Understanding the Impact of Network Infrastructure Changes using Large-Scale Measurement Platforms

Vaibhav Bajpai

JACOBS
UNIVERSITY

Understanding the Impact of Network Infrastructure Changes using Large-Scale Measurement Platforms

by Vaibhav Bajpai

A thesis submitted in partial fulfillment of the requirements for the degree of

Doctor of Philosophy in Computer Science

Dissertation Committee:

Prof. Dr. Jürgen Schönwälder
Jacobs University Bremen, Germany

Dr. Kinga Lipskoch
Jacobs University Bremen, Germany

Prof. Dr. Filip De Turck
University of Ghent, Belgium

Date of Defense: May 30, 2016

DECLARATION

I, hereby declare that I have written this PhD thesis independently, unless where clearly stated otherwise. I have used only the sources, the data and the support that I have clearly mentioned. This PhD thesis has not been submitted for conferral of degree elsewhere. I confirm that no rights of third parties will be infringed by the publication of this thesis.

Bremen, Germany, May 30, 2016

Vaibhav Bajpai

This thesis is dedicated to my mom for her love, endless support and encouragement

ACKNOWLEDGMENTS

I would like to express my gratitude to my supervisor Jürgen Schönwälder for providing me constant feedback and support throughout the entire duration of my doctoral research. I would also like to thank my thesis committee consisting of Jürgen Schönwälder, Kinga Lipskoch and Filip De Turck for guiding and supporting my doctoral research.

I am grateful to my co-authors: Steffie Jacob Eravuchira, Saba Ahsan, Radek Krejčí, Jörg Ott and Jürgen Schönwälder with whom I learned to be a productive collaborator. Special thanks to: Sam Crawford, Philip Eardley, Trevor Burbridge, Arthur Berger, Daniel Karrenberg, Robert Kisteleki, Al Morton, Frank Bulk, Dan Wing and Andrew Yourtchenko for providing valuable and constructive feedback for improving my manuscripts. I also thank all the anonymous reviewers that helped improve this research work during the peer-review process.

I further thank Sam Crawford, Jamie Mason and Karen Davis from SamKnows for providing me constant technical support on the SamKnows infrastructure. My research on measuring IPv6 performance wouldn't have been possible without their help. I also thank Daniel Karrenberg, Philip Homburg, Robert Kisteleki, Emile Aben and Vesna Manojlovic from RIPE NCC for supporting my research work using the RIPE Atlas measurement platform.

I also like to thank Philip Eardley, Trevor Burbridge, Andrea Soppera, Olivier Bonaventure and all Leone and Flamingo project colleagues that promoted my growth as a researcher. I am also grateful to Christian Kaufmann, Marco Hogewoning and Fred Baker for giving me an opportunity to present my research at RIPE and IETF meetings.

I also would like to thank all the volunteers who hosted a SamKnows probe for this research activity (sorted by alphabetical order): Aiko Pras, Alex Buie, Andrea Soppera, Antonio Prado, Antonio Querubin, Bart Van Der Veer, Bayani Benjamin Lara, Brandon Ross, Chiang Fong Lee, Chris Baker, Clinton Work, Dario Ercole, Edoardo Martelli, Emile Aben, Eric Vyncke, Erik Taraldsen, Faruk Sejdic, Frank Bulk, Fuminori Tany Tanizaki, Heinrich Stamerjohanns, Javier Henderson, Jens Hoffmann, Jesse Sowell, Joel Maslak, Jon Thompson, Jordi Palet, Josh Hoge, Juan Cerezo, Juan Cordero, Jürgen Schönwälder, Kawashima Masanobu, Krunal Shah, Lucas do Amaral Saboya, Marco Sommani, Marian Neagul, Martin Neitzel, Masaki Tagawa, Mat Ford, Mathew Newton, Matthieu Bouthors, Michael Carey, Michael Richardson, Michael Van Norman, Mikael Abrahamsson, Mike Taylor, Mircea Suciu, Nick Chettle, Nishal Goburdhan, Ole Troan, Owen DeLong, Pedro Tumusok, Per Olsson, Peter Bulckens, Phillip Remaker, Radek Krejčí, Richard Patterson, Ryan Rawdon, Saba Ahsan, Sergey Sarayev, Steffie Jacob Eravuchira, Steinar Haug, Stephen Strowes, Steve Bauer, Tero Marttila, Thibault Cholez, Tim Chown, Tim Martin, Tobias Oetiker, Torbjorn Eklov, Trond Hastad, Vlad Ungureanu, Wouter de Vries, Yasuyuki Kaneko

My immeasurable gratitude to my mother and my brother Gaurav Bajpai for providing me continuous encouragement and for proof-reading my thesis.

This work was supported by the European Community's Seventh Framework Programme (FP7/2007-2013) grant no. 317647 (Leone). This work was also partly funded by Flamingo, a Network of Excellence project (ICT- 318488) supported by the European Commission under its Seventh Framework Programme.

ABSTRACT

A number of large-scale network measurement platforms have emerged in the last few years. These platforms have deployed thousands of measurement probes at strategic locations within the access and backbone networks and at residential gateways. The primary goal of these efforts is typically to measure the performance of broadband access networks and to help regulators sketch better policy decisions.

In this dissertation we expand the goal further by using large-scale measurement platforms to understand the impact of network infrastructure changes. We utilise probes deployed at the edge of the network to measure: a) IPv6 performance and b) access network performance. This dissertation largely provides three main contributions:

- **Survey on Internet Performance Measurement Platforms:** Initially, measurement platforms were deployed to measure the topology of the Internet. Such topology measurement platforms have been surveyed in the past [1, 2, 3]. In the last couple of years, this focus has evolved towards the measurement of network performance. This has been supported by the deployment of a number of performance measurement platforms. We provide a survey of such Internet performance measurement platforms [4]. For each performance measurement platform, we present its coverage, scale, lifetime, deployed metrics and measurement tools, architecture and overall research impact. Furthermore, we discuss standardization efforts that are currently being pursued in this space.

- **Measuring IPv6 Performance:** A large focus of IPv6 measurement studies in the past has been on measuring IPv6 adoption [5, 6, 7] on the Internet. However, there has been very little to no study [8] on measuring IPv6 performance. We measure IPv6 performance from the edge of the network to popular content services on the Internet. We present metrics, measurement tools, measurement insights and experience from studying geographically varied IPv6 networks. We provide a comparison of how content delivery [9, 10] over IPv6 compares to that of IPv4. We also identify and document glitches in this content delivery that can help improve user experience over IPv6. Our longitudinal observations also identify areas of improvements [11] in the standards work for the IPv6 operations community at the IETF.

- **Measuring Access Network Performance:** Last-mile latency is a key broadband network performance indicator. However little is known [12, 13] about the characteristics of last-mile latency in access networks. We perform a characterization of last-mile latency by time of day, by subscriber location, by broadband product subscription and by access technology used by the DSL modem. We show that DSL deployments not only tend to enable interleaving on the last-mile, but also employ multiple depth levels that change over time. Our characterization of last-mile latency can be used by simulation studies to model DSL, cable and fibre access links in the future.

JOURNAL PUBLICATIONS

1. Vaibhav Bajpai, Steffie Jacob Eravuchira, Jürgen Schönwälder. Lessons Learned from using the RIPE Atlas Platform for Measurement Research ACM Computer Communication Review (CCR) [Editorial], July 2015 **[ISI Impact Factor:** 1.40**,** 2015**]**. http://dx.doi.org/10.1145/2805789.2805796

2. Vaibhav Bajpai, Jürgen Schönwälder. A Survey on Internet Performance Measurement Platforms and Related Standardization Efforts. IEEE Communications Surveys & Tutorials (COMST), April 2015: **[ISI Impact Factor:** 9.22**,** 2015**]**. http://dx.doi.org/10.1109/COMST.2015.2418435

CONFERENCE PUBLICATIONS

1. Vaibhav Bajpai, Jürgen Schönwälder. IPv4 versus IPv6 - Who connects faster? IFIP Networking Conference, May 2015 **[Acceptance Rate:** 23.3%**,** 47/202**]**. http://dx.doi.org/10.1109/IFIPNetworking.2015.7145323

2. Saba Ahsan, Vaibhav Bajpai, Jörg Ott, Jürgen Schönwälder. Measuring YouTube from Dual-Stacked Hosts. Passive and Active Measurement Conference (PAM), March 2015 **[Acceptance Rate:** 27%**,** 27/100**]**. Also presented at the IRTF / ISOC Workshop on Research and Applications of Internet Measurements (RAIM), October 2015. http://dx.doi.org/10.1007/978-3-319-15509-8_19

3. Vaibhav Bajpai, Johannes Schauer, Jürgen Schönwälder. NFQL: A Tool for Querying Network Flow Records. IFIP/IEEE International Symposium on Integrated Network Management (IM). May 2013. http://ieeexplore.ieee.org/xpl/freeabs_all.jsp?arnumber=6573045

WORKSHOP PUBLICATIONS

1. Vaibhav Bajpai, Jürgen Schönwälder. Measuring the Effects of Happy Eyeballs. ACM / IRTF / ISOC Applied Networking Research Workshop (ANRW), July 2016. **[Acceptance Rate:** 60%**,** 18/30**]**. http://dx.doi.org/10.1145/2959424.2959429

2. Vaibhav Bajpai, Jürgen Schönwälder. Understanding the Impact of Network Infrastructure Changes using Large-Scale Measurement Platforms. Autonomous Infrastructure, Management & Security (AIMS), June 2013. **[Acceptance Rate:** 50%**,** 07/14**]**. http://dx.doi.org/10.1007/978-3-642-38998-6_5

POSTERS

1. Vaibhav Bajpai, Radek Krejčí. Managing SamKnows Probes using NETCONF. IEEE/IFIP Network Operations and Management Symposium (NOMS), May 2014. http://dx.doi.org/10.1109/NOMS.2014.6838279.

2. Vaibhav Bajpai, Jürgen Schönwälder. NETCONF Interoperability Lab. IEEE/IFIP Network Operations and Management Symposium (NOMS), May 2014. http://dx.doi.org/10.1109/NOMS.2014.6838278.

3. Vaibhav Bajpai, Jürgen Schönwälder. Measuring TCP Connection Establishment Times of Dual-Stacked Web Services. Conference on Network and Service Management (CNSM), October 2013. http://dx.doi.org/10.1109/CNSM.2013.6727822

IETF INTERNET-DRAFTS

1. Jürgen Schönwälder, Vaibhav Bajpai. A YANG Data Model for LMAP Measurement Agents. IETF Internet Draft (I-D), July 2016 **[Working Group Document]**. http://tools.ietf.org/html/draft-ietf-lmap-yang-05.

2. Jürgen Schönwälder, Vaibhav Bajpai. Using RESTCONF with LMAP Measurement Agents. IETF Internet Draft (I-D), July 2016 **[Working Group Document]**. http://tools.ietf.org/html/draft-ietf-lmap-restconf-03.

3. Marcelo Bagnulo, Trevor Burbridge, Sam Crawford, Juergen Schoenwaelder, Vaibhav Bajpai. Large MeAsurement Platform Protocol. IETF Internet Draft (I-D), September 2014 **[Expired Document]**. http://tools.ietf.org/html/draft-bagnulo-lmap-http-03.

4. Vaibhav Bajpai, Jürgen Schönwälder. Measuring the Effects of Happy Eyeballs. IETF Internet Draft (I-D), July 2013 **[Expired Document]**. http://tools.ietf.org/html/draft-bajpai-happy-01. Also published as a RIPE Labs Article, June 2013. https://goo.gl/1mERyV

EVENT REPORTS

1. Vaibhav Bajpai, Arthur W. Berger, Philip Eardley, Jörg Ott, Jürgen Schönwälder. Global Measurements: Practice and Experience (Report on Dagstuhl Seminar #16012). ACM Computer Communication Review (CCR), April 2016 **[ISI Impact Factor:** 1.40**,** 2015**]**. http://dx.doi.org/10.1145/2935634.2935641. Dagstuhl Reports, Volume 6, Issue 1, May 2016. http://dx.doi.org/10.4230/DagRep.6.1.15

2. Vaibhav Bajpai, Jürgen Schönwälder. A Report on the 1st NMRG Workshop on Large Scale Network Measurements. Journal of Network and Systems Management (JNSM), September 2014 **[ISI Impact Factor:** 1.08**,** 2015**]**. http://dx.doi.org/10.1007/s10922-014-9328-2

TALKS

1. Vaibhav Bajpai, Steffie Jacob Eravuchira, Jürgen Schönwälder, Sam Crawford. Measuring Webpage Similarity from Dual-Stacked Hosts. RIPE 72 Plenary, Copenhagen, May 2016. https://ripe72.ripe.net/archives/video/126

2. Vaibhav Bajpai, Steffie Jacob Eravuchira, Jürgen Schönwälder, Robert Kisteleki, Emile Aben. Vantage Point Selection for IPv6 Measurements: Benefits and Limitations of RIPE Atlas Tags. MAT Working Group Meeting at RIPE 72, Copenhagen, May 2016. https://ripe72.ripe.net/archives/video/167

3. Steffie Jacob Eravuchira, Vaibhav Bajpai, Jürgen Schönwälder. Measurement Research within the Python3 Ecosystem. MAT Working Group Meeting at RIPE 69, London, November 2014. https://ripe69.ripe.net/archives/video/10125

4. Vaibhav Bajpai, Steffie Jacob Eravuchira, Jürgen Schönwälder. Lessons Learned From Using the RIPE Atlas Platform for Measurement Research. MAT Working Group Meeting at RIPE 68, Warsaw, May 2014. https://ripe68.ripe.net/archives/video/240

5. Vaibhav Bajpai, Jürgen Schönwälder. Measuring TCP Connection Establishment Times of Dual-Stacked Web Services. IRTF NMRG Workshop on Large-Scale Network Measurements at CNSM 2013, Zürich, October 2013. http://goo.gl/3ACsAA

6. Vaibhav Bajpai, Jürgen Schönwälder. Measuring the Effects of Happy Eyeballs. v6ops Working Group Meeting at IETF 87, Berlin, July 2013. http://ietf.org/proceedings/87/slides/slides-87-v6ops-8.pdf

7. Vaibhav Bajpai, Jürgen Schönwälder. Measuring the Effectiveness of Happy Eyeballs. IPv6 Working Group Meeting at RIPE 66, Dublin, May 2013. https://ripe66.ripe.net/archives/video/1208

TUTORIALS

1. Vaibhav Bajpai, Nikolay Melnikov. Large-Scale Measurement Platforms. Autonomous Infrastructure, Management & Security (AIMS), Barcelona, June 2013. http://cnds.eecs.jacobs-university.de/tutorials

CONTENTS

Part I

INTRODUCTION

In Chapter 1, we present the research statement with research questions and associated approach. Research contributions that incubate out of this work are enlisted and an outline is presented on how to read this dissertation.

