg. Network	2	203.21.37.2	lis255.atm1-0.central.saard.net	7910
Aus Backbone	3	202.139.32.205	atm5-0-0-4.ia4.optus.net.au	7879
Aus Backbone	4	202.139.1.205	n/a	7860
US Backbone	**5**	**192.65.89.242**	**hssi11-0-0.la1.optus.net.au**	**3858**
US Backbone	**5**	**192.65.89.238**	**hssi9-0-0.la1.optus.net.au**	**3951**
US Backbone	6	157.130.227.181	34.ATM0-0-0.GW1.LAX4.ALTER.NET	7813
US Backbone	7	146.188.248.106	121.ATM2-0.XR1.LAX4.ALTER.NET	4562
US Backbone	8	146.188.249.2	192.ATM2-0.TR2.LAX2.ALTER.NET	4563
US Backbone	9	146.188.137.53	111.ATM7-0.TR2.EWR1.ALTER.NET	4560
US Backbone	**10**	**146.188.176.85**	**196.ATM7-0.XR2.EWR1.ALTER.NET**	**3056**
US Backbone	**10**	**146.188.176.81**	**196.ATM6-0.XR2.EWR1.ALTER.NET**	**1499**
US Backbone	**11**	**146.188.177.169**	**192.ATM9-0-0.GW3.EWR1.ALTER.NET**	**2086**
US Backbone	**11**	**146.188.177.161**	**192.ATM8-0-0.GW3.EWR1.ALTER.NET**	**2471**
US Backbone	**12**	**157.130.12.250**	**Serial3-1-1.GW1.BLM1.ALTER.NET**	**6183**
US Backbone	**12**	**157.130.12.70**	**Serial1-1-1.GW1.BLM1.ALTER.NET**	**1583**
Braz Backbone	**13**	**198.3.149.6**	**n/a**	**3378**
Braz Backbone	**13**	**198.3.149.10**	**n/a**	**4448**
Braz Backbone	14	200.230.0.102	ebt-P2-0-gsr01.spo.embratel.net.br	7820
Braz Backbone	15	200.230.0.105	ebt-P12-0-0-dist03.spo.embratel.net.br	7827
Reg. Network	**16**	**200.211.93.242**	**uol-A5-0-0-2-dist03.spo.embratel.net.br**	**561**
Reg. Network	**16**	**200.211.95.238**	**uol-A5-0-0-1-dist03.spo.embratel.net.br**	**7258**
Remote LAN	17	200.230.198.57	n/a	7793

Table 19: 2nd Brazil Typical Path – 11/11 ← →27/11

Node Location	Node	IP Field	DNS Field	RTT Occurrences
Local Network	1	129.127.180.253	vlan0180.atm2-0.pancho.net.adelaide.edu.au	7901
Reg. Network	2	203.21.37.2	lis255.atm1-0.central.saard.net	7910
Aus Backbone	3	202.139.32.205	atm5-0-0-4.ia4.optus.net.au	7879
Aus Backbone	4	202.139.1.205	n/a	7860
US Backbone	**5**	**192.65.89.242**	**hssi11-0-0.la1.optus.net.au**	**3858**
US Backbone	**5**	**192.65.89.238**	**hssi9-0-0.la1.optus.net.au**	**3951**
US Backbone	6	157.130.227.181	34.ATM0-0-0.GW1.LAX4.ALTER.NET	7813
US Backbone	**7**	**146.188.248.102**	**121.ATM2-0.XR2.LAX4.ALTER.NET**	**3055**
US Backbone	**7**	**146.188.248.110**	**121.ATM3-0.XR2.LAX4.ALTER.NET**	**225**
US Backbone	8	146.188.248.250	293.ATM2-0.TR1.LAX2.ALTER.NET	3278
US Backbone	9	146.188.137.49	111.ATM7-0.TR1.EWR1.ALTER.NET	3277
US Backbone	**10**	**146.188.176.77**	**200.ATM7-0.XR1.EWR1.ALTER.NET**	**1859**
US Backbone	**10**	**146.188.176.73**	**200.ATM6-0.XR1.EWR1.ALTER.NET**	**1342**
US Backbone	**10**	**146.188.176.65**	**100.ATM6-0.XR1.EWR1.ALTER.NET**	**72**
US Backbone	11	146.188.177.165	193.ATM8-0-0.GW3.EWR1.ALTER.NET	3271
US Backbone	**12**	**157.130.12.70**	**Serial1-1-1.GW1.BLM1.ALTER.NET**	**1583**
US Backbone	**12**	**157.130.12.250**	**Serial3-1-1.GW1.BLM1.ALTER.NET**	**6183**
Braz Backbone	**13**	**198.3.149.6**	**n/a**	**3378**
Braz Backbone	**13**	**198.3.149.10**	**n/a**	**4448**
Braz Backbone	14	200.230.0.102	ebt-P2-0-gsr01.spo.embratel.net.br	7820
Braz Backbone	15	200.230.0.105	ebt-P12-0-0-dist03.spo.embratel.net.br	7827
Reg. Network	**16**	**200.211.93.242**	**uol-A5-0-0-2-dist03.spo.embratel.net.br**	**561**
Reg. Network	**16**	**200.211.95.238**	**uol-A5-0-0-1-dist03.spo.embratel.net.br**	**7258**
Remote LAN	17	200.230.198.57	n/a	7793

