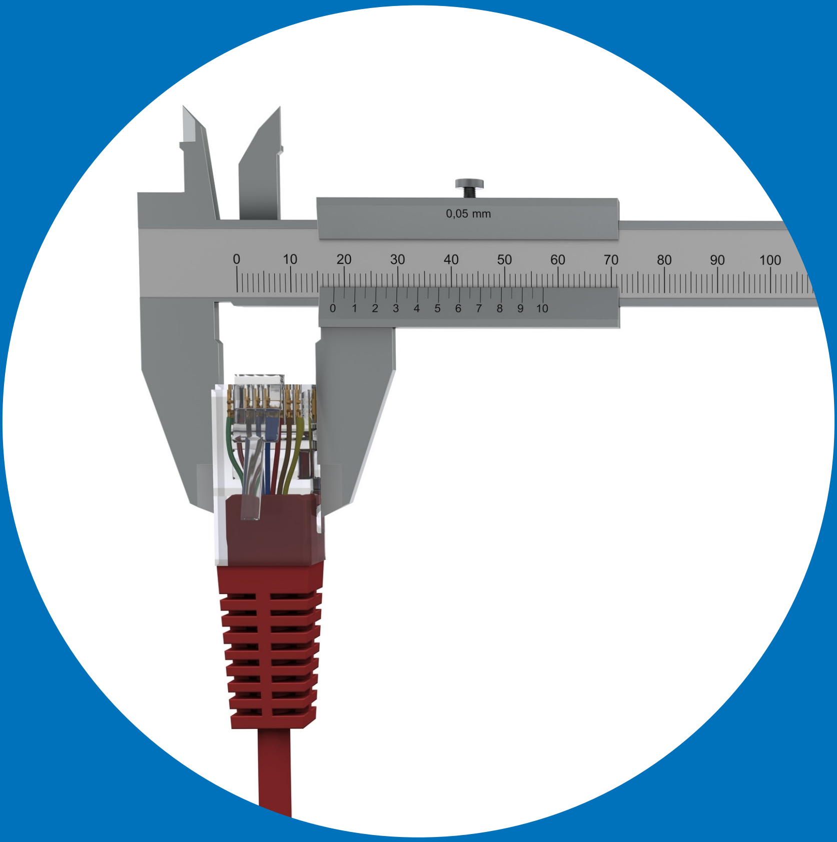
Department of Computer Science and Engineering

# Calibration and Analysis of Enterprise and Edge Network Measurements

**Boris Nechaev**



Aalto University

DOCTORAL DISSERTATIONS

Aalto University publication series
**DOCTORAL DISSERTATIONS** 201/2013

# Calibration and Analysis of Enterprise and Edge Network Measurements

**Boris Nechaev**

A doctoral dissertation completed for the degree of Doctor of Science (Technology) to be defended, with the permission of the Aalto University School of Science, at a public examination held at the lecture hall T2 of the school on December 13th, 2013, at 12 noon.

**Aalto University**
**School of Science**
**Department of Computer Science and Engineering**

**Supervising professor**
Professor Antti Ylä-Jääski

**Thesis advisors**
Professor Vern Paxson
Professor Andrei Gurtov

**Preliminary examiners**
Professor Paul Barford, University of Wisconsin, USA
Dr. Renata Teixeira, Laboratoire d’Informatique de Paris 6 (LIP6), France

**Opponent**
Dr. Walter Willinger, Niksun Inc, USA

Aalto University publication series
**DOCTORAL DISSERTATIONS** 201/2013

ISBN 978-952-60-5466-7
ISBN 978-952-60-5467-4 (pdf)
ISSN-L 1799-4934
ISSN 1799-4934 (printed)
ISSN 1799-4942 (pdf)
http://urn.fi/URN:ISBN:978-952-60-5467-4

Unigrafia Oy
Helsinki 2013

Finland

Publication orders (printed book):
boris.nechaev@hiit.fi

NORDIC ECOLABEL
441 697
Printed matter

**Aalto University**

## Abstract

**Aalto University, P.O. Box 11000, FI-00076 Aalto** www.aalto.fi

**Author**
Boris Nechaev

**Name of the doctoral dissertation**
Calibration and Analysis of Enterprise and Edge Network Measurements

**Publisher** School of Science

**Unit** Department of Computer Science and Engineering

**Series** Aalto University publication series DOCTORAL DISSERTATIONS 201/2013

**Field of research** Network Measurements

**Manuscript submitted** 9 September 2013 **Date of the defence** 13 December 2013

**Permission to publish granted (date)** 5 November 2013 **Language** English

☐ **Monograph** ☒ **Article dissertation (summary + original articles)**

### Abstract

With the growth of the Internet over the past several decades, the field of Internet and network measurements has attracted the attention of many researchers. Doing the measurements has allowed a better understanding of the inner workings of both the global Internet and its specific parts. But undertaking a measurement study in a sound fashion is no easy task. Given the complexity of modern networks, one has to take great care in anticipating, detecting and eliminating all the measurement errors and biases.

In this thesis we pave the way for a more systematic calibration of network traces. Such calibration ensures the soundness and robustness of the analysis results by revealing and fixing flaws in the data. We collect our measurement data in two environments: in a medium-sized enterprise and at the Internet edge. For the former we perform two rounds of data collection from the enterprise switches. We use the differences in the way we recorded the network traces during the first and second rounds to develop and assess the methodology for five calibration aspects: measurement gain, measurement loss, measurement reordering, timing, and topology. For the dataset gathered at the Internet edge, we perform calibration in the form of extensive checks of data consistency and sanity.

After calibrating the data, we engage in the analysis of its various aspects. For the enterprise dataset we look at TCP dynamics in the enterprise environment. Here we first make a high-level overview of TCP connection characteristics such as termination status, size, duration, rate, etc. Then we assess the parameters important for TCP performance, such as retransmissions, out-of-order deliveries and channel utilization. Finally, using the Internet edge dataset, we gauge the performance characteristics of the edge connectivity.

**Keywords** calibration, sound network measurements, enterprise switch measurements, edge networks

**ISBN (printed)** 978-952-60-5466-7 **ISBN (pdf)** 978-952-60-5467-4

**ISSN-L** 1799-4934 **ISSN (printed)** 1799-4934 **ISSN (pdf)** 1799-4942

**Location of publisher** Helsinki **Location of printing** Helsinki **Year** 2013

**Pages** 154 **urn** http://urn.fi/URN:ISBN:978-952-60-5467-4

*To my parents*

# Preface

During the journey towards the moment when I could proudly present this dissertation to a wide audience, I have met and received support from many wonderful people. I would like to thank all those who knowingly or unknowingly contributed to the completion of this dissertation.

I am grateful to my supervising professor Antti Ylä-Jääski for signing my Application for Doctoral Studies form all those years ago – this is how the journey officially started. During the years of my study and especially during the dissertation preparation phase, Antti was always there for me, ready to help me tackle administrative hurdles.

I am extremely honored to have Vern Paxson as my dissertation instructor. Vern was the one who encouraged me to start working on the topic of this thesis and then continually provided crucial scientific guidance. He was ever eager to answer even my most stupid questions, and by doing so taught me most of the data analysis skills I now possess. Vern shaped my understanding of research process and demonstrated the values a true scientist must always follow. He is my scientist role model.

Most of the time during my PhD studies I spent at HIIT, and my second instructor Andrei Gurtov was the one who invited me there. I thank Andrei for providing me with the financial support for doing my research.

The wonderful time at HIIT was made possible by my amazing colleagues. Dmitry Korzun was one of the first passionate scientists I have met in my career. His critical thinking skills and hardworking attitude will always be an example for me. I had the pleasure to co-author multiple papers with Dmitriy Kuptsov. Thank you for all the fun we had – it's been a great ride! Andrey Lukyanenko always astonished me with his deep knowledge of mathematics, so I always knew whom to contact when I struggled with a math problem. At HIIT I also had the pleasure to work and study with Andrey Khurri, Katerina Vorobyeva, Miika Komu, Samu

Varjonen, Alexander Gogolev, Joakim Koskela, Oleg Ponomarev, Tatiana Polishchuk, Ramya Sri Kalyanaraman and Ilya Nikolaevskiy.

During my PhD study years, I paid three research visits to the International Computer Science Institute (ICSI), Berkeley, California. These visits proved to be pivotal for my doctoral studies. Not only was the topic of this dissertation conceived in the first visit, but also during the following stays I had wonderful opportunities to broaden the research topic and eventually bring it to completion. I would like to thank Mark Allman for his enormous support in working on the joint papers. Mark's unique sense of humor always helped me move forward during tough times. I greatly enjoyed collaborating with Christian Kreibich, Robin Sommer, Nicholas Weaver and Gregor Maier. I am grateful to both the HIIT and ICSI administrative staff for always being eager to help me in any matters.

My research visits to ICSI were funded by FICNIA. Thus, I am grateful to Martti Mäntylä, Jussi Kangasharju, Scott Shenker and Kristiina Karvonen, who approved by eligibility for the visits. I would like to thank Kristiina Karvonen for the discussions, which helped me navigate the waters of the scientific world. I am also grateful to the FIGS doctoral programme for the funding I received at some points during my PhD studies.

I thank the pre-examiners of my dissertation Paul Barford and Renata Teixeira for finding the time to review my work and help improve it, as well as Walter Willinger for agreeing to serve as my opponent.

My gratitude goes to all my friends who sincerely supported me and never tired of asking when I was going to graduate. Here you go!

I wish to thank my parents and my family for all the love and support they gave to me. My father Igor Nechaev is the one who opened for me the door to the breathtakingly thrilling world of science and technology. My mother Natalya Nechaeva is my life-long tutor and moral compass. Mother, Father, I dedicate this dissertation to you.

Finally, I want to thank the single most important person in my life – my wife Assel Mukhametzhanova. It is her support, patience, love, and faith in me that made all this possible. I love you!

Helsinki, November 12, 2013,

Boris Nechaev

# Contents

# List of Publications

This thesis consists of an overview and of the following publications which are referred to in the text by their Roman numerals.

**I** Boris Nechaev, Vern Paxson, Mark Allman, Andrei Gurtov. On Calibrating Enterprise Switch Measurements. In *Proceedings of the ACM SIGCOMM / USENIX Internet Measurement Conference (IMC'09)*, Chicago, Illinois, USA, pp. 143–155, November 2009.

**II** Boris Nechaev, Vern Paxson, Mark Allman, Mike Bennett, Andrei Gurtov. Towards Methodical Calibration: A Case Study of Enterprise Switch Measurements. *ICSI Technical Report TR-13-005*, September 2013.

**III** Boris Nechaev, Mark Allman, Vern Paxson, Andrei Gurtov. A Preliminary Analysis of TCP Performance in an Enterprise Network. In *Proceedings of the USENIX Internet Network Management Conference on Research on Enterprise Networking (INM / WREN'10)*, San Jose, CA, USA, 2010.

**IV** Christian Kreibich, Nicholas Weaver, Boris Nechaev, Vern Paxson. Netalyzr: Illuminating The Edge Network. In *Proceedings of the ACM SIGCOMM / USENIX Internet Measurement Conference (IMC'10)*, Melbourne, Australia, pp. 246–259, November 2010.

# Author's Contribution

### Publication I: "On Calibrating Enterprise Switch Measurements"

The author of this thesis is the lead ideologist of the paper. He performed the extensive initial exploration of the network traces, which led to the idea for the paper. He was also actively engaged in almost all the calibration and analysis efforts described in the publication.

### Publication II: "Towards Methodical Calibration: A Case Study of Enterprise Switch Measurements"

The author of this thesis is the lead ideologist of the paper. He was involved in the collection of the enterprise switch traces and executed all calibration procedures except for calibration of measurement reordering.

### Publication III: "A Preliminary Analysis of TCP Performance in an Enterprise Network"

The author of this thesis is an active contributer to the paper. He prepared the trace dataset for the analysis and contributed to all the analysis aspects described in the publication.

### Publication IV: "Netalyzr: Illuminating The Edge Network"

The author of this thesis was responsible for the whole calibration effort described in the publication. He also performed the analysis of results for several Netalyzr tests.

# 1. Introduction

During the last decade the Internet has become a vital part of our society. The pervasive web of its connections penetrates almost all aspects of our lives. Billions of devices, ranging from tiny sensors to personal computers to supercomputing clusters, use the Internet to communicate. The Internet carries personal data, news, commercial information, financial transactions, entertainment media, and many other types of information. Data flows in a multitude of formats: audio, video, text, compressed, cyphered, etc.

Millions of individuals, companies, enterprises and government agencies rely on the Internet in their operations. The importance of the Internet in our everyday lives dictates high demands for its availability. For instance, even a break in operations lasting as little as tens of minutes may lead to considerable monetary losses for a financial institution or a popular website.

The ability to assure stable working of the Internet relies heavily on our ability to understand this system. Understanding of its structure and dynamics is required to pin-point the cause of a problem and eliminate it fast. Also, only in a system which is well understood, is it possible to avoid the same problem appearing again. Unfortunately, there are several factors that severely complicate the task of studying the Internet.

One of the main reasons which makes it difficult to understand how the Internet operates is its size. When the Internet was conceived as a research project, it consisted of only a handful of nodes. The next step was connecting dozens of research institutes and universities. The explosive growth started in the early 90's, when households and enterprises began joining the network. According to the International Telecommunication Union, in 2011 the Internet was used by 2.5 billion people [74]. In the same year, Cisco estimated the amount of global IP traffic to be

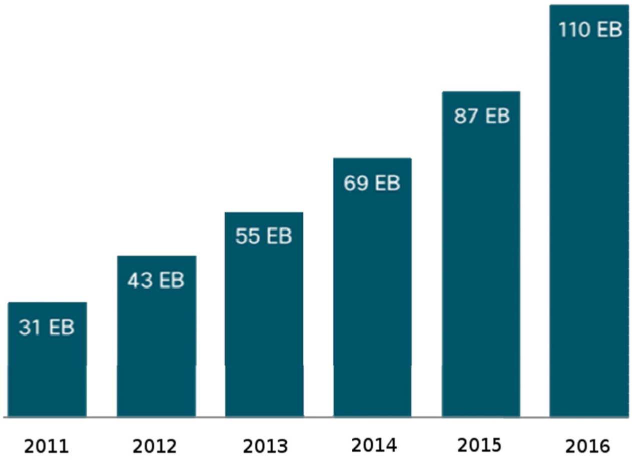


**Figure 1.1.** Internet IP traffic growth, Exabytes per month [33].

31 Exabytes (or 31 million Terabytes) per month [33]. The same study predicted no signs of slowing down and projected the amount of global IP traffic would reach 110 Exabytes per month by 2016.

The huge size of the modern Internet leads to several consequences, which impede easy understanding of the Internet's structure and properties. First, there is no centralized control over the Internet, i.e. no single entity which oversees its functioning and development. In part, this is due to the fact that with its big size and pervasiveness, the Internet is a highly decentralized system, spanning all continents and most countries. Thus, there is no authority that would possess a "map of the Internet" or a detailed plan of its components and their functions.

Another consequence of the Internet's immense size is the variety of user and protocol behavior. Even though the IETF produces RFCs and Internet drafts that should unambiguously specify a protocol's inter-operation [20], the implementation details of different vendors still vary [58, 132, 97]. This may lead to unpredictable results, when deployed in the wild. User behavior also differs greatly due to interests, technical proficiency, age, culture, and other sociodemographic reasons. Technical diversity implies that the numerous devices connected to the Internet have different computational and power resources. All these factors result in the diverse traffic patterns seen in the Internet. The size of the Internet also implies that even highly improbable scenarios see the light of the day in the Internet sooner rather than later.

The final main reason complicating our understanding of the Internet is its dynamic nature. The Internet changes fast over time on various levels. It may be a change in topology, e.g. when a new high bandwidth link is installed or a new Internet Exchange Point is joined by many ISPs. Alternatively, it may be a new protocol which drastically alters usage modes (such as the invention of HTTP [47]), or even a small-scale spike in the amount of traffic, for instance, due to public interest in a popular event. No matter how comprehensive a study of some aspect of the Internet is, it is in danger of being not fully representative, and is destined to get outdated quickly. This means that the quest for understanding the Internet has to be a never-ending endeavor.

In the long tradition of science originating in ancient Greece, the first step in studying and understanding a complex object or a system is the act of observation [50]. Aristotle, and many generations of natural scientists after him, performed careful observation of stars, planets, societies, animals, micro-organisms, elementary particles, etc. The observation step has several objectives: to collect experimental data (or do measurements, i.e. collect quantitative data), to discover phenomena unexplainable within the current scientific paradigm, and to confirm or refute existing theories. Even though the Internet was designed by humans and, therefore, seemingly, must have known properties, in its current state it is so big and complex that it requires constant observations to study its mechanics. Consequently, the discipline of network measurements emerged to tackle this problem.

Network and Internet measurements, which is the broad domain of this thesis, is a wide area of computer network research with many tasks. The primary task is collecting network data e.g. in the form of `tcpdump` traces or network delay measurements. Next, a researcher must pre-process the collected data, which will put it into a coherent form and allow the performance of the next step. In this thesis we call the act of ensuring the soundness of the data *calibration* of the dataset. Analysis of calibrated data is aimed at discovering unanticipated phenomena such as protocol malfunctions and inefficiencies, routing anomalies, congested links, etc. Besides finding unusual phenomena, another important goal of network measurements is to obtain reliable characteristics of the Internet, such as a typical topology, distribution of flow sizes and rates, dynamics of traffic matrices, etc. All these characteristics help in simulating the Internet and building analytical models by providing realistic values obtained in measurements to be used as model or simulation parameters.

The scope of Internet measurements is certainly not limited to its technical aspects. Social and demographic statistics is important to government agencies, ISPs, companies, and various different enterprises. For instance, an ISP may be interested in following the trends in the popularity of certain applications since the wide usage of some applications such as Napster, KaZaa and BitTorrent can have big impact on the amount and type of network traffic flowing via the ISP. A company owning a website would in most cases be very interested in the demographic distribution of its visitors so that it can improve its marketing efforts, or in case of a large corporation, make a decision to build data centers geographically closer to its customers. The degree of Internet connectivity among the population can be used by international organizations as one of the metrics for scoring the country's quality of life and technological level.

Internet measurements vary in type, location, scale, the component of the Internet they focus on and other characteristics. Typical topics and objects considered in network measurements are: user behavior, protocol dynamics, topology, traffic, applications, infrastructure, end-hosts, performance, etc. The two main types of network measurements are active and passive measurements. The former relies on active involvement in the data collection process, usually in the form of injecting traffic into the network. For instance, here one may send ICMP probes or fully saturate a link to examine its bottleneck bandwidth. Passive measurements are limited to capturing existing network traffic without interfering. The examples of passive measurements are recording BGP updates and capturing all network traffic flowing in the network for later analysis. It is not uncommon for Internet measurement studies to incorporate both active and passive methods.

It is possible to perform measurements at various locations in the Internet. Researchers can obtain the broadest view when capturing traffic in the Internet backbone. To do so they can place a capturing apparatus inside a bigger Tier 1, a smaller regional ISPs, their peering links or Internet Exchange Points. One a smaller scale, one can record network traffic at enterprises, campuses, data centers and local area networks. In each of the above settings, be it an Autonomous System, an enterprise or a LAN, it is possible to place monitoring equipment in the core of the given network or at its boundary. Each of the two has its benefits: the former gives a more comprehensive view of network activity inside the organization, while the latter allows the capture of traffic entering and leaving the network.

Obviously, the location and type of network measurements are defined by the goals of the study. For instance, one of the key objectives of backbone measurements is capacity planning. Routine monitoring of link utilization helps to optimally allocate bandwidth in an ISP and accurately calculate the amount of over-provisioning required in the network to meet the service level agreements (SLAs). Such measurements also allow the detection of failures in links at various levels. Monitoring inside enterprises or data centers may, for example, pursue the questions of characterizing user behavior or specific properties of transport protocols. Some enterprises also perform active and passive monitoring of their networks to identify security threats and detect hosts infected with Internet worms.

On the smallest scale, network measurements and data collection can be part of a controlled experiment. Such studies can be executed in a small wired or wireless LAN, or a testbed such as ORBIT [124] and PlanetLab [116]. Such experiments are typically aimed at studying the properties and performance of a new protocol or a modification of an existing one.

Performing sound Internet measurements is a challenging task [113]. There are a multitude of factors that can interfere with the measurement process and corrupt the results. For instance, the capturing apparatus may not be recording the data in its entirety (i.e. some data may get lost or be discarded at the filtering stage due to a bug) or it may be recording data in a wrong format. This may require reconstruction of the missing data if possible, or repeating the measurements if reconstruction is impossible and the missing data is crucial for further analysis. Sometimes conversion from the current data format to a format suitable for analysis is required. Alternatively, the amount of data may be big enough to severely complicate its handling and analysis. In such cases it is necessary to either discard some of the data or aggregate it. Additionally, measurements may have precision and accuracy issues, contain biases, be self-inconsistent, etc. These issues often go unnoticed during the data collection phase. Therefore, it is very important to perform sanity checks and properly calibrate the data before moving on to the analysis phase.

## 1.1 Research Questions, Scope and Methodology

The main broad research question we investigate in this thesis is how to properly collect, calibrate, and analyze network traces. These three tasks are intimately interconnected. For instance, the way the data is collected imposes certain limits on what conclusions can be drawn from it later. It also determines what calibration techniques have to be applied to the collected network traces before the proper analysis of the data becomes possible.