1 INTRODUCTION

A *large-scale measurement platform* is an infrastructure of dedicated hardware probes that periodically run network measurements tests. These platforms have been deployed to satisfy specific use-case requirements. For instance, a number of platforms (such as CAIDA Archipelago [14], DIMES [15] and iPlane [16]) emerged in the past to accurately map the network topology of the Internet. Several years of research efforts has matured this area. Recently we have seen a shift towards deployment of performance measurement platforms that provide network operational support (such as RIPE Atlas [17, 4] and PerfSONAR [4]) and measure fixed-line (such as SamKnows [4] and BISmark [18]) networks. This has been motivated by the emerging need to not only assess the broadband quality but also to verify service offers against contractual agreements.

Marc Linsner *et al.* in [19] (2015) describe three use-cases that motivate large-scale broadband measurements: Internet Service Provider (ISP), consumers and regulators. An ISP would like to use broadband measurements not only to identify, isolate and fix problems in access networks, but also to evaluate the Quality of Service (QoS) experienced by its users. Public measurement data in addition helps the ISP benchmark its product and peek into its competitor's insights. The consumers, on the other hand, would like to use these measurements to confirm whether the ISP is adhering to its Service-Level Agreement (SLA) offerings. The user can also use these measurement insights to audit and diagnose network problems in its own private internal network. The measurement insights will eventually become useful to network regulators. The Federal Communications Commission (FCC), the national regulator in the United States, has launched a campaign [20] with an intent to use measurement datasets to study and compare multiple broadband provider offerings. Ofcom, the national regulator in the United Kingdom, has been using such datasets [21] as input to frame better policies to help regulate the broadband industry in the past.

Sundaresan *et al.* [22] (2011) have used measurement data from a swarm of deployed SamKnows probes to investigate the throughput and latency of access network links across multiple ISPs in the United States. They have analyzed this data together with data from their own Broadband Internet Service Benchmark (BISmark) platform [18] to investigate different traffic shaping policies enforced by ISPs and to understand the bufferbloat [23] phenomenon. The empirical findings of this study have been repraised by Canadi *et al.* in [24] (2012) where they use crowdsourced data from `speedtest.net` to compare both results. The primary aim of all these activities is to measure the performance and reliability of broadband access networks and facilitate the regulators with research findings to help them make policy decisions.

Contents

1.1 MOTIVATION

In this dissertation, we expand the goal by using large-scale measurement platforms to *understand the impact of network infrastructure changes*. We utilise probes deployed at the edge of network to measure: a) IPv6 performance and b) access network performance.

a) **Measuring IPv6 Performance:** In the past, IPv6 measurement studies were focussed on measuring IPv6 adoption [5, 6, 7] on the Internet. This involved measuring addressing, naming, routing and reachability aspects of IPv6. However, there has been very little work on measuring the performance of delivered services over IPv6. This has largely been due to lack of the availability of content over IPv6. This changed significantly during the span of this dissertation work as a cascading effect of a number of events. For one, the World IPv6 Launch day in 2012 [25] gathered several notable content providers to start providing services over both IPv4 and IPv6. This was also driven by the rapidly exhausting pool of IPv4 address space. As of today, 4/5 RIRs: APNIC (in Apr 2011), RIPE (in Sep 2012), LACNIC (in Jun 2014), and ARIN (in Sep 2015) have exhausted their IPv4 address pool [26] and consequently LIRs now receive allocations from within the last available IPv4 /8 address block. As a result of this depletion, within a span of 3 years, a number of large IPv6 broadband rollouts have also happened [9]. These efforts have eventually led to an increased global adoption of IPv6. Fig. 1 shows how IPv6 adoption jumped during the span of this dissertation work from around 0.85% (as of Sep 2012) to around 9.8% (as of Mar 2016) according to Google's IPv6 adoption statistics [27]. These numbers demonstrate that IPv6 is no longer an optional IP stack protocol. However, there has been very little to no study [8] on measuring IPv6 performance. This dissertation, fills this gap to measure IPv6 performance of operational dual-stacked content services from 80 dual-stacked vantage points.

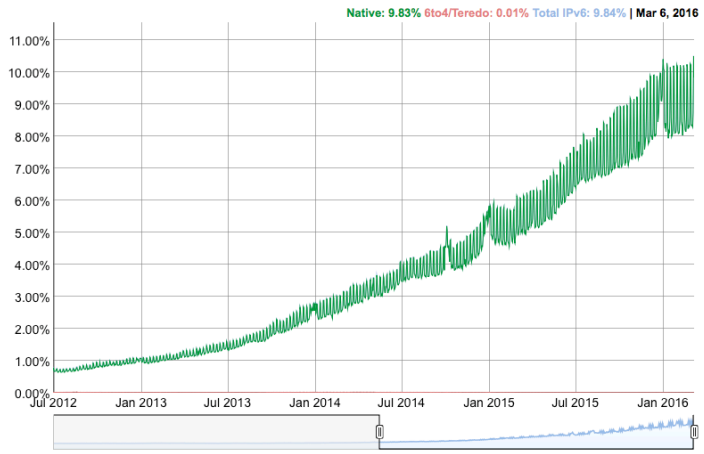


Figure 1: *Timeline of IPv6 adoption over duration of this dissertation work. IPv6 adoption jumped from around 0.85% (as of Sep 2012) to around 9.8% (as of Mar 2016) as seen by Google IPv6 adoption statistics: https://www.google.com/intl/en/ipv6/statistics.html*

b) **Measuring Access Network Performance:** Measurement Studies [28] (2013) performed using the BISmark [4, 18] platform, have shown that latency becomes a critical factor impacting Quality of Experience (QoE) in networks where downstream throughput exceeds 16Mb/s. This has driven content providers to deploy content caches [29, 30] in service provider networks to move the content as close [31, 32] to the edge as possible. Furthermore, recent [33] (2015) and upcoming standards [34, 35] (2015) cater to this requirement to target operation at a much reduced latency. It was recently shown [28] (2013) that last-mile latency is a major contributor to end-to-end latency and it contributes heavily to Domain Name System (DNS) lookup and page load times. Last-mile latency is becoming a key broadband network performance indicator. However little is known [12, 13] (2011, 2007) about the characteristics of last-mile latency in access networks. In this dissertation we perform a deeper investigation of last-mile latency to close this gap.

1.2 RESEARCH STATEMENT

This dissertation is largely divided into 3 parts: a) A survey on large-scale Internet measurement platforms, b) measuring IPv6 performance and c) measuring access network performance. In this section we present the research questions separately for each part:

a) **Internet Performance Measurement Platforms:** Recently we have seen a trend towards the deployment of Internet performance measurement platforms that provide network operational support and measure fixed-line and mobile access networks. This has been motivated by the emerging need to not only assess the broadband quality but also to verify service offers against contractual agreements. Platforms (such as CAIDA Ark [36]) focussing on inferring the Internet topology have been surveyed in the past [1, 2, 3]. Metrics and tools usually employed in active measurements have also been surveyed [37, 38]. However, there has been no survey on Internet performance measurement platforms. We want to know:

> RQ − 1 : *What is the state-of-the-art in Internet performance measurement platforms? What is the coverage, scale, lifetime, deployed metrics and measurement tools, architecture and overall research impact of such performance measurement platforms? What standardization efforts are currently being pursued in this space?*

The research question, RQ − 1 is discussed in Part II.

b) **Measuring IPv6 Performance:** The default address selection policy [39] has been designed to favour native IPv6 connections in dual-stacked networks. We want to know:

> RQ − 2 : *Do users experience benefit (or an added penalty) when connecting to popular dual-stacked websites over IPv6?*

> RQ − 3 : *How do websites centralize over Content Distribution Networks (CDN) infrastructure for IPv4 and IPv6 content delivery? Is there disparity in the availability of CDN caches over IPv4 and IPv6?*

The research questions RQ − 2 and RQ − 3 are discussed in Chapter 8. In order to address RQ − 2 and RQ − 3, we introduce a metric that measures Transmission Control Protocol (TCP) connection establishment times. By

repeated execution of such a test, we are able to collect time series of TCP connect times that provide us with insights on how IPv6 connectivity to websites compares to IPv4 connectivity.

The Internet Engineering Task Force (IETF) has also developed solutions to promote a healthy IPv4 and IPv6 co-existence. The Happy Eyeballs (HE) algorithm [40] (2012) for instance, provides recommendations to application developers to help prevent bad user experience in situations where IPv6 connectivity is broken. The algorithm when combined with the default address selection policy [39] (2012), tends to give a noticeable advantage (300 ms) to connections made over IPv6. The HE timer value was chosen during a time (2012) when broken IPv6 connectivity was quite prevalent, which made applications stall for several seconds before attempting a connection over IPv4. The broken IPv6 connectivity has been largely attributed to failures caused by Teredo [41] and 6to4 relays [42]. Today, IPv6 adoption has reached 10.2% (native) with Teredo/6to4 at around 0.01% according to Google IPv6 adoption statistics [27] (as of Feb 2016). In such a changed landscape, the effect of the HE timer value (300 ms) on the overall experience of a dual-stacked user remains largely unclear. We want to know:

> RQ − 4 : *What are the percentage of cases where HE makes a bad decision of choosing IPv6 when it's slower. Furthermore, in such situations what is the amount of imposition (in terms of latency impact) a dual-stacked user has to pay as a result of the high HE timer value.*

This is critical since applications on top of TCP not only apply HE in scenarios where IPv6 connectivity is broken, but also in scenarious where IPv6 connectivity is comparable. The fragmentation of the algorithm due to the high HE timer value is visible in implementations today. For instance, Firefox (since v15) [43] and Opera (since v12.10) [44] by default use parallel TCP connections over IPv4 and IPv6. Google Chrome (since v11) [45] uses a 300 ms timer value while Apple (since OS X 10.11 and iOS 9) [46] uses a considerably smaller 25 ms timer value in favor of IPv6 connections. Note, these values are arbitrarily chosen. We want to know:

> RQ − 5 : *What is the right HE timer value that provides the same preference levels over IPv6 as is today but also reduces the performance penalty in situations where IPv6 is considerably slower.*

Research questions, RQ − 4 and RQ − 5 are discussed in Chapter 9. Using a 3-years long (2013 - 2016) dataset of TCP connection establishment times obtained from our metric, we are able to calculate decisions a HE enabled application would have taken. We are also able to experiment with variations of the HE algorithm and propose changes to it.

Nadi Sarrar *et al.* in [47] (2012) recently studied the application mix of traffic before and after the World IPv6 Day. They showed that IPv6 traffic is largely dominated by services running over HTTP and that YouTube is the primary service over HTTP that contributes heavily to large volumes of IPv6 traffic. We want to know:

> RQ − 6 : *Do users experience benefit (or an added penalty) when streaming YouTube videos over IPv6? How do failure rates compare over IPv4 and IPv6? What factors contribute towards the performance difference? Is there disparity in the availability of Google Global Caches (GGC) over IPv4 and IPv6?*

The research question, RQ − 6 is discussed in Chapter 10. In order to address RQ − 6, we run two kinds of measurements: speed tests and YouTube

tests. Each test is run over IPv4 and then IPv6 separately allowing us to draw performance comparison over each address family.

The content providers need to ensure that the content delivered over IPv4 and IPv6 is identical. This is a two-step process, whereby the content provider has to begin by providing an AAAA record of the service endpoint (or the upfront load balancer) to the DNS resolvers. The end-host then must be able to receive the same content when requesting services from the resolved IPv6 endpoint. The IPv6 adoption studies have mostly focussed on the first step by measuring the amount of AAAA entries in DNS resolvers. The similarity of the content served over IPv4 and IPv6 has not been measured in practice. We want to know:

> RQ − 7 : *How similar are the webpages accessed over IPv6 to their IPv4 counterparts? Is it that most of the content providers provide an AAAA entry but only serve a landing page when a request is made over IPv6, or is the content delivery over both routes the same for all the services?*

> RQ − 8 : *In situations where the content is dissimilar over IPv4 and IPv6, what factors contribute to the dissimilarity?*

Research questions, RQ − 7 and RQ − 8 are discussed in Chapter 11. In order to address RQ − 7 and RQ − 8, we develop and deploy an active test (`simweb`) that uses well-known content and service complexity metrics [48] to quantify the level of webpage similarity. In situations where there is a dissimilarity we also perform a causal analysis and identify sources responsible for the difference.

c) **Measuring Access Network Performance:** Recent studies [28] (2013) have shown that latency becomes a critical factor impacting quality of experience in networks where downstream throughput exceeds 16Mb/s. Recently it was shown [28] that last-mile latency is a major contributor to this end-to-end latency and it contributes heavily to DNS lookup and page load times. Last-mile latency is becoming a key broadband network performance indicator today and factors affecting last-mile latency need further investigation. We want to know:

> RQ − 9 : *Should last-mile latency measurements include latencies within the home network? How to account for queuing delay caused by bufferbloat on home routers when measuring last-mile latencies?*

> RQ − 10 : *What characteristic value of last-mile latency can be used by simulation studies to model DSL, cable and fibre access links?*

Prior knowledge [12, 13] (2011, 2007) has shown that cable users in the US experience lower last-mile latencies than Digital Subscriber Line (DSL) users due to interleaving effects. We want to know:

> RQ − 11 : *Do service providers employ multiple interleaving depth levels? Do these depth levels vary over time?*

> RQ − 12 : *Do last-mile latencies vary by time of day? Do they vary by subscriber location? Do they vary by broadband product subscription and the access technology used by the DSL modem?*

Research questions, RQ − 9 to RQ − 12 are discussed in Chapter 13. In order to address RQ − 9 to RQ − 12, we utilise two month-long traceroute

datasets. The first dataset has been obtained from 696 residential RIPE Atlas probes deployed in 19 different network service providers in the US and the EU. The second dataset has been obtained from 1245 SamKnows [4] probes deployed in 9 network service providers in the UK. The latencies observed as part of the traceroute measurement allow to capture the last-mile latency characteristics of these service provider networks.