Table 20: 1st South Africa Typical Path - Whole period

Node Location	Node	IP Field	DNS Field	PTT Occurrences
Local Network	1	129.127.180.253	vlan0180.atm2-0.pancho.net.adelaide.edu.au	7920
Reg. Network	2	203.21.37.2	lis255.atm1-0.central.saard.net	7930
Aus Backbone	3	202.139.32.205	atm5-0-0-4.ia4.optus.net.au	7898
Aus Backbone	4	202.139.1.197	n/a	7915
US Backbone	**5**	**192.65.89.234**	**hssi11-0-0.sf1.optus.net.au**	**2740**
US Backbone	**5**	**192.65.89.246**	**hssi9-0-0.sf1.optus.net.au**	**1536**
US Backbone	**5**	**192.65.89.230**	**hssi4-0-0.sf1.optus.net.au**	**2941**
US Backbone	**5**	**192.65.89.226**	**Serial4-1-0.sf1.optus.net.au**	**689**
US Backbone	6	207.124.109.57	g-sfd-br-02-f12-0.gn.cwix.net	7891
US Backbone	7	206.142.242.130	sfd-7513-2-f9-0.cwix.net	7882
US Backbone	8	207.124.107.73	nyd-7513-1-a11-0-2.cwix.net	7884
US Backbone	9	206.142.243.3	g-nyd-br-01-f5-0.gn.cwix.net	7886
US Backbone	10	207.124.127.2	g-nyd-br-02-fe9-0.gn.cwix.net	7881
	11	**207.124.125.106**	**n/a**	**7879**
	12	**195.44.136.9**	**n/a**	**7879**
	13	**195.44.128.2**	**n/a**	**6502**
	14	194.6.80.252	bgp-gw2.cwci.net	6510
	15	**195.44.50.158**	**n/a**	**1608**
	15	**195.44.50.162**	**n/a**	**1652**
	15	**195.44.50.154**	**n/a**	**1628**
	15	**195.44.50.150**	**n/a**	**1631**
	16	196.25.0.81	ndf-core2.wc.saix.net	6507
	17	196.25.0.22	cbs-core.wc.saix.net	6519
Reg. Network	18	196.25.3.22	scien-data-gw.wc.saix.net	6501
Remote LAN	19	196.25.18.137	sun.new.co.za	6493

Table 21: 2nd South Africa Typical Path - 24/11<-->30/11

Node Location	Node	IP Field	DNS Field	RTT Occurrences
Local Network	1	129.127.180.253	vlan0180.atm2-0.pancho.net.adelaide.edu.au	7920
Reg. Network	2	203.21.37.2	lis255.atm1-0.central.saard.net	7930
Aus Backbone	3	202.139.32.205	atm5-0-0-4.ia4.optus.net.au	7898
Aus Backbone	4	202.139.1.197	n/a	7915
US Backbone	**5**	**192.65.89.234**	**hssi11-0-0.sf1.optus.net.au**	**2740**
US Backbone	**5**	**192.65.89.246**	**hssi9-0-0.sf1.optus.net.au**	**1536**
US Backbone	**5**	**192.65.89.230**	**hssi4-0-0.sf1.optus.net.au**	**2941**
US Backbone	**5**	**192.65.89.226**	**Serial4-1-0.sf1.optus.net.au**	**689**
US Backbone	6	207.124.109.57	g-sfd-br-02-f12-0.gn.cwix.net	7891
US Backbone	7	206.142.242.130	sfd-7513-2-f9-0.cwix.net	7882
US Backbone	8	207.124.107.73	nyd-7513-1-a11-0-2.cwix.net	7884
US Backbone	9	206.142.243.3	g-nyd-br-01-f5-0.gn.cwix.net	7886
US Backbone	10	207.124.127.2	g-nyd-br-02-fe9-0.gn.cwix.net	7881
	11	**207.124.125.106**	**n/a**	**7879**
	12	**195.44.136.9**	**n/a**	**7879**
	13	**195.44.137.186**	**n/a**	**753**
	13	**195.44.128.10**	**n/a**	**611**
	14	**195.44.128.6**	**n/a**	**750**
	14	**195.44.56.37**	**n/a**	**615**
	15	194.6.80.252	bgp-gw2.cwci.net	1378
	16	**195.44.50.150**	**n/a**	**325**
	16	**195.44.50.154**	**n/a**	**356**
	16	**195.44.50.162**	**n/a**	**345**
	16	**195.44.50.158**	**n/a**	**357**
	17	196.25.0.81	ndf-core2.wc.saix.net	1375
	18	196.25.0.22	cbs-core.wc.saix.net	1376
Reg. Network	19	196.25.3.22	scien-data-gw.wc.saix.net	1376
Remote LAN	20	196.25.18.137	sun.new.co.za	2009