Due to the large scope of the above research question and the diversity of scenarios and configurations in the Internet, it is impossible to devise a general set of techniques that would guide data collection, calibration and analysis in all environments. Therefore, we limit the scope of this thesis to enterprise and edge networks. The techniques and methodology we develop are mostly applicable to active and passive network measurements conducted in these types of networks because many details of data collection and calibration are closely related to the topological and usage pattern specifics of these networks. However, other types of networks and scenarios can bear certain similarities to enterprise and edge networks, which means that some of the findings described in this thesis can be used there. For instance, our enterprise traces were collected from switches, which means that the same techniques can be utilized in modern data centers, which are built using a large number of commodity switches.

Thus, the *first research question* we address in this thesis is: **How should one collect network traces in enterprise and edge environments?** For enterprise measurements we further narrow down this question to switch-based measurements, i.e. we ask what are the possible ways of collecting packet-level data at the enterprise switches. To address this question we propose and evaluate two capturing schemas, with which we perform two rounds of data collection in the same enterprise. Each of the two ways has its benefits and pitfalls, thus requiring different calibration procedures. Moreover, the design of the second schema provided us with the ground truth not available during the first round of data collection, which allowed us to verify some of the calibration techniques developed for the first round traces. For the edge measurements, we describe the infrastructure of Netalyzr and explain how we collected the Internet edge dataset.

The *second research question* in this thesis is: **How should one perform calibration of the collected enterprise and edge data?** We consider calibration of network traces to be our most prominent contribution. Calibration of enterprise and edge network measurements is the main theme connecting all the work presented in this thesis. Researchers often underestimate the importance of this step, while neglecting it can have detrimental consequences for the whole research project. In this thesis we propose and validate calibration techniques for multiple trace features. We describe the methodology of preprocessing the traces, untangling ambiguities, and eliminating unwanted artifacts. To convince the reader of the importance of this step, we demonstrate the consequences of not performing the calibration procedures.

The *third research question* we investigate in this thesis is: **What analysis can be done for the calibrated enterprise and edge network traces?** The general goal here is to find and characterize interesting phenomena in the monitored networks. For the enterprise dataset we choose to do a preliminary analysis of TCP performance. And for the edge dataset we perform a broad characterization of edge connectivity properties.

The main methodological tool used in our work is exploratory data analysis [143]. Exploratory data analysis is a set of techniques that allow us to gain a broad understanding of the data. Unlike confirmatory data analysis, it usually does not involve explicit hypothesis testing, but is rather used to uncover interesting and relevant trends and phenomena from the data. It is also commonly used to inform possible models that would describe the data and generate hypotheses about various phenomena in it. Exploratory data analysis relies heavily on data visualization techniques and in most cases is an iterative process, where the next iteration is suggested by the outcome of the previous one. Technically, exploratory data analysis consists of going through all or most of the variables in the dataset, studying their distributions and trying to identify relationships between them. This process often uncovers unwanted artifacts or missing data, which has to be accounted for with the calibration process.

Due to the high volumes of data we possess, we relied heavily on scripting and automation of our calibration and data analysis process. We stored the enterprise raw network traces in `PCAP` format. To transform the raw traces to a readable format, we used `tcpdump`[1], `ipsumdump`[2] and

[1] `http://www.tcpdump.org/`

[2] `http://www.read.seas.harvard.edu/~kohler/ipsumdump/`

our custom scripts and tools. For the analysis, we extensively utilized Bro [111] for reconstructing TCP flows and R language [122] for handling, studying and visualizing the data. We stored the Netalyzr data in a custom format, but same as with the enterprise data, we used R language for its analysis.

## 1.2 Contributions

This thesis is a summary of four publications. Here we briefly outline the contributions of each publication. We provide more elaborate summaries in Chapter 3.

Publication I describes the first round of switch-based data collection in the enterprise. Here we engage in calibration of the measurement loss, measurement gain, timing, and the topology aspects of the collected enterprise dataset. Finally, in this paper we perform a minor analysis of the calibrated traces.

Publication II takes a bigger step towards proposing, describing and applying the methodical calibration of network traces. In this paper we propose a methodology for progressive and iterative calibration. We demonstrate the efficiency of the methodology by performing calibration of the switch-based enterprise dataset collected during the second round of data acquisition. In addition, we assess the accuracy of the calibration techniques proposed in Publication I as well as tackle the calibration of a new aspect: measurement reorder.

Publication III presents our study of the high-level characteristics of TCP performance in the enterprise network. In particular, we assess the amount of successful TCP connections, application mix, TCP throughput, and prevalence of retransmissions, out-of-order delivery, and packet corruption in TCP flows.

Publication IV studies the properties of Internet edge connectivity. Here we design a data collection infrastructure capable of simultaneously receiving the measurement data from multiple end-users. Next, we calibrate the recorded dataset and analyze multiple aspects of the Internet edge connectivity.

## 1.3 Structure of the Thesis

The thesis is further structured as follows. In Chapter 2 we lay out the background considerations relevant to our work. Here we also give an extensive overview of the related work performed by other researchers. Further, in Chapter 3 we summarize the results obtained in our publications, thus giving the answers to the research questions stated in Section 1.1. Finally, Chapter 4 concludes the thesis, after which the original publications follow.

# 2. Background and Related Work

This chapter covers the background material of Internet and network measurements in general (Section 2.1), and dives into the specifics of network trace collection, calibration and analysis. We present the background and related work on general calibration techniques used in the network measurements research in Section 2.2. Next, in Section 2.3 we explain the specifics of network measurements in enterprise networks. We move to the last area of our research—edge networks—in Section 2.4. Section 2.5 summarizes this chapter.

## 2.1 Internet and Network Measurements

We start presenting the background work by reviewing the methodological aspects of network measurements. We describe common goals and existing types of Internet measurements, various practical issues, best practices, and supplementary techniques such as sampling and anonymization.

### 2.1.1 Common Goals and Best Practices

Internet measurements can pursue a multitude of goals. Most commonly the goal is to understand the structure, specifics and the dynamics of network topology, traffic, protocol, application or service. A study can focus on one or several facets of the object under investigation: security, performance, privacy, economics, user behavior, etc. The focus can be either on answering a certain question about the system or exploring and documenting its general properties. Ultimately, the results of network measurement studies will help to investigate flaws and improve the performance of the networked systems, protocols, applications and services.

A common way to study a system is to model its behavior. For a system as big and complex as the Internet, it is hard or impossible to build a mathematical model which would fully incorporate all its features and inter-operations [48, 149]. Such models have to focus on a single aspect or component of the Internet, such as the dynamics of a single protocol (e.g. TCP [87, 103]) or application [35, 120, 10, 144]; caching [84, 131]; topology [99, 46]; traffic matrices [45, 25]; queuing [19, 140]; etc. Network measurement studies help to model the Internet and its parts in several ways. First, some of the studies explicitly introduce empirical models, which usually take the form of describing the distribution of a random variable (e.g. flow size or duration) based on observed data [108, 89, 27]. Second, network measurements provide realistic input parameters and serve as "reality checks" for simulations. This is the reason why Floyd et al. [48, 49] promote interdependence and complementarity of simulations and measurements. And finally, data collected in real settings is often used to validate theoretical models.

Even though network measurement studies vary in many ways, it is still possible to discern commonalities and work out the best practices in collecting, calibrating and analyzing network measurement data. Paxson [113] summarizes strategies for executing Internet measurements in a sound way. These strategies are applicable to the majority of network measurement studies. One of the recommendations Paxson gives in his work is to accompany data collection with comprehensive meta-data and annotations, which can help to recreate the details of the measurement process, both by the members of the research team and also other researchers in the event that they get access to the data. In case of long-running measurements, it is beneficial to have an automated monitoring system that does early detection of data collection errors, performs data quality checks and alerts researchers in case of anomalies.

An important requirement for any data measurement study is to ensure that the whole research process is reproducible. It often happens that it is necessary to re-run the analysis from the beginning. The reason for this may be a bug or an error discovered at one of the analysis stages, or a new question which to be answered requires a fairly similar to the original, but still slightly modified analysis. Reproducibility can be ensured by breaking down the whole analysis process into stages and creating scripts for each stage. Ideally, the measurement study should not involve any manual work, firstly because it is very easy to make a mistake when handling data manually, and secondly because it is easy to forget the chain of actions in case the result has to be reproduced. Intermediary data produced by calibration and analysis scripts may be deleted due to lack of storage space, but it is always important to preserve the original data. Then this data in conjunction with all the scripts can be used at any time to reproduce the results of the study.

The majority of network measurement studies rely on either active or passive measurement methodology. Active measurements inject traffic into the network for the purposes of measurement. Most commonly, active measurements are used for checking availability of a service or a protocol [67, 121], path discovery [142, 88] or bandwidth measurement [119, 128]. The tools that allow performance of these tasks include: `ping`, `traceroute`, `NMAP`, `dig`, and other protocol-specific utilities. One of the challenges of active measurements is ensuring that the injected traffic does not bias the results [118]. A contrary approach is passive measurements, where the traffic generated by applications and users is captured at some vantage point. Passive measurements allow less experimental control over the measurements, but do not directly interfere with the measured environment. Passive measurements are used for a broad spectrum of tasks from characterizing traffic properties to latency estimations [7, 106, 64, 44, 75].

The Internet was designed as a packet-switched network, which makes capturing packets—the atomic entities carrying information—the primary method of doing measurement studies. In earlier days when local networks consisted mostly of unmanaged hubs, it was easy to capture all packets floating inside the network since they were broadcasted to every end-host. With the spread of network switches it became necessary to use packet splitters, commonly called network taps or mirroring functionality in switches or routers to capture packets. The growing complexity of networks and increase of available bandwidth boosted the usage of specialized or cleverly utilized hardware [36, 21] and software [57]. Alternatively, the researchers could move to non-packet based measurements. A natural way to aggregate individual packets is to group them into flows—sequences of packets exchanged between two end-hosts according to a higher level protocol such as TCP. Flow-based monitoring provides benefits in simplicity of handling and analyzing measurement data [68, 129].

One of the hurdles of executing measurement studies is legal and ethical barriers. With the goal of protecting online privacy, the laws of many countries prohibit network monitoring and recording the user's private information [133]. Even for the cases where the laws do not restrict network measurements, a certain set of rules shaping the boundaries of ethically acceptable actions should be present. The issues concerning network measurement etiquette are discussed in [6]. Since performing measurements is a very fruitful and much needed undertaking, researchers have developed techniques that help alleviate legal issues. For instance, it is possible to avoid capturing any data that could potentially harm the user's privacy, e.g. by stripping any sensitive information. Another way is to employ data anonymization techniques [150, 105, 105, 93]. The appeal of these techniques is in the fact that they protect privacy without eliminating information that may be useful in analysis. An example anonymization approach is to replace IP addresses that can reveal the user's identity with random identifiers, which nevertheless allow the grouping of packets sent by a user into flows.

### 2.1.2 Measurement Vantage Points

The Internet with its tiered structure (Figure 2.1) has numerous distinctive points for conducting measurements. One way to categorize network measurement studies is by the location of the monitoring apparatus. The typical locations are: Internet backbone, Internet Exchange Point, smaller ISP, enterprise, data center, campus, local area network. Each of these locations comprises a number of network devices where traffic can be captured: edge and internal routers, switches, middleboxes, modems, servers, hosts, hand-held devices, etc. In the following paragraphs we outline specifics and the common goals of measurements in these locations.

The Internet backbone consists of the largest (mostly Tier-1) ISPs, where due to the huge amounts of data traversing the channels, a great wealth of observations can be made in a very short time. Because of their business model, ISPs are very interested in finding optimal routing policies for the network traffic they carry. In general, the main goal of backbone measurements is capacity planning — the act of ensuring that the network has enough bandwidth to satisfy current and future demands. Understanding traffic structure and dynamics is instrumental in successful network provisioning. [16] and [2] are the examples of efforts aimed at understanding the details of traffic dynamics inside large ISPs. Other large entities in the Internet are Internet Exchange Points (IXPs) — places where ISPs can exchange traffic without sending it through an upstream ISP. In [95, 8, 3] the authors analyze the traffic seen at IXPs.

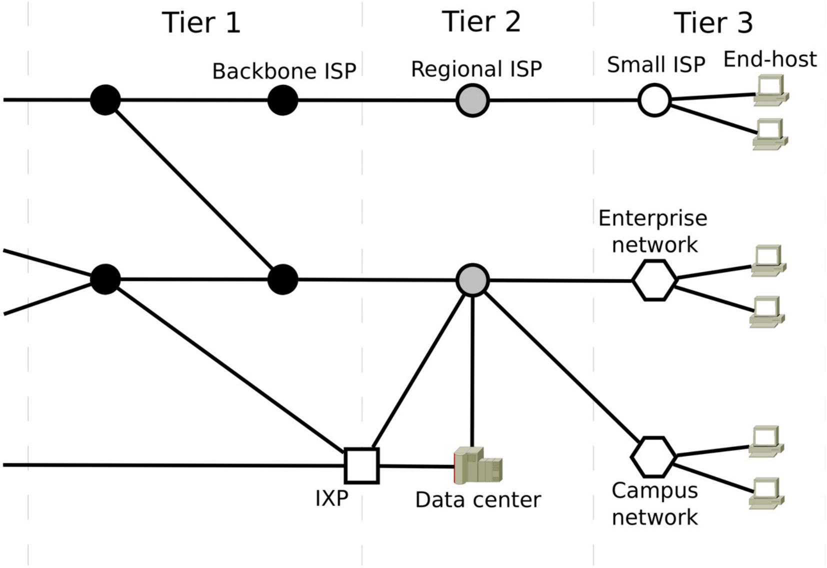


**Figure 2.1.** Tiered Internet structure.

Enterprises, data centers and campuses are similar to ISPs in the sense that they are usually independent Autonomous Systems (ASes), i.e. organizational entities which control a number of IP prefixes and define routing policies within their borders. However, the main difference between ISPs and these institutions, as well as among the institutions themselves, is network usage patterns dictated by their operational goals. For instance, enterprises and campuses often consist of many end-hosts used by employees or students, while data centers comprise many commodity worker machines that do computations and store data. Measurements in enterprise networks is one of the key topics of this thesis, and therefore we elaborate on it later in Section 2.3. In campus networks, which are structurally similar to enterprise networks, the main focus of measurement studies is usually on traffic composition [52, 29] and network configuration [17, 82]. Data centers have recently become a hot topic of networking research due to their importance in large Internet companies' business. The network measurement community has investigated the traffic characteristics of data centers [15, 14] and their problems such as the root causes of most common failures, incast, non-optimal TCP performance, etc [5, 54].

Within the organizational entities listed above, capturing network data may be done at various vantage points. Deploying monitoring infrastructure at gateway and backbone routers in an ISP gives good visibility in terms of containing traffic of a large number of users. Also, due to big volumes of traffic typically traversing these routers per unit of time researchers can collect a lot of data in a short time at this vantage point [24, 63]. An alternative, but closely related vantage point is switches. The main advantage of switch-based traffic capturing is that it allows to observe intra-subnet traffic. We used this vantage point in most of the papers presented in this thesis, and we give its detailed description in Section 3.1. End-hosts are also widely used to collect network data. Even though such vantage point placement limits the ability to observe traffic of the neighboring machines, it allows the probing of different network paths going through distinct ISPs [112, 114]. The topic of such measurements is discussed in detail in Section 2.4.

One important type of Internet measurement studies is those that perform broad-scale measurements of the wide area network (WAN). This type of Internet measurements is challenging, since due to the Internet's size and diversity it is hard to achieve good representativeness of the results. One has to perform measurements at multiple locations or otherwise obtain data which captures the multitude of the Internet's components. A popular way to study WAN and the Internet topology is to leverage the Internet's several distributed and centralized databases, which allow it to function properly. The two most prominent ones are BGP and DNS. The importance of these protocols sparked much research dedicated to understanding and improving them [59, 13]. BGP data is often used for inferring and studying the topology of the Internet [101, 123, 66]. DNS infrastructure has several layers: root servers, secondary servers, ISP resolvers, and end-hosts. Researchers have studied DNS from most of these perspective: with configuration files obtained from root servers [76], at the intermediate level [77], and from the end-host perspective [4, 102, 86].

Researchers often use end-hosts for surveying the WAN or other end-hosts. In [67] the authors performed a census of the visible Internet by sending probes to the whole IPv4 address space. [85] describes the experiences and best practices of implementing an Internet-wide scanner capable of, for example, surveying end-hosts for specific protocol support or OS fingerprinting. Running wide-scale studies or experimenting with a new protocol may require the usage of tens of hosts. Since finding such a big number of machines for experimental and research purposes may prove a troublesome task, there exists an open platform PlanetLab [115], which provides hundreds of hosts scattered around the globe, and is available to researchers upon fulfilling a few simple requirements. Finally, CAIDA hosts a network telescope [98] which constantly monitors unsolicited traffic in the Internet.

## 2.2 Calibration

Internet measurements as virtually any other field of human activity is prone to errors that may arise from a variety of sources. The dominating source is perhaps human error since, although usually being automated, network measurements inevitably require the attention of a human being, be it the task of designing and maintaining measuring apparatus or analyzing and interpreting results. Besides explicit mistakes where the researcher's actions directly lead to erroneous outcomes, e.g. bugs in the code or misjudgments in understanding the data, there are unforeseen factors that may also easily undermine the quality of results. Such factors often arise in Internet measurements due to the nature of the object in question—the Internet nowadays is a huge and extremely diverse system binding together a plethora of devices, operating systems, middleware, network protocols and applications.

Obviously, as any other scientific pursuit, network measurement research calls for objectivity and maximum correctness of the results. The ways of achieving these goals have been already discussed in the community [113]. Additionally, some of the tools and common practices used in other fields can help in building confidence over soundness of results. For instance, following the best practices in software development, data warehousing and the proper use of statistical analysis reduces the chance of error. However, this is not always sufficient since statistical techniques, for example, useful in identifying outliers and providing results that can be mathematically shown to be correct, are nevertheless useless in the face of systematic errors. In fact, the absence of systematic bias in measurements is one of the fundamental requirements of statistical analysis and can be generalized as the assumption of the overall correctness of measurements.

Ensuring the soundness of complex enough measurement studies may require more than just relying on proper tools and techniques. We believe that such studies must make a concentrated effort to search for possible errors and inconsistencies in data collection, processing and analysis. We call this process calibration.

### 2.2.1 Definition and General Workflow

In narrow terms, calibration means simply tuning a measurement apparatus by comparing its recordings with the one considered to reveal ground truth. For instance, calibration in the conventional meaning may include comparing two thermometers, one already known to give accurate readings, and the other being under scrutiny. The readings are made at various temperatures, and if a difference in values is found, it indicates that the latter device has a systematic bias. Such bias usually appears due to imperfectness of the manufacturing process, e.g. varying quality of constituents and their composition. In case systematic bias is identified, the readings from the ground truth thermometer are used to adjust the second one to account for the bias. Applicability of the conventional notion of calibration to Internet measurements is explored by Sommers [136].

In our work we extend the above general definition. We use the word *"calibration"* not simply as a synonym for doing comparison with the ground truth, but as a more general term denoting the efforts of ensuring the correctness of collecting, processing and analyzing network measurement data. We believe that this part of measurement studies is frequently unduly omitted. In the majority of cases, researchers either fail to consider the quality of their dataset or are blindly convinced of its correctness. The illusion of correctness is commonly based on intuition (often wrong) or on ad-hoc and limited probing of the data. In our opinion performing calibration is crucial for the success and soundness of every network measurement study. And even within the tight page limits of many publication venues, the authors ought to least mention that they have engaged in the data calibration procedure in order to bolster the reader's confidence in the results of the paper.

Ideally, calibration efforts should start already at the stage of data collection. In most network measurement studies the data collection period is stretched over many days or even months, and usually produces very large amounts of data. Early and systematic assessment of the quality of incoming data helps in spotting measurement errors as early as possible. In its simplest form, the early assessment may include logging of errors reported by the capturing software, or, for example, in case of packet-based measurements validation of packet losses, by recording the `tcpdump`'s *"dropped by kernel"* messages. More advanced techniques may be study-specific, but some examples of metrics for packet-based measurements include the number of IP and TCP checksum errors, the number of packets per unit of time, and the approximate share of higher level protocols. Discovering errors during capturing may lead to several courses of actions, from adjusting the measurement process to account for the imperfections to terminating and restarting the whole data acquisition routine. In the best scenario, the researchers do not have to alter the capturing process, but, nevertheless, they do become aware of the caveats associated with the data, and so have an opportunity to start planning ways to fix the observed imperfections during the data processing stage.

The majority of calibration actions are performed after the data is collected, i.e. during the data processing phase. The general goal is to have the data prepared for further analysis. Researchers can start by validating their implicit assumptions about the data. For instance, for packet-based network traces such an assumption may be that all the packets have monotonically increasing timestamps, i.e. are recorded in a correct order. The list of the assumptions is usually dictated by the goals of the study. For instance, if the goal is to monitor a number of end-hosts, then it makes sense to verify that the traces have traffic from all of these and no other hosts. Alternatively, if researchers set out to survey a certain IP range, it is useful to check that indeed all the addresses from the range were queried the right number of times.