1.3 RESEARCH CONTRIBUTION

The dissertation work provides the following research contributions:

Chapter 2 - 7: Internet Performance Measurement Platforms: We provide a taxonomy of Internet performance measurement platforms based on their deployment use-case: a) platforms deployed at the periphery of the Internet that measure performance over fixed-line access networks, b) platforms that measure performance over mobile access networks, c) platforms deployed largely within the core of the Internet that help provide network operational support. We present a survey of these Internet performance measurement platforms, and provide a comprehensive review of their features and research impacts with an exploration on standardization efforts that will help make these measurement platforms interoperable. This contribution is based on the following publication:

- Vaibhav Bajpai, Jürgen Schönwälder. A Survey on Internet Performance Measurement Platforms and Related Standardization Efforts. IEEE Communications Surveys & Tutorials (COMST), April 2015: **[ISI Impact Factor:** 9.22**,** 2015**]**. http://dx.doi.org/10.1109/COMST.2015.2418435

Chapter 8: Measuring TCP Connect Times: a) We define an active metric and a corresponding implementation (`happy`) to measure TCP connection establishment times alongwith a list of top 100 dual-stacked websites processed from Amazon 1M Alexa entries. b) We identify CDN deployments and content-caches over IPv4 and IPv6 in service provider networks using Border Gateway Protocol (BGP)-based clusters processed from Internet Protocol (IP) endpoints seen from globally distributed SamKnows vantage points. A quantification of disparity in IPv4 and IPv6 clusters is also made available. c) We present distributions of TCP connect times over an year-long dataset to compare IPv4 and IPv6 performance over each CDN cluster and d) We discuss special cases such as `www.bing.com` globally stopping IPv6 services in 2013, and Google CDN blacklisting resolvers that inhibit some hosts from receiving their services over IPv6. These contributions are based on the following publications:

- Vaibhav Bajpai, Jürgen Schönwälder. IPv4 versus IPv6 - Who connects faster? IFIP Networking Conference, May 2015 **[Acceptance Rate:** 23.3%**,** 47/202**]**. http://dx.doi.org/10.1109/IFIPNetworking.2015.7145323
- Vaibhav Bajpai, Jürgen Schönwälder. Measuring TCP Connection Establishment Times of Dual-Stacked Web Services. Conference on Network and Service Management (CNSM) Poster Session, October 2013 http://dx.doi.org/10.1109/CNSM.2013.6727822

Chapter 9: Measuring Effects of Happy Eyeballs: a) We show that TCP connect times to popular websites over IPv6 have considerably improved

over time. As of Jan 2016, 5% of these websites are faster over IPv6 with 90% being atmost 1 ms slower. b) Only around 1% of the TCP connect times over IPv6 were ever above the HE timer value (300 ms), which leaves around 2% chance for IPv4 to win a HE race towards these websites. As such, 99% of these websites prefer IPv6 connections more than 98% of the time and c) Although absolute TCP connect times (in ms) are not that far apart in both address families, HE with 300 ms timer value tends to prefer slower IPv6 connections in around 90% of the cases. A lowering of the HE timer value to 150 ms gives us a margin benefit of 10% while retaining same preference levels over IPv6. These contributions are based on the following publications:

- Vaibhav Bajpai, Jürgen Schönwälder. Measuring the Effects of Happy Eyeballs. ACM / IRTF / ISOC Applied Networking Research Workshop (ANRW), July 2016. **[Acceptance Rate:** 60%**,** 18/30**]**. http://dx.doi.org/10.1145/2959424.2959429
- Vaibhav Bajpai, Jürgen Schönwälder. Measuring the Effects of Happy Eyeballs. IETF Internet Draft (I-D), July 2013 **[Expired Document]**. http://tools.ietf.org/html/draft-bajpai-happy-01. Also published as a RIPE Labs Article, June 2013. https://goo.gl/1mERyV

Chapter 10: Measuring YouTube: a) Success rates of streaming a stall-free version of a video over IPv6 have improved over time. b) A HE race during initial TCP connection establishment leads to a strong (more than 97%) preference to stream audio and video content over IPv6. c) Even though clients prefer streaming videos over IPv6, the observed performance over IPv6 is worse. We witness consistently higher TCP connection establishment and startup delays (100 ms or more) over IPv6. d) Furthermore, we observe consistently lower achieved throughput both for audio and video streams over IPv6, although the throughput difference has improved over time. e) We observe less than 1% stall rates over both address families and stall durations tend to have reduced over the years. Due to lower stall rates, bitrates that can be reliably streamed over both address families are comparable. However in situations where a stall does occur, 80% of the samples experience stall durations that are at least 1s longer over IPv6. f) We also witness that 97% of our probes receive content delivery through a content cache over IPv4 while only 5% receive it from a content cache over IPv6.

- Vaibhav Bajpai, Saba Ahsan, Jürgen Schönwälder, Jörg Ott. Measuring YouTube over IPv6 (Under Review).
- Saba Ahsan, Vaibhav Bajpai, Jörg Ott, Jürgen Schönwälder. Measuring YouTube from Dual-Stacked Hosts. Passive and Active Measurement Conference (PAM), March 2015 **[Acceptance Rate:** 27%**,** 27/100**]**. Also presented at the IRTF / ISOC Workshop on Research and Applications of Internet Measurements (RAIM), October 2015. http://dx.doi.org/10.1007/978-3-319-15509-8_19

Chapter 11: Measuring Web Similarity: a) `simweb`: A tool for measuring webpage similarity over IPv4 and IPv6. The tool is written in C and open-sourced for the measurement community. b) 14% of the ALEXA top 100 dual-stacked websites exhibit dissimilarity in the *number* of fetched webpage elements with 6% showing more than 50% difference. 94% of dual-stacked websites exhibit dissimilarity in *size* with 8% showing atleast 50% difference. This dissimilarity in number and size of elements negatively impacts webpages fetched over IPv6. c) 27% of dual-stacked websites have some fraction

of webpage elements that fail over IPv6 with 9% of the websites having more than 50% webpage elements that fail over IPv6. Worse, 6% announce AAAA entries in the DNS but no content is delivered over IPv6 when an HTTP request is made. d) Failure rates are largely affected by DNS resolution error on images, javascript and CSS content delivered from both same-origin and cross-origin sources. This contribution is joint work with Steffie Jacob Eravuchira. This chapter and the following publication is a condensed version of her masters thesis [49].

- Steffie Jacob Eravuchira, Vaibhav Bajpai, Jürgen Schönwälder, Sam Crawford. Measuring Web Similarity from Dual-Stacked Hosts. (Under Review)

Chapter 12: RIPE Atlas Vantage Point Selection: a) We show that system tags have improved the vantage point selection process by exhibiting a case study on selecting dual-stacked probes for IPv6 measurement studies and b) We extend the tagging effort to allow automated tagging of popular user tags. This will eliminate the need for probe hosts to manually tag their probes. We validate our results against the ground truth obtained from user-tagged probes. These contributions are based on the following publication:

- Vaibhav Bajpai, Steffie Jacob Eravuchira, Jürgen Schönwälder, Robert Kisteleki, Emile Aben. Vantage Point Selection for IPv6 Measurements: Benefits and Limitations of RIPE Atlas Tags (Under Review)

Chapter 13: Revisiting Last-mile Latency: a) The home network latency makes a discernible contribution and therefore should not be accounted when measuring last-mile links. b) Some Customer-premises Equipment (CPE)s rate limit Internet Control Message Protocol (ICMP) responses to Time to Live (TTL) expiry. Latencies towards these CPEs should not be accounted for baseline measurements. c) DSL service providers not only enable interleaving, but also dynamically adapt the depth levels with time. d) Last-mile latency is considerably stable over time and not affected by diurnal load patterns. e) Last-mile latencies for DSL center at around 16 ms, with cable at around 8 ms, and fibre deployments at around 4 ms. f) Subscribers of some US cable providers experience considerably different last-mile latencies across the US east (centered at around 32 ms) and west coast (centered at around 8 ms) and g) Last-mile latencies decrease with increase in broadband product. Very-high-bit-rate DSL (VDSL) deployments show last-mile latencies lower than Asymmetric DSL (ADSL)2/ADSL2+. These contributions are based on the following publication:

- Vaibhav Bajpai, Steffie Jacob Eravuchira, Jürgen Schönwälder. Last-mile Latency of Broadband Access Networks (Under Review)

Chapter 14: Lessons Learned from using RIPE Atlas a) We identify that v3 probes are more suitable for performance (such as latency) measurements than older versions (v1 and v2) that suffer load issues. b) We demonstrate how measurement-based studies that require higher coverage of network origins benefit more from RIPE Atlas than those that require high probe density within each network and c) We describe two use-cases where measurement platforms can benefit from one another: SamKnows probes are cross-traffic aware (unlike RIPE Atlas probes) and RIPE Atlas probes do not aggregate latencies over each `traceroute` hop (unlike SamKnows probes) both of which when disabled can heavily impact measurement results. These contributions are based on the following publication:

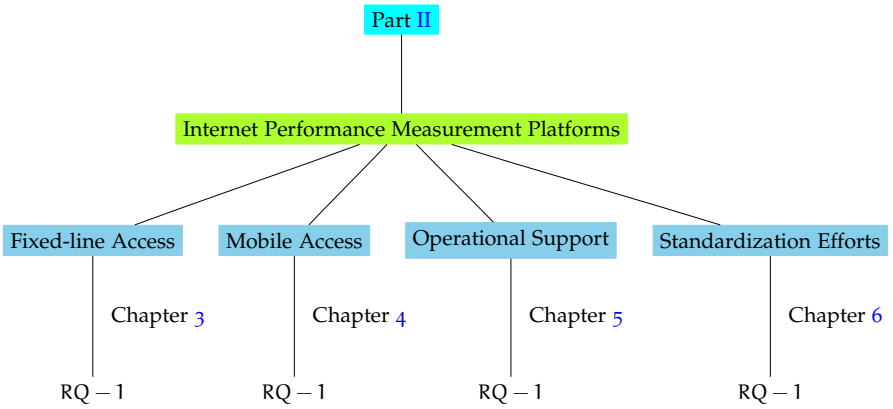


Figure 2: *An outline of the survey on Internet performance measurement platforms and related standardization efforts. This part of the thesis covers* RQ – 1.

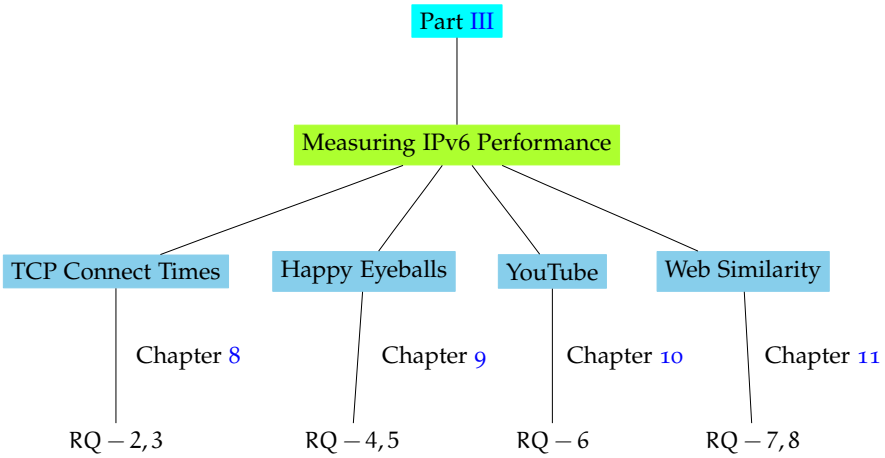


Figure 3: *A research outline on measuring IPv6 performance. We compare TCP connect times and similarity of webpages. We also measure the effects of HE and performance of YouTube streaming over IPv6. This part of the thesis covers* RQ – 2 *to* RQ – 8.

- Vaibhav Bajpai, Steffie Jacob Eravuchira, Jürgen Schönwälder. Lessons Learned from using the RIPE Atlas Platform for Measurement Research. ACM Computer Communication Review (CCR) [Editorial], July 2015 **[ISI Impact Factor:** 1.40**,** 2015**]**. http://dx.doi.org/10.1145/2805789.2805796

1.4 HOW TO READ THIS THESIS

The structure of the thesis follows directly from the research questions. For instance, Fig. 2 shows the outline where research question RQ – 1 is discussed. This part is relevant for parties who build and maintain large-scale measurement platforms. This part may also prove useful to early researchers to get acquainted with the background (see Chapter 3, 4, 5) in measurement-based research. Parties involved in large-scale measurement standardization activities (see Chapter 6) may also find this part useful.

Fig. 3 shows the outline of our research on measuring IPv6 performance. This covers research questions, RQ – 2 to RQ – 8 and includes metrics, open-

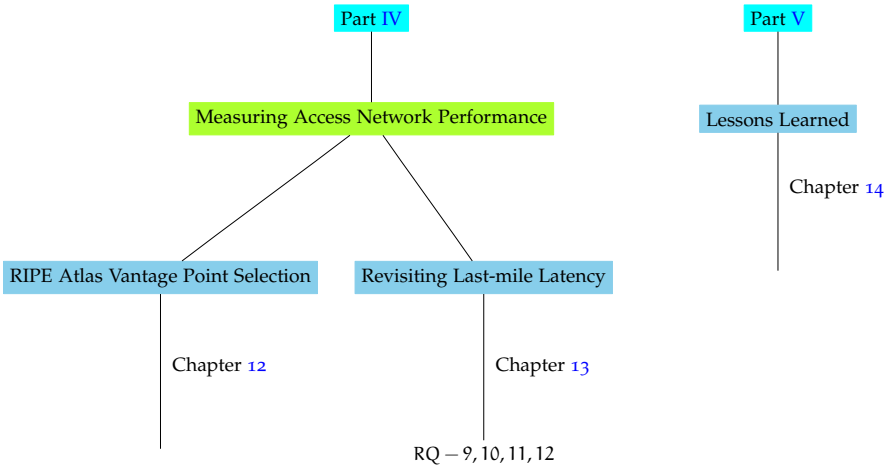


Figure 4: *A research outline on measuring access network performance. We present a methodology to select RIPE Atlas home probes and use it to measure last-mile latency of home networks. Lessons learned from using RIPE Atlas and SamKnows are also discussed. This part of the thesis covers* RQ – 9 *to* RQ – 12.

source measurement tools, insights from measurement data analysis and experience from studying geographically varied IPv6 networks. This part is relevant for network operators that are either in the process of or are in early stages of IPv6 deployment. It also provides content providers insights towards how their service delivery over IPv6 compares (see Chapter 8 and 10) against IPv4. In the process, it also identifies glitches in the content delivery (see Chapter 11) that once patched can help improve user experience over IPv6. Furthermore, our longitudinal observations may also help drive related standards work (see Chapter 9) in the IETF in the future.