Table 22: 1st Zimbabwe Typical Path - 03/11 ← → 9/11

Node Location	Node	IP Field	DNS Field	RTT Occurrences
Local Network	1	129.127.180.253	vlan0180.atm2-0.pancho.net.adelaide.edu.au	7963
Reg. Network	2	203.21.37.2	lis255.atm1-0.central.saard.net	7969
Aus Backbone	3	202.139.32.205	atm5-0-0-4.ia4.optus.net.au	7939
Aus Backbone	4	202.139.1.205	n/a	2417
US Backbone	**5**	**192.65.89.242**	**hssi11-0-0.la1.optus.net.au**	**1208**
US Backbone	**5**	**192.65.89.238**	**hssi9-0-0.la1.optus.net.au**	**1190**
US Backbone	6	157.130.227.181	34.ATM0-0-0.GW1.LAX4.ALTER.NET	2416
US Backbone	7	146.188.248.102	121.ATM2-0.XR2.LAX4.ALTER.NET	2414
US Backbone	8	146.188.249.25	193.ATM9-0-0.GW2.LAX4.ALTER.NET	2409
US Backbone	**9**	**No details**	**No details**	
US Backbone	10	207.45.222.181	if-1-1.core1.LosAngeles.Teleglobe.net	2408
US Backbone	11	207.45.222.25	if-6-2.core1.NewYork.Teleglobe.net	2407
US Backbone	12	207.45.223.49	if-0-0.core1.Scarborough.Teleglobe.net	1840
Can Backbone	13	207.45.223.38	if-0-2.core1.Montreal.Teleglobe.net	1840
Can Backbone	14	207.45.221.131	if-0-0-0.bb1.Montreal.Teleglobe.net	1940
Zimb Backbone	15	199.202.55.190	ix-11-4.bb1.Montreal.Teleglobe.net	2314
Reg. Network	16	194.133.159.26	n/a	2310
Remote LAN	17	196.7.224.44	tucows.harare.iafrica.com	2289

Table 23: 2nd Zimbabwe Typical Path - 9/11 ← → 11/11

Node Location	Node	IP Field	DNS Field	RTT Occurrences
Local Network	1	129.127.180.253	vlan0180.atm2-0.pancho.net.adelaide.edu.au	7963
Reg.Network	2	203.21.37.2	lis255.atm1-0.central.saard.net	7969
Aus Backbone	3	202.139.32.205	atm5-0-0-4.ia4.optus.net.au	7939
Aus Backbone	4	202.139.1.205	n/a	2417
US Backbone	**5**	**192.65.89.242**	**hssi11-0-0.la1.optus.net.au**	**1208**
US Backbone	**5**	**192.65.89.238**	**hssi9-0-0.la1.optus.net.au**	**1190**
US Backbone	6	157.130.227.181	34.ATM0-0-0.GW1.LAX4.ALTER.NET	2416
US Backbone	7	146.188.248.102	121.ATM2-0.XR2.LAX4.ALTER.NET	2414
US Backbone	8	146.188.249.25	193.ATM9-0-0.GW2.LAX4.ALTER.NET	2409
US Backbone	**9**	**No details**	**No details**	
US Backbone	10	207.45.222.181	if-1-1.core1.LosAngeles.Teleglobe.net	2408
US Backbone	11	207.45.222.25	if-6-2.core1.NewYork.Teleglobe.net	2407
Can Backbone	12	207.45.223.61	if-1-1.core1.Montreal.Teleglobe.net	555
Can Backbone	13	207.45.221.131	if-0-0-0.bb1.Montreal.Teleglobe.net	558
Zim Backbone	14	199.202.55.190	ix-11-4.bb1.Montreal.Teleglobe.net	557
Reg. Network	15	194.133.159.26	n/a	558
Remote LAN	16	196.7.224.44	tucows.harare.iafrica.com	556

Table 24: 3rd Zimbabwe Typical Path - 11/11← → 15/11

Node Location	Node	IP Field	DNS Field	RTT Occurrences
Local Network	1	129.127.180.253	vlan0180.atm2-0.pancho.net.adelaide.edu.au	7963
Reg. Network	2	203.21.37.2	lis255.atm1-0.central.saard.net	7969
Aus Backbone	3	202.139.32.205	atm5-0-0-4.ia4.optus.net.au	7939
Aus Backbone	4	202.139.1.197	n/a	5541
US Backbone	**5**	**192.65.89.246**	**hssi9-0-0.sf1.optus.net.au**	**747**
US Backbone	**5**	**192.65.89.230**	**hssi4-0-0.sf1.optus.net.au**	**2157**
US Backbone	**5**	**192.65.89.234**	**hssi11-0-0.sf1.optus.net.au**	**2179**
US Backbone	**5**	**192.65.89.226**	**Serial4-1-0.sf1.optus.net.au**	**445**
US Backbone	6	205.174.74.165	n/a	5064
US Backbone	7	134.24.45.5	atm10-0-1.sfo-bb2.cerf.net	902
US Backbone	8	134.24.29.197	pos0-2-155M.sfo-bb3.cerf.net	903
US Backbone	9	134.24.32.90	pos4-0-0-155M.sjc-bb3.cerf.net	903
US Backbone	**10**	**No details**		
US Backbone	11	207.45.222.234	if-4-0.core1.Denver.Teleglobe.net	903
US Backbone	12	207.45.222.226	if-3-0.core1.Chicago3.Teleglobe.net	904
US Backbone	13	207.45.222.209	if-8-1.core1.NewYork.Teleglobe.net	904
Can Backbone	14	207.45.220.5	if-5-1.core1.Montreal.Teleglobe.net	900
Can Backbone	15	207.45.221.131	if-0-0-0.bb1.Montreal.Teleglobe.net	1206
Zim Backbone	16	199.202.55.190	ix-11-4.bb1.Montreal.Teleglobe.net	3512
Reg. Network	17	194.133.159.26	n/a	3509
Remote LAN	18	196.7.224.44	tucows.harare.iafrica.com	3560