Even though each study would require a unique calibration procedure, there are several common points that can or should be applied to most datasets. One of the common examples of a calibration process is to convert the data from the raw format into a format more suitable for analysis. In the case of packet traces this can be a conversion from `tcpdump` traces to a human readable format, e.g. with each text line corresponding to a single packet. Another common calibration task is to filter out unwanted data points. For instance, if the goal of the study is to analyze the traffic mix seen in the network, then the researchers have to make sure that the traces do not contain any traffic not belonging to the network in question, such as the unsolicited background radiation traffic received by the network edge device from the Internet. It is also important to assess the consistency of the data. This task may include identifying missing, ambiguous or anomalous values, investigating apparent outliers, checking ranges and distributions of the variables, ensuring that the mutual exclusiveness of certain values is honored or that the fractions add up to a correct total. While checking for anomalous values, one can also make

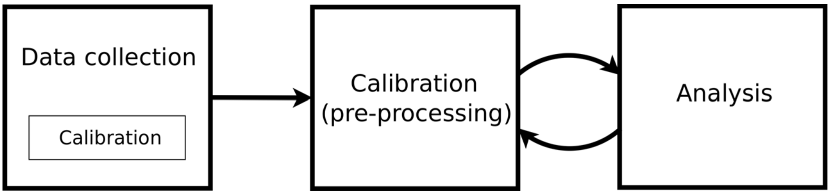


**Figure 2.2.** Typical network measurement study workflow.

sure that the observed values are consistent with the corresponding preconditions. For instance, the source code logic or the network conditions may forbid certain values under certain circumstances, and therefore seeing these values may hint at an inconsistency. During the calibration procedures, it is crucial to document and store all the relevant produced metadata since one may find it very helpful when doing further analysis.

While doing calibration at the preprocessing stage is an important undertaking and can help greatly in confirming data consistency, it is unfortunately impossible to envision all possible difficulties or list all the implicit or explicit assumptions that have to be asserted or refuted. In practice it often happens that flaws in the dataset are discovered during the analysis phase. This fact produces a loop-like research pattern (see Figure 2.2), where upon discovering an error the researcher has to go back to the calibration phase and only after completing it again returns to the analysis, during which another flaw may be found. Iterating this loop is what ensures sound and reliable research results. The calibration and analysis loop may not only work in the form of calibration procedures initiated by discovering data flaws during the analysis. It may also go the other way round, when calibration efforts offer valuable insights into the nature of the data and even guide further analysis. In this light, one may view calibration as an initial step in the analysis. In extreme cases the loop may also contain a return to the earliest step: re-collecting the data.

In general it is hard to draw a solid line between the processes of calibration and analysis. Many researchers consider data quality checks a part of the analysis. Though we argue that a dedicated calibration effort produces far more robust results. It is tempting to think that calibration can be done on the fly, during the analysis. Unfortunately such a mindset often leads to numerous ad hoc patches and hacks. For instance, instead of fixing the data programmatically, one may do the changes manually, which may leave numerous similar data points unfixed. Or the knowledge of the observed flaws may be “swept under the rug”, i.e. fully omitted in the hope that they are not too numerous and will not have much effect on the validity of results. We believe that a similar thinking leads to the failure of many researchers to report the actions dedicated to assuring the quality of their data in the papers. However, our experience demonstrates that even a short mention of such actions significantly reinforces the reader's confidence in the robustness of the reported findings.

We should note that as in a high-quality analysis, thorough calibration is a very domain-specific task. This means that one has to possess considerable expertise in the area in order to successfully pursue a sound calibration effort. For instance, one cannot simply hope that good programming skills or an outstanding knowledge of statistical techniques would allow him or her to perform calibration of the data collected in an unfamiliar domain. Even though such a person would be able to spot simple arithmetical inconsistencies like fractions not adding up to a correct total, not possessing the domain-specific experience would prevent the researcher from identifying deeper flaws, such as incorrect or inconsistent values.

Calibration is a tedious task, and unfortunately no universal algorithm would fit all the cases that exist. If one wishes to do a thorough calibration, then besides following the general guidelines provided above, there is no other way but to engage in a scrupulous verification of the data at all levels. Ideally, one has to double-check the results at every step of data manipulation. It also helps a lot to cross-check the newly produced data with similar data, but generated in a different and preferable independent way. For a thorough calibration one may find it useful to acquire a bit of a paranoid mindset, anticipating a flaw or an inconsistency at each step. Keeping the raw dataset, all the scripts that produce the intermediate and final results, and the notes that describe the thinking behind the whole process is crucial in ensuring reproducibility of the calibration and analysis tasks.

We previously noted that it is beneficial to start calibrating the data already during the data collection phase. However, in some cases it is possible to go even further and perform extensive analysis of possible data collection pitfalls prior to actual measurements. In [51] the authors set out to develop an accurate measurement system for in-context measurements of cellular network performance. Towards this goal, they obtained a dataset of anonymized flows from an operating production network, which was primarily used to identify usage patterns and look for biases capable of skewing analysis results. To corroborate and refine the findings, the authors also collected similar data in a controlled environment. Since the latter featured known conditions, it proved to be a very useful source of ground truth information. The outcomes of the study are the list of observations found in the two datasets and, more importantly, the recommendations that guided the authors' design of a robust and accurate measurement system. Even though one is never safe from unforeseen measurement artifacts, performing a calibration effort even before designing the monitoring system is a big step towards sound network measurements.

### 2.2.2 Packet and Flow Granularity

In this thesis we mostly work with packet traces. Thus, the papers discussing collection and calibration of such traces are most relevant to us. Paxson [109] describes a tool designed for analysis of TCP behavior using packet traces. The work recognizes the need for the thorough calibration of the packet capturing procedure, specifically packet filters. The main difference between the Paxson's study and ours is in the data collection specifics: many of the issues not present in his work arise in our case because we record packets from multiple switch ports simultaneously. Still, the four calibration aspects (drops, additions, resequencing and timing) investigated in the paper are very similar to some of those presented in this thesis. Detecting packet drops is done in a very similar way—by relying on kernel reports and TCP sequencing dynamics. We go further than Paxson [109] in battling additions, or *gain* in our terminology, which is manifested in the appearance of duplicates. In our traces we also observe re-sequencing effects, even though they appear for different reasons. Timing calibration is very important since it directly influences the majority of analysis undertakings, and for this reason we investigate it in detail from several perspectives. In addition, in our work we explored topology calibration, which echoes the vantage point issues considered by Paxson.

A coarser level of granularity in studying network traces can be achieved by collecting flows instead of packets. The main benefit of such approach is that the aggregation of packets into flows produces a smaller amount of data, which simplifies data storage and management. Sommer and Feldmann [135] set out to assess the accuracy of the data collected with NetFlow as compared to corresponding packet traces. This is an example of calibrating a new measurement technique by comparing its outputs with the ground truth. The authors' conclusion is that NetFlow is in general a valuable tool for network measurements.

In a similar spirit, but in a deeper fashion, Cunha et al. [34] investigate the accuracy of an alternative to NetFlow – J-flow. The authors conduct the study in a controlled testbed, which provides the benefits of the ground truth. The study investigates two of the three main flow measurement parameters: inactive timeout (the amount of time a flow should wait for another packet) and active timeout (the maximum time a flow can stay in the router's cache). The third parameter—sampling rate—is not investigated, because the authors set out to check the accuracy of flow collection under the best possible settings; in pursuit of this goal, the sampling mechanism was turned off. In conclusion, the authors identify two major measurement artifacts of J-flow. The first is a timing-related systematic bias, the pronounced artificial periodic pattern and inaccurate flow start times occurring due to the router's coarse inactive and active timeout precision. The second is the measurement gaps created when a task with a higher priority than flow capturing occupies the router's CPU. This demonstrates the drawbacks of using commodity hardware for network measurements as opposed to specialized hardware.

Wallerich et al. [147] set themselves the challenging goal of assessing persistence of flows by examining NetFlow traces. The obvious difficulty is the aggregated nature of flow traces. The assumption of a constant throughput for the whole flow lifetime is unfounded and, therefore, has to be confirmed or refuted in an elaborate experiment. The larger part of work in [147] is dedicated to this calibration effort. The main instrument in asserting the throughput constancy assumption is binning. The authors reconstruct flows from packet traces and split them into bins, with each bin having an associated computed throughput rate. Then the distribution properties of these bins are compared with constant NetFlow bin rates. The general conclusion of the paper is that it is possible to use NetFlow traces for the task defined above. The degree of inaccuracy can be controlled using time and flow aggregation.

### 2.2.3 Topology

One of the calibration facets we engage with in our work is topology related questions. Topology is an important element of network measurement studies since it is one of the determining factors of the network traffic composition and properties. On a global Internet scale, the peering relationships between Autonomous Systems (see Figure 2.3) are of big in-

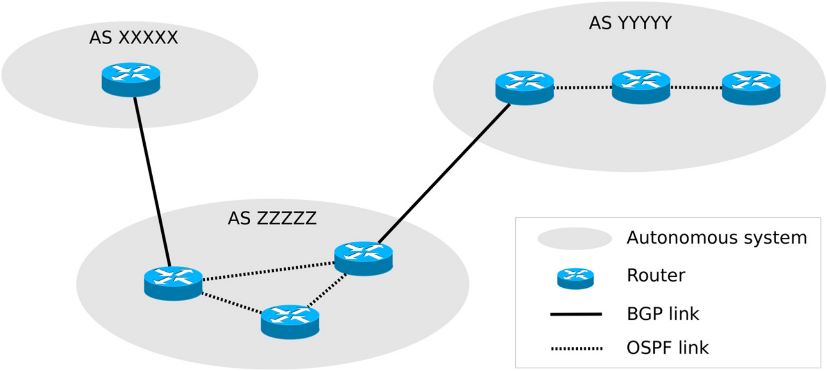


**Figure 2.3.** Links between and within Autonomous Systems.

terest. Commonly, these relationships are studied using BGP data, as reported by several vantage points in the Internet. Chang et al.[26] raise the question of accuracy of such an approach, and more specifically inquires if there are hidden ASs, uninferable from the commonly used BGP data. Such a question resonates with our efforts of identifying hidden topological elements—switches—in the enterprise network. The authors augment the widely used Oregon datasets with several other sources of AS connectivity information to conclude that such a composition allows the revelation of hidden peering relationships. The importance of scrutinizing wide-spread assumptions is the general lesson of the paper, and is also one of the central elements of any calibration undertaking.

The work in [94] also relates to BGP. Here, the authors make use of BGP beacons to study the properties of BGP dynamics such as vendor implementation specifics, route flap damping, BGP updates inter-arrival times, and BGP convergence delay. As in our case, the authors discovered that the raw BGP beacon data was not readily suitable for these types of analysis. Therefore, they ventured into an explicit calibration procedure. The calibration actions the authors go through are: *baselining*, which is similar to duplicate suppression in our work; *signal identification*, which includes grouping semantically similar BGP updates; and *noise filtering* to delete artificial and bogus updates. The paper also demonstrates the importance of the calibration step by signifying the changes of observed signal delay and signal duration distributions before and after the noise cleaning—the difference in some cases reached 50% of the original value of signal delay and duration.

Continuing the AS interconnections topic, and specifically the verification of accuracy and correctness of its measurements, [127] takes a more formal modeling approach for estimating the number of peering links missing from their measurements. The authors adapt the "capture-recapture" technique used in biology to estimate the population of a species. The main difficulties in adapting were the measurement independence and unit homogeneity assumptions, which do not hold for the AS link measurements. Meeting these assumptions, even in their weaker forms, required calibration of the dataset, mainly removing the monitors that violated them. Nevertheless, the paper introduces an accurate technique for assessing measurement loss (e.g. due to sampling) in active measurement studies.

Inferring enterprise subnets from the MAC and IP addresses of end-hosts and routers was one of the subtasks of topology calibration in our work. Gunes and Sarac [61] explore the same task on a broader level, in the wide area network. In addition, unlike our passive approach, the paper makes use of active `traceroute` probes to verify the existence of subnets. Also, before the actual subnet inference task, the authors perform a number of trace calibration actions, such as filtering and combining of path traces, resolving anonymous routers, etc. More in the spirit of our topology inference based on passive monitoring is the work by Eriksson et al. [44]. One of the main tasks discussed in the paper is the reconstruction of missing measurements, which is a common calibration element in many studies.

### 2.2.4 Timing

One of the biggest and most discussed aspects of calibration is timing. The reason is that almost any network measurement incorporates the time component, which makes the quality of such measurements directly dependent on the associated timing fidelity. The seminal work which tackles the problem of timing calibrating was done by Paxson [110]. The paper considers errors in measuring network delay for packets traversing the wide-area network. The core problem is that the delay has to be calculated from the recordings of two different clocks: on the sender and on the receiver. A seemingly simple solution of synchronizing the two clocks unfortunately works only on large time scales (hours and days), whereas packet transit times are far smaller than that and typically reside in the millisecond or even microsecond ranges. After studying the observed timings in their traces, the authors identified two major sources of error: abrupt clock adjustments, when a clock suddenly alters its settings, and

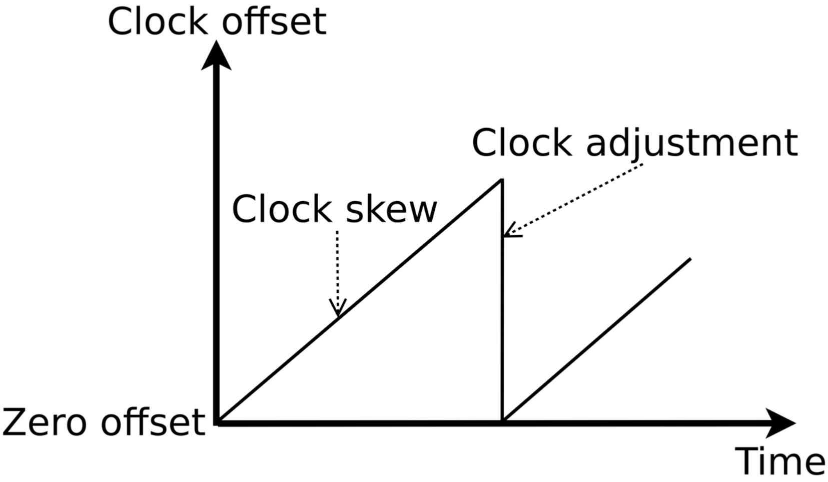


**Figure 2.4.** Relative offset between two clocks and clock synchronization.

relative clock skew, which manifests itself as two clocks running with different rates (see Figure 2.4). The paper presents algorithms to account for both of these artifacts. In our work we go further in studying the timing fidelity of packet traces, first by investigating sub-millisecond timings, and second by observing new timing errors in a different environment.

In addition to Paxson's work [110] there are many other papers that address the problem of calibrating one-way delay and round-trip time measurements in different settings. For instance, Donnelly et al. [42] compare active and passive delay measurements in the wide-area network. Such calibration is an important step in understanding the limitations of deployed active measurement systems. Yin et al. [151] propose using passive monitoring to calibrate active delay measurements on the fly. Kandula an Mahajan [79] demonstrates how to eliminate sampling biases in path latency measurements, and Lee et al. [83] develop methods for latency interpolation, namely reconstructing latency from crude measurements on sub-millisecond intervals.

The two papers that further address the problem of timing quality are those of Pasztor et al. [107], and Veitch et al. [146]. In the former the authors focus on ensuring the accuracy of a single clock. They design a new software clock with a nanosecond resolution, and low and predictable skew. One shortcoming of such a clock is that being based on counting CPU cycles, it still needs to be calibrated by adjusting the duration of each such cycle. The second work provides a solution for robust clock synchronization over the Internet. In addition to proposing novel ways of using NTP routines, the authors examine the influence of the negative effects which arise during practical usage of the system: packet loss, server outages, route changes, temperature and network congestion.

### 2.2.5 Miscellaneous

In general, the use of active probing for collecting network data is as demanding of thorough calibration as almost any other type of network measurement. Unsurprisingly, this fact is even more evident on a big scale, such as considered in [67] where the authors set out to survey the whole Internet. As they rightfully state, any system complex enough to be able to do that is inevitably susceptible to imperfections. Recognizing this, the authors perform an assessment of the sources of errors, the limitations of their own methodology and the causes of inaccuracy. The success of active probing studies is closely related to the choice of the probing protocol, which determines how network elements will respond to the probes they receive. Next, the authors study the possible bias introduced by the choice of measurement location or vantage point. Concerns about multi-homed hosts and routers are not specific to this study since their presence would skew most similar measurements. Finally, the authors investigate the effects of packet loss on their results, and devise remedies for loss events.

Above we already mentioned several examples of loss related calibration. It can include detection of loss, estimating its degree, and reconstructing lost entities. Another example of estimating the amount of loss and accounting for it is given in [148]. The paper studies worm propagation dynamics, and reliable measurements of the number of infected machines can be obtained only after evaluating congestion drops induced by aggressive worm scanning. The authors use an epidemic model of worm propagation, the early stage of which can be modeled using an exponential distribution. Under the assumption that late stage worm propagation deviates from the exponential distribution due to worm-induced congestion loss, the authors are able to reconstruct the scan arrival rate for the late infectees.

Barford and Sommers [12] compare the accuracy of passive and active packet loss measurements in both a controlled laboratory and live network environment. The main outcomes are the high accuracy of passive (SNMP-based) measurements and the limitations of active probing and models of packet loss based on it, especially at low loss rates.

In a narrow sense, calibration of a device or a tool encompasses measuring the deviation of the tool's readings from the known ground truth and adjusting the tool to eliminate the systematic bias. Sommers et al [137] follow this methodology in calibrating two bandwidth estimation tools. The authors recognize the importance of performing tool calibration in a controlled, yet realistic environment. In achieving this goal, they propose designing a flexible laboratory testbed environment, which would allow both robustly affirming the ground truth and performing complex experiments with realistic settings. Careful study of the two existing bandwidth measurement tools allowed the authors to identify the sources of inaccuracy in them. This knowledge enabled the creation of a refined algorithm, which was able to properly take into account the negative effects of cross traffic on bandwidth estimation.

SNMP is widely used for a range of network administration tasks, including link load estimation. Since it is known, however, that SNMP has a set of limitations, the accuracy of such projections is an open question. To fill the gap, [126] performs calibration of the SNMP readings obtained in the Abilene network. The authors find that even though the majority of errors are small, they nevertheless follow a heavy-tailed distribution. The investigation revealed that large errors often could be attributed to data collection problems, such as measurement loss.

An interesting angle on detection of measurement artifacts is presented in [11]. So far we have discussed only the manual means of calibration, i.e. when researchers are actively engaged in the process. Obviously, a disadvantage of manual calibration is that it is too much dependent on human experience, expertise and prudence. An alternative proposed in [11] is to use automated anomaly detection, mostly using wavelet analysis. One type of anomaly the authors managed to identify was measurement related artifacts, i.e. those induced by problems with the data collection infrastructure. Not without its own limitations, this approach nevertheless offers an interesting addition to a common set of manual calibration techniques.

## 2.3 Enterprise Networks

A larger part of the work presented in this thesis is dedicated to the calibration and analysis of enterprise network traces. Conventionally, an enterprise is defined as a company or an organization with for-profit, nonprofit, governmental or educational operations. From our perspective it is important to recognize that an enterprise is an entity under centralized control, unlike the Internet. Daily operations in modern enterprises require the use of a network as a means of connecting the hosts within the enterprise and accessing external resources. The precursors of the current enterprise networks at the dawn of the Internet were small local area networks (LANs), mostly deployed in research institutions. These LANs evolved into complex and often mission-critical networks maintained in numerous enterprises around the globe.

The main motivation for studying enterprise networks is the quest for improving their reliability and performance, as well as reducing the operating cost. The related work we review below approaches these goals from various angles. Some of the papers analyze traffic composition, locality and link load, which upon spotting inefficiencies may result in switching down certain internal services or in altering the topology by adding, moving or upgrading network elements to provide higher performance and eliminate bottlenecks. A related topic is user profiling and traffic pattern classification, which enables modeling of the network evolution and forecasting the future needs of the network. A large body of work is devoted to fast and accurate troubleshooting of the network problems that arise in enterprises—one of the most important tasks of enterprise network operators is to ensure break-free network operation or at least minimize its downtime. Other important topics which we cover here and which can benefit from network measurements are security and energy efficiency in enterprise networks.

### 2.3.1 General Characteristics

Each enterprise network has two main components: internal infrastructure and connection to the WAN (see Figure 2.5). In the simplest cases, the former can be merely a single router providing Internet access to a handful of hosts. A middle-sized enterprise may have hundreds or thousands of hosts and a number of internal servers. For instance, a company can have its email, web and storage servers deployed physically inside its premises. Such an enterprise usually has one or two border routers, several internal routers and numerous switches. A big enterprise network may consist of tens of thousands and, in rare cases, hundreds of thousands of hosts, and it is not uncommon for such an enterprise to be multi-sited with offices scattered over different countries or continents. The company's sites are usually connected to permanent VPN tunnels to

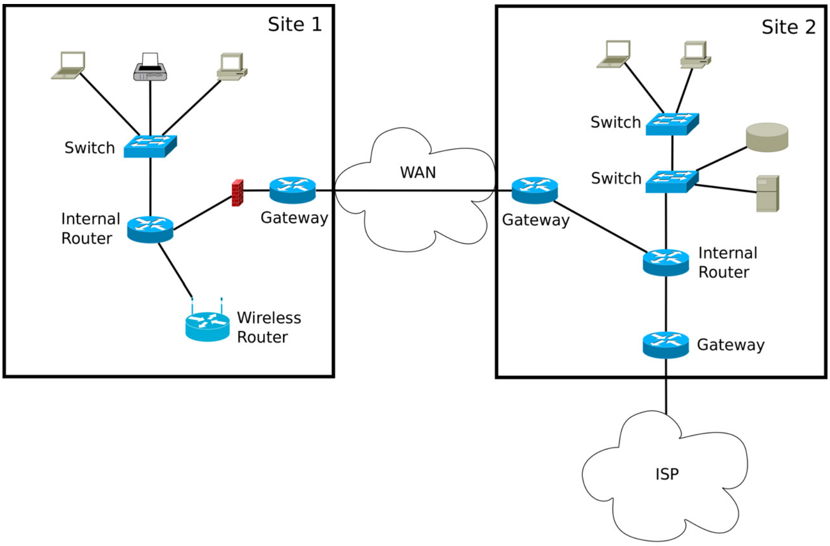


**Figure 2.5.** An enterprise network.

give the employees secure access to all internal resources. Large enterprises are very likely to have a vast infrastructure of internal services, and their internal network architecture tends to be very complex.