Fig. 4 shows the outline of our research on measuring access network performance. This covers research questions, RQ – 9 to RQ – 12. Our vantage point selection methodology (see Chapter 12) to identify home probes in the RIPE Atlas platform can serve as a good starting point for future broadband measurement studies using the RIPE Atlas platform. This part extends our understanding of last-mile latency witnessed by home users. CDN providers that attempt to optimise content delivery towards the edge of the network may benefit from the identified characteristics (see Chapter 13) of the last-mile. This work will also benefit service providers since it promotes the possibility of caching popular content near to the CPE to further eliminate the bottlenecks induced by last-mile latency. This research may also serve as possible input for ongoing standardization efforts (such as Quick UDP Internet Connection (QUIC) [34] and Transport Layer Security (TLS) 1.3 [35]) within the IETF that attempt to target operations at much reduced latency. Lessons learned from pursuing this part of the research may also prove valuable (see Chapter 14) to the wider measurement community in general.

Part II

INTERNET PERFORMANCE MEASUREMENT PLATFORMS

A number of Internet measurement platforms have emerged in the last few years. These platforms have deployed thousands of probes at strategic locations within access and backbone networks and behind residential gateways. In this part we provide a taxonomy of these measurement platforms on the basis of their deployment use-case. We describe these platforms in detail by exploring their coverage, scale, lifetime, deployed metrics and measurement tools, architecture and overall research impact. We conclude by describing current standardization efforts to make large-scale performance measurement platforms interoperable.

In Chapter 2 we describe the scope of the survey and related background research that paved way for large-scale Internet measurement platforms. In Chapter 3 and 4, we cover platforms that measure performance on fixed-line and mobile access networks. Chapter 5 surveys platforms that perform measurements to provide support to network operators and the scientific community. We explore upcoming efforts to standardize components of a measurement infrastructure to make these measurement platforms interoperable in Chapter 6. We conclude with a discussion of collaboration amongst these platforms, usage of measurement facilitators, timeline of the surveyed work and an overall summary in Chapter 7.

2 INTRODUCTION

An Internet measurement platform is an infrastructure of dedicated probes that periodically run network measurement tests on the Internet. These platforms have been deployed to satisfy specific use-case requirements. Fig. 5 provides a taxonomy of these platforms based on their deployment use-case. For instance, a number of early measurement studies utilized these platforms to understand the macroscopic network-level topology of the Internet. Several years of research efforts have matured this area and led to a number of algorithms that decrease the complexity of such topology mapping efforts. Recently we have seen a shift towards deployment of performance measurement platforms that provide network operational support and measure fixed-line and mobile access networks. This has been motivated by the emerging need to not only assess the broadband quality but also to verify service offers against contractual agreements. For instance, the FCC, the national regulator in the United States, has launched a campaign [20] with an intent to use the gathered measurement dataset to study and compare multiple broadband provider offerings in the country. The Office of Communications (Ofcom), the national regulator in the United Kingdom, has already

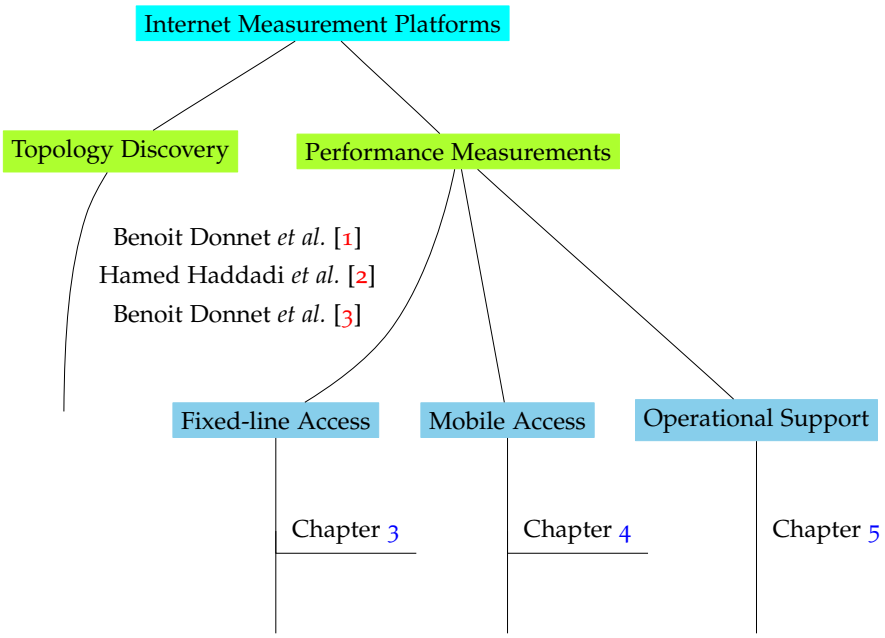


Figure 5: *A graph representing the taxonomy of Internet measurement platforms. They can largely be divided into two classes: topology discovery (labels depicting references to earlier surveys) and performance measurements. We further subdivide performance measurement platforms into three classes depending on their deployment use-case: measurements within fixed-line access networks, mobile access networks and measurements to provide operational support. Labels indicate sections where we survey them in detail.*

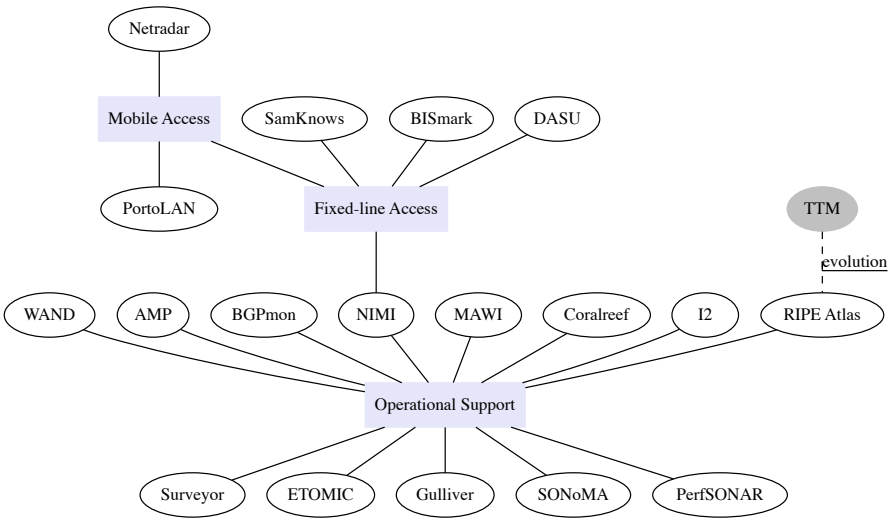


Figure 6: *A graph representing the taxonomy (in purple) of Internet performance measurement platforms (in white) based on their deployment use-case. Greyed out measurement platforms have been superseded by their successors. We only survey currently active measurement platforms from within this set. Tables 4, 5 and 6 provide a summary of this survey.*

been using similar datasets [21] as input to frame better broadband policies. Such initiatives are being run to help regulate the broadband industry.

We focus our survey on these Internet performance measurement platforms, and provide a comprehensive review of their features and research impacts with an exploration on standardization efforts that will help make these measurement platforms interoperable. Platforms focussing on inferring the network topology have been surveyed in the past [1, 3]. Techniques used to mine the active measurement data to model and generate the Internet topology have been surveyed as well [2]. Metrics and tools usually employed in such active measurements have also been surveyed [37, 38]. Therefore, we do not survey topology discovery platforms such as Archipelago [36], DIMES [15] and iPlane [16], but refer the reader to the aforementioned surveys.

There are platforms deployed by academic consortiums and government bodies to allow researchers to achieve geographical and network diversity for their network research. PlanetLab [50] for instance is a platform to support development and testing of new network services [51] but is specifically not a measurement platform. In fact for many types of measurements, PlanetLab is rather unusable due to unpredictable load issues and the tendency of nodes to be located in national research networks. Measurement Lab (M-Lab) [52] on the other hand, is primarily a server infrastructure that is designed to support active measurements and facilitate exchange of large-scale measurement data. Its resource allocation policies encourage active measurement tools to utilize M-Lab servers as a sink of measurement traffic and as a repository to hold measurement results. We define such infrastructures separately as measurement facilitators and do not survey them in this work. This is to allow a more longitudinal analysis of platforms we have scoped our survey to. We also survey only currently active performance measurement

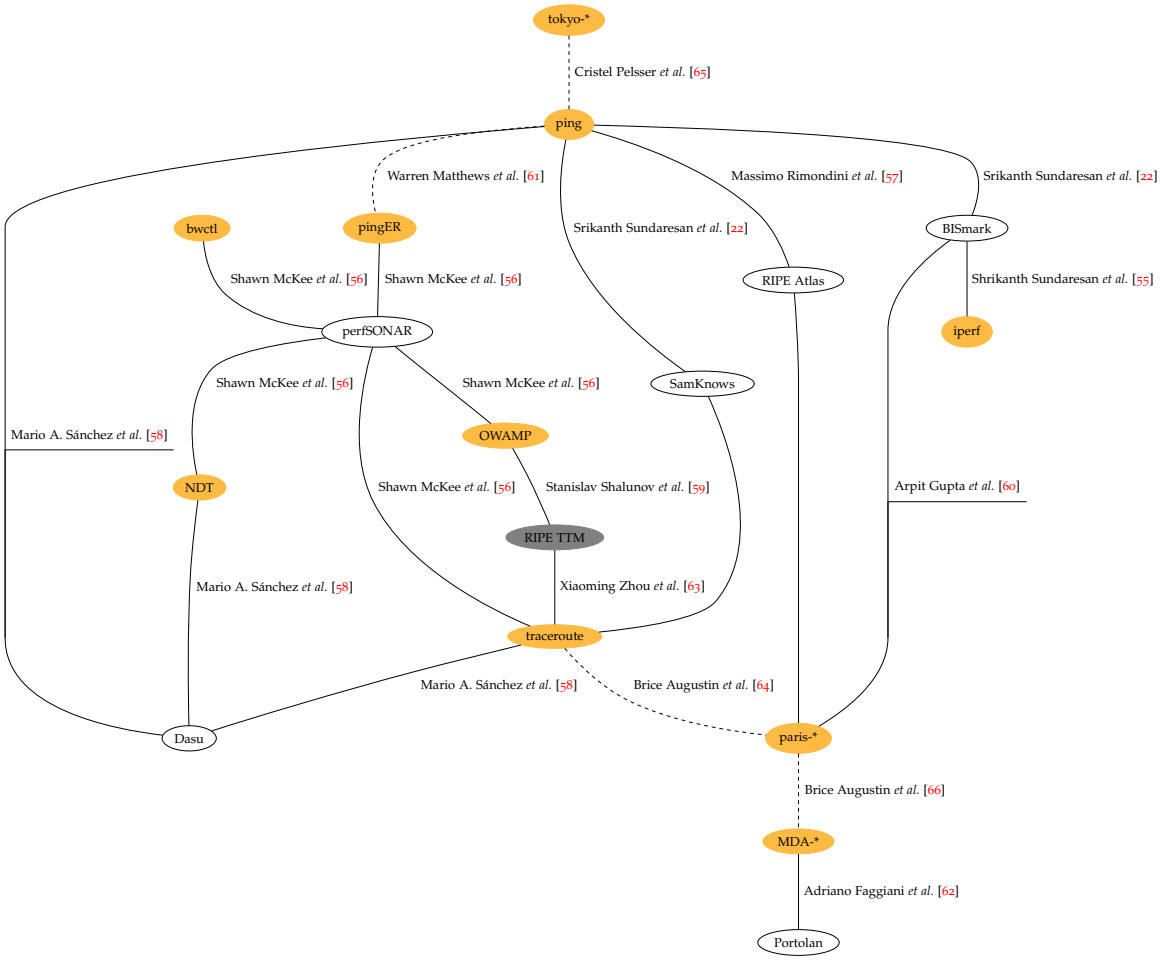


Figure 7: *A graph representing common tools (in gold) used by Internet performance measurement platforms (in white). Tools that are specifically used by only one platform are not included in this graph, but are described in the survey. Greyed out measurement platforms have been decommissioned and superseded by their successors. Dotted lines indicate an evolution of the tool along with the research paper that describes this evolution marked in labelled edges. Straight lines connect a measurement platform with a tool, along with the labelled edges that mark the research paper that describes how they use it.*

platforms. We refer the reader to [53] for a survey and a webpage [54] maintained by Cooperative Association for Internet Data Analysis (CAIDA) on measurement platforms that have existed in the past.

Fig. 6 provides a high-level overview of currently deployed Internet performance measurement platforms. We provide a taxonomy based on their deployment use-case: a) platforms deployed at the periphery of the Internet that measure performance over fixed-line access networks, b) platforms that measure performance over mobile access networks, c) platforms deployed largely within the core of the Internet that help provide network operational support. These platforms, although disparate in their scope, utilize a rather popular list of measurement tools to achieve their objectives. Fig. 7 provides a representation of common measurement tools used by the Internet performance measurement platform ecosystem.

2.1 BACKGROUND

We start with early studies that predate the performance measurement platforms era. Multiple techniques ranging from remote probing and passive monitoring to running one-off software-based probes were being employed to infer network performance. We provide a brief survey of these techniques.