Table 25: 4th Zimbabwe Typical Path - 15/11 ← → 28/11

Node Location	Node	IP Field	DNS Field	RTT Occurrences
Local Network	1	129.127.180.253	vlan0180.atm2-0.pancho.net.adelaide.edu.au	7963
Reg. Network	2	203.21.37.2	lis255.atm1-0.central.saard.net	7969
Aus Backbone	3	202.139.32.205	atm5-0-0-4.ia4.optus.net.au	7939
Aus Backbone	4	202.139.1.197	n/a	5541
US Backbone	**5**	**192.65.89.246**	**hssi9-0-0.sf1.optus.net.au**	**747**
US Backbone	**5**	**192.65.89.230**	**hssi4-0-0.sf1.optus.net.au**	**2157**
US Backbone	**5**	**192.65.89.234**	**hssi11-0-0.sf1.optus.net.au**	**2179**
US Backbone	**5**	**192.65.89.226**	**Serial4-1-0.sf1.optus.net.au**	**445**
US Backbone	**6**	**205.174.74.165**	**n/a**	**5064**
US Backbone	7	12.127.11.29	gbr2-a30s14.sffca.ip.att.net	2613
US Backbone	8	12.122.1.13	gbr1-p60.sffca.ip.att.net	2606
US Backbone	9	12.122.2.70	gbr2-p50.la2ca.ip.att.net	2604
US Backbone	10	12.123.28.141	br1-p3120.la2ca.ip.att.net	2603
US Backbone	11	192.205.32.90	att-gw.la.teleglobe.net	2605
US Backbone	12	207.45.222.181	if-1-1.core1.LosAngeles.Teleglobe.net	2601
US Backbone	13	207.45.222.25	if-6-2.core1.NewYork.Teleglobe.net	2604
Can Backbone	14	207.45.223.61	if-1-1.core1.Montreal.Teleglobe.net	2609
Can Backbone	15	207.45.221.163	if-1-0-0.bb1.Montreal.Teleglobe.net	2307
Zim Backbone	16	199.202.55.190	ix-11-4.bb1.Montreal.Teleglobe.net	3512
Reg. Network	17	194.133.159.26	n/a	3509
Remote LAN	18	196.7.224.44	tucows.harare.iafrica.com	3560

Table 26: 5th Zimbabwe Typical Path - 11/11 & 18/11 ← → 20/11

Node Location	Node	IP Field	DNS Field	RTT Occurrences
Local Network	1	129.127.180.253	vlan0180.atm2-0.pancho.net.adelaide.edu.au	7963
Reg.Network	2	203.21.37.2	lis255.atm1-0.central.saard.net	7969
Aus Backbone	3	202.139.32.205	atm5-0-0-4.ia4.optus.net.au	7939
Aus Backbone	4	202.139.1.197	n/a	5541
US Backbone	**5**	**192.65.89.246**	**hssi9-0-0.sf1.optus.net.au**	**747**
US Backbone	**5**	**192.65.89.230**	**hssi4-0-0.sf1.optus.net.au**	**2157**
US Backbone	**5**	**192.65.89.234**	**hssi11-0-0.sf1.optus.net.au**	**2179**
US Backbone	**5**	**192.65.89.226**	**Serial4-1-0.sf1.optus.net.au**	**445**
US Backbone	6	205.174.74.165	n/a	5064
US Backbone	7	12.127.11.45	gbr1-a31s11.sffca.ip.att.net	472
US Backbone	8	12.122.2.70	gbr2-p50.la2ca.ip.att.net	479
US Backbone	9	12.123.28.141	br1-p3120.la2ca.ip.att.net	478
US Backbone	10	192.205.32.90	att-gw.la.teleglobe.net	481
US Backbone	11	207.45.222.181	if-1-1.core1.LosAngeles.Teleglobe.net	483
US Backbone	12	207.45.222.25	if-6-2.core1.NewYork.Teleglobe.net	483
Can Backbone	**13**	**207.45.220.5**	**if-5-1.core1.Montreal.Teleglobe.net**	**75**
Can Backbone	**13**	**207.45.223.61**	**if-1-1.core1.Montreal.Teleglobe.net**	**402**
Can Backbone	14	207.45.221.163	if-1-0-0.bb1.Montreal.Teleglobe.net	381
Zim Backbone	15	199.202.55.190	ix-11-4.bb1.Montreal.Teleglobe.net	2314
Reg. Network	16	194.133.159.26	n/a	2310
Remote LAN	17	196.7.224.44	tucows.harare.iafrica.com	2289