Since enterprise networks are characterized by centralized control, it is very common for a large enough enterprise to have a single Autonomous System number assigned to it. It is typical for an Autonomous System to have all routers within its boundaries using a single routing protocol. Nowadays, the most commonly used internal routing protocols in enterprises are IS-IS, OSPF and EIGRP [93]. The border (also called gateway) routers serve the links to other ASes and upstream ISPs and use either a default route in simpler cases or BGP in more complex setups for communication with external networks. It is important to note that the concepts of Autonomous Systems and enterprise networks overlap only partially. First, not all enterprise networks have an associated AS number, which usually happens if they are a part of a bigger network and share the same routing policies, and therefore assigning them an AS number is not justified. Second, the concept of an Autonomous System is more generic — besides enterprises, AS numbers are assigned to ISPs.

As we mentioned earlier, we limit the scope of this thesis to two environments, one of them being enterprise networks. Specifically, this means that we do not consider other types of networks, such as ISPs. The main distinction between the two is in purpose, composition and usage patterns of the network. Unlike an enterprise where the network is typically used to provide connectivity to employees, the core business of an ISP is to sell Internet access to enterprises and private individuals. This distinction leads to a large difference in constituents: ISPs do not have many end-hosts in their network, they are capable of carrying larger amounts of traffic than enterprise networks, and the profile of their internal services has little in common with typical enterprise services.

Other types of modern networks that we do not cover in this thesis are data center networks. Data centers, especially those utilizing commodity hardware, emerged very recently in response to the growing need for large scale storage and computational power. They still continue to develop and in high likelihood did not assume their final shape yet. As with ISPs, the purpose and structure of data centers is very different from those of an enterprise. The current trend in designing data center infrastructure is to use commodity network equipment, which makes cheap switches the main building block of data center networks. Since our work with enterprise traces is focused on data collection at switches, we believe that our findings are applicable to similar tasks in modern data centers.

Finally, we note that there is little difference between campus and enterprise networks. Indeed, our network traces were collected at a research laboratory, Lawrence Berkeley National Laboratory. While it is not exactly a campus network, conceptually it is not far from one. The similarity of enterprise and campus networks is defined by their purpose: providing connectivity to internal users. There may be slight usage differences between the two, but at least structurally they can be viewed as one type of network.

### 2.3.2 Vantage Points

Observing a typical enterprise network structure reveals three main candidates for data collection vantage points: routers, switches and end-hosts. One may pick an internal or gateway router for obtaining a high-level view of the network, with the former capturing a majority of local traffic and the latter observing all the packets leaving and entering the enterprise. The drawback of router-based monitoring is the limited visibility of intra-subnet traffic. Since routers usually mark boundaries of subnets and forward only IP traffic, much of the chatter happening inside a subnet may remain unseen to them. This problem can be addressed by switch-based monitoring—the approach we chose in most of our papers. Traces collected at routers and switches may be both packet and flow based, the latter typically being in NetFlow or Jflow format. The final vantage point in an enterprise is the end-host. These may include desktops and laptops used by employees, and enterprise servers (e.g. web server or email server). The main advantage of data collection at end-hosts is full visibility of the users' network activity. In some cases it is also possible to gather additional information from the host's operating system (application-specific data, system calls, etc) to supplement the network traces. However, there are several drawbacks in end-host data collection. One is that the researcher has to write portable software which will run on various hardware architectures and operation systems. Another is connected to administrative hurdles: it is hard to incentivize a large enough number of people to install data collection software on their machines. This issue is not present in switch and router based monitoring, where one can monitor big portions of the network with little effort.

Ensuring representativeness of the data—one of the main challenges of the whole Internet measurement field—is not alien to enterprise traffic analysis. Obviously, the biggest question is whether the results obtained in a single enterprise would apply to another enterprise. Admittedly, this is far from always being the case, which, however, should not deter the community from studying enterprise networks. In any case, the number of enterprise networks in the world is far too large to try to conceive a holistic study that would embody all of them. And in a situation when obtaining network traces from multiple enterprises is nearly impossible for a single researcher, the best we can do is publish the results characterizing a single network and then "compare notes".

But the problem of representativeness is certainly also present even when collecting traffic from a single enterprise. As we discussed earlier a large enterprise may easily comprise tens of thousands of machines, ensuing a huge variety of services, applications, protocols and usage modes. The question is how to take a trustworthy snapshot of such a big and complex system. Firstly, a successful analysis of an enterprise network requires a systematic and carefully designed effort. Ideally, one should monitor all the existing components of the network. But since this is clearly not possible for networks over a certain size, one has to employ sampled data collection. As is widely known from the theory of experiment design, the samples must be randomized. When applied to our domain, this means that the end-hosts or switch ports we choose to monitor must be selected randomly. This is easier to do with switch and router based studies since in end-host-based studies the population of employees who opt for installing the monitoring software may easily be biased towards tech-savvy personnel. Second, one has to ensure that the data was being collected long enough to capture all diurnal variations in usage and traffic patterns. And finally, one must thoroughly identify and account for biases during the calibration stage of the study.

### 2.3.3 Traffic Characteristics

One of the earliest attempts to characterize LAN traffic was undertaken by Gusella [62]. He used a single UNIX machine to collect traffic in a 10 Mb/s Ethernet campus network of the University of California at Berkeley. The traces spanned three weeks and consisted of packet headers. First, the paper analyzes overall traffic properties, such as network load, distributions of packet length and interarrival time, protocol mix, locality and Ethernet contention. Next, the author makes an analysis of the following specific protocols: TCP, Network Disk Protocol and Network File System Protocol. In its conclusion, the paper offers suggestions on network capacity planning for networks with diskless workstations.

The work by Pang et al. [104] is usually considered seminal for modern enterprise traffic analysis. It is close in spirit to [62]. Pang et al. also give a high-level overview of traffic characteristics and then investigate the properties of specific applications and protocols. In contrast to [62], the network traces in this work were collected from the internal routers of the Lawrence Berkeley National Laboratory. Before engaging in the analysis, the authors had to calibrate the network traces by removing scanning traffic, which constituted 4-18% of connections across datasets and therefore could skew the results. The analysis starts with a broad breakdown of observed applications. Continuing into locality analysis, the authors make an interesting observation that the variety of internal application usage is higher than that of the WAN. Further, they give the detailed characteristics of the traffic of these applications such as the connection success rate, duration, size, and so on for web, email, DNS, NetBios, NFS and backup services. Finally, the paper confirms that the enterprise network is mostly underutilized.

Both in [60] and [55] the authors set out to collect enterprise traces on end-hosts. Data collection software was downloaded and installed by volunteer employees, and the majority of end-hosts in these works were laptops. The focus of the first paper is on enterprise network health as measured by the share of non-useful flows. A non-useful flow is defined as an unsuccessful flow which failed to establish a meaningful communication between the initiator and the responder. In the case of TCP, such flows would not have successfully completed a three-way handshake. For UDP they would not see traffic in both directions. The biggest limitation of such a simplistic definition is that it misses other types of non-useful flows that managed to establish a proper connection, but did not carry any traffic or were terminated before completing a logical application layer operation. As a general result, the authors have identified 34% of non-useful flows. Of the outgoing connection failures, 90% happened within the first four minutes of acquiring a new IP address, which is explained by a user moving from one locale to another (e.g. from a public hotspot to the enterprise network) and the host trying to re-establish old connections. The major outcome of the paper is that environmental awareness can decrease the number of failures in mobile enterprise devices.

The second paper ([55]) further explores the question of host locality. The authors recognize three locales: inside the enterprise, with VPN outside of the enterprise, and without VPN outside of the enterprise. The goal is to derive and compare user profiles in these three environments. The main finding is that the diversity is high both for the single user's behavior in different environments and also the behavior of different users in a single environment. The inability to derive a “typical user” profile leads to complications in creating automated anomaly detection systems that usually recognize anomalies as outliers beyond a certain threshold derived from the normal behavioral profile. The possibility for improvement of such systems lies in personalized environmentally aware anomaly detection.

One of the sub-tasks of enterprise traffic analysis is the analysis of specific protocols operating in enterprise networks. In [145] the authors look at media conferencing traffic in a multi-branched enterprise. The traces they collected consisted of IP video and audio call logs. The main contribution is the study of the impact of various factors on call quality. The authors consider network environment (wired vs. wireless connection, VPN vs. unprotected Internet access, inter- vs. intra-branch communication), QoS services (VLANs and DiffServ DSCP) and branch provisioning policies. In line with the authors' findings, the highest impactor of low call quality is packet loss, which can be mitigated among other things by assigning a separate VLAN ID to the VoIP traffic and prioritizing the traffic with DiffServ DSCP bits.

VPN is commonly used in enterprises as a mechanism for ensuring a secure connection to the enterprise core network from an insecure public environment. The need to efficiently interconnect multiple offices of a single enterprise with VPN led to the invention of multicast VPN (MVPN). It is especially actively used by VPN providers—companies that sell VPN connectivity services to multi-site enterprises. Such services may also be provided by ISP providers. In [81] Karpilovsky et al. collected MVPN traffic at the egress routers of a Tier-1 ISP. The authors study general multicast flow characteristics: rates, durations and throughput, as well as the characteristics that are multicast specific, such as the number of receivers. They describe a peculiar finding that most multicast communication can be replaced with unicast flows without hindering performance, but at the same time decreasing the amount of state in routers.

In the beginning of this section we mentioned the routing protocols commonly used inside ASes and enterprise networks: OSPF, IS-IS and EIGRP. A good understanding of their workings allows for reliable and stable network operation. Shaikh et al. [130] study the dynamics of Link State Advertisement (LSA)—one of the building blocks of OSPF. LSAs are broadcasted between routers to allow each of them to build a consistent global view of the network topology. The focus of the study is on finding a baseline for the frequency of LSA packets and analyzing the anomalies observed to be beyond the inferred baseline. Maltz et al. [93] make a detailed examination of how routing protocols are used in operational networks.

### 2.3.4 Behavior Profiling

Traffic and user behavior profiling in an enterprise is a common goal of network measurement studies [43, 139, 28, 96]. These papers take a formal approach to profiling by creating formal classification models. En-Najjary et al. [43] use a logistic regression classifier based on three features (direction of the packet, PUSH flag, packet size) extracted from the first four packets of the flow. They build classifiers for the 10 ports which account for more than 80% of traffic in their traces. The results demonstrate high accuracy and precision for classifying both encrypted and non-encrypted traffic.

Unlike [43] which classifies flows, [139] creates a model for classifying hosts by their behavior. The main idea here is to utilize the similarity in the sets of hosts with which a machine communicates. The intuitive premise is that hosts which perform similar functions should communicate with about the same set of nodes. Though due to high behavior variation, two runs of the above algorithm may produce different groupings. To overcome this issue, the authors supplement the model with a next step: correlation of grouping results. Two groupings produced with different runs of the grouping step are correlated with each other to yield more robust logical host role classification results.

Both [28] and [96] apply node behavior profiling for the purposes of security. After deriving normal communication patterns using historical data, it becomes possible to detect abnormal activity such as malicious scanning or worm propagation. The features used in such classifications usually include the transport layer 5-tuple as well as the number of packets and bytes sent/received. In practice, the number of false positives can be adjusted by the model parameters. But usually its level is high enough to require human inspection of the triggered security incidents.

### 2.3.5 Troubleshooting

An important task faced by all enterprise network operators is troubleshooting network failures. Often due to the complexity of enterprise networks it may be hard to pin down the root of the problem. To address this issue [9] proposes the Sherlock system—a multi-level fault inference engine. The core idea is to form an Inference Graph of dependencies between all network and IT infrastructure components. Sherlock is user-centric, i.e. it only reports problems that directly affect user perceived performance as opposed to merely reporting diagnostics data. The first step in initializing the Sherlock system is forming the Inference Graph. This is done automatically by monitoring the hosts' activity to determine the services they use and supplementing this with the information about networking elements. When the system detects abnormally high response times for requests, it produces a list of possible failure chains in the Inference Graph ranked by likelihood. The network operators can then easily investigate the localized root cause of the failure.

Kandula et al [80] also rely on the dependency graph of enterprise network components to identify failure culprits. One of the challenges solved in the paper is failure root cause detection with a fine-grained level of granularity. Unlike [9] which operates at host level, [80] considers process and configuration entry level of granularity. Another challenge faced in the paper is to infer the problem cause without much knowledge of application-specific interactions. The latter is achieved by relying on a system that tracks historical interdependencies between components to predict the likelihood of impacting the behavior of other components in the future.

### 2.3.6 Wireless Networks

As the popularity of Wi-Fi networks increased, they began getting installed in enterprises. All the advantages of 802.11 networks: wireless medium, mobility of hosts, ease of deployment—may also be seen as factors complicating their troubleshooting. Cheng et al [30] tackle the problem of enterprise wireless network troubleshooting with the focus on data transfer delays. The authors created models for sources of delays from physical to transport layers, including the MAC layer and mobility mechanisms. These models are used to infer the state of the system at times of abnormal delays when the corresponding system parameters cannot be measured directly. One of the findings in the paper is that in most cases no single phenomenon at a single layer is enough to explain the observed performance degradation, and hence a holistic analysis approach is required to fully grasp the source of the problem.

The two other papers that investigate the topic of 802.11 networks in an enterprise are [31] and [100]. The former introduces Jigsaw—a distributed wireless monitoring system for capturing data from physical, link, network and transport layers. The challenges of trace calibration faced by the authors echo our own. First, the contemporaneous traces collected at multiple locations have to be merged, which raises the question of clock synchronization discussed in Section 2.2.4. The authors use broadcasted packets seen by multiple receivers as reference frames or beacons. Next, similarly to our work the authors eliminate duplicates — multiple instances of a single physical packet. To finalize the trace assembly, Jigsaw performs link and transport layer reconstruction.

Even a medium sized enterprise network may require deployment of a large number of wireless access points. Though this will increase the coverage and the maximum number of concurrent users, it does not directly solve the problem of improving the wireless network capacity. Murty et al [100] propose DenseAP—a centralized system that, given a dense deployment of wireless access points, would improve overall network performance by controlling which access points the clients associate with. The centralized controller is also able to force a client to move to another access point with less load than its current one. In addition, it intelligently controls channel allocation for the pool of access points.

## 2.4 Edge Networks

Alongside data collection, calibration and analysis in enterprise networks, in this thesis we also investigate the application of the same questions in relation to edge networks.

### 2.4.1 General Characteristics

The term *edge network* refers to the outermost components of the Internet infrastructure as seen from the perspective of the Internet core. Two main entities comprising the Internet edge are last mile ISPs and home networks (see Figure 2.6). The edge ISPs are typically small Tier-3 ISPs whose business model consists mainly of buying bandwidth from the upstream ISP and selling access to the Internet to households and small companies.

There is a multitude of network access technologies for providing the last mile access to end users. In the early days, dial-up access [71] with a modem making the call over a public phone line was the most widely used way to connect to the Internet. With the growth of interest in the Internet from the general public, ISPs had to develop technologies providing higher connection speeds than dial-up modems. These technologies got the collective name of *broadband Internet access*. One of the first broadband technologies which gained a lot of popularity in the 90's was ISDN [73], which similarly to dial-up connections relied upon telephone wires. The tradition of using telephone networks for Internet access continued with the invention of DSL [70]. Cable connection [22] provides an alternative to copper telephone wires—it utilizes coaxial cable television infrastructure. One of the last arrivals among broadband technologies was optical fiber connections [72].

In addition to the wired Internet access technologies in the recent years, we have seen the rise of wireless broadband Internet. The families of protocols in 2G, 3G and 4G networks [1] provide Internet connectivity to billions of mobile devices. Several ISPs around the globe have recently started providing households and small businesses with WiMAX [69] broad-

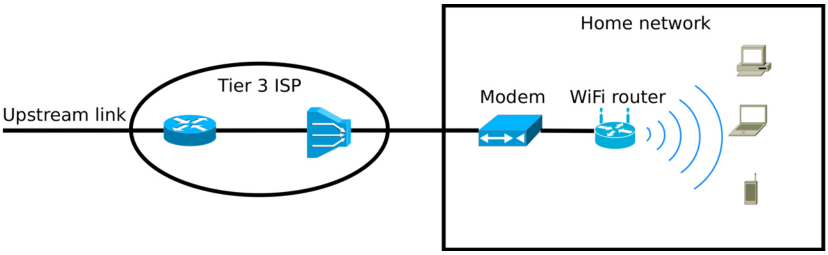


**Figure 2.6.** An edge network.

band Internet access. It is a common consensus that reliance on wireless Internet access technologies will continue to grow in the future.

As we already mentioned, broadband subscribers are mostly private individuals and small companies. Even though there is usually only one link between the Tier 3 ISP and the subscriber, the link can be used by several people. After the link arrives at a *gateway* such as an ADSL or a cable modem, the connection can be shared either using Ethernet or Wi-Fi. The former case requires an Ethernet router or switch, and the latter a WiFi access point; there are also hybrid devices combining both alternatives. Residents of a modern household can connect a variety of devices to the Internet: desktop computer, laptop, tablet, mobile phone, TV, network-attached storage, sensors, etc.

It may seem that in some aspects enterprise and edge network are similar. After all, both receive Internet access via a link to an ISP and distribute the connection among end users. The core difference lies in the scale, which produces multiple differences between the two types of networks. Since enterprise networks are much larger, they require a higher bandwidth link, which limits the choice of access technologies. But most importantly, unlike simple home LANs, enterprise networks have a complex topology. An enterprise may have several internal routers, multiple switches and Wi-Fi access points, various servers, and hundreds or thousands of hosts. Managing all this complexity usually requires a dedicated enterprise IT department. Having relevant expertise, the IT personnel can configure the network and service to operate optimally, while edge subscribers often simply rely on the default settings of their devices. In addition, enterprise networks may have security policies and internal services not commonly found in home networks. And finally, the usage modes in the two types of networks are very different since the first usually involves the work environment, while the latter is, in most cases, used during leisure time.

### 2.4.2 Link Characteristics

Edge networks have attracted much attention from researchers because their performance has a big impact on the quality of service perceived by the end users. But due to the multitude of factors at play, it is not always straightforward to analyze the performance. Settings at the ISP equipment, home gateway, switch or wireless router, end-host operating system can all influence the quality of the Internet connection. The details of application, transport and network layer protocols also play a big role in the perceived Internet connection latency. And of course, the characteristics of the link between the ISP and the subscriber's premises, such as bandwidth and modulation technology, set the upper bound on the connection speed. In the remainder of this section, we review the research work done previously on the topic of edge network performance and characteristics.

With the widespread appearance of broadband links, it became important to understand their characteristics and identify the ways in which they are different from the well-studied backbone, enterprise and campus links. [40] focuses on DSL and cable links, which were and still remain the most dominant broadband technologies. The authors performed active measurements by sending probes from measurement hosts under their control to selected broadband end-hosts. They chose measuring link bandwidth, packet latencies, jitter and loss as the goal of the study. During and after the measurement trace collection phase, the authors performed extensive calibration of their methodology and verified numerous assumptions about the methodology. The findings reveal low packet loss in DSL and cable ISPs, long-term stability of link bandwidths and high last-hop delay in DSL connections. Interestingly, the authors managed to detect traffic shaping by ISPs, which manifested itself in the form of abrupt bandwidth changes.

Unlike the previous study, many researchers approach the problem of analyzing last-mile link characteristics from the user's local network. The vantage point may be at a home gateway, modem or end-host machine, but the general idea is to observe link behavior from the subscriber's point of view. We employ the same approach in our work with Netalyzr. In [138] the authors had access to a deployment of 4000 home gateways. They identified traffic shaping by the ISP, congestion, and the influence of the link technology as the main sources of throughput variability. In regard to packet latency, the most dominant factors included intervening traffic from other applications, buffering and access link quality. [23] studied the same link characteristics: throughput and latency, but relied on crowd-sourced data of a Flash-based openly available connection performance measurement tool. The free nature of the tool allowed collection of the data not only from the US, but from many other countries in the world. [23] confirms most of the results obtained in [138], but also describes several new findings not observed in the previous work. For example, the authors report clusters of low performance links and a correlation between latency and distance from the end-host and the measurement server.

Given the importance of bandwidth as an edge link performance indicator, many researchers and companies have created free and commercially-available speed measurement tools. Goga and Teixeira [56] give an overview of existing bandwidth estimation techniques and compare their performance. The most commonly used approach floods the link with the intention of reaching the maximum available throughput. The researchers have also developed other techniques such as the sending of packet trains and deducing the available bottleneck bandwidth from the dispersion observed at the receiving server. Analysis of the traces collected in a semi-controlled broadband environment and a fully controlled testbed showed that some of the tools can underestimate the bandwidth by more than 60%. The reason for this lies mostly in the inability of some types of home gateways to sustain high packet forwarding rates.

The authors of [18] take an interesting approach to conducting link and ISP characterization measurements from end-hosts. They suggest using bandwidth intensive tools such as peer-to-peer file sharing applications for this purpose. Specifically, they explore the possibility of doing this with an extension to a BitTorrent client. This approach satisfies three requirements: scalability, continuity of monitoring, and the view from the edge.