The curiosity to understand the performance of the Internet from a user's vantage point led to the development of techniques that remotely probe fixed-line access networks. Marcel Dischinger *et al.* in [67] for instance, inject packet trains and use responses received from residential gateways to infer broadband link characteristics. They show that the last-mile is a bottleneck in achieving high throughput and last-mile latencies are mostly affected by large router queues. Aaron Schulman *et al.* in [68] use PlanetLab [50] vantage points to remotely send `ping` probes to measure connectivity of broadband hosts in severe weather conditions. They found that network failure rates are four times more likely during thunderstorms and two times more likely during rainy conditions in parts of the United States.

Karthik Lakshminarayanan *et al.* in [69] deployed an active measurement tool, PeerMetric to measure P2P network performance experienced by broadband hosts. Around 25 hosts volunteered across 9 geographical locations for a period of 1 month. During this period, they observed significantly assymetric throughput speeds and poor latency-based peer-selections adopted by P2P applications.

Matti Siekkinen *et al.* in [70] investigate a day long packet trace of 1300 DSL lines. They observed throughput limitations experienced by end users. On further analysis they identified the root-cause to be P2P applications that were self-imposing upload rate limits. These limits eventually were hurting download performance. In a similar study, Gregor Maier, *et al.* in [71] analyzed packet-level traces from a major European ISP covering 20K DSL customers. They used this data to study typical session durations, application mixes, TCP and performance characteristics within broadband access networks. They use the same dataset in [72] and go further to quantify Network Address Translation (NAT) deployments in residential networks. They observed that around 90% of these DSL lines were behind NAT, 10% of which had multiple hosts active at the same time.

These studies led to the development of a number of software-based solutions such as `speedtest.net` that require explicit interactions with the broadband customer. Marcel Dischinger *et al.* in [73] for instance, describe Glasnost, a tool that can help end-users detect whether the ISP implements any application blocking or throttling policies on their path. The tool was used to perform a measurement study to detect BitTorrent differentiation amongst 350K users across 5.8K ISPs. Partha Kanuparthy *et al.* in [74] describe ShaperProbe, which is a similar tool that can also help detect traffic shaping policies implemented by the ISP. Christian Kreibich *et al.* in [75], describe the `netalyzr` tool that communicates with a farm of measurement servers to probe key network performance and diagnostic parameters of the broadband user. The tool can detect outbound port filters, hidden Hypertext Transfer Protocol (HTTP) caches, DNS and NAT behaviors, path Maximum Transmission Unit (MTU), bufferbloat issues and IPv6 support. Mohan Dhawan *et al.* in [76] describe Fathom, a Firefox extension that provides a number of measurement primitives to enable development of measurement tools using Javascript. Fathom has been used to port the java applet based `netalyzr` tool into native Javascript. Lucas DiCioccio *et al.* in [77] introduce HomeNet

Profiler, a tool similar to `netalyzr` that performs measurements to collect information on a set of connected devices, running services and wireless characteristics of a home network.

The accuracy of these software-based measurement tools has recently been under scrutiny. For instance, Oana Goga *et al.* in [78] evaluate the accuracy of bandwidth estimation tools. They found that tools such as `pathload` [79] that employ optimized probing techniques can underestimate the available bandwidth capacity by more than 60%. This happens because home gateways cannot handle high-probing rates used by these methods. Another study by Weichao Li *et al.* in [80] investigates the accuracy of measurements using HTTP-based methods. They found discernible delay overheads which are not taken into account when running such measurements. These overheads also vary significantly across multiple browser implementations and make the measurements very hard to calibrate.

These inadequacies have ushered rapid deployment of measurement platforms that have specifically been designed to accurately measure broadband performance. These platforms use dedicated hardware-based probes and can run continuous measurements directly from behind a residential gateway requiring minimal end-user participation.

3 FIXED-LINE ACCESS

There are three stakeholders involved in an effort to measure performance within an access network: ISPs, consumers and regulators. Marc Linsner *et al.* in [19] enlist and describe their respective use-cases. For instance, an ISP would like to use broadband measurements to not only identify, isolate and fix problems in its access network, but also to evaluate the QoS experienced by its users. The data made public through such a measurement activity will also help the ISP benchmark its product and peek into its competitor's performance. Consumers, on the other hand, would like to use these measurements as a yardstick to confirm whether the ISP is adhering to its SLA offers. The user can also use these measurement insights to audit and diagnose network problems in its own home network. The insights resulting from these measurements are useful to network regulators. They can use them to compare multiple broadband provider offerings and frame better policies to help regulate the broadband industry.

Contents

3.1 SAMKNOWS

SamKnows is a company specializing in the deployment of hardware-based probes that performs continuous measurements to assess broadband performance. These probes are strategically [81] deployed within access networks and behind residential gateways. Fig. 8 provides an overview of the architecture of the SamKnows measurement platform.

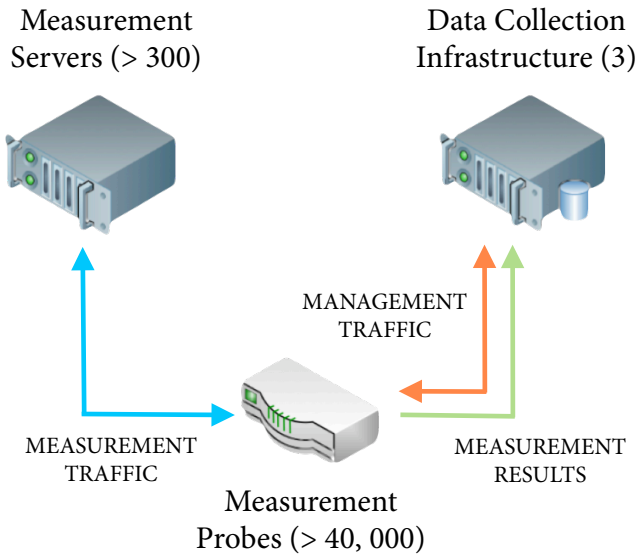


Figure 8: *An architecture of the SamKnows measurement platform. A measurement probe is managed by a Data Collection Server (DCS) from which it receives software updates and measurement schedules. Probes periodically run measurements against custom SamKnows measurement servers. Measurement results are pushed to nearby DCS on an hourly window:* `http://goo.gl/mh5Qu7`

.

3.1.1 *Scale, Coverage and Timeline*

SamKnows started in 2008, and in seven years they have deployed around 70K probes all around the globe. These probes have been deployed in close collaborations with 12 ISPs and 6 regulators: a) FCC, United States, b) European Commission (EC), European Union, c) Canadian Radio-Television Commission (CRTC), Canada, d) Ofcom, United Kingdom, e) Brazilian Agency of Telecommunications (Anatel), Brazil, f) Infocomm Development Authority of Singapore (IDA), Singapore.

3.1.2 *Hardware*

The probes are typical off-the-shelf TP-Link router devices (with earlier generations using Linksys, Netgear, and PC Engines hardware) that have been flashed with a custom snapshot of OpenWrt firmware. The firmware has been made open-source with a GPL licence [82]. The probes function only as an ethernet bridge and all routing functionality has been stripped off the firmware. The wireless radio is used to monitor the cross-traffic to make sure active measurements are only run when the user is not aggressively using the network. The probe never associates to any wireless access point. As such, there is no IP-level configuration provisioned on the wireless port. Due to privacy concerns, the probe neither runs any passive measurements nor does it ever look into the user's traffic crossing the network.

3.1.3 *Metrics and Tools*

Probes typically measure end-to-end latency, last-mile latency, latency-under-load, forward path, end-to-end packet loss, upstream and downstream throughput and goodput, end-to-end jitter, network availability, webpage download, Voice over IP (VoIP), Peer to Peer (P2P), DNS resolution, email relays, File Transfer Protocol (FTP) and video streaming performance. The raw measurement results sent by the probes are archived in geographically distributed and sharded MySQL instances. Hourly, daily and weekly summaries of the data are precomputed and stored in MySQL as well, to allow for rapid generation of reports. On specific measurement panels, where measurements are conducted in close collaboration with the ISP, the results are also validated against service-tier information. The obtained measurement reports are viewable via the SamKnows performance monitoring dashboards [83]. Hosts also receive monthly email report cards giving an overview of their broadband performance. iOS [84] and Android [85] smartphone apps have been released for Brazil, Europe and US regions.

3.1.4 *Architecture*

The active measurement tests and their schedules are remotely upgradeable by the Data Collection Server (DCS). The DCS functions both as a controller and as a measurement collector. The communication with the DCS is only server-side authenticated and encrypted over TLS. Probes typically measure against a custom SamKnows measurement server. These are servers that only respond to measurement traffic and do not store any measurement results. There are around 300 such measurement servers deployed around the globe. The locality of these servers is critical to the customer, and therefore Round-Trip Time (RTT) checks are periodically made by the probe to make sure that the probe is measuring against the nearest measurement server. Measurement servers can either be deployed within the ISP (called on-net test nodes) or outside the access network (called off-net test nodes).

3.1.5 *Research Impact*

Ofcom and FCC regularly publish their regulator reports on broadband performance using the SamKnows platform. These publicly available datasets have actively been utilized in multiple studies. Steven Bauer *et al.* in [86] for instance, use the FCC dataset to measure the subtle effects of Powerboost. They show how the scheduling of measurement tests needs to be improved to make sure different tests remain independent. They also show how the warm-up period used in the SamKnows throughput test needs a fair treatment to take the Powerboost effects into account. Zachary S. Bischof *et al.* in [87] demonstrate the feasibility of crowdsourced ISP characterization through data gathered from BitTorrent users. They used the Ofcom dataset to compare and validate their results. Zachary S. Bischof *et al.* in [88] go further to show how BitTorrent data can be used to accurately estimate latency and bandwidth performance indicators of a user's broadband connection. They used the FCC dataset to validate their results for users in the AT&T network. Giacomo Bernardi *et al.* in [89] describe `BSense`, a software-based broadband mapping framework. They compare their results by running a `BSense` agent from a user's home that also participates in SamKnows broadband measurements. They performed evaluation for a period of two-weeks and

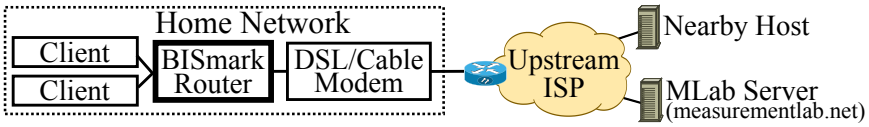


Figure 9: *An architecture of the BISmark measurement platform. A measurement probe is wired behind a DSL or a cable modem. The probe can run both active and passive measurements. Measurement servers are source/sinks of measurement traffic. They are primarily M-Lab servers. A management server is used to remotely administer probes and collect measurement results [92].*

obtained comparable results. Igor Canadi *et al.* in [24] use the crowd-sourced data from `speedtest.net` to measure broadband performance. They use the FCC dataset to validate their results. Daniel Genin *et al.* in [90] use the FCC dataset to study the distribution of congestion in broadband networks. They found that DSL networks suffer from congestion primarily in the last-mile. Cable networks on the other hand are congested elsewhere, and with a higher variability. Vaibhav Bajpai *et al.* in [9] deploy SamKnows probes within dual-stacked networks to measure TCP connection establishment times to a number of popular services. They observed that websites clustering behind CDN deployments are different for IPv4 and IPv6. Using these clusters they show how CDN caches are largely absent over IPv6. They go further in [91] where they study effects of the happy eyeballs algorithm. They show how a 300ms advantage imparted by the algorithm leaves 1% chance for a client to prefer connections over IPv4. They show how this preference impacts user experience in situations where an IPv6 happy eyeballed winner is slower than IPv4. Saba Ahsan *et al.* take this further in [10] to show how TCP connection establishment times to YouTube media servers makes the happy eyeballs algorithm prefer a connection over IPv6 even when the measured throughput over IPv4 is better. This results in lower bit rates and lower resolutions when streaming a video than can be achieved if streamed over IPv4. They show how this is due to the disparity in the availability of YouTube content caches which are largely absent over IPv6.

3.2 BISMARK

BISmark [18] is an initiative by Georgia Tech to develop an OpenWrt-based platform for broadband performance measurement. The platform is similar to SamKnows as shown in Fig. 9. The probes primarily run active measurements. Passive measurements, however, can be enabled on a case by case basis by providing written consents. This is necessary to ensure volunteers are aware of the risk of exposing personally identifiable information.

3.2.1 *Scale, Coverage and Timeline*

BISmark started in 2010 and in five years they have deployed around 420 measurement probes on a global scale. Although more than 50% of the probes are deployed in developed countries, a significant effort has recently been made to increase the geographical diversity of the platform as shown in Fig. 10. A real-time snapshot of the coverage is also available on the network dashboard [93].

Figure 10: *The coverage of the BISmark measurement platform as of Feb 2015. The green and red dots represent connected (around 119) and disconnected probes respectively:* `http://networkdashboard.org`.

3.2.2 *Hardware*

BISmark uses off-the-shelf Netgear routers that have been custom flashed with an OpenWrt firmware. The firmwares run a measurement overlay that is composed of a number of active measurement tools and scripts that have been packaged by the BISmark team. The entire BISmark software-suite has been open-sourced through a GPL v2 licence [94]. The probe unlike that of a SamKnows probe is a full-fledged router. The probe by default provides wireless access points on both 2.4 GHz and 5 GHz radio interfaces.

3.2.3 *Metrics and Tools*

The probes support both active and passive measurements. All probes actively measure end-to-end latency, last-mile latency, latency under load, end-to-end packet loss, access-link capacity, upstream and downstream throughput, and end-to-end jitter. Occasionally, they also send special heartbeat packets to report their online status and uptime information to BISmark management servers. The metrics are measured using popular specialized tools. For instance, probes run ShaperProbe [74] to measure the access link capacity, `iperf` to measure the upstream and downstream throughput, D-ITG [95] to measure jitter and packet loss, `paris-traceroute` [64] to measure forward and reverse path between probes and M-Lab servers, and Mirage [28] to measure the webpage load time. On explicit volunteer consent, probes can also run some passive measurements. For instance, probes can count the number of wired devices, devices associated on a wireless link, and number of wireless access points in the vicinity. Probes also passively measure packet and flow statistics, DNS responses and Media Access Control (MAC) addresses. The obtained measurement results and overall statistics are available via the network dashboard.