Table 27: 6th Zimbabwe Typical Path - 24/11 ← → 28/11

Node Location	Node	IP Field	DNS Field	RTT Occurrences
Local Network	1	129.127.180.253	vlan0180.atm2-0.pancho.net.adelaide.edu.au	7963
Reg. Network	2	203.21.37.2	lis255.atm1-0.central.saard.net	7969
Aus Backbone	3	202.139.32.205	atm5-0-0-4.ia4.optus.net.au	7939
Aus Backbone	4	202.139.1.197	n/a	5541
US Backbone	**5**	**192.65.89.246**	**hssi9-0-0.sf1.optus.net.au**	**747**
US Backbone	**5**	**192.65.89.230**	**hssi4-0-0.sf1.optus.net.au**	**2157**
US Backbone	**5**	**192.65.89.234**	**hssi11-0-0.sf1.optus.net.au**	**2179**
US Backbone	**5**	**192.65.89.226**	**Serial4-1-0.sf1.optus.net.au**	**445**
US Backbone	6	205.174.74.165	n/a	5064
US Backbone	7	4.1.142.253	s2-0-0.paloalto-cr18.bbnplanet.net	1054
US Backbone	8	4.0.3.85	p3-2.paloalto-nbr2.bbnplanet.net	1051
US Backbone	9	4.0.5.65	p1-0.paloalto-nbr1.bbnplanet.net	1054
US Backbone	10	4.0.6.45	p3-3.paix-bi1.bbnplanet.net	1057
US Backbone	**11**	**No details**		
US Backbone	12	207.45.222.234	if-4-0.core1.Denver.Teleglobe.net	1048
US Backbone	13	207.45.222.226	if-3-0.core1.Chicago3.Teleglobe.net	1045
US Backbone	14	207.45.222.209	if-8-1.core1.NewYork.Teleglobe.net	1047
Can Backbone	15	207.45.223.61	if-1-1.core1.Montreal.Teleglobe.net	1052
Can Backbone	16	207.45.221.163	if-1-0-0.bb1.Montreal.Teleglobe.net	1053
Zim Backbone	17	199.202.55.190	ix-11-4.bb1.Montreal.Teleglobe.net	1062
Reg. Network	18	194.133.159.26	n/a	1058
Remote LAN	19	196.7.224.44	tucows.harare.iafrica.com	1133

Table 28: 7th Zimbabwe Typical Path - 27/11 ← → 30/11

Node Location	Node	IP Field	DNS Field	RTT Occurrences
Local Network	1	129.127.180.253	vlan0180.atm2-0.pancho.net.adelaide.edu.au	7963
Reg. Network	2	203.21.37.2	lis255.atm1-0.central.saard.net	7969
Aus Backbone	3	202.139.32.205	atm5-0-0-4.ia4.optus.net.au	7939
Aus Backbone	4	202.139.1.197	n/a	5541
US Backbone	**5**	**192.65.89.246**	**hssi9-0-0.sf1.optus.net.au**	**747**
US Backbone	**5**	**192.65.89.230**	**hssi4-0-0.sf1.optus.net.au**	**2157**
US Backbone	**5**	**192.65.89.234**	**hssi11-0-0.sf1.optus.net.au**	**2179**
US Backbone	**5**	**192.65.89.226**	**Serial4-1-0.sf1.optus.net.au**	**445**
US Backbone	6	207.124.109.57	g-sfd-br-02-f12-0.gn.cwix.net	476
US Backbone	7	206.142.242.130	sfd-7513-2-f9-0.cwix.net	471
US Backbone	8	207.124.107.73	nyd-7513-1-a11-0-2.cwix.net	472
US Backbone	9	206.142.243.3	g-nyd-br-01-f5-0.gn.cwix.net	472
US Backbone	10	207.124.127.2	g-nyd-br-02-fe9-0.gn.cwix.net	470
US Backbone	11	157.130.19.25	Serial12-1-1.GW4.NYC4.ALTER.NET	466
US Backbone	12	146.188.179.162	146.ATM3-0.XR1.NYC4.ALTER.NET	465
US Backbone	13	146.188.180.33	189.ATM8-0-0.GW7.NYC4.ALTER.NET	466
US Backbone	14	157.130.6.34	teleglobe-ny3.customer.ALTER.NET	465
US Backbone	15	207.45.221.65	if-9-0.core1.NewYork.Teleglobe.net	465
Can Backbone	16	207.45.223.61	if-1-1.core1.Montreal.Teleglobe.net	465
Can Backbone	17	207.45.221.131	if-0-0-0.bb1.Montreal.Teleglobe.net	465
Zim Backbone	18	199.202.55.190	ix-11-4.bb1.Montreal.Teleglobe.net	471
Reg. Network	19	194.133.159.26	n/a	471
Remote LAN	20	196.7.224.44	tucows.harare.iafrica.com	517

Table 29: Summary of Performance findings from Throughput Experiment[1]

Application Server	Total Samples	Performance < 64 Kbps	< 64 Kbps (% Total)	"Timed Out"	"No response before 25 s"
South Australia	425	None	0	1	2
Victoria	425	30[2]	7.05	None	1
USA West Coast	259	None	0	2	1
USA East Coast	259	12	4.63	1	3
Hong Kong	425	21	4.94	2	1
Israel	425	106	24.94	None	4
Germany	256	194	75.78	6	1
England	258	18	6.97	2	42[3]
Argentina	253	208	82.21	7	6
Brazil	256	192	75	5	8
South Africa	254	177	69.68	5	20
Zimbabwe	252	221	87.69	4	8

[1] We do not consider samples that have "timed out" & "server not responding before 25 seconds" messages between 6 AM and 7 AM on November 14. During that period, a router memory upgrade was deployed in our University's router and the Internet connections were interrupted.
[2] All transfer rates below 64 Kbps happened after the Application Server was re-configured for a maximum of 256 Kbps per session.
[3] 40 of the 42 "No response" messages happened between late hours on Nov 19 and Nov 22, what suggests a server response re-configuration.