### 2.4.3 Traffic Properties

Prediction of traffic growth rate is an important task since it allows backbone and local ISPs to plan ahead for the increase in capacity demand, thereby producing a roadmap for upgrading their infrastructure. In [32] the researchers study the trend of residential traffic growth and shed light on the main driving factors behind the pace of growth. Their work focuses on six major Japanese ISPs, together provisioning 42% of Japan's Internet traffic. They compare the traces collected at a three year interval (in 2005 and 2008), with Fiber-To-The-Home as the main last-mile technology. The authors articulate an interesting finding that the traffic growth is fairly moderate and is slower than the ISP capacity growth. Their explanation for this peculiar phenomenon relies on the observation that the dominant peer-to-peer traffic volumes increase slowly, while the video and other rich media traffic have yet to show considerable growth in the following years.

As a consequence of bandwidth usage growth being slower than available capacity growth, one may assume that users do not fully utilize their residential connections. Siekkinen et al. [134] set out to confirm or refute this hypothesis by looking at a 24-hour packet trace of several thousand ADSL subscribers. First, they confirmed the commonly accepted assertion that peer-to-peer file sharing applications are the biggest contributor to the traffic volumes in edge networks. Second, the authors studied link utilization and found that it was very low, with 80% of clients most of the time consuming less than 20% of their downstream bandwidth. Finally, and most interestingly, the paper sheds light on the root causes of low channel utilization. The primary limiting factor turned out to be the peer-to-peer file sharing applications themselves. Many users limit the upload rate in such applications, which leads to slow download rates among their peers.

Maier et al. [90] dive deeper into the structure and characteristics of residential traffic. They had a perspective similar to [32]—they collected traffic at a major European ISP. The traces, though, contained traffic only from one Internet access technology: DSL. Hence, many of the findings reported by the authors have roots in this particular technology. For instance, the paper reports short DSL session durations, with a median of up to 30 minutes, which means that the end-hosts change their IP addresses very frequently. Startlingly, the authors found that the latency of the DSL channel dominates the overall latency of user connections. In addition, corroborating [134], they showed that users rarely fully utilize the available bandwidth. Though being a more recent study, [90] contradicts the findings of earlier works by stating that the share of peer-to-peer traffic in residential connections is small compared to HTTP traffic. A closer look at the composition of HTTP flows reveals heavy usage of video and other media assets as well as direct downloads of archived files from file sharing websites.

Continuing the topic of video traffic, Gill et al. [53] explore YouTube traffic as seen from the edge. The work tries to identify the characteristics of YouTube traffic such as usage patterns, distribution of video popularity, file properties, and transfer modes. In addition, the authors contrast YouTube traffic with other types of traffic and predict the consequences of video traffic growth for edge network and ISP infrastructure. One such consequence is the appropriateness of caching for rich media traffic.

The work in [78] attempts to generalize analysis of edge traffic by introducing a statistical framework (eXpose) capable of automatically learning the communication rules of the protocols and applications operating in the network. The framework does not need guidance in discerning the rules and makes no prior assumptions about the current state of affairs in the network, which makes it suitable for many present and future scenarios. In communication rule extraction, eXpose relies on the simple idea that several flows consistently appearing together within a certain time frame must be logically connected. Applying the framework to the network traces collected at the edge, the authors managed to detect and troubleshoot unexpected protocols and applications, malicious traffic and infected hosts, and also configuration errors. Similarly to eXpose, [117] employs a statistical approach to traffic classification. The authors chose the perspective of an ADSL ISP and collected traces from three ISPs in France. Their main motivation lied in decreasing the share of flows unrecognized by deep packet inspection tools. The findings showed that the developed technique worked well in a single location, but suffered from over-fitting when applied in another network.

Two other works that analyze certain aspects of residential broadband traffic are [91] and [92]. The former studies the malicious activity of end-hosts such as contacting botnet command and control servers, scanning, spamming, and visiting risky websites. It also considers the hosts' security hygiene, which manifests itself in regularly downloading software updates and deploying anti-virus software. The authors make an interesting finding that maintaining good security hygiene does not correlate strongly with the probability of an end user getting infected with malware. Instead, the deciding factor here seems to be the tendency of the user to visit risky websites. Maier et al. [92] use some of the network traces from [91] to study the prevalence of NAT usage among DSL subscribers. The work relies on two methods of NAT detection: observing packet TTL values and the HTTP user-agent header field. The finding that 90% of DSL lines utilize NATs suggests that using IP addresses as user identifiers is problematic.

### 2.4.4 Miscellaneous

Households constitute the majority of Internet access consumers at the edge. A modern home network usually consists of a modem, gateway or router, and multiple devices sharing the connection. Interoperation of these components may lead to unexpected consequences for the perceived quality of service. To study the properties of such interoperation, the authors of [38] created a testbed emulating a typical home network, consisting of a gateway, a phone, a TV and two computers. The experiments included multiple scenarios of combinations of active devices with both Ethernet and Wi-Fi connectivity. The results suggest that performance of the home network and especially competing traffic from several devices has a high impact on end-to-end performance. Another common source of performance degradation in home networks are modems, routers and gateways. [65] experimentally evaluates several dozen widely available home gateway devices in an attempt to categorize their protocol support and implementation details. The authors observed no consistency in the way the devices handle various aspects of the widely used transport and network layer protocols. For instance, few devices honored IETF recommended timeout values, which can lead to performance harming discrepancies in protocol implementations among gateways and operating system networking stacks.

Most of the research projects and commercial tools use a browser as a platform for measuring performance of the edge network. Modern browsers support several technologies that can be used for this purpose: JavaScript, Flash, Java applet, etc. Since all of these have different restrictions, the work in [125] focuses on exploring the feasibility of detailed home network performance analysis with the most promising of the technologies—Java applet. The authors succeeded in obtaining information about the user's machine configuration (operating system and browser versions), DNS settings and performance, wireless connectivity, available bandwidth, and high-level information about other devices present in the home network. Netalyzr used the same approach, but featured many more performance tests implemented using a Java applet.

The work in [37] suggests using another technology—UPnP—to collect information about home devices. UPnP proved to be useful in determining such aspects as link capacity, presence of cross traffic, packet loss and buffer sizes. However, the authors discovered that only a small fraction of home networks have UPnP gateways, and in many of those that do, UPnP exhibited strange and bogus behavior. DiCioccio et al. [39] rely on both Netalyzr and HomeNet Profiler to collect a wide range of measurements about home networks. Unlike most other tools that run in a browser, HomeNet Profiler is packaged as an executable JAR file, thus allowing the researchers to access native operating system settings required for performing specific types of measurements. In particular, the authors focus on discovering the devices and services operating in the home network, and analyzing the characteristics of Wi-Fi connections. Having collected the data from more than 1600 homes in France, the authors observed a high variation in the number of devices per home.

For many years network neutrality remained a hot topic of discussion. At its core, the discourse revolved around the question of whether it is acceptable for the ISPs to treat various traffic types differently. For instance, an ISP may want to allow more bandwidth for some application or even fully block another. The question of network neutrality is most pressing at the network edge since smaller ISPs are the ones most interested in traffic differentiation, while end users in most cases oppose the notion of throttling and, moreover, blocking any applications. In our work we profiled ISPs by the number of blocked connections on Windows, MsSQL and SMTP ports. We have also observed a few cases of interrupted downloads of files with extensions `.exe`, `.mp3` and `.torrent`. Dischinger et al. [41] study network neutrality by focusing solely on BitTorrent blocking. Their measurement methodology is very similar to Netalyzr—they use a Java applet to emulate a BitTorrent flow. The authors identified evidence of BitTorrent blocking for 8.2% of the users who ran the test. Blocking happened mostly in the USA and Singapore and was performed only by few ISPs. Interestingly, in the vast majority of cases, the ISPs blocked only upstream BitTorrent traffic, identifying it by BitTorrent messages, hinting at the use of deep packet inspection for this purpose. In [141] the researchers go even further and instead of focusing on a single application, implement a general discrimination detection system capable of detecting network neutrality violation, even without any a priori knowledge.

## 2.5 Summary

In this chapter we have covered the background material for network measurements in general and our areas of interest in particular. We started by presenting the general aspects of network measurements: common goals, available vantage points, types of measurements, means and tools for collecting network trace data, the requirements and best practices for a sound network measurement study, and finally, legal and ethical issues.

Next, we moved on to the core topic of this thesis, calibration of network traces. We have described how we extend the notion of calibration as compared with the conventional meaning employed in other areas of science and technology. In application to network measurements, we have provided the justification for the necessity of calibrating the collected network traces as early as possible in the course of the research study. Then, we described the main features of the iterative calibration methodology. Finally, we made an extensive overview of the studies fully or partially dedicated to calibration of network traces.

After presenting the general principles of calibration, we moved onto describing enterprise and edge network environments. For enterprise networks, we covered their general characteristics and structure, common monitoring vantage points, traffic and user behavior characteristics, and the ways of troubleshooting these networks. For edge networks, we provided the background and related work on their general characteristics, as well as link and traffic properties.

# 3. Summary of Results

In this chapter we present the published contributions of this thesis and give answers to the research questions stated in Section 1.1. The main contribution is the methodology for conducting sound measurements in enterprise and edge networks. In addition, we describe our efforts in the analysis of TCP performance in an enterprise network and our findings on the properties of edge connectivity. We conclude the chapter with some open questions and suggestions for future work.

## 3.1 Data Collection in the Enterprise and at the Edge

In this section we present the designs of the capturing apparatus we used to collect our datasets in the enterprise and edge environments. We also compare the two enterprise capturing infrastructure designs and present the advantages and disadvantages of each of them.

Before we could start recording network data in the enterprise, we had to choose a vantage point for placing our monitoring hardware. Among several alternatives such as end-hosts, switches, internal routers and border routers, we chose to deploy our measurement apparatus at the switch level. Firstly, this offered us visibility of intra-subnet traffic inaccessible for router-based monitoring, and secondly, the logistics of simultaneously collecting data flowing to and from multiple machines was so much easier compared to deploying monitoring infrastructure at the end-hosts. All in all, we found that switch based monitoring is an economical and efficient way of network trace capturing in an enterprise environment.

We performed the data collection procedure twice at Lawrence Berkeley National Laboratory (LBL)—a medium-sized enterprise with several thousand hosts. We gathered the first round of data from October 2005 through March 2006, and the second round from November 2009 through

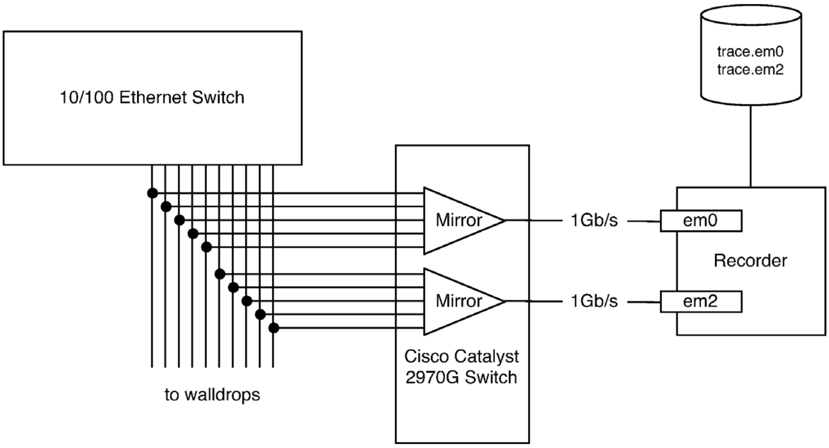


**Figure 3.1.** LBL1 capturing infrastructure.

February 2010. The two sets of traces are similar in that they both capture switch activity, but are different in a number of important details. In what follows we refer to the former and the latter collections of data as LBL1 and LBL2.

In Figure 3.1 we demonstrate the apparatus we used to capture LBL1 traces in Publication I. First, we attached network taps to the Ethernet cables connecting a production switch to the end-hosts. The taps act as a splitter, mirroring all packets to our monitoring switch Cisco Catalyst 2970G. The monitoring switch gathered the incoming mirrored packets into two batches each fed by 5 taps. Two tcpdump processes running on a recording machine with two interfaces (em0 and em2) saved the batches into their respective files on the hard disk. The LBL network operators who assisted us in collecting the data moved the apparatus to a new set of 10 ports every 24 hours. In total the LBL1 dataset consists of 50 pairs of traces, with 869M packets and about 400 GB of payload.

Figure 3.2 shows that the LBL2 capturing infrastructure discussed in Publication II is very similar to that of LBL1. Here we used a pair of monitoring switches to record traffic flowing between a production switch and the end-hosts. The first main difference from the LBL1 setup is that we recorded each traffic direction separately. Thus, we ended up with a pair of uni-directional traces containing the network activity of 10 switch ports, while in LBL1 each trace in a pair contained the bi-directional activity of 5 ports. The second difference in LBL2 was in tagging each monitored port with a unique VLAN tag, unlike LBL1 where we multiplexed all ports. In LBL2 we recorded 1625M packets totaling 957 GB of payload.

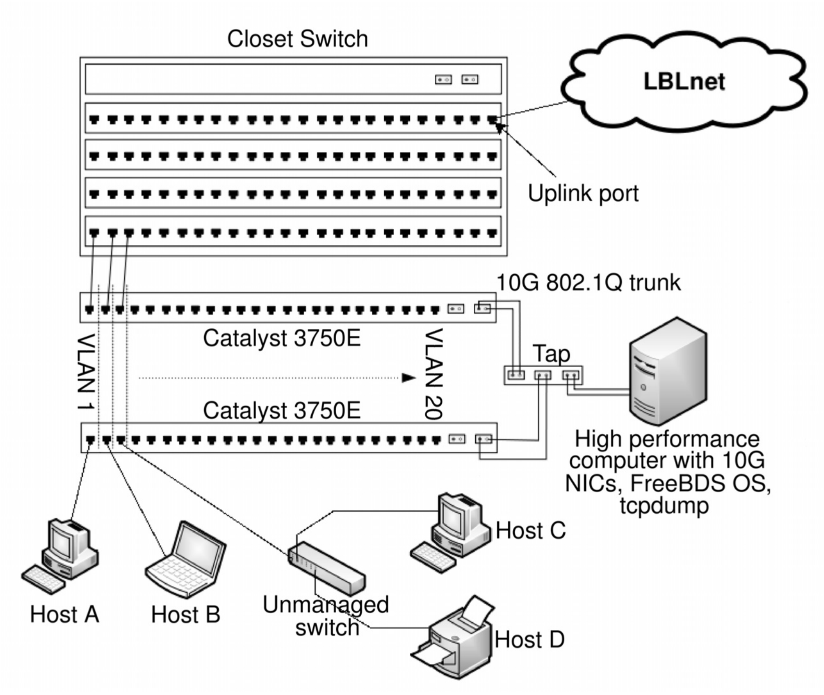


**Figure 3.2.** LBL2 capturing infrastructure.

Each of the two capturing apparatus designs had its own strengths and weaknesses. Multiplexing in LBL1 all monitored ports into one trace file made it hard to distinguish the communication of an individual monitored host. It also produced duplicates of broadcasted packets, which we had to suppress in order to prevent skewing the results of our protocol mix analysis. However, our choice to tag each monitored host with a unique VLAN ID in LBL2 provided us with the ground truth for verifying some of the duplicate suppression and topology calibration techniques developed in Publication I. The uni-directional nature of LBL2 traces also enabled us to easily figure out which of the two communicating hosts we monitor—in LBL1 this was not straightforward, and became possible only after some calibration effort.

However, the same benefit of having uni-directional traces also produced a drawback. In LBL1 we recorded all packets in each flow in the correct order. However, since in LBL2 we captured each flow direction with a separate tap, it became possible for the packets to get reordered. Such reordering was manifested, for instance, in the TCP ACK packets appearing before the respective data sequence packets. This happened because each tap and the associated recording machine network interface had its own queues, and with more traffic in one direction the data packet could spend more time in the queue, thus getting saved later than its ACK packet. We discuss all the above issues in the next section, where we present the results of calibrating both LBL1 and LBL2 datasets and validating the accuracy of the calibration techniques using the ground truth obtained with the LBL2 traces.

Now, turning to data collection in the edge environment, we present the architecture of Netalyzr described in detail in Publication IV. Netalyzr is a network measurement and debugging service run on demand by end-users. The Netalyzr architecture consists of a Java applet which users run in their browser, and a backend service to which the applet connects in order to measure various aspects of Internet connectivity. The backend runs multiple servers used in the tests: an HTTP server, DNS server, TCP and UDP echo server, bandwidth measurement server, path MTU measurement server, etc. To ensure scalability, the number of hosts running the backend service reached 20 during the high load periods. Finally, all the communication happening between the user's browser and the backend machines is recorded for further analysis both in the form of human-readable logs and in `tcpdump` trace files.

## 3.2 Calibration of the Enterprise Dataset

We consider the methodology for collecting and calibrating measurements in enterprise networks to be one of the main contributions of this thesis. Our calibration efforts started in Publication I during our initial attempt to analyze the LBL1 traces. Very soon we realized that the traces contain artifacts which may skew or invalidate the analysis results. For instance, since we aggregated the data from multiple the switch ports, we saw duplicates of a packet appearing very closely to each other in time. Had we proceeded without dealing with the duplicates, their presence could have heavily skewed the observed traffic mix.

Doing calibration for the LBL1 traces has taught us much about the way we should capture and process switch network traces. Building on this experience, in Publication II we present a methodology for *progressive* and *iterative* calibration of enterprise switch traces. Progressive calibration postulates that the whole calibration effort should be split into stages, with each stage reflecting an individual logical calibration procedure. Furthermore, the stages are usually dependent on each other, i.e. successful execution of one stage makes it possible to perform the next stage. As for the iterative nature of calibration—which we have already elaborated on in Section 2.2.1—it implies that the researcher may often need to revisit certain calibration tasks, even after they have been seemingly resolved.

Our experience with the LBL1 dataset led us to perform the first calibration step of LBL2 in the form of initial sanity checks of the traces already during the data capturing process. In addition, we utilized the two described earlier key differences in the LBL2 capturing apparatus to validate many of the calibration techniques developed for LBL1. Also, several problems with the LBL2 traces led us to consider new types of artifacts not present in LBL1. Mostly, the calibration aspects we dealt with were measurement induced, i.e. explicitly or implicitly produced by our measurement infrastructure. In total, while working with the LBL traces, we considered five calibration aspects: *gain*, *loss*, *reordering*, *timing* and *topology*.

### 3.2.1 Measurement Gain

*Measurement gain* happened to be one of the first artifacts that we encountered. By the term *gain* we denote the appearance in the traces of unwanted or erroneous entities such as additional packets. As we described earlier, each LBL1 trace contains data from up to 5 network taps. The production switch to which we attached the taps could, under certain circumstances, broadcast a packet, thus creating in our traces multiple copies of a single packet. The switch could do so for one of the following reasons: the packet was destined for either the Ethernet broadcast or a multicast address, or the switch's forwarding table did not contain a rule for mapping the MAC address in the packet to a proper switch port, and therefore the switch needed to flood the packet to all ports. We call such packet duplicates *phantoms*.

The problem with phantoms is that they can unduly skew the overall traffic mix. Also, if we try to analyze the traces from the perspective of a single end-host, the presence of multiple copies of a packet may be confusing. For instance, the production switch will typically broadcast the first packet in a flow if it has not seen any communication between the two hosts for a certain period of time. In case of TCP this will be the SYN packet, and so the TCP connection reconstruction engine will easily be confused into believing that the connection originator has tried to initiate several TCP connections, when in reality the originator has sent only one

SYN packet. In the end, we would have been left with a wrong count of failed TCP connections, for which only the SYN, but no other handshake packets, were seen. To prevent these and other problems caused by phantoms, we made a decision to suppress packet duplicates.

The most naïve way to suppress the duplicates would be to simply scan the trace packet by packet and discard all copies immediately following the first instance of the packet. In reality, however, it may indeed happen that a host sends several copies of the same packet (e.g. ARP) one after another, thereby eliminating additional true packets using the above method. Thus, we decided to use time intervals between duplicates as the basis for determining if the packet is a phantom. Initially, we tried a 15 usec threshold: we discarded a packet if it was within 15 usec of the previous identical packet. Unfortunately, we discovered that the 15 usec threshold was too narrow with many duplicates remaining unsuppressed. Thus, we faced the task of determining a proper threshold.

The trade-off in the suppression threshold lies in the number of false positives versus false negatives, i.e. packets that get flagged as duplicates, when in truth they represent true network events and duplicate packets that the suppression mechanism fails to discard. In this terminology the 15 usec threshold discussed above turned out to produce too many false negatives. Thus, in order to estimate the proper value for the threshold, we first needed to find packets for which we could be sure that there would be no identical packets truly sent by a host. We called such packets *sole-sourced* and identified them by extracting all Ethernet broadcast packets that had only up to 5 appearances in the trace. Since broadcast packets are always replicated by the switch, this gave us the guarantee that these were indeed copies of a single packet sent only once in the trace. Next, we analyzed time intervals between the sole-sourced packets and found that 99.998% of them lie below 5 msec. To verify that this value is not broad enough to produce multiple false positives, we counted the number of times we observed the same payload more than 5 times across all broadcast packets. We found 150 such cases out of 7.8M unique broadcast packet payloads (0.002%). Thus, we concluded that 5 msec was a threshold with acceptably low false positive and false negative rates.

We used the 5 msec rule to suppress duplicates in the LBL1 traces, where the above sole-sourced heuristics were the only way to determine the rule's accuracy. In LBL2, however, we had each packet tagged with a VLAN ID, which provided a good opportunity for verifying the accuracy of the technique as a whole and of the threshold in particular. To do so, in Publication II we repeated the sole-sourced based false positives estimation only to find that their number jumped to 158K out of 12.1M unique payloads (1.3%). This led us to rethink the suppression algorithm. We proposed and compared three suppression schemes: (i) time-based only used in Publication I; (ii) packet-based, where, in addition to the time threshold, we also made sure that the suppression stops when the number of duplicates reaches the number of connected taps; and (iii) VLAN ID-based, where upon hitting a repeating VLAN ID for the same packet, we concluded that we had encountered another true packet. As expected, both the second and the third algorithms decreased the number of false positives. However, when putting this number into perspective, we found that in about 99% of cases the original time-based only algorithm would suffice.