3.2.4 *Architecture*

The BISmark architecture consists of measurement probes, a management server and several measurement servers. The management server functions

both as a controller and as a measurement collector. Measurement servers are strategically deployed targets used by active measurement tools. These are primarily M-Lab servers hosted by Google. The measurement probe periodically sends User Datagram Protocol (UDP) control packets to the controller. This punches a hole in the gateway's NAT and allows the controller to push configuration and software updates.

3.2.5 *Research Impact*

Srikanth Sundaresan *et al.* in [96] use the BISmark platform to identify a collection of metrics that affect the performance experienced by a broadband user. They show that such a *nutrition label* provides more comprehensive information, and must be thus advertised by an ISP in its service plans to increase transparency. Hyojoon Kim *et al.* in [97] use the BISmark platform to demonstrate how broadband users can monitor and manage their usage caps. It proposes an OpenFlow control channel to enforce usage policies on users, applications and devices. Srikanth Sundaresan *et al.* in [22, 55] use the BISmark platform to investigate the throughput and latency of access network links across multiple ISPs in the United States. They analyze this data together with data publicly available from the SamKnows/FCC study to investigate different traffic shaping policies enforced by ISPs and to understand the bufferbloat phenomenon. Swati Roy *et al.* in [98] use the BISmark platform to measure end-to-end latencies to M-Lab servers and Google's anycast DNS service. They propose an algorithm to correlate latency anomalies to subsets of the network path responsible for inducing such changes. They observed low last-mile latency issues, with higher middle-mile issues in developing regions, indicating scope of improvement along peering links. Srikanth Sundaresan *et al.* in [28, 99, 100] use the BISmark platform to measure web performance bottlenecks using Mirage, a command-line web performance tool. They show that latency is a bottleneck in access networks where throughput rates exceed 16Mbits/s. They also show how last-mile latency is a significant contributor both to DNS lookup times and time to first byte. They demonstrate how these bottlenecks can be mitigated by up to 53% by implementing DNS and TCP connection caching and prefetching on a residential gateway. Sarthak Grover *et al.* in [92] use the BISmark platform to perform a longitudinal measurement study on home network properties. They use continuously running active and passive measurements to study home network availability, infrastructure and usage patterns. They show how network usage behavior patterns differ across countries in developed and developing regions, how the 2.4 GHz wireless spectrum is significantly more crowded (specially in developed countries) when compared to the 5 GHz wireless spectrum, and how majority of the home traffic is destined to only few destinations. Marshini Chetty *et al.* in [101] use the BISmark platform to measure fixed and mobile broadband performance in South Africa. They show how broadband users do not get advertised rates, how throughputs achievable on mobile networks are higher when compared to fixed networks, and how latency to popular web services is generally high. Arpit Gupta *et al.* in [60] go further and study ISP peering connectivities in Africa. Using `paris-traceroute` they show how local paths detour via remote Internet Exchange Point (IXP)s in Europe leading to increased latencies to popular web services. They also show how ISPs either are not present or do not peer at local IXPs due to economic disincentives. Srikanth Sundaresan *et al.* in [18] reflect upon the success of BISmark by discussing design decisions faced

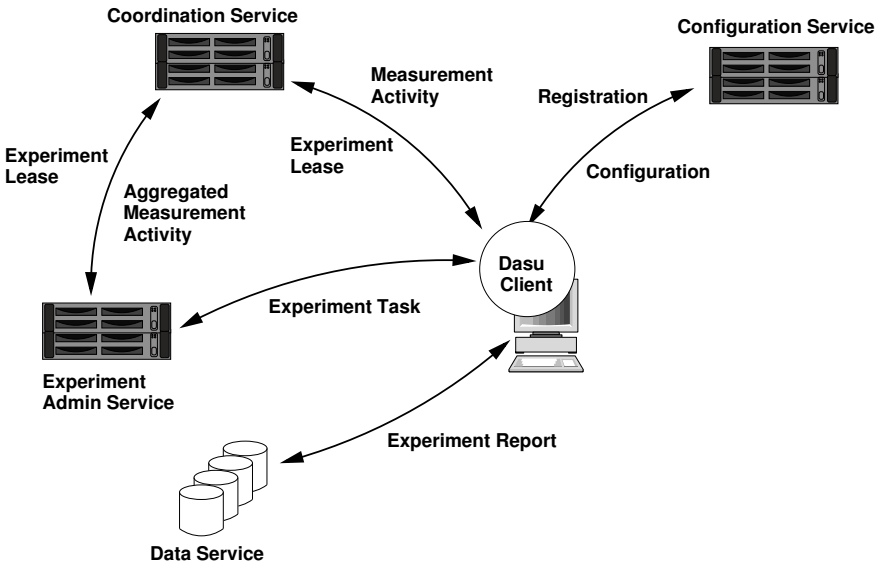


Figure 11: *An architecture of the Dasu measurement platform. A client on startup registers with a coordination service to retrieve configuration settings and the location of the measurement collecter. The client periodically contacts the EA service to retrieve a set of assigned measurement tasks. Once the tasks are assigned, the client contacts the coordination service to pick up a lease to start measurements. Measurement results are eventually pushed to the data service. The configuration, coordination and EA service together function as a controller, while the data service functions as a measurement collector [58].*

during the implementation work. A summary of research projects using this platform and on-going experiments are enumerated. Lessons learned during the four-year deployment effort are also described. Srikanth Sundaresan *et al.* in [102] use passively collected packet traces from a subset of BISmark probes to study the relationship between wireless and TCP performance metrics on user traffic. They show how with an increase in access link capacity, wireless performance starts to play an increasing role on achievable TCP throughput. They show how the wireless performance is affected more over the 2.4 GHz spectrum (when compared with 5 GHz spectrum) where the latency impacts are worse with higher retransmission rates. They also show how latency inside a home wireless network contributes significantly towards end-to-end latency.

3.3 DASU

Dasu is an initiative by the Northwestern University to develop a software-based measurement platform that allows network experimentation from the Internet's edge. The platform started with an objective to perform broadband characterization from home, but it has evolved into facilitating end-users to identify service levels offered by their ISP. Fig. 11 provides an architecture of the Dasu measurement platform. The platform allows clients to run both active and passive measurements.

3.3.1 *Scale, Coverage and Timeline*

Dasu started in 2010 and in five years they have around 100K users connected behind around 1.8K service networks. These users are located around the globe and span around 166 countries as shown in Fig. 12.

3.3.2 *Hardware*

Dasu is a software plugin that hooks into the Vuze/Azureus BitTorrent client application. Vuze is chosen for its increasing popularity and its modular architecture that easily allows installation of third-party plugins. Vuze also seamlessly handles software updates for installed plugins. For users that do not use BitTorrent, a standalone client is also available online in its current beta stage [103]. The platform prefers a software-based approach to not only eliminate the cost factor involved in deployed hardware probes, but also to increase the control, flexibility and low-barrier to adoption of software-based models.

3.3.3 *Metrics and Tools*

The platform allows the clients to perform both active and passive measurements. The BitTorrent plugin passively collects per-torrent (number of TCP resets, upload and download rates), application-wide (number of active torrents, upload and download rates) and system-wide statistics (number of active, failed, and closed TCP connections). The client is composed of multiple probe modules that allow active measurements. These probe modules actively measure end-to-end latency, forwarding path, HTTP GET, DNS resolution and upstream and downstream throughput. `ping` is used to measure end-to-end latency, `traceroute` for capturing the forwarding path and Network Diagnostic Tool (NDT) to measure upstream and downstream throughput. Active measurements are scheduled using a `cron`-like scheduler. All the clients synchronize their clocks using Network Time Protocol (NTP). This allows synchronization of a task that covers multiple clients. To allow a finer synchronization, clients can establish a persistent TCP connection to the coordination server. Each measurement runs in its own Java Virtual Machine (JVM) sandboxed environment with a security manager that applies policies similar to those applied to unsigned Java applets. The configuration files sent by the server are digitally signed. All client-server communications are also encrypted over a secure channel. The client also monitors resources such as CPU, network bandwidth, memory and disk usage to make sure measurements only run when the resource utilization is below a certain threshold. The client employs watchdog timers to control CPU utilization. It uses `netstat` to monitor the network activity and couples it with the maximum bandwidth capacity estimate retrieved from NDT to control bandwidth utilization. It also assigns quota limits to control memory and disk space utilization.

3.3.4 *Architecture*

The Dasu architecture consists of a distributed collection of clients, a measurement controller composed of the configuration, coordination, and Experiment Admin (EA) service and a measurement collector called the data service. A client on bootstrap registers with a configuration service to retrieve a set of

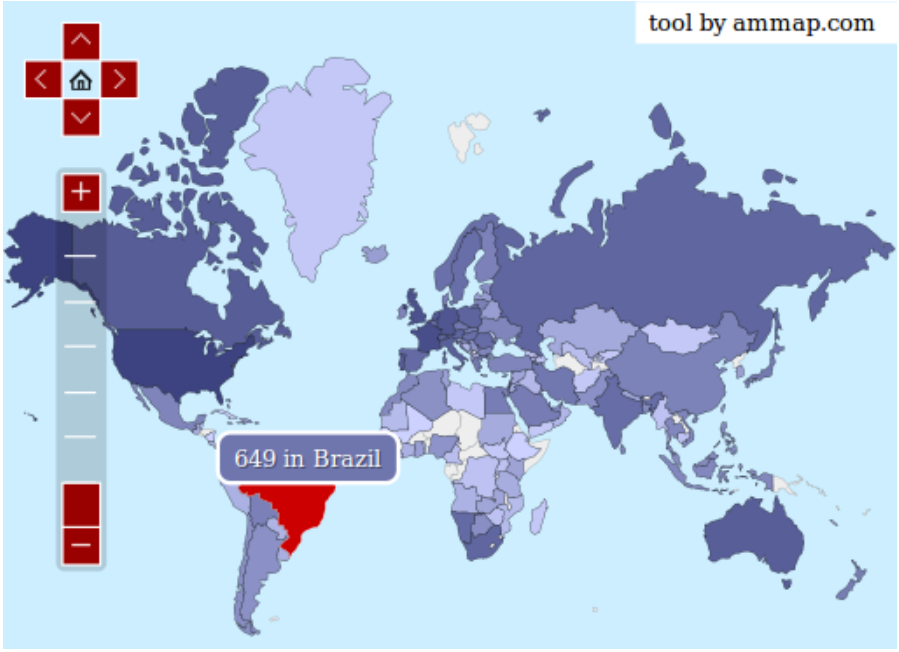


Figure 12: *The network coverage of the Dasu measurement platform as of Feb 2015. The different shades of blue indicate the number of clients participating in the measurement:* `http://goo.gl/nqshJM`

.

configuration settings. These settings assign duration and frequency of measurement operations and instruct which coordination and data service must this client use in future interactions. The client periodically polls the EA service to retrieve measurement tasks. The measurement tasks are defined using a rule-based declarative model. A set of rules describe a program, while a set of programs form a measurement task. The EA service assigns measurement tasks to clients based on the requirements and client characteristics. The client must pickup a lease from the coordination service before it can start measurements for an assigned task. Leases are used to ensure fine-grained control of the measurement infrastructure. Leases grant budgets, which are upper bounds on the number of measurement queries a client can run at specific point in time. These budgets are elastic and can vary dynamically depending on the aggregated load of the measurement infrastructure. The EA service is composed of a primary EA server and several secondary EA servers. The primary EA service ensures that the aggregated measurement activity is within defined bounds. This is used to set values for the elastic budgets for specific leases. Secondary EA services then are responsible for allocating these leases to the coordination service. The coordination service hands out these leases to clients when they contact them. The coordination service runs on top of the PlanetLab infrastructure to ensure replication and high availability. The collected measurement results are finally pushed to the data service.

3.3.5 *Research Impact*

Mario A. Sánchez *et al.* in [104] introduce Dasu as a platform that can crowdsource ISP characterization from the Internet's edge. They describe how it can

capture end user's view by passively monitoring user-generated BitTorrent traffic from the host application. They specifically show how measurement rule specifications are defined and how they trigger measurement tests from within the client application. Zachary S. Bischof *et al.* in [87] demonstrate the feasibility of this approch by analyzing data gathered from 500K BitTorrent users. They show how this data can be used to a) infer service levels offered by the ISP, b) measure the diversity of broadband performance across and within regions of service, c) observe diurnal patterns in achieved throughput rates, d) measure visibility of DNS outage events, and e) relatively compare broadband performance across ISPs. They used the SamKnows/Ofcom dataset to compare and validate their results. They go further in [88] to show how this approach can be used to accurately estimate latency and bandwidth performance indicators of a user's broadband connection. They measure last-mile latencies of AT&T subscribers and validate their results using the SamKnows/FCC dataset. They also validate the soundedness of their throughput measurements by comparing BitTorrent throughputs against those obtained by the NDT tool. Mario A. Sánchez *et al.* in [58, 105] describe the design and implementation of the platform alongwith a coverage characterization of its current deployment. They use the platform to present three case studies: a) measuring Autonomous System (AS)-level asymmetries between Dasu and PlanetLab nodes, b) studying prefix-based peering arrangements to infer AS-level connectivities, and c) measuring the performance benefits of DNS extensions. They go further in [106] to leverage Universal Plug and Play (UPnP) to study home device characteristics from 13K home users. They use the Digital Living Network Alliance (DLNA) specification to further categorize the UPnP devices. They also utilize received traffic counters and couple them with the data collected through their client's passive monitoring tools to identify whether the cross-traffic originates locally from another application or from entirely another device. Zachary S. Bischof *et al.* in [107] use a 23-months long Dasu and SamKnows/FCC dataset to study broadband markets; particularly the relationship between broadband connection characeteritics, service retail prices and user demands. They show how the increase in broadband traffic is driven more by increasing service capacities and broadband subscriptions, and less by user demands to move up to a higher service-tiers. They also find a strong correlation between capacity and user demands and show how the relationship tends to follow the law of diminishing returns.

4 MOBILE ACCESS

A number of platforms have recently emerged that specifically focus on measuring performance in mobile access networks. The challenges faced by these platforms are very different from platforms that operate on fixed-line networks. Factors such as signal strength, device type, radio type, frequency of handovers and positioning information of cellular devices need to be taken into account when doing measurements. The service plans on these mobile devices are also very restrictive, and measurements need to ensure that they take usage caps into account when generating network traffic. Additionally the measurements run on top of cellular devices. These devices are not homogenous, but rather run varying flavors of mobile operating systems. The measurement overlay needs to specifically be developed for each mobile platform.