Table 30: Path Instability Analysis

Application Server	Total # Nodes	Total # Samples	Average # Nodes	COMPARE Output	# Changes after Filtering[1]	$\left(\frac{\text{Filtered Changes}}{\text{Samples}}\right)\%$	Path Instability (ε)
South Australia	40436	5686	7.11	24	4	0.07	0.56
Victoria	34171	5689	6	8	8	0.14	1.33
USA West Coast	103155	5688	18.13	30	24	0.42	1.32
USA East Coast	91460	5689	16.07	292	27	0.47	1.68
Hong Kong	45637	5687	8	9	9	0.16	1.12
Israel	97931	5688	17.21	88	48	0.84	2.79
Germany	114361	5689	20.1	41	38	0.67	1.89
England	86088	5688	15.14	73	25	0.44	1.65
Argentina	95048	5686	16.17	240	16	0.28	0.99
Brazil	96439	5628	17.13	35	12	0.21	0.70
South Africa	108066	5646	19.14	34	19	0.34	0.99
Zimbabwe	99719	5696	17.5	61	40	0.70	2.28

[1] This field shows all changes but changes within nodes returning "host/net unreachable" messages, changes in nodes with variation in IP number having similar DNS information, changes in node due to a load balance implementation.

Table 31: USA West Coast (Minimum Delay Found: 300.33 ms)
Number of Nodes: 14

Path	ADL-SYD	SYD-SF	SF-LA	LA-SF	SF-LA	LA-SF	SF-SYD	SYD-ADL	Propagation Delay	Total Delay (Propagation +Transmission)
Distance (Km)	1152	11960	559	559	559	559	11960	1152		
Path A	Fibre	Fibre	Fibre	Fibre	Fibre	Fibre	Fibre	Fibre	142.30	161.90
Path B	**Fibre**	**Satellite**	**Fibre**	**Fibre**	**Fibre**	**Fibre**	**Fibre**	**Fibre**	**332.50**	**352.10**
Path C	Fibre	Satellite	Fibre	Fibre	Fibre	Fibre	Satellite	Fibre	522.70	542.30

Table 32: Hong Kong (Minimum Delay Found: 170.00 ms)
Number of Nodes: 8

Path	ADL-SYD	SYD-HK	HK-LA	LA-SYD	SYD-ADL	Propagation Delay	Total Delay (Propagation + Transmission)
Distance (Km)	1152	7394	11645	12085	1152		
Path A	**Fibre**	**Fibre**	**Fibre**	**Fibre**	**Fibre**	**167.14**	**178.34**
Path B	Fibre	Fibre	Fibre	Satellite	Fibre	356.72	367.92
Path C	Fibre	Fibre	Satellite	Fibre	Fibre	358.92	370.12
Path D	Fibre	Satellite	Fibre	Fibre	Fibre	380.17	391.37
Path E	Fibre	Satellite	Satellite	Fibre	Fibre	571.95	583.145
Path F	Fibre	Satellite	Fibre	Satellite	Fibre	569.75	580.94
Path G	Fibre	Fibre	Satellite	Satellite	Fibre	548.49	559.69
Path H	Fibre	Satellite	Satellite	Satellite	Fibre	761.52	772.72

Table 33: Israel (Minimum Delay Found: 471.00 ms)
Number of Nodes: 17

Path	ADL-SYD	SYD-LA	LA-NY	NY-TV	TV-NY	NY-LA	LA-SYD	SYD-ADL	Propagation Delay	Total Delay (Propagation + Transmission)
Distances	1152	12085	3937	9112	9112	3937	12085	1152		
Path A	Fibre	Fibre	Fibre	Fibre	Fibre	Fibre	Fibre	Fibre	262.86	279.86
Path B	**Fibre**	**Fibre**	**Fibre**	**Fibre**	**Fibre**	**Fibre**	**Satellite**	**Fibre**	**452.44**	**469.44**
Path C	Fibre	Fibre	Fibre	Satellite	Fibre	Fibre	Satellite	Fibre	656.88	673.88
Path D	Fibre	Fibre	Fibre	Satellite	Fibre	Fibre	Satellite	Fibre	861.32	878.32
Path E	**Fibre**	**Fibre**	**Fibre**	**Fibre**	**Satellite**	**Fibre**	**Fibre**	**Fibre**	**467.3**	**484.30**
Path F	Fibre	Satellite	Fibre	Fibre	Fibre	Fibre	Satellite	Fibre	642.01	659.01
Path G	Fibre	Satellite	Fibre	Satellite	Fibre	Fibre	Satellite	Fibre	846.45	863.45
Path H	Fibre	Satellite	Fibre	Satellite	Satellite	Fibre	Satellite	Fibre	1050.89	1067.89
Path I	Fibre	Fibre	Fibre	Satellite	Satellite	Fibre	Fibre	Fibre	671.74	688.74