To once again show the importance of phantom suppression, in Publication I we demonstrated the changes in relative protocol shares before and after calibration. In many cases we found substantial differences, with the largest change in the relative share of a protocol reaching 40%.

### 3.2.2 Measurement Loss

The second calibration aspect we tackled was the opposite of gain: *measurement loss*. In general, measurement loss represents the failure of our measurement apparatus to properly record a packet mirrored from its real copy in the production network. Such loss can take place in a number of places due to a variety of reasons. For instance, a cable tap may fail to replicate a packet; the buffer at the measurement switch can fill up, leading to packet drops; similarly, NIC, PCI bus, kernel or `tcpdump` buffers can become full; finally, even though unlikely, it is possible for the packets to get corrupted due to transmission bit errors. Ideally we should try to distinguish between these types of losses to pin-point the mechanism responsible for the bulk of missing packets. Unfortunately, in practice this is very hard to do. Nevertheless, we should at least try to estimate the amount of measurement loss in our traces. It is important to know the degree of missing data since this builds up confidence in the validity of our analysis results.

We try to detect measurement loss in Publication I and Publication II using five different techniques. The first two methods, leveraging *orphans* and *phantoms*, we apply only to `LBL1`. Since we gathered the `LBL1` traces into contemporaneous pairs, we expect all broadcast packets to appear at about the same time in each trace in a pair. We call the packet which misses its pair an orphan. Counting orphans is a simplistic technique, but nevertheless can provide us with a lower bound for the measurement loss. We have found that about 50% of our trace pairs have no orphans, and that there were only 797 orphans across all traces, which yields a 0.007% measurement loss rate. One of the benefits of measurement gain is in allowing us to use phantoms in loss estimation. To do so, we detected overall stable periods of replication level with a short-lived drop by one duplicate. To elaborate, since multiplexing at the measurement switch produces the number of duplicates equal to the number of active monitored end-hosts, we can plausibly assume that the duplication level would stay stable until one of the monitored machines is turned on or off. Thus, a short-lived deviation, in the form of the replication level decreasing by one, would indicate a measurement loss event. We have found that two-thirds of our traces contain such events, with the total loss rate amounting to 0.08%.

The next two approaches to estimating the measurement loss ratio are dependent on TCP dynamics. The TCP standard specifies that each data sequence has to be acknowledged by the receiver. Therefore, failure to see the acknowledgment indicates either an error in the TCP engine operation or a measurement loss event. Given the low plausibility of the former explanation, we treat such occurrences as the latter. There are two ways to estimate measurement loss using TCP flows: by counting the number of acknowledged, but missing data packets and by counting the number of missing bytes in these packets. We have found that there is a good agreement between the loss rates derived from the number of packets and the amount of bytes. And we have observed that the measurement loss rates are on the same level for `LBL1` and `LBL2`: the total ratio of missing bytes across all trace files tops at 0.1-0.12% and rarely in an individual trace does the loss rate exceed 1%.

The final source of measurement loss information is the `tcpdump` log output. When recording the `LBL2` traces, the LBL network operators saved the log files before moving the measurement apparatus to the next set of ports. We found this useful since `tcpdump` reports the number of packets dropped by kernel during the capturing process. We have found that the `tcpdump` loss rate and the rates derived from TCP acknowledge gaps are close to each other on a per-trace basis.

### 3.2.3 Measurement Reordering

Next, we turn to the *measurement reordering* aspect of calibration. We have first discovered it when trying to quantify the degree of measurement loss in the LBL2 traces. Upon manual inspection of several loss incidents, we found out that in they fact constitute measurement reordering. For instance, we observed TCP data packets coming immediately after their respective acknowledgment packets. As we describe in Publication II, measurement reordering in LBL2 is the consequence of recording each direction of the traffic separately. In this scenario, it is feasible that the two capturing channels would not produce perfectly aligned traces. For example, at one moment, a buffer in one of the channels can contain more packets than the other, which will lead to a delay in timestamping of the packets in that buffer. It is impossible to fix the reordering for the protocols that have no order indicators in packet headers. However, there is such information in TCP packets, and therefore we set out to reconstruct the original packet order by flipping the positions of wrongly ordered data packets with their respective ACKs. We have identified that roughly 6.9M (0.5%) of all packets in LBL2 required flipping. After performing the flipping procedure for these cases, we have found that there still were cases of incorrect packet order, and therefore we had to run the procedure several times until we were left with a handful of marginal cases where weird end-host TCP behavior prevented us from determining the correct ordering of the packets in the TCP flow.

Calibration of measurement reordering can be used to demonstrate what may happen if one does not engage in elaborate calibration of the traces. If we had not performed the procedure described in this section, we would have counted the reordered packets as measurement loss. Indeed, the absence of the correct TCP data packet before its respective ACK would be indicative of a measurement gap. We have observed that after performing the flipping procedure, the apparent amount of measurement loss decreased significantly.

### 3.2.4 Timing Fidelity

One of the most important calibration aspects in many studies is *timing fidelity* as the time component plays a central role in many types of analyses and, therefore, has a direct influence on their accuracy and correctness. In fact, we paid a lot of attention to timing fidelity in Publication

I and Publication II. First, we performed simple consistency checks, such as making sure that the timestamps progress monotonically in our traces. We did this post-hoc for the LBL1 traces, and already in the phase of data collection for LBL2. Next, we assessed the consequences of merging the trace pairs into a single trace file. In the case of LBL1, we had two contemporaneous traces containing different sets of switch ports. For each pair we identified sole-sourced packets and looked at the time difference between the copies of each such packet with the goal of determining how well the two traces in the pair were aligned with each other. We have found as a result that the distribution of the time differences was symmetric, meaning that there was no process that would cause a systematic shift for any of the recording pipelines. In absolute values, 99.8% of time differences lay below 152 usec.

As we discussed earlier, in LBL2 each trace file in a pair contained unidirectional traffic. Therefore, the consequences of merging a pair turned out to be totally different and more problematic than in LBL1. We have already described how we dealt with measurement reordering, but did not yet quantify the degree of timestamp shift that occurred after we had performed the reordering procedure. To summarize the disturbance caused by fixing the measurement reordering: out of 6.9M TCP packets that were flipped, the flipping interval was below one millisecond in 98.9% of cases.

Next, in Publication II we delved deeper into the timing details of TCP connections. We chose to study the fidelity of RTTs in TCP, because TCP provides a natural way to calculate round-trip times. For a TCP handshake consisting of SYN, SYN+ACK and ACK packets, one simply has to measure the time elapsed from observing the SYN packet until recording the ACK packet to get an RTT measurement. The big advantage of this method is its independence of the exact vantage point position: since it captures the full handshake cycle, it yields the same result whether the vantage point is close to the sender or to the receiver. In the case of measuring the RTT of the TCP data and its respective ACK packets, we have to ensure that the vantage point was positioned close to the sender since, otherwise, the measured value would not contain the time it took for both packets to traverse the network.

Upon extracting RTTs for LBL1 TCP handshakes, we were puzzled by the peculiar clustering the RTTs revealed. For both intra-subnet and inter-subnet connections we observed sharp clustering of RTT values around factors of 125 usec. The main question that arises here is whether such clustering is indeed a real network effect, e.g. emerging as a manifestation of an 8 KHz timer or is a consequence of our imperfect measurement process. In trying to answer this question, we have split TCP handshakes into components and study them separately, and have also checked interarrival times between full-sized packets. To our regret we had to conclude that the quantization in LBL1 must have been produced by our capturing apparatus. This conclusion led us to the hypothesis that the apparatus may have imposed the 125 usec timing precision, which unfortunately turned out not to be the case since we observed multiple RTTs with values far from multiples of 125 usec. Fortunately, we have observed no signs of a similar quantization phenomenon in the LBL2 traces.

Our ability in LBL2 to isolate the traffic from a single switch port and the knowledge of what hosts we monitored allowed us to observe individual TCP connections, while at the same time being aware of how close the vantage point was to the connection originator or responder. We used this opportunity to study the peculiar connections for which we monitored both the originator and the responder. We have found 11.5K such flows, and have called them double-monitored connections. The main benefit of such flows lies in the ability to split the RTTs into components. For instance, the time between the moments when a SYN packet is registered at the vantage points close to the sender and then the responder, gives us the network component of the delay. Similarly, the difference between the timestamps of the SYN and the SYN-ACK packets, both observed at the vantage point close to the responder, reflects how much time the end-host spent processing the SYN packet and generating the SYN-ACK packet. To our surprise, we have found network delays to be smaller than end-host delays, suggesting that in a network with small latencies the TCP engine processing contributes the bulk of the RTT.

Given the quantization effects present in LBL1 and the consequences of fixing measurement reordering in LBL2, we had to conclude that in both of our trace sets the timings smaller than 1 msec are unreliable. To obtain high fidelity in the sub-millisecond range, one has to employ specialized equipment and capturing process.

### 3.2.5 Topology

As the final calibration aspect of the enterprise switch traces, we tackled network *topology* or *layout*. Seemingly, in a controlled environment such as LBL, where the network administrators possess in-depth knowledge of the network's structure and operation modes, one would not expect to find much need for calibration. But our experience proved otherwise. Not only in this and other cases is calibration worth doing to strengthen the researcher's confidence in the validity of the results and assert assumptions, but also when dealing with the topology information we were able to discover network elements whose existence came as a surprise to the LBL network operators.

In both Publication I and Publication II, we started topology calibration by identifying the subnet boundaries in the `LBL1` and `LBL2` trace collections. Knowing the subnets is important for other types of analysis. For instance, while assessing timing fidelity, we found it useful to split the traffic into intra- and inter-subnet categories since intra-subnet traffic does not include the effects produced by such complex devices as routers. We found that the easiest way to identify the subnet boundaries was to first single out the MAC and IP addresses of the internal routers. We did so by extracting the MAC addresses of the hosts with IPs external to LBL. Such packets necessarily had to be sent by the LBL routers and would therefore have the router's MAC address. Next, we removed all the traffic involving the routers' IP addresses and for the remaining traffic calculated the smallest accommodating subnet ranges. In most cases the subnets fell into /22 or /23 ranges. To corroborate our findings, we showed them to the LBL network administrators, who confirmed that they are largely correct, barring several cases where our subnets were too narrow due to the inactivity of the hosts in the true broader range.

After determining the subnets we turned to identification of monitored hosts. For `LBL1` this was a non-trivial task, and we had to develop a heuristic technique for this purpose. In short, we built a communication graph with the hosts' MAC addresses in the nodes, and starting from the router's MAC address colored the nodes `green` or `red`, alternating the color with each step. Thus, the nodes in the `red` category belonged to the same class of remote hosts as the router, and the `green` nodes were the hosts we monitored. In `LBL2` the task of extracting the monitored hosts was straightforward: we had to simply take the MAC addresses of the packet senders in the the upstream direction. This gave us the ground truth for assessing the accuracy of the graph coloring technique developed for `LBL1`. In `LBL2` it successfully identified 97% of monitored hosts. The 3% of false negatives were mostly due to those hosts that had no or very little bidirectional communication, thus virtually disconnecting them from the other hosts in the communication graph.

While determining the number of monitored hosts for LBL1 in Publication I, we stumbled upon a puzzling observation that a single trace file seemingly had more monitored MAC addresses than the number of capturing taps. We envisioned several mundane explanations for this effect such as end systems possessing several MAC addresses or different hosts being connected to the port at different times. However, after some consideration we refuted these and other simplistic explanations. For instance, we found it fair to conclude that ubiquitous use of several MAC addresses by the same machine is an infeasible scenario. As for the hypothesis of multiple machines getting connected and disconnected to the port, we split each trace into 15-minute intervals and counted the number of MAC addresses appearing together in each of the intervals. In almost all the traces we have observed at least one interval where all the deduced monitored hosts appeared together.

The only remaining feasible explanation was that some of the monitored switch ports led to a hub or a switch which connected several hosts. In Publication I we confirmed the existence of such hidden switches by detecting ARP requests between pairs of monitored hosts not followed by ARP replies. Such a pattern occurred since ARP requests are issued after a long silence, and, therefore, the hidden switch would not have the requested MAC address in its forwarding table and would therefore broadcast the ARP packet, which we would capture on the tapped link. However, the ARP reply would be sent directly to the requesting host via the hidden switch, and so we would not observe it on the tap. This methodology suggested that roughly 42 out of the 100 LBL1 traces contained hidden switches. In LBL2 we also found more monitored hosts than the number of taps. In this dataset, the presence of VLAN tags allowed us to obtain the ground truth for the number of monitored hosts, but not the presence of hidden switches. Though when applied to LBL2, the methodology that we developed in Publication I yielded 51 out of the 102 traces with a hidden switch, which is in good agreement with the results for LBL1.

All in all, we concluded that the comparison of the enterprise trace calibration techniques developed in Publication I with the ground truth provided by LBL2 in Publication II suggested that the majority of them are adequate and applicable in similar capturing setups.

### 3.3 Calibration of the Edge Dataset

In Publication IV we dealt with a different environment than the one described so far. Here we collected the traces at the edge end-hosts from all over the world. The Netalyzr Java applet in the end-host's browser ran a number of tests between the end host and our servers. Unlike our enterprise traces, we not only stored `tcpdump` files here, but also recorded the measurement session data in a custom format.

Even though the specific calibration tasks in Publication IV differed from those in Publication I and Publication II, they nevertheless served the same goal: to assert or refute the explicit and implicit assumptions about the ways we collected and formatted the data. Also, calibration helped in spotting subtle flaws and inconsistencies in the data capable of partially skewing or completely invalidating our analysis results. The main difficulty with the Netalyzr dataset lay in the broad range of measured parameters and in the vast variety of failure modes able to occur at the edge of the wide area network or en route to our back-end servers.

Our extensive calibration efforts for the Netalyzr dataset included assessing consistency for each of the tests the Java applet ran, which comprised checking the ranges of the measured variables, spotting the impossible values, investigating and explaining the outliers, detecting ambiguous or outright bogus values, ensuring that mutual exclusiveness is honored, and that fractions add up to the correct total. In addition we made sure that the restrictions imposed by the specifics of the Netalyzr test are reflected and honored in the data point values. Finally we investigated the measured parameters for systematic errors. To our relief we did not identify any major flaws, barring minor inconsistencies and ambiguities.

One more calibration aspect we dealt with in Publication IV was measurement biases, which are undesirable due to their ability to skew the results towards a particular user population group. In our data we confirmed five major biases: referral, ISP, OS, browser, and DNS resolver. The users of Netalyzr discovered the service through technical blogs and news aggregation websites, which created the referral bias. This in turn spurred the ISP bias since some of the users chose to run the Netalyzr tests to investigate the specific problems of several of the ISPs discussed in the blog posts. Since the audience of such blogs is mostly technically savvy people, we faced the "geek bias", which manifested itself in the distribution of OS and browser usage different from that of the general population of the Internet. And finally, technical users often choose to use OpenDNS or Google DNS as their resolvers instead of the default ISP's DNS resolvers. There is not much we could do to fix the above biases besides reporting them to the reader. Nevertheless, we utilized the biases for studying some of the reported problems of specific ISPs.

## 3.4 Analysis of the Datasets

Having finished the calibration of our traces, we continued to the analysis stage. As with calibration, our analysis comprises two major parts: the first concerning the enterprise environment and the second focusing on the Internet edge.

### 3.4.1 Enterprise Network Measurements

As the first step of the analysis of the enterprise dataset, we have studied the locality properties of the traffic. Specifically, in Publication I we have divided the traffic of LBL1 into three locality classes: intra-subnet, inter-subnet, and external. Intra-subnet traffic includes the flows that stay strictly inside a broadcast domain. Inter-subnet flows are those that traverse the internal LBL routers, and in detecting such flows we used the knowledge of the subnet boundaries derived during the topology calibration step. And unlike the two previous locality classes, the external traffic had one of the communicating end points outside of the LBL, so identifying such flows was easy for us since we knew the IP ranges allocated to the LBL. In studying traffic locality we calculated the ratio of the flows staying inside a subnet or within the enterprise. In particular, we tried to determine if the traffic locality distribution has a clear and consistent pattern across many subnets. Such knowledge could prove useful for the network engineers in designing the network architecture.

Our findings suggest high variance in the relative shares of the traffic locality classes in different traces. For instance, while in some traces more than 95% of the traffic stays within the subnet, in the others we observe the contrary, with the same percentage of packets flowing to other subnets. Next, we briefly drilled down to the protocol mix in the localities. As expected, the external traffic consisted mostly of TCP. In the inter-subnet communications, UDP slightly prevails with a median share of 58% across the traces. Inside the subnet, the shares of TCP and UDP range from 1% to 99%, thus yielding no clear pattern. Interestingly, we have also observed a significant portion of non-IP traffic in the intra-subnet locality (a median of 29%). The main non-IP protocols were ARP, LLC and IPX.

As the next step in our analysis of the enterprise traces, in Publication III we turn to the study of TCP performance in LBL1. First, we make a high-level overview of the TCP connection termination statuses. We set the goal of finding out how often the connections succeed in being established and terminated properly, and if they fail, then with what status. We defined and derived the statuses of the TCP connections using Bro [111]. For instance, we denote by SF the connections that saw full SYN handshake and FIN termination sequences, and by REJ the connections that had a SYN packet immediately followed by a RST packet from the responder. In total we observed 13 various connection states. The SF connections constituted 58% of all connections in LBL1, but the proportion varied considerably from trace to trace. We found REJ to be another significant connection status: it amounted to 30% of all connections. However, roughly 80% of these rejected connections originated from a single host, which turned out to be a legitimate security scanner deployed by the LBL network operators.

Further, we ventured into characterizing the application level traffic mix, the distribution of size and the locality scope of the TCP connections. The most prevalent applications by the number of bytes and connections in LBL1 were HTTP, NetBIOS, NFS, EPMapper, SSH, Portmap, Dantz backup service, and Warewulf Security Monitor. We have also observed a significant number of connections on non-standard ports that we failed to identify. Interestingly, the distribution of the application prevalence in terms of bytes and number of connections is unbalanced, i.e. typically an application type can be heavy either in number of bytes or number of connections, but not both. We have also found variance in the locality scope: some of the applications predominantly stay inside a subnet, some mostly communicate across subnets, and some do both. To give a high-level connections' size characterization, the median amount of bytes transferred per connection was 2 KB, and the distribution is highly skewed: 15 flows (or 0.004% of all flows) were responsible for transferring 57% of all the bytes in our traces. The percentage rises to 90% if we consider the top 160 flows.

As a final step in analyzing the enterprise traffic, we go through a number of TCP performance characteristics. We found only 583 TCP packets with checksum errors, which most likely originated from isolated hardware malfunctions. We observe reordering based on IP ID sequence numbers in 0.1% of connections, which is nearly negligible compared to the analogous numbers reported for wide-area traffic. 0.5% of the flows contained retransmissions, with the majority experiencing no more than three retransmitted packets per flow. We found the median and the 99th percentile of maximum flight size—the biggest distance between the sender's outstanding sequence number and the acknowledgment point—to be 214 and 5296 bytes respectively. Assuming the average bandwidth-delay product of 12.5 KB in the enterprise network, this finding hints at bandwidth under-utilization. As for the flow rates, 50% of the connections turned out to be faster than 100 KBps, with the transfers inside subnets an order of magnitude faster than the inter-subnet flows.

### 3.4.2 Edge Network Measurements

In Publication IV we cover a wide range of characteristics of the edge host Internet connections performance. The *Netalyzr* tool allowed the users to test the quality of their Internet connection and learn about its various properties by running a Java applet in their browser. In the process of executing the tests, we have collected the data from the measurement sessions. In total we have obtained 130,000 such sessions from 99,000 unique public IP addresses all over the world.

We started by looking at the multiple network-layer properties of the traffic. We see a wide-spread use of NATs, 90% of all sessions indicated their presence. 30% of the NATs use port renumbering, in 90% of cases performing it in a monotonically increasing fashion. IPv6 support is scarce: only 4.8% of hosts were IPv6-enabled. Further, we have spotted problems with IP fragmentation, with 8% of hosts not able to properly send 2000 byte UDP packets. Across all the ISPs, the average download and upload bandwidths were 6.7 Mbps and 2.7 Mbps respectively. We observed the highest download bandwidth in the sessions which, according to the user reports, ran in their place of work, the lowest in public spaces, with home networks in the middle. We also saw the evidence of packet replication in 2% and reordering in 33% of sessions in the download tests.

An interesting topic we pursued with the Netalyzr dataset was that of the network uplink buffering. To infer the buffer capacity, we send a burst of UDP packets with the intent of filling up the buffer. Then the sustained sending rate multiplied by the delay gives us the uplink buffer capacity. Following this path, we observe multiple cases of excessing uplink buffering, with the common buffer sizes equal to a power of two (e.g. 128 KB or 256 KB). Buffers of such size can lead to significant additional latency. For instance, with a 1 Mbps uplink the full buffer will induce a latency over 1 sec, and even with a high-speed uplink connection of 8 Mbps, the latency will be 250 ms.