Contents

4.1 NETRADAR

Netradar is a mobile measurement platform operated by Aalto University. The objective is not just to run tests and present measurement results to the end-user, but also to provide an automated reasoning of the perceived results. Towards this end, Netradar runs measurements that cover a wide-range of key network performance indicators to be able to do analysis that can provide a rationale behind the observations.

4.1.1 *History*

Netradar is a successor to the Finish specific mobile measurement platform, Nettitutka [108]. Nettitutka started in early 2011. The platform was designed to serve the local user population in Finland, and therefore measurements were targeted to a single server located within the Finnish University and Research Network (FUNET). With the increasing popularity of the platform, Nettitutka has been replaced by Netradar.

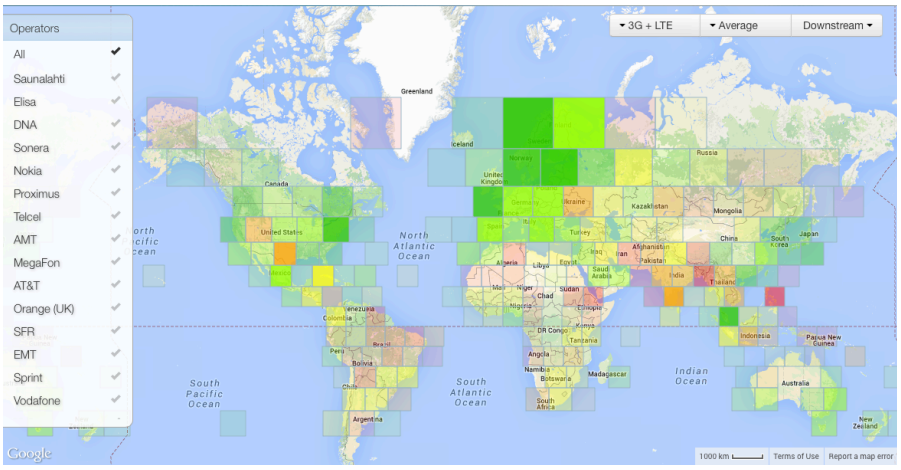


Figure 13: *The coverage of the Netradar measurements as of Feb 2015. The quality is measured based on network download and upload speeds, latency and signal strength:* `https://goo.gl/NVwNmP`. *The threshold intervals used to define different colors on the map are described here:* `https://goo.gl/TzgAjQ`

.

4.1.2 *Scale, Coverage and Timeline*

Netradar started in 2012 and in three years they have performed around 3.8M measurements from mobile devices. The client itself has been installed 150K times on a wide variety of (around 5K) mobile handsets. Fig. 13 shows the geographical coverage of these measurements.

4.1.3 *Hardware*

The Netradar measurement platform is a software client that can be installed on bare-bones smartphone devices. The client is available for Google Android, Apple iOS, Nokia Meego, Symbian, BlackBerry, Microsoft Windows and Sailfish phones. The measurement capability of each platform is identical with minor differences. For instance, iOS does not expose signal strength details that can be utilized by the Netradar platform.

4.1.4 *Metrics and Tools*

Netradar performs both active and passive measurements. Passive measurements report parameters such as signal strength, operating system, device type, radio type, positioning information, handovers using base station ID, and vendor information. Active measurements include measuring latency and TCP goodput using upload and download speed tests. Handovers, signal strength and location information are also measured during an active measurement. Each measurement tags measurement result with timestamps at millisecond resolution. The speed test measurements are run for 10 seconds on a single TCP connection against the closest Netradar measurement server. The speed test results are stored with a resolution of 50ms. The speed test also skips the first 5 seconds as a warmup phase to skip TCP slow-start. Internet disconnectivity is also recorded to map the distribution of best-connectivity areas. Netradar uses GPS, wireless, cellular, and IP address information to accurately map the positioning information of a device. The latency mea-

surements run over UDP both before and after a speed test measurement. Netradar also uses TCP statistics to store RTT values during the speed test measurement.

4.1.5 *Architecture*

Netradar relies on a client-server based architecture. Servers are measurement targets that are deployed in the cloud and globally distributed. Clients measure against closest measurement servers. The measurement result databases and web servers are replicated to achieve scalability. The number of instances are scaled by real-time monitoring of server load. The number of simultaneous connections to a server instance is also limited by a threshold.

4.1.6 *Research Impact*

Sebastian Sonntag *et al.* in [109] use the Netradar platform to study various parameters that affect bandwidth measurements in mobile devices. They show how the used radio technology and signal strength are the most significant factors affecting bandwidth. They also describe how the bandwidth is cut by a third, due to poor provisioning and congestion at the cell tower. The device type and frequency of handovers are also limiting factors. They go further in [110] to study the correlation between signal strength and other network parameters. They show how signal strength has low correlation to TCP goodput. They show how taking the time of the day and motion speed parameters into account still does not increase this correlation. As such, coverage maps drawn using signal strength as a parameter are limited. They provide recommendations on the tile size and on using TCP goodput as a parameter for drawing these coverage maps. Le Wang *et al.* in [111] show how the energy consumption of mobile devices is suboptimal when browsing web content both over wireless and cellular networks. They present an energy-efficient proxy system, that utilizes bundling of web content, Radio Resource Control (RRC) state based header compression and selective content compression to reduce the operating power of mobile devices during web access.

4.2 PORTOLAN

Portolan is a crowd-sourced mobile measurement platform operated by the University of Pisa and the Informatics and Telematics Institute of the Italian National Research Council. The objective is twofold: a) provide a comprehensive mapping of the signal strength coverage over the globe and b) facilitate topology mapping efforts at the AS-level by contributing measurements from mobile devices. Fig. 14 provides an overview of the architecture of the Portolan measurement platform.

4.2.1 *Scale, Coverage and Timeline*

Portolan started in 2012 and in three years they have around 300 active users all around the globe as shown in Fig. 15. The concentration is higher in Italy from where the platform originated.

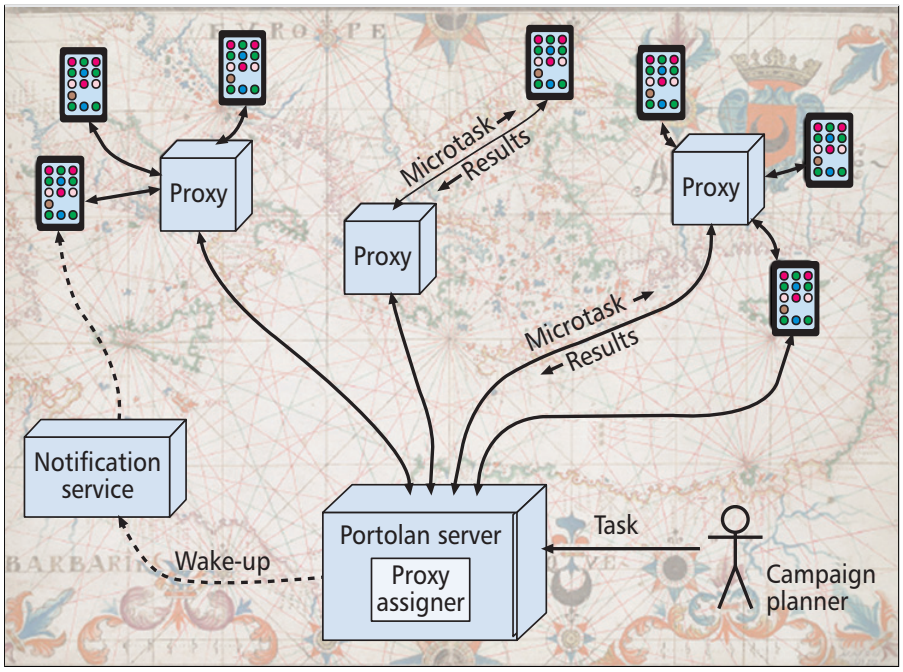


Figure 14: *The architecture of the Portolan measurement platform. A human prepares a XML specification of a measurement campaign and deploys it on a central server. The server validates the specification and bifurcates it into a set of microtasks. Microtasks are handed out to regional proxies who mediate the deployment of measurement instructions and collection of results between mobile devices and the central server [112].*

4.2.2 *Hardware*

The Portolan measurement platform utilizes a software client that one can install on stock smartphone devices. It currently supports Google Android, however a client for Apple iOS is in the works. The client itself has received around 8 version releases [113]. The client treats the mobile device as a sensor that can measure network-related properties. The client is therefore subdivided into multiple measurement subsystems. Each subsystem measures a particular network property and is described using a SensorML specification [114].

4.2.3 *Metrics and Tools*

The platform supports both active and passive measurements. It actively measures latency, forwarding path (both at the IP and AS level), and achievable bandwidth. It passively scans available wireless networks, signal strength and cell coverage. It also periodically runs a traffic shaping detection tool to check if your bittorrent traffic is treated differently. Portolan uses `SmartProbe` [115] to measure the achievable bandwidth and `MDA-traceroute` [66] to capture the forwarding path. The implementation has been modified to utilize UDP-based probing using the `IP_RECVERR` socket option to perform `traceroute` measurements without superuser privileges. It is also made multi-threaded to utilize multiple sockets to parallelize the probing operation. These adaptations however limit the possibility of performing fingerprinting-based alias-resolution on the client side. As such, alias-resolution is performed

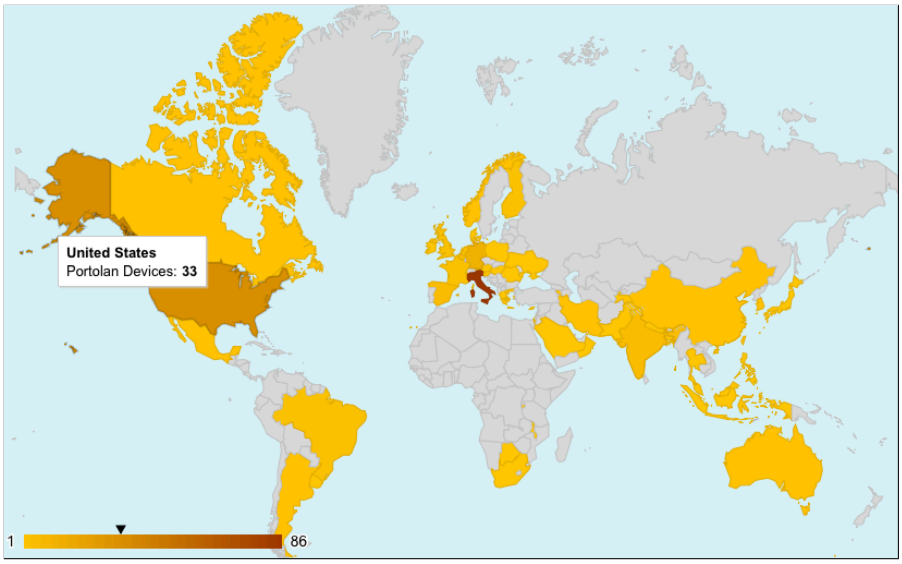


Figure 15: *The network coverage of the Portolan measurement platform as of Oct 2014. The different shades of brown indicate the number of clients participating in the measurement:* `http://portolan.iet.unipi.it`.

in a post-processing stage by the server. Not more than 200 measurements are run per day. This limitation is enforced to ensure that Portolan does not consume roughly more than 2MB/day on `traceroute` measurements. The signal strength results must be geo-referenced using the device's Global Positioning System (GPS). In order to avoid draining the battery, Portolan does not actively enable the GPS but waits to reuse the location information when the user (or an application started by the user) enables it. Portolan suspends all activity when the battery level goes below 40%. The server-side components are written as Java Servlets running on Apache Tomcat.

4.2.4 *Architecture*

Portolan is based on a centralized architecture. A central server acts both as a controller and as a measurement collector. However, in order to achieve scalability, a number of regional proxies have been deployed to mediate the deployment of measurement instructions and retrieval of measurement results from a set of geographically clustered mobile devices. Proxies are deployed at a country-level resolution, given mobile devices tend to show a quasi-static behavior at this granularity. Each mobile device is identified in the system using a pseudo-randomly generated ID. These IDs are assigned to a regional proxy by a proxy assigner implemented within the central server. A measurement campaign is formally described in a Extensible Markup Language (XML) specification by a human and submitted to the central server, where it is validated and decomposed into a set of loosely-coupled instructions, called *microtasks*. These microtasks are then shipped to regional proxies for local deployment. The microtasks are pulled (and not pushed) by mobile devices. This call-home mechanism allows devices to traverse the NAT. However high-priority microtasks can also be directly pushed to devices by the central server. The server uses the Google Cloud Messaging (GCM) service as a notification service to push high-priority microtasks as network events. The notification service is also used to tune device polling

intervals to adapt to the number of the devices associated with a regional proxy. The XML specification of a measurement consists of the type of metric, source and target destination lists, duration, metric parameters and an urgent flag. The validation of the specification is performed using the Sensor Planning Service (SPS) component, while the Sensor Observation Service (SOS) component is used to retrieve measurement results. These components are standards specified within the Sensor Web Enablement (SWE) framework [116]. The polling beacon messages piggyback the device's location, IP address, battery status, network load and base station ID. Regional proxies use this as a guideline to choose mobile devices for a specific microtask.

4.2.5 *Research Impact*

Adriano Faggiani *et al.* in [62] present their idea on smartphone-based crowdsourced measurements. They describe the design of such a measurement system, alongwith details on the implementation and validation of running `MDA-traceroute` measurements from an Android device. Enrico Gregori *et al.* in [114] describe the implementation of the Portolan measurement platform alongwith preliminary results. They present how they use standards defined in the SWE framework to treat mobile devices as sensors to provision measurement tasks and retrieve measurement results. They perform a preliminary study on measuring the AS-level topology using this platform. They run validations using ground-truth data obtained from network operators, and evaluate their results against publicly available AS topology datasets. Francesco Disperati *et al.* in [115] present SmartProbe, a link capacity estimation tool that is tailored for mobile devices. It is an adaptation of the packet-train based tool, PBProbe [117], for wireless and wired networks. Portolan uses it to measure acheivable bandwidth from mobile devices. Adriano Faggiani *et al.* in [113] share their experiences in building such a measurement platform. The challenges involve factors such as human involvement in a control loop, limited resources of mobile devices, handling big data, and motivating users to participate in measurements. They go further in [112] to describe their motivation behind choosing a crowdsourced-based monitoring approach. They illustrate opportunities and challenges that come with this approach, alongwith use-case scenarios where this could prove beneficial. They briefly describe the measurement platform with measurement results.