Table 34: Germany (Minimum Delay Found: 437.33 ms)
Number of Nodes: 21

Path	ADL-SYD	SYD-SF	SF-NY	NY-FR	FR-NY	NY-SF	SF-SYD	SYD-ADL	Propagation Delay	Total Delay (Propagation +Transmission)
Distances	1152	11960	4129	6199	6199	4129	11960	1152		
Path A	Fibre	Fibre	Fibre	Fibre	Fibre	Fibre	Fibre	Fibre	234.4	263.8
Path B	**Fibre**	**Fibre**	**Fibre**	**Fibre**	**Fibre**	**Fibre**	**Satellite**	**Fibre**	**424.6**	**454**
Path C	Fibre	Fibre	Fibre	Satellite	Fibre	Fibre	Satellite	Fibre	643.605	673.005
Path D	Fibre	Fibre	Fibre	Satellite	Satellite	Fibre	Satellite	Fibre	862.61	892.01
Path E	**Fibre**	**Fibre**	**Fibre**	**Fibre**	**Satellite**	**Fibre**	**Fibre**	**Fibre**	**453.405**	**482.805**
Path F	Fibre	Satellite	Fibre	Fibre	Fibre	Fibre	Satellite	Fibre	614.8	644.2
Path G	Fibre	Satellite	Fibre	Satellite	Fibre	Fibre	Satellite	Fibre	833.805	863.205
Path H	Fibre	Satellite	Fibre	Satellite	Satellite	Fibre	Satellite	Fibre	1052.81	1082.21
Path J	Fibre	Fibre	Fibre	Satellite	Satellite	Fibre	Fibre	Fibre	672.41	701.81

Table 35: England (Minimum Delay Found: 450.66 ms)
Number of Nodes: 15

Path	**ADL-SYD**	**SYD-LA**	**LA-WN**	**WN-LN**	**LN-WN**	**WN-LA**	**LA-SYD**	**SYD-ADL**	**Propagation Delay**	**Total Delay (Propagation +Transmission)**
Distances	1152	12085	3692	5903	5903	3692	12085	1152		
Path A	Fibre	Fibre	Fibre	Fibre	Fibre	Fibre	Fibre	Fibre	228.32	249.32
Path B	**Fibre**	**Fibre**	**Fibre**	**Fibre**	**Fibre**	**Fibre**	**Satellite**	**Fibre**	**417.90**	**438.90**
Path C	Fibre	Fibre	Fibre	Satellite	Fibre	Fibre	Satellite	Fibre	638.38	659.38
Path D	Fibre	Fibre	Fibre	Satellite	Satellite	Fibre	Satellite	Fibre	858.86	879.86
Path E	**Fibre**	**Fibre**	**Fibre**	**Fibre**	**Satellite**	**Fibre**	**Fibre**	**Fibre**	**448.80**	**469.80**
Path F	Fibre	Satellite	Fibre	Fibre	Fibre	Fibre	Satellite	Fibre	607.47	628.47
Path G	Fibre	Satellite	Fibre	Satellite	Fibre	Fibre	Satellite	Fibre	827.96	848.96
Path H	Fibre	Satellite	Fibre	Satellite	Satellite	Fibre	Satellite	Fibre	1048.44	1069.44
Path I	Fibre	Fibre	Fibre	Satellite	Satellite	Fibre	Fibre	Fibre	669.29	690.29

Table 36: Argentina (Minimum Delay Found: 788.00 ms)
Number of Nodes: 17

Path	ADL-SYD	SYD-LA	LA-ATL	ATL-ORL	ORL-MIA	MIA-ORL	ORL-BUE	BUE-ORL	ORL-MIA	MIA-ORL	ORL-ATL	ATL-LA	LA-SYD	SYD-ADL	Propagation Delay	Total Delay (Propagation + Transmission)
Dist	1152	12085	3110	646	329	329	7360	7360	329	329	646	3110	12085	1152		
Path A	Fibre	Fibre	Fibre	Fibre	Fibre	Fibre	Fibre	Fibre	Fibre	Fibre	Fibre	Fibre	Fibre	Fibre	281.21	305.01
Path B	Fibre	Fibre	Fibre	Fibre	Fibre	Fibre	Fibre	Fibre	Fibre	Fibre	Fibre	Fibre	Satellite	Fibre	470.78	494.58
Path C	**Fibre**	**Fibre**	**Fibre**	**Fibre**	**Fibre**	**Fibre**	**Satellite**	**Fibre**	**Fibre**	**Fibre**	**Fibre**	**Fibre**	**Satellite**	**Fibre**	**683.98**	707.78
Path D	Fibre	Fibre	Fibre	Fibre	Fibre	Fibre	Satellite	Satellite	Fibre	Fibre	Fibre	Fibre	Satellite	Fibre	897.18	920.98
Path E	Fibre	Fibre	Fibre	Fibre	Fibre	Fibre	Fibre	Satellite	Fibre	Fibre	Fibre	Fibre	Fibre	Fibre	494.41	518.21
Path F	Fibre	Satellite	Fibre	Fibre	Fibre	Fibre	Fibre	Fibre	Fibre	Fibre	Fibre	Fibre	Satellite	Fibre	660.36	684.16
Path G	**Fibre**	**Satellite**	**Fibre**	**Fibre**	**Fibre**	**Fibre**	**Satellite**	**Fibre**	**Fibre**	**Fibre**	**Fibre**	**Fibre**	**Satellite**	**Fibre**	**873.56**	**897.36**
Path H	Fibre	Satellite	Fibre	Fibre	Fibre	Fibre	Satellite	Satellite	Fibre	Fibre	Fibre	Fibre	Satellite	Fibre	1086.76	1110.56
Path I	**Fibre**	**Fibre**	**Fibre**	**Fibre**	**Fibre**	**Fibre**	**Satellite**	**Satellite**	**Fibre**	**Fibre**	**Fibre**	**Fibre**	**Fibre**	**Fibre**	**707.61**	**731.41**