Next, we turn to services reachability as manifested by blocking, proxying and caching connections. We observe wide-spread blocking of NetBIOS, SMB, RPC and MsSQL ports, all likely due to security measures. Also wide-spread SMTP blocking is usually implemented to prevent spam sending. Apparent FTP blocking is most likely explained by the inability of simple NATs to handle separate FTP data and control connections. HTTP connections revealed proxying in 8.4% of sessions. Many of these connections had an in-path HTTP proxy, and some of them had a proxy explicitly configured in the browser. Caching of an image happened fairly rarely: it manifested itself only in 5.1% of sessions.

We devoted much effort to the DNS measurements, investigating the properties of glue policy, AAAA queries, EDNS, MTU, NXDOMAIN wildcarding, DNS proxies, and reliability of resolving the important DNS names. In 21% of sessions, resolvers accepted glue records from the `Additional` field, and in 25% of session they accepted A records corresponding to CNAMEs. IPv6 queries are clearly on the rise, but their number is still fairly small: we observed them in 13% of sessions. Of the sessions, 52% had EDNS-aware resolvers and 49% had DNSSEC enabled. NXDOMAIN wildcarding was present in 29% of sessions. In only 1.3% of sessions did we deduce an in-path DNS proxy, with many NATs also containing a DNS proxy. We tested DNS reliability by resolving well-known domain names. We found low general DNS reliability, likely due to packet loss, but high reliability for the DNS authorities.

As a final analysis in Publication IV, we do the detailed ISP characterization of the top 20 ISPs in the Netalyzr dataset. For each ISP, we calculated the amount of port blocking per protocol, the presence of different types of DNS wildcarding, and use of PPPoE.

## 3.5 Open Questions and Future Work

Having recapped the main results of our publications, we now discuss the possible directions of future research and the open questions raised by our work.

In Publication I, Publication II and Publication IV, we have explored calibration in enterprise and edge networks. We have limited our efforts to only a few specific ways of collecting the data, and therefore merely scratched the surface. There are numerous other types of networks where one can collect network traces that will no doubt require calibration. Our experience suggests that, at least in part, the required calibration actions will be dictated by the specifics of the network environment and the capturing apparatus. Thus, one of the future research directions is to explore calibration in other types of networks. Even more interestingly, one could pursue the challenging open question of identifying those common calibration modes that are largely independent of the capturing environment and are, therefore, applicable in a multitude of scenarios. If one succeeds in deriving such systematic calibration aspects, then it will become possible to implement automated calibration tools that will contribute to the overall correctness of networking research studies.

Even though in Publication III we have engaged in the analysis of TCP functioning in enterprise networks, we have merely provided a high-level overview. It is clear that after finishing the calibration, we can now use our traces to go much further in characterizing TCP dynamics in the enterprise environment. The goal here would be to uncover unusual details, artifacts and inefficiencies of the TCP protocol. Such findings could potentially lead to adjusting the TCP standard to better suit modern networks. Enterprise networks usually have high throughputs (e.g., in our traces we saw 100 Mbps or 1 Gbps connectivity) which, even with low latencies, would lead to high bandwidth-delay products. It would, therefore, be interesting to determine whether the current TCP implementations succeed in saturating the channels in enterprise networks.

Another important research direction that remained unexplored in our study of enterprise networks is the use of wireless connectivity. Nowadays, many companies install Wi-Fi access points, and the vendors of mobile devices and laptops incorporate Wi-Fi support into their products. All of this makes wireless networking an important factor in enterprise and other environments. The network operators at LBL informed us that during the period when we captured our traces this institution was not providing wireless network connectivity. We are confident, however, that collecting wireless traces would also raise some interesting calibration questions. Moreover, studying such traces would provide a valuable insight into the composition of wireless traffic, its usage modes and performance issues.

Our work in Publication IV covered a wide range of edge connectivity tests. But the list is by no means comprehensive. For instance, we can extend the support for more elaborate testing of IPv6 connectivity, DNS, path MTU and future protocols as they emerge. Another open question is illuminating the internals of home networks. One can also implement similar in spirit automated tests for environments other than the edge.

# 4. Conclusions

Internet and network measurements in the past two decades have emerged as a powerful tool for monitoring, troubleshooting and improving the Internet and its various parts. Researchers and network engineers have performed numerous measurements in multiple locations ranging from end-hosts to enterprise gateways to backbone routers. Doing this has yielded a valuable insight into the details of how the networks operate, illuminated existing inefficiencies and suggested paths for enhancing the infrastructure and the protocols.

Performing sound network measurements and deriving robust results is no easy task. First, the researcher has to carefully design the capturing apparatus. Then, after the data has been collected, it is often necessary to validate its consistency and do sanity checks. Finally, the researcher has to analyze the data and draw significant and novel conclusions.

In this thesis we studied all of the three above steps in two distinct types of networks: in a medium-sized enterprise and at the Internet edge. In the process we pursued three high-level goals: developing a robust way for collecting network traces, proposing a methodology for calibrating them, and, finally, analyzing the calibrated traces.

In the enterprise network we conducted two rounds of data collection from the switches. We structured the way we recorded each round of the network traces in a different way. The differences in the second dataset provided us with the ground truth for validating the efficiency of the calibration efforts we developed while working with the first dataset.

We consider the proposed methodology for calibrating enterprise network traces to be one of the main contributions of this thesis. In total, we considered five calibration aspects: measurement gain, measurement loss, measurement reordering, timing, and topology. Using the second dataset and the ground truth it provided, we demonstrated that, overall, our calibration algorithms performed well, thus proving the usefulness and robustness of our methodology.

Having properly calibrated the datasets, we turned to the analysis of the TCP dynamics in the enterprise environment. First, we provided a high-level characterization of TCP connection statuses by calculating the number of successful, rejected and reset connections. Next, we did an overview of connection characteristics such as size, duration and rate. And as the last step, we studied the performance of TCP flows by calculating the amount of retransmissions, out-of-order deliveries, corrupted packets and channel utilization.

For the second networking environment—the Internet edge—we have developed a connectivity performance measuring tool. With end users running the tool in the browsers on their home and work machines, we have collected measurement sessions from all over the world. For this dataset we have performed calibration in the form of high-level checks of data consistency and sanity. Further, we analyzed the multiple performance characteristics of the edge connectivity.

We believe that the contributions described in this thesis will prove useful to the community. We trust that our demonstration of the importance of calibration in performing sound network measurements will convince researchers to engage in more thorough calibration of their data. We also hope that the foundations laid down in this thesis will mature into the methodology of systematic calibration of network traces in diverse network environments.

# Bibliography

[1] 3RD GENERATION PARTNERSHIP PROJECT (3GPP). About 3GPP. `http://www.3gpp.org/About-3GPP`.

[2] ADHIKARI, V. K., JAIN, S., AND ZHANG, Z.-L. Youtube traffic dynamics and its interplay with a tier-1 isp: an isp perspective. In *Proceedings of the 10th ACM SIGCOMM conference on Internet measurement* (2010), IMC'10, pp. 431–443.

[3] AGER, B., CHATZIS, N., FELDMANN, A., SARRAR, N., UHLIG, S., AND WILLINGER, W. Anatomy of a large european ixp. In *Proceedings of the ACM SIGCOMM 2012 conference on Applications, technologies, architectures, and protocols for computer communication* (2012), SIGCOMM'12, pp. 163–174.

[4] AGER, B., MÜHLBAUER, W., SMARAGDAKIS, G., AND UHLIG, S. Comparing dns resolvers in the wild. In *Proceedings of the 10th ACM SIGCOMM conference on Internet measurement* (2010), IMC'10, pp. 15–21.

[5] ALIZADEH, M., GREENBERG, A., MALTZ, D. A., PADHYE, J., PATEL, P., PRABHAKAR, B., SENGUPTA, S., AND SRIDHARAN, M. Data center tcp (dctcp). In *Proceedings of the ACM SIGCOMM 2010 conference* (2010), SIGCOMM'10, pp. 63–74.

[6] ALLMAN, M., AND PAXSON, V. Issues and etiquette concerning use of shared measurement data. In *Proceedings of the 7th ACM SIGCOMM conference on Internet measurement* (2007), IMC'07, pp. 135–140.

[7] ARLOS, P., FIEDLER, M., AND NILSSON, A. A. A distributed passive measurement infrastructure. In *Proceedings of the 6th international conference on Passive and Active Network Measurement* (2005), PAM'05, pp. 215–227.

[8] AUGUSTIN, B., KRISHNAMURTHY, B., AND WILLINGER, W. Ixps: mapped? In *Proceedings of the 9th ACM SIGCOMM conference on Internet measurement conference* (2009), IMC'09, pp. 336–349.

[9] BAHL, P., CHANDRA, R., GREENBERG, A., KANDULA, S., MALTZ, D. A., AND ZHANG, M. Towards highly reliable enterprise network services via inference of multi-level dependencies. In *Proceedings of the 2007 conference on Applications, technologies, architectures, and protocols for computer communications* (2007), SIGCOMM'07, pp. 13–24.

[10] BARBERA, M., LOMBARDO, A., SCHEMBRA, G., AND TRIBASTONE, M. A Markov model of a freerider in a BitTorrent P2P network. In *Global*

*Telecommunications Conference, 2005. GLOBECOM '05. IEEE* (Dec 2005), vol. 2.

[11] BARFORD, P., KLINE, J., PLONKA, D., AND RON, A. A signal analysis of network traffic anomalies. In *Proceedings of the 2nd ACM SIGCOMM Workshop on Internet measurment* (2002), IMW'02, pp. 71–82.

[12] BARFORD, P., AND SOMMERS, J. Comparing probe- and router-based packet-loss measurement. *IEEE Internet Computing 8*, 5 (Sept. 2004), 50–56.

[13] BEN HOUIDI, Z., MEULLE, M., AND TEIXEIRA, R. Understanding slow bgp routing table transfers. In *Proceedings of the 9th ACM SIGCOMM conference on Internet measurement conference* (2009), IMC'09, pp. 350–355.

[14] BENSON, T., AKELLA, A., AND MALTZ, D. A. Network traffic characteristics of data centers in the wild. In *Proceedings of the 10th ACM SIGCOMM conference on Internet measurement* (2010), IMC'10, pp. 267–280.

[15] BENSON, T., ANAND, A., AKELLA, A., AND ZHANG, M. Understanding data center traffic characteristics. In *Proceedings of the 1st ACM workshop on Research on enterprise networking* (2009), WREN'09, pp. 65–72.

[16] BHATTACHARYYA, S., DIOT, C., AND JETCHEVA, J. Pop-level and access-link-level traffic dynamics in a tier-1 pop. In *Proceedings of the 1st ACM SIGCOMM Workshop on Internet Measurement* (2001), IMW'01, pp. 39–53.

[17] BIN TARIQ, M., MANSY, A., FEAMSTER, N., AND AMMAR, M. Characterizing vlan-induced sharing in a campus network. In *Proceedings of the 9th ACM SIGCOMM conference on Internet measurement conference* (2009), IMC'09, pp. 116–121.

[18] BISCHOF, Z. S., OTTO, J. S., SÁNCHEZ, M. A., RULA, J. P., CHOFFNES, D. R., AND BUSTAMANTE, F. E. Crowdsourcing isp characterization to the network edge. In *Proceedings of the first ACM SIGCOMM workshop on Measurements up the stack* (2011), W-MUST'11, pp. 61–66.

[19] BOHACEK, S., HESPANHA, J. A. P., LEE, J., AND OBRACZKA, K. A hybrid systems modeling framework for fast and accurate simulation of data communication networks. In *Proceedings of the 2003 ACM SIGMETRICS international conference on Measurement and modeling of computer systems* (2003), SIGMETRICS'03, pp. 58–69.

[20] BRADNER, S. Key words for use in RFCs to Indicate Requirement Levels. RFC 2119, March 1997.

[21] BRAUN, L., DIDEBULIDZE, A., KAMMENHUBER, N., AND CARLE, G. Comparing and improving current packet capturing solutions based on commodity hardware. In *Proceedings of the 10th ACM SIGCOMM conference on Internet measurement* (2010), IMC'10, pp. 206–217.

[22] CABLE TELEVISION LABORATORIES, INC. DOCSIS Specifications. `http://www.cablelabs.com/cablemodem/specifications/specifications30.html`.

[23] CANADI, I., BARFORD, P., AND SOMMERS, J. Revisiting broadband performance. In *Proceedings of the 10th ACM SIGCOMM conference on Internet measurement* (2012), IMC'12.

[24] CARTER, K. M., LIPPMANN, R. P., AND BOYER, S. W. Temporally oblivious anomaly detection on large networks using functional peers. In *Proceedings of the 10th ACM SIGCOMM conference on Internet measurement* (2010), IMC'10, pp. 465–471.

[25] CASAS, P., VATON, S., FILLATRE, L., AND CHONAVEL, L. Efficient methods for traffic matrix modeling and on-line estimation in large-scale IP networks. In *Teletraffic Congress, 2009. ITC 21 2009. 21st International* (Sept 2009), pp. 1–8.

[26] CHANG, H., GOVINDAN, R., JAMIN, S., SHENKER, S. J., AND WILLINGER, W. Towards capturing representative as-level internet topologies. *Comput. Netw. 44*, 6 (Apr. 2004).

[27] CHANG, H., JAMIN, S., MAO, Z. M., AND WILLINGER, W. An empirical approach to modeling inter-as traffic matrices. In *Proceedings of the 5th ACM SIGCOMM conference on Internet Measurement* (2005), IMC'05.

[28] CHANG, S., AND DANIELS, T. E. Correlation based node behavior profiling for enterprise network security. In *Proceedings of the 2009 Third International Conference on Emerging Security Information, Systems and Technologies* (2009), SECURWARE'09, pp. 298–305.

[29] CHEN, X., JIN, R., SUH, K., WANG, B., AND WEI, W. Network performance of smart mobile handhelds in a university campus wifi network. In *Proceedings of the 2012 ACM conference on Internet measurement conference* (2012), IMC'12, pp. 315–328.

[30] CHENG, Y.-C., AFANASYEV, M., VERKAIK, P., BENKÖ, P., CHIANG, J., SNOEREN, A. C., SAVAGE, S., AND VOELKER, G. M. Automating cross-layer diagnosis of enterprise wireless networks. In *Proceedings of the 2007 conference on Applications, technologies, architectures, and protocols for computer communications* (2007), SIGCOMM'07, pp. 25–36.

[31] CHENG, Y.-C., BELLARDO, J., BENKÖ, P., SNOEREN, A. C., VOELKER, G. M., AND SAVAGE, S. Jigsaw: solving the puzzle of enterprise 802.11 analysis. In *Proceedings of the 2006 conference on Applications, technologies, architectures, and protocols for computer communications* (2006), SIGCOMM'06, pp. 39–50.

[32] CHO, K., FUKUDA, K., ESAKI, H., AND KATO, A. Observing slow crustal movement in residential user traffic. In *Proceedings of the 2008 ACM CoNEXT Conference* (2008), CoNEXT'08, pp. 12:1–12:12.

[33] CISCO. The Zettabyte Era. `http://www.cisco.com/en/US/solutions/collateral/ns341/ns525/ns537/ns705/ns827/VNI_Hyperconnectivity_WP.pdf`, 2011.

[34] CUNHA, I., SILVEIRA, F., OLIVEIRA, R., TEIXEIRA, R., AND DIOT, C. Uncovering artifacts of flow measurement tools. In *Proceedings of the 10th International Conference on Passive and Active Network Measurement* (2009), PAM'09, pp. 187–196.

[35] De Cicco, L., and Mascolo, S. A Mathematical Model of the Skype VoIP Congestion Control Algorithm. *Automatic Control, IEEE Transactions on 55*, 3 (March 2010), 790 –795.

[36] Degioanni, L., and Varenni, G. Introducing scalability in network measurement: toward 10 gbps with commodity hardware. In *Proceedings of the 4th ACM SIGCOMM conference on Internet measurement* (2004), IMC'04, pp. 233–238.

[37] DiCioccio, L., Teixeira, R., May, M., and Kreibich, C. Probe and pray: using upnp for home network measurements. In *Proceedings of the 13th international conference on Passive and Active Measurement* (2012), PAM'12, pp. 96–105.

[38] DiCioccio, L., Teixeira, R., and Rosenberg, C. Impact of home networks on end-to-end performance: controlled experiments. In *Proceedings of the 2010 ACM SIGCOMM workshop on Home networks* (2010), HomeNets'10, pp. 7–12.

[39] DiCioccio, L., Teixeira, R., and Rosenberg, C. Measuring home networks with homenet profiler. In *Proceedings of Passive and Active Measurement Conference* (2013), PAM'13.

[40] Dischinger, M., Haeberlen, A., Gummadi, K. P., and Saroiu, S. Characterizing residential broadband networks. In *Proceedings of the 7th ACM SIGCOMM conference on Internet measurement* (2007), IMC'07, pp. 43–56.

[41] Dischinger, M., Mislove, A., Haeberlen, A., and Gummadi, K. P. Detecting bittorrent blocking. In *Proceedings of the 8th ACM SIGCOMM conference on Internet measurement* (2008), IMC'08, pp. 3–8.

[42] Donnelly, S., Graham, I., and Wilhelm, R. Passive calibration of an active measurement system. In *Passive and Active Measurements Workshop* (April 2001), PAM'01.

[43] En-Najjary, T., and Urvoy-Keller, G. A first look at traffic classification in enterprise networks. In *Proceedings of the 6th International Wireless Communications and Mobile Computing Conference* (June 2010), IWCMC'10, pp. 764–768.

[44] Eriksson, B., Barford, P., Nowak, R., and Crovella, M. Learning network structure from passive measurements. In *Proceedings of the 7th ACM SIGCOMM conference on Internet measurement* (2007), IMC'07, pp. 209–214.

[45] Erramill, V., Crovella, M., and Taft, N. An independent-connection model for traffic matrices. In *Proceedings of the 6th ACM SIGCOMM conference on Internet measurement* (2006), IMC'06, pp. 251–256.

[46] Faloutsos, M., Faloutsos, P., and Faloutsos, C. On power-law relationships of the Internet topology. In *Proceedings of the conference on Applications, technologies, architectures, and protocols for computer communication* (1999), SIGCOMM'99, pp. 251–262.

[47] FIELDING, R., GETTYS, J., MOGUL, J., FRYSTYK, H., MASINTER, L., LEACH, P., AND BERNERS-LEE, T. Hypertext Transfer Protocol – HTTP/1.1. RFC 2616, June 1999.

[48] FLOYD, S., AND KOHLER, E. Internet research needs better models. *SIGCOMM Comput. Commun. Rev. 33*, 1 (Jan. 2003), 29–34.

[49] FLOYD, S., AND PAXSON, V. Difficulties in simulating the internet. *IEEE/ACM Trans. Netw. 9*, 4 (Aug. 2001), 392–403.

[50] GAUCH, H. G. *Scientific Method in Practice*. Cambridge University Press, 2002.

[51] GEMBER, A., AKELLA, A., PANG, J., VARSHAVSKY, A., AND CACERES, R. Obtaining in-context measurements of cellular network performance. In *Proceedings of the 10th ACM SIGCOMM conference on Internet measurement* (2012), IMC'12.

[52] GEMBER, A., ANAND, A., AND AKELLA, A. A comparative study of handheld and non-handheld traffic in campus wi-fi networks. In *Proceedings of the 12th international conference on Passive and active measurement* (2011), PAM'11, pp. 173–183.

[53] GILL, P., ARLITT, M., LI, Z., AND MAHANTI, A. Youtube traffic characterization: a view from the edge. In *Proceedings of the 7th ACM SIGCOMM conference on Internet measurement* (2007), IMC'07, pp. 15–28.

[54] GILL, P., JAIN, N., AND NAGAPPAN, N. Understanding network failures in data centers: measurement, analysis, and implications. In *Proceedings of the ACM SIGCOMM 2011 conference* (2011), SIGCOMM'11, pp. 350–361.

[55] GIROIRE, F., CHANDRASHEKAR, J., IANNACCONE, G., PAPAGIANNAKI, K., SCHOOLER, E., AND TAFT, N. The Cubicle vs. The Coffee Shop: Behavioral Modes in Enterprise End-Users. In *Proc. PAM* (2008).

[56] GOGA, O., AND TEIXEIRA, R. Speed measurements of residential internet access. In *Proceedings of the 13th international conference on Passive and Active Measurement* (2012), PAM'12, pp. 168–178.

[57] GONZALEZ, J., AND PAXSON, V. pktd: A packet capture and injection daemon. In *Passive and Active Measurement conference* (2003), PAM'03.

[58] GOPINATH, K. N., BHAGWAT, P., AND GOPINATH, K. An empirical analysis of heterogeneity in ieee 802.11 mac protocol implementations and its implications. In *Proceedings of the 1st international workshop on Wireless network testbeds, experimental evaluation & characterization* (2006), WiNTECH'06, pp. 80–87.

[59] GREGORI, E., IMPROTA, A., LENZINI, L., ROSSI, L., AND SANI, L. On the incompleteness of the as-level graph: a novel methodology for bgp route collector placement. In *Proceedings of the 2012 ACM conference on Internet measurement conference* (2012), IMC'12, pp. 253–264.

[60] GUHA, S., CHANDRASHEKAR, J., TAFT, N., AND PAPAGIANNAKI, K. How healthy are today's enterprise networks? In *ACM SIGCOMM/USENIX Internet Measurement Conference* (2008).

[61] GUNES, M. H., AND SARAC, K. Inferring subnets in router-level topology collection studies. In *Proceedings of the 7th ACM SIGCOMM conference on Internet measurement* (2007), IMC'07, pp. 203–208.

[62] GUSELLA, R. A measurement study of diskless workstation traffic on an Ethernet. *IEEE Transactions on Communications 38*, 9 (Sept. 1990).

[63] GYARMATI, L., STANOJEVIC, R., SIRIVIANOS, M., AND LAOUTARIS, N. Sharing the cost of backbone networks: cui bono? In *Proceedings of the 2012 ACM conference on Internet measurement conference* (2012), IMC'12, pp. 509–522.

[64] HALEPOVIC, E., PANG, J., AND SPATSCHECK, O. Can you get me now?: estimating the time-to-first-byte of http transactions with passive measurements. In *Proceedings of the 2012 ACM conference on Internet measurement conference* (2012), IMC'12, pp. 115–122.

[65] HÄTÖNEN, S., NYRHINEN, A., EGGERT, L., STROWES, S., SAROLAHTI, P., AND KOJO, M. An experimental study of home gateway characteristics. In *Proceedings of the 10th ACM SIGCOMM conference on Internet measurement* (2010), IMC'10, pp. 260–266.

[66] HE, Y., FALOUTSOS, M., KRISHNAMURTHY, S. V., AND CHROBAK, M. Obtaining provably legitimate internet topologies. *IEEE/ACM Trans. Netw. 20*, 1 (Feb. 2012), 271–284.

[67] HEIDEMANN, J., PRADKIN, Y., GOVINDAN, R., PAPADOPOULOS, C., BARTLETT, G., AND BANNISTER, J. Census and survey of the visible internet. In *Proceedings of the 8th ACM SIGCOMM conference on Internet measurement* (2008), IMC'08, pp. 169–182.

[68] IANNACCONE, G., DIOT, C., GRAHAM, I., AND MCKEOWN, N. Monitoring very high speed links. In *Proceedings of the 1st ACM SIGCOMM Workshop on Internet Measurement* (2001), IMW'01, pp. 267–271.

[69] IEEE STANDARDS ASSOCIATION. 802.16m-2011 - IEEE Standard for Local and metropolitan area networks Part 16. http://standards.ieee.org/findstds/standard/802.16m-2011.html.

[70] INTERNATIONAL TELECOMMUNICATION UNION. Asymmetric digital subscriber line (ADSL) transceivers. http://www.itu.int/rec/T-REC-G.992.1/en.

[71] INTERNATIONAL TELECOMMUNICATION UNION. Data communication over the telephone network. http://www.itu.int/rec/T-REC-V/en.

[72] INTERNATIONAL TELECOMMUNICATION UNION. Gigabit-capable passive optical networks. http://www.itu.int/rec/T-REC-G.984.1/en.

[73] INTERNATIONAL TELECOMMUNICATION UNION. Integrated services digital network. http://www.itu.int/rec/T-REC-I.

[74] INTERNATIONAL TELECOMMUNICATION UNION. The World in 2011: ITC Facts and Figures. http://www.itu.int/ITU-D/ict/facts/2011/material/ICTFactsFigures2011.pdf, Geneva, 2011.

[75] JOHN, W., TAFVELIN, S., AND OLOVSSON, T. Review: Passive internet measurement: Overview and guidelines based on experiences. *Comput. Commun. 33*, 5 (Mar. 2010), 533–550.

[76] KALAFUT, A. J., GUPTA, M., COLE, C. A., CHEN, L., AND MYERS, N. E. An empirical study of orphan dns servers in the internet. In *Proceedings of the 10th ACM SIGCOMM conference on Internet measurement* (2010), IMC'10, pp. 308–314.

[77] KALAFUT, A. J., SHUE, C. A., AND GUPTA, M. Understanding implications of dns zone provisioning. In *Proceedings of the 8th ACM SIGCOMM conference on Internet measurement* (2008), IMC '08, pp. 211–216.

[78] KANDULA, S., CHANDRA, R., AND KATABI, D. What's going on?: learning communication rules in edge networks. In *Proceedings of the ACM SIGCOMM 2008 conference on Data communication* (2008), SIGCOMM'08, pp. 87–98.

[79] KANDULA, S., AND MAHAJAN, R. Sampling biases in network path measurements and what to do about it. In *Proceedings of the 9th ACM SIGCOMM conference on Internet measurement conference* (2009), IMC'09, pp. 156–169.

[80] KANDULA, S., MAHAJAN, R., VERKAIK, P., AGARWAL, S., PADHYE, J., AND BAHL, P. Detailed diagnosis in enterprise networks. In *Proceedings of the ACM SIGCOMM 2009 conference on Data communication* (2009), SIGCOMM'09, pp. 243–254.

[81] KARPILOVSKY, E., BRESLAU, L., GERBER, A., AND SEN, S. Multicast redux: a first look at enterprise multicast traffic. In *ACM workshop on Research on enterprise networking* (Aug. 2009), WREN'09, pp. 55–64.

[82] KIM, H., BENSON, T., AKELLA, A., AND FEAMSTER, N. The evolution of network configuration: a tale of two campuses. In *Proceedings of the 2011 ACM SIGCOMM conference on Internet measurement conference* (2011), IMC'11, pp. 499–514.

[83] LEE, M., DUFFIELD, N., AND KOMPELLA, R. R. Not all microseconds are equal: fine-grained per-flow measurements with reference latency interpolation. In *Proceedings of the ACM SIGCOMM 2010 conference* (2010), SIGCOMM'10, pp. 27–38.

[84] LEHRIEDER, F., DAN, G., HOSSFELD, T., OECHSNER, S., AND SINGEORZAN, V. Caching for BitTorrent-like P2P systems: a simple fluid model and its implications. *IEEE/ACM Trans. Netw. 20*, 4 (Aug. 2012), 1176–1189.

[85] LEONARD, D., AND LOGUINOV, D. Demystifying service discovery: implementing an internet-wide scanner. In *Proceedings of the 10th ACM SIGCOMM conference on Internet measurement* (2010), IMC'10, pp. 109–122.

[86] LIANG, J., JIANG, J., DUAN, H., LI, K., AND WU, J. Measuring query latency of top level dns servers. In *Proceedings of the 14th international conference on Passive and Active Measurement* (2013), PAM'13, pp. 145–154.

[87] LOW, S. H. A duality model of TCP and queue management algorithms. *IEEE/ACM Trans. Netw. 11*, 4 (Aug. 2003), 525–536.

[88] LUCKIE, M., AND STASIEWICZ, B. Measuring path mtu discovery behaviour. In *Proceedings of the 10th ACM SIGCOMM conference on Internet measurement* (2010), IMC'10, pp. 102–108.

[89] MAH, B. An empirical model of http network traffic. In *INFOCOM '97. Sixteenth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings IEEE* (apr 1997), vol. 2, pp. 592–600.

[90] MAIER, G., FELDMANN, A., PAXSON, V., AND ALLMAN, M. On dominant characteristics of residential broadband internet traffic. In *Proceedings of the 9th ACM SIGCOMM conference on Internet measurement conference* (2009), IMC'09, pp. 90–102.

[91] MAIER, G., FELDMANN, A., PAXSON, V., SOMMER, R., AND VALLENTIN, M. An assessment of overt malicious activity manifest in residential networks. In *Proceedings of the 8th international conference on Detection of intrusions and malware, and vulnerability assessment* (2011), DIMVA'11, pp. 144–163.

[92] MAIER, G., SCHNEIDER, F., AND FELDMANN, A. Nat usage in residential broadband networks. In *Proceedings of the 12th international conference on Passive and active measurement* (2011), PAM'11, pp. 32–41.

[93] MALTZ, D. A., ZHAN, J., XIE, G., ZHANG, H., HJÁLMTÝSSON, G., GREENBERG, A., AND REXFORD, J. Structure preserving anonymization of router configuration data. In *Proceedings of the 4th ACM SIGCOMM conference on Internet measurement* (2004), IMC'04, pp. 239–244.

[94] MAO, Z. M., BUSH, R., GRIFFIN, T. G., AND ROUGHAN, M. Bgp beacons. In *Proceedings of the 3rd ACM SIGCOMM conference on Internet measurement* (2003), IMC'03, pp. 1–14.

[95] MCCREARY, S., AND CLAFFY, K. Trends in wide area ip traffic patterns - a view from ames internet exchange. In *ITC Specialist Seminar* (Monterey, CA, Sep 2000).

[96] MCDANIEL, P., MERWE, J. V. D., SEN, S., AIELLO, B., SPATSCHECK, O., AND KALMANEK, C. Enterprise security: a community of interest based approach. In *In Proc. NDSS* (2006).

[97] MEDEIROS, J. A. P. S., DE MEDEIROS BRITO JÚNIOR, A., AND PIRES, P. S. M. A qualitative survey of active TCP/IP fingerprinting tools and techniques for operating systems identification. In *Proceedings of the 4th international conference on Computational intelligence in security for information systems* (2011), CISIS'11, pp. 68–75.

[98] MOORE, D., SHANNON, C., VOELKER, G., AND SAVAGE, S. Network Telescopes: Technical Report. Tech. rep., Cooperative Association for Internet Data Analysis (CAIDA), Jul 2004.

[99] MÜHLBAUER, W., FELDMANN, A., MAENNEL, O., ROUGHAN, M., AND UHLIG, S. Building an AS-topology model that captures route diversity. In

*Proceedings of the 2006 conference on Applications, technologies, architectures, and protocols for computer communications* (2006), SIGCOMM'06, pp. 195–206.

[100] MURTY, R., PADHYE, J., CHANDRA, R., WOLMAN, A., AND ZILL, B. Designing high performance enterprise wi-fi networks. In *Proceedings of the 5th USENIX Symposium on Networked Systems Design and Implementation* (2008), NSDI'08, pp. 73–88.

[101] OLIVEIRA, R., PEI, D., WILLINGER, W., ZHANG, B., AND ZHANG, L. The (in)completeness of the observed internet as-level structure. *IEEE/ACM Trans. Netw. 18*, 1 (Feb. 2010), 109–122.

[102] OTTO, J. S., SÁNCHEZ, M. A., RULA, J. P., AND BUSTAMANTE, F. E. Content delivery and the natural evolution of dns: remote dns trends, performance issues and alternative solutions. In *Proceedings of the 2012 ACM conference on Internet measurement conference* (2012), IMC'12, pp. 523–536.

[103] PADHYE, J., FIROIU, V., TOWSLEY, D., AND KUROSE, J. Modeling TCP throughput: a simple model and its empirical validation. In *Proceedings of the ACM SIGCOMM '98 conference on Applications, technologies, architectures, and protocols for computer communication* (1998), SIGCOMM'98, pp. 303–314.

[104] PANG, R., ALLMAN, M., BENNETT, M., LEE, J., PAXSON, V., AND TIERNEY, B. A first look at modern enterprise traffic. In *ACM SIGCOMM/USENIX Internet Measurement Conference* (2005).

[105] PANG, R., AND PAXSON, V. A high-level programming environment for packet trace anonymization and transformation. In *Proceedings of the 2003 conference on Applications, technologies, architectures, and protocols for computer communications* (2003), SIGCOMM'03, pp. 339–351.

[106] PAPAGEORGE, P., MCCANN, J., AND HICKS, M. Passive aggressive measurement with mgrp. In *Proceedings of the ACM SIGCOMM 2009 conference on Data communication* (2009), SIGCOMM'09, pp. 279–290.

[107] PÁSZTOR, A., AND VEITCH, D. Pc based precision timing without gps. In *Proceedings of the 2002 ACM SIGMETRICS international conference on Measurement and modeling of computer systems* (2002), SIGMETRICS'02, pp. 1–10.

[108] PAXSON, V. Empirically derived analytic models of wide-area tcp connections. *IEEE/ACM Trans. Netw. 2*, 4 (Aug. 1994), 316–336.

[109] PAXSON, V. Automated packet trace analysis of TCP implementations. In *Proc. SIGCOMM* (1997).

[110] PAXSON, V. On calibrating measurements of packet transit times. In *Proc. SIGMETRICS* (June 1998).

[111] PAXSON, V. Bro: a System for Detecting Network Intruders in Real-Time. *Computer Networks 31*, 23-24 (1999), 2435–2463.

[112] PAXSON, V. End-to-end internet packet dynamics. *IEEE/ACM Trans. Netw. 7*, 3 (June 1999), 277–292.

[113] PAXSON, V. Strategies for sound internet measurement. In *Proceedings of the 4th ACM SIGCOMM conference on Internet measurement* (2004), IMC'04, pp. 263–271.

[114] PAXSON, V., MAHDAVI, J., ADAMS, A., AND MATHIS, M. An architecture for large scale internet measurement. *Communications Magazine, IEEE 36*, 8 (aug 1998), 48–54.

[115] PETERSON, L., BAVIER, A., FIUCZYNSKI, M. E., AND MUIR, S. Experiences building planetlab. In *Proceedings of the 7th symposium on Operating systems design and implementation* (2006), OSDI'06, pp. 351–366.

[116] PETERSON, L., MUIR, S., ROSCOE, T., AND KLINGAMAN, A. PlanetLab Architecture: An Overview. Tech. Rep. PDN–06–031, PlanetLab Consortium, May 2006.

[117] PIETRZYK, M., COSTEUX, J.-L., URVOY-KELLER, G., AND EN-NAJJARY, T. Challenging statistical classification for operational usage: the adsl case. In *Proceedings of the 9th ACM SIGCOMM conference on Internet measurement conference* (2009), IMC'09, pp. 122–135.

[118] PORTOLES-COMERAS, M., CABELLOS-APARICIO, A., MANGUES-BAFALLUY, J., BANCHS, A., AND DOMINGO-PASCUAL, J. Impact of transient csma/ca access delays on active bandwidth measurements. In *Proceedings of the 9th ACM SIGCOMM conference on Internet measurement conference* (2009), IMC'09, pp. 397–409.

[119] PRASAD, R., DOVROLIS, C., MURRAY, M., AND CLAFFY, K. Bandwidth estimation: metrics, measurement techniques, and tools. *Netwrk. Mag. of Global Internetwkg. 17*, 6 (Nov. 2003), 27–35.

[120] QIU, D., AND SRIKANT, R. Modeling and performance analysis of BitTorrent-like peer-to-peer networks. In *Proceedings of the 2004 conference on Applications, technologies, architectures, and protocols for computer communications* (2004), SIGCOMM'04, pp. 367–378.

[121] QUAN, L., HEIDEMANN, J., AND PRADKIN, Y. Trinocular: Understanding internet reliability through adaptive probing. In *Proceedings of the ACM SIGCOMM Conference* (August 2013), p. to appear.

[122] R CORE TEAM. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2012. ISBN 3-900051-07-0.

[123] RASTI, A. H., MAGHAREI, N., REJAIE, R., AND WILLINGER, W. Eyeball ases: from geography to connectivity. In *Proceedings of the 10th ACM SIGCOMM conference on Internet measurement* (2010), IMC'10, pp. 192–198.

[124] RAYCHAUDHURI, D., SESKAR, I., OTT, M., GANU, S., RAMACHANDRAN, K., KREMO, H., SIRACUSA, R., LIU, H., AND SINGH, M. Overview of the ORBIT radio grid testbed for evaluation of next-generation wireless network protocols. In *Wireless Communications and Networking Conference, 2005 IEEE* (March 2005), vol. 3, pp. 1664–1669.

[125] RITACCO, A., WILLS, C., AND CLAYPOOL, M. How's my network? a java approach to home network measurement. In *Computer Communications and Networks, 2009. ICCCN 2009. Proceedings of 18th Internatonal Conference on* (aug. 2009), pp. 1 –7.

[126] ROUGHAN, M. A Case Study of the Accuracy of SNMP Measurements. In *Journal of Electrical and Computer Engineering* (2010), vol. 2010.

[127] ROUGHAN, M., TUKE, S. J., AND MAENNEL, O. Bigfoot, sasquatch, the yeti and other missing links: what we don't know about the as graph. In *Proceedings of the 8th ACM SIGCOMM conference on Internet measurement* (2008), IMC'08, pp. 325–330.

[128] SEDIGHIZAD, M., SEYFE, B., AND NAVAIE, K. Mr-bart: Multi-rate available bandwidth estimation in real-time. *J. Netw. Comput. Appl. 35*, 2 (Mar. 2012), 731–742.

[129] SEKAR, V., REITER, M. K., AND ZHANG, H. Revisiting the case for a minimalist approach for network flow monitoring. In *Proceedings of the 10th ACM SIGCOMM conference on Internet measurement* (2010), IMC'10, pp. 328–341.

[130] SHAIKH, A., ISETT, C., GREENBERG, A., ROUGHAN, M., AND GOTTLIEB, J. A case study of ospf behavior in a large enterprise network. In *Proceedings of the 2nd ACM SIGCOMM Workshop on Internet measurment* (2002), IMW'02, pp. 217–230.

[131] SHI, L., AND ZHANG, Y. Optimal Model of Web Caching. In *Proceedings of the 2008 Fourth International Conference on Natural Computation - Volume 07* (2008), ICNC'08, pp. 362–366.

[132] SHU, G., AND LEE, D. Network Protocol System Fingerprinting - A Formal Approach. In *INFOCOM 2006. 25th IEEE International Conference on Computer Communications. Proceedings* (April 2006), pp. 1–12.

[133] SICKER, D. C., OHM, P., AND GRUNWALD, D. Legal issues surrounding monitoring during network research. In *Proceedings of the 7th ACM SIGCOMM conference on Internet measurement* (2007), IMC'07, pp. 141–148.

[134] SIEKKINEN, M., COLLANGE, D., URVOY-KELLER, G., AND BIERSACK, E. W. Performance limitations of adsl users: a case study. In *Proceedings of the 8th international conference on Passive and active network measurement* (2007), PAM'07, pp. 145–154.

[135] SOMMER, R., AND FELDMANN, A. Netflow: information loss or win? In *Proceedings of the 2nd ACM SIGCOMM Workshop on Internet measurment* (2002), IMW'02, pp. 173–174.

[136] SOMMERS, J. *Calibrated Network Measurement*. PhD thesis, University of Wisconsin-Madison, 2007.

[137] SOMMERS, J., BARFORD, P., AND WILLINGER, W. Laboratory-based calibration of available bandwidth estimation tools. In *Elsevier Microprocessors and Microsystems Journal* (2007), vol. 31.

[138] SUNDARESAN, S., DE DONATO, W., FEAMSTER, N., TEIXEIRA, R., CRAWFORD, S., AND PESCAPÈ, A. Broadband internet performance: a view from the gateway. In *Proceedings of the ACM SIGCOMM 2011 conference* (2011), SIGCOMM'11, pp. 134–145.

[139] TAN, G., POLETTO, M., GUTTAG, J., AND KAASHOEK, F. Role classification of hosts within enterprise networks based on connection patterns. In *Proceedings of the annual conference on USENIX Annual Technical Conference* (2003), ATEC'03.

[140] TANG, A., ANDREW, L. L. H., JACOBSSON, K., JOHANSSON, K. H., HJALMARSSON, H., AND LOW, S. H. Queue dynamics with window flow control. *IEEE/ACM Trans. Netw. 18*, 5 (Oct. 2010), 1422–1435.

[141] TARIQ, M. B., MOTIWALA, M., FEAMSTER, N., AND AMMAR, M. Detecting network neutrality violations with causal inference. In *Proceedings of the 5th international conference on Emerging networking experiments and technologies* (2009), CoNEXT'09, pp. 289–300.

[142] TOZAL, M. E., AND SARAC, K. Tracenet: an internet topology data collector. In *Proceedings of the 10th ACM SIGCOMM conference on Internet measurement* (2010), IMC'10, pp. 356–368.

[143] TUKEY, J. *Exploratory Data Analysis*. Addison-Wesley Series in Behavioral Science. Addison-Wesley Publishing Company, 1977.

[144] URGAONKAR, B., PACIFICI, G., SHENOY, P., SPREITZER, M., AND TANTAWI, A. An analytical model for multi-tier internet services and its applications. In *Proceedings of the 2005 ACM SIGMETRICS international conference on Measurement and modeling of computer systems* (2005), SIGMETRICS'05, pp. 291–302.

[145] VASUDEVAN, V., SENGUPTA, S., AND LI, J. A first look at media conferencing traffic in the global enterprise. In *Conference on Passive and Active Network Measurement* (April 2009), PAM'09, pp. 133–142.

[146] VEITCH, D., BABU, S., AND PÀSZTOR, A. Robust synchronization of software clocks across the internet. In *Proceedings of the 4th ACM SIGCOMM conference on Internet measurement* (2004), IMC'04, pp. 219–232.

[147] WALLERICH, J., DREGER, H., FELDMANN, A., KRISHNAMURTHY, B., AND WILLINGER, W. A methodology for studying persistency aspects of internet flows. *SIGCOMM Comput. Commun. Rev. 35*, 2 (Apr. 2005), 23–36.

[148] WEI, S., AND MIRKOVIC, J. Correcting congestion-based error in network telescope's observations of worm dynamics. In *Proceedings of the 8th ACM SIGCOMM conference on Internet measurement* (2008), IMC'08, pp. 125–130.

[149] WILLINGER, W., AND PAXSON, V. Where mathematics meets the internet. *Notices of the American Mathematical Society* (1998), 961–970.

[150] XU, J., FAN, J., AMMAR, M., AND MOON, S. B. On the design and performance of prefix-preserving ip traffic trace anonymization. In *Proceedings of the 1st ACM SIGCOMM Workshop on Internet Measurement* (2001), IMW'01, pp. 263–266.

[151] Yin, J., Cai, Z., Zhao, W., and Liu, X. Passive calibration of active measuring latency. In *Proceedings of the 4th international conference on Networking - Volume Part II* (2005), ICN'05, pp. 746–753.

Aalto-DD 201/2013

ISBN 978-952-60-5466-7
ISBN 978-952-60-5467-4 (pdf)
ISSN-L 1799-4934
ISSN 1799-4934
ISSN 1799-4942 (pdf)

**Aalto University**
**School of Science**
**Department of Computer Science and Engineering**
**www.aalto.fi**