5 OPERATIONAL SUPPORT

A number of Internet performance measurement platforms have been deployed with the goal to provide operational support to network operators. These platforms are being utilized by the operators to help diagnose and troubleshoot their network infrastructure. A large number of the probes within these platforms are therefore not deployed at the edge but within the core of the Internet.

Contents

5.1 RIPE ATLAS

RIPE Atlas is a measurement infrastructure deployed by the RIPE Network Coordination Centre (RIPE NCC). It consists of thousands of hardware probes distributed all around the globe. These probes specifically perform only active measurements. The infrastructure has been designed with a goal to provide operational support to Local Internet Registry (LIR)s. Fig. 16 provides an overview of the architecture of the RIPE Atlas measurement platform.

5.1.1 *History*

RIPE Atlas is a successor to the RIPE Test Traffic Measurement Service (TTM). RIPE TTM is a legacy measurement platform that started in 1997 [118] and was designed to provide standardized measurements for one-way delay and one-way packet loss between probes. The platform had around 100 TTM boxes [119] distributed globally as shown in Fig. 17. The probes continuously measured one-way delay, packet loss, jitter, root-nameserver reachability, routing statistics, GPS satellite conditions and PMTU discovery. In addition, each TTM box was running `traceroute` measurements to one another. The platform was decommissioned on 1st July 2014 in favour of the RIPE Atlas platform.

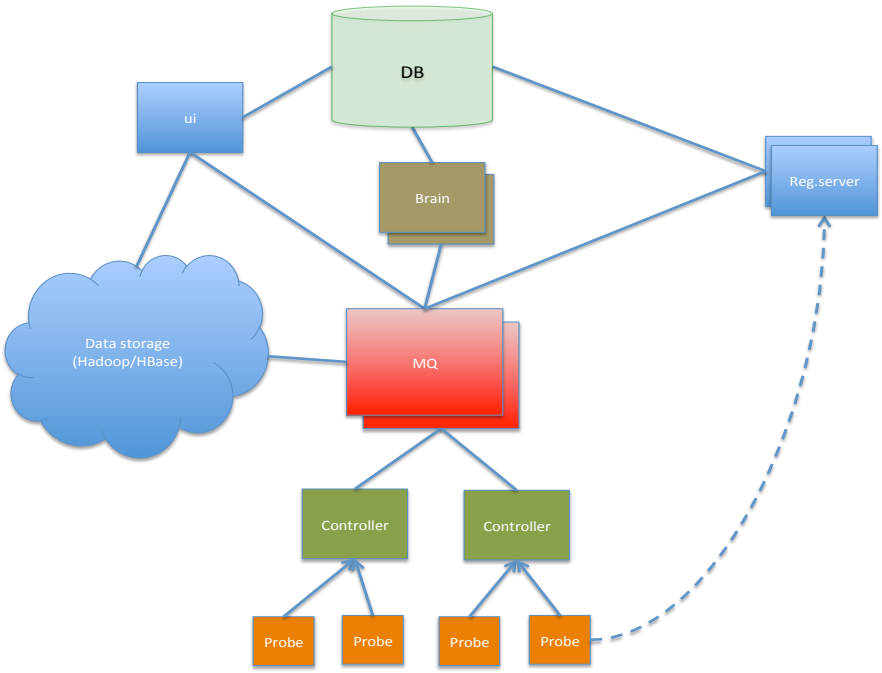


Figure 16: *The architecture of the RIPE Atlas platform. A measurement probe on bootstrap learns about the location of its controller by securely connecting to a registration server. The controller on receiving the initial request sends measurement schedules and software updates to the probe. The probe ships the measurement results to the controller. The brain supplements the results with information from third-party sources. The aggregated results are queued up to be later processed by Hadoop jobs and archived in HBase stores:* `http://goo.gl/X8C8GG`

5.1.2 *Scale, Coverage and Timeline*

RIPE Atlas started in 2010 [120] and in five years RIPE has deployed around 12K hardware probes all around the globe as shown in Fig. 18. A large number of these probes have been deployed by network operators in their internal network. These probes are situated within access networks and at the core. A discernible number of enthusiasts do volunteer to host a probe at their home. As as result, quite a number of probes are also connected behind a residential gateway.

5.1.3 *Hardware*

The hardware probes have evolved over the years. The first and second generations were a custom hardware built around a Lantronix XPort Pro module. The limitations of the hardware led to a third generation probe running on top of an off-the-shelf TP-Link wireless router. Although the third generation is much more capable than the previous iterations, the firmware running on all the three variants is exactly the same. The measurement firmware runs on top of OpenWrt and has been open-sourced with a GPLv2 licence [121]. All wireless capabilities have been stripped off the firmware for privacy reasons. In addition to the probes, RIPE also deploys RIPE Atlas anchors [122]. Anchors are dedicated servers running the RIPE Atlas firmware. Fig. 19 shows the deployment coverage of these anchors. Anchors can serve both

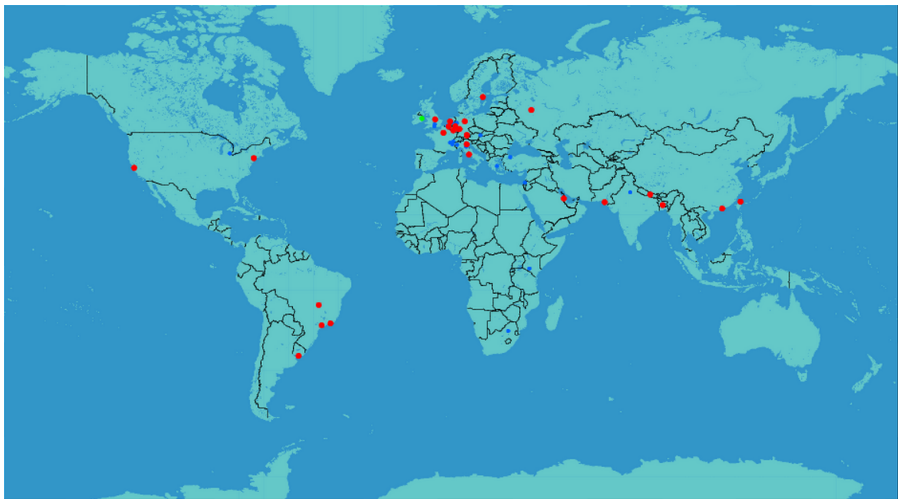


Figure 17: *The coverage of the legacy RIPE TTM measurement platform as of Feb 2015. The red dots represent active probes:* `http://goo.gl/2lVqHS`

as a source and sink of measurement traffic. Anchors when acting as probes can run a large number of measurements in parallel. The regular probes can also schedule measurements targetted to these anchors, which serve as powerful targets to handle a large number of measurement requests. This way, anchors help provide information on regional connectivity and reachability. The RIPE NCC also periodically schedules baseline measurement to an anchor, called *anchoring measurements* from a batch of several hundred regular probes and every other anchor to continously measure regional reachability.

5.1.4 *Metrics and Tools*

The probes only run active measurements [123]. They perform RTT, `traceroute`, HTTP and Secure Sockets Layer (SSL) queries to a number of preconfigured destinations as built-in measurements. They also specifically run RTT measurements to the first and second hop alongside DNS queries to DNS root servers. All of these built-in measurements are run both over IPv4 and IPv6. The probes also send their local uptime, total uptime, uptime history and current network configuration information periodically to measurement controllers. The measurement tools are adaptations of the standard UNIX utilities available in busybox. The measurement code has been modified to make measurements run in an event-driven manner using `libevent` and to make them output the measurement results in JavaScript Object Notation (JSON) format. These modifications have resulted in: `evping`, `evtraceroute`, `evtdig` and `evhttpget`. The platform also includes an evented scheduler, `eperd`, which is similar to `cron` but with added capabilities: a) The scheduler in addition to the start time, can also take a stop time and runtime frequency of a test, b) it also adds jitter to make sure not all measurements start running at the same time, and c) it runs tests as separate functions and not as separate processes to overcome limitations of the Memory Management Unit (MMU). A non-evented version of the scheduler, `perd` is used to periodically run the SSL measurement test, `sslgetcert` and ship measurement results over HTTP. A `eooqd` daemon is used to provision one-off measurements (measurements that execute only once). A RIPE Atlas roadmap page [124] describes the

Figure 18: *The coverage of the RIPE Atlas measurement platform as of Feb 2015. The green, red and grey slices represent connected (around 7.7K), disconnected and abandoned probes respectively. Around 12K probes have been deployed in total:* https://atlas.ripe.net/results/maps/network-coverage

future plans on deployment of newer metrics and measurement tools. The RIPE NCC is using measurement results to provide Internet scale latency and reachability maps [125] as a community service.

5.1.5 *Architecture*

The RIPE Atlas architecture consists of measurement probes, a registration server and several controllers. A probe bootstraps by securely connecting to a registration server. The address of the registration server and keys are hardwired on the probe. All of the communications are initiated by mutual authentication over two reverse `ssh` channels. These channels run on port 8080 to easily traverse firewalls. The registration server on a successful connection directs the probe to a nearby controller. The decision is based on the geographical proximity and overall availability of the controller. The controller, on receiving a request from the probe, sends a measurement schedule on one `ssh` channel, and sets up a periodic wait to receive measurement results on another `ssh` channel. The scheduling decisions are made by the controller based on the available measurement capacity and geographical proximity of the probe. The controller is also responsible for shipping software updates to the probe. There are less than 500 probes associated per controller [126]. The intermediate measurement results are queued up by RabbitMQ to be later archived in HBase measurement stores. The brain is responsible for running parallel Hadoop jobs to process these measurement results and incorporate information from BGP data sources. A central database is used to keep administrative information, measurement metadata, recent measurement results and credit stores. A user-interface is available to check status of the probes, measurement results and credit accumulation points. RIPE Atlas architecture also provides the capability to run custom measurements, User Defined Measurement (UDM). The ability to provision UDMs has been available since the launch of the platform. Running a UDM

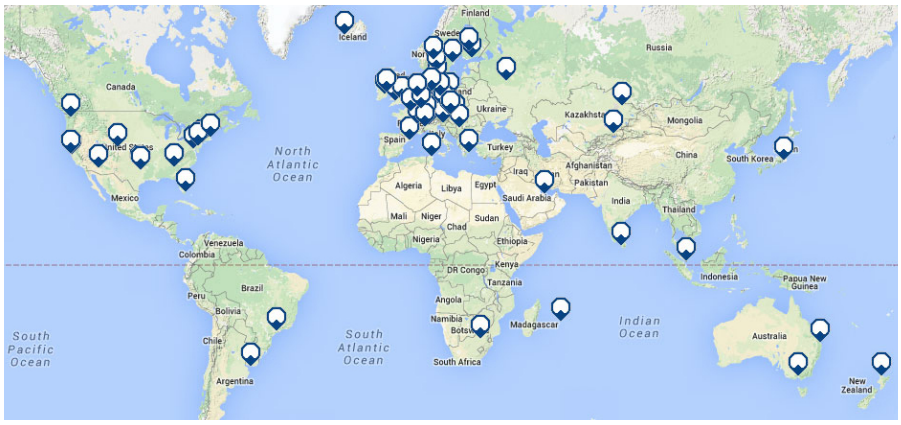


Figure 19: *The coverage of the RIPE Atlas anchors as of Feb 2015. Around 100 anchors have been deployed in total:* `https://goo.gl/1ff9hV`. *A list of deployed anchors and anchoring measurements is available here:* `https://goo.gl/N4GH2j`

.

consumes credits, which are earned by either hosting or sponsoring probes. RIPE Atlas also provides a REST-based API [127] to not only provision such UDMs, but also retrieve measurement results programatically. Measurement results produced from within RIPE Atlas are made publicly available with an immutable reference, the measurement ID. This enables one to publish raw datasets to enable reproducible research. As a result, the platform is starting to gain traction within the academic community.

5.1.6 *Research Impact*

The RIPE NCC regularly publishes results derived from the RIPE Atlas measurement platform. These articles [128] range from studying an event (e.g. Hurricane and Superstorm Sandy), to troubleshooting issues (e.g. debogonising 128.0/16, BGP route filtering of IPv6 /48) to understanding the infrastructure changes (IPv6 reachability testing).

Independent researchers have also used RIPE Atlas for measurement-based research. For instance, Massimo Candela *et al.* in [129] demonstrate a system, called `TPLAY` that can be used to visualize `traceroute` measurements performed by the RIPE Atlas probes. The visualization is a radial representation of a clustered graph where routers are vertices and clusters are adminitrative domains. Massimo Rimondini *et al.* in [57] present an automated matching method to evaluate the impact of BGP routing changes on network delays. They verify the effectiveness of the method on publicly available BGP data from RIPE Routing Information Service (RIS) and RTT data from the RIPE Atlas platform. Andra Lutu *et al.* in [130] use the BGP Visibility Scanner [131] to categorize the visibility of announced IPv6 prefixes. They run `traceroute` measurements from the RIPE Atlas platform to measure the reachability of the categorized Limited-Visibility Prefixes (LVP) and Dark Prefixes (DP). They show that LVP are generally reachable, however DP are largely not. Nevil Brownlee *et al.* in [132] study patterns in `traceroute` responses caused by routing changes as seen by a cluster of RIPE Atlas probes. They use a combination of edit-distance and uncommon-distance measures to cluster probes. Adriano Faggiani *et al.* in [133] utilize the p2c-distance metric [134] to show how traceroute measurement infrastructures along with BGP

route-collectors can increase the AS-level topology coverage by 48.5%. Collin Anderson *et al.* in [135] use RIPE Atlas to study censorship events in Turkey and Russia. They ran hourly DNS, `traceroute` and `SSL` connectivity tests towards social media websites to study content restrictions and blocking strategies employed during censorship events. Marco Di Bartolomeo *et al.* in [136] introduce an *empathy* relationship between `traceroute