Table 37: Brazil (Minimum Delay Found: 691.00 ms)
Number of Nodes: 17

Path	**ADL-SYD**	**SYD-LA**	**LA-NJ**	**NJ-SP**	**SP-NJ**	**NJ-LA**	**LA-SYD**	**SYD-ADL**	**Propagation Delay**	**Total Delay (Propagation + Transmission)**
Distances	1152	12085	3922	7695	7695	3922	12085	1152		
Path A	Fibre	Fibre	Fibre	Fibre	Fibre	Fibre	Fibre	Fibre	248.54	272.34
Path B	Fibre	Fibre	Fibre	Fibre	Fibre	Fibre	Satellite	Fibre	438.12	461.915
Path C	**Fibre**	**Fibre**	**Fibre**	**Satellite**	**Fibre**	**Fibre**	**Satellite**	**Fibre**	**649.64**	**673.44**
Path D	Fibre	Fibre	Fibre	Satellite	Satellite	Fibre	Satellite	Fibre	861.16	884.96
Path E	Fibre	Fibre	Fibre	Fibre	Satellite	Fibre	Fibre	Fibre	460.06	483.86
Path F	**Fibre**	**Satellite**	**Fibre**	**Fibre**	**Fibre**	**Fibre**	**Satellite**	**Fibre**	**627.69**	**651.49**
Path G	Fibre	Satellite	Fibre	Satellite	Fibre	Fibre	Satellite	Fibre	839.215	863.015
Path H	Fibre	Satellite	Fibre	Satellite	Satellite	Fibre	Satellite	Fibre	1050.74	1074.54
Path I	**Fibre**	**Fibre**	**Fibre**	**Satellite**	**Satellite**	**Fibre**	**Fibre**	**Fibre**	**671.59**	**695.39**

Table 38: South Africa (Minimum Delay Found: 604.33 ms)
Number of Nodes: 19

Path	ADL-SYD	SYD-SF	SF-NY	NY-CT	CT-NY	NY-SF	SF-SYD	SYD-ADL	Propagation Delay	Total Delay (Propagation +Transmission)
Distances	1152	11960	4129	12572	12572	4129	11960	1152		
Path A	Fibre	Fibre	Fibre	Fibre	Fibre	Fibre	Fibre	Fibre	298.13	324.73
Path B	**Fibre**	**Fibre**	**Fibre**	**Fibre**	**Fibre**	**Fibre**	**Satellite**	**Fibre**	**488.33**	**514.93**
Path C	**Fibre**	**Fibre**	**Fibre**	**Satellite**	**Fibre**	**Fibre**	**Satellite**	**Fibre**	**675.47**	**702.07**
Path D	Fibre	Fibre	Fibre	Satellite	Satellite	Fibre	Satellite	Fibre	862.61	889.21
Path E	Fibre	Fibre	Fibre	Fibre	Satellite	Fibre	Fibre	Fibre	485.27	511.87
Path F	Fibre	Satellite	Fibre	Fibre	Fibre	Fibre	Satellite	Fibre	678.53	705.13
Path G	Fibre	Satellite	Fibre	Satellite	Fibre	Fibre	Satellite	Fibre	865.67	892.27
Path H	Fibre	Satellite	Fibre	Satellite	Satellite	Fibre	Satellite	Fibre	1052.81	1079.41
Path I	Fibre	Fibre	Fibre	Satellite	Satellite	Fibre	Fibre	Fibre	672.41	699.01

Table 39: Zimbabwe (Minimum Delay Found: 851.33 ms)
Number of Nodes: 16

Path	ADL-SYD	SYD-LA	LA-NY	NY-MTL	MTL-HRR	MTL-HRR	MTL-NY	NY-LA	LA-SYD	SYD-ADL	Propagation Delay	Total Delay (Propagation+ Transmission)
Distances	1152	12085	3922	510	12518	12518	510	3922	12085	1152		
Path A	Fibre	Fibre	Fibre	Fibre	Fibre	Fibre	Fibre	Fibre	Fibre	Fibre	301.87	325.67
Path B	Fibre	Fibre	Fibre	Fibre	Fibre	Fibre	Fibre	Fibre	Satellite	Fibre	491.44	515.24
Path C	Fibre	Fibre	Fibre	Fibre	Satellite	Fibre	Fibre	Fibre	Satellite	Fibre	678.86	702.66
Path D	**Fibre**	**Fibre**	**Fibre**	**Fibre**	**Satellite**	**Satellite**	**Fibre**	**Fibre**	**Satellite**	**Fibre**	**861.16**	**884.97**
Path E	Fibre	Fibre	Fibre	Fibre	Fibre	Satellite	Fibre	Fibre	Fibre	Fibre	489.28	513.08
Path F	Fibre	Satellite	Fibre	Fibre	Fibre	Fibre	Fibre	Fibre	Satellite	Fibre	681.02	704.82
Path G	**Fibre**	**Satellite**	**Fibre**	**Fibre**	**Satellite**	**Fibre**	**Fibre**	**Fibre**	**Satellite**	**Fibre**	**868.43**	**892.23**
Path H	Fibre	Satellite	Fibre	Fibre	Satellite	Satellite	Fibre	Fibre	Satellite	Fibre	1055.84	1079.64
Path I	Fibre	Fibre	Fibre	Fibre	Satellite	Satellite	Fibre	Fibre	Fibre	Fibre	676.69	700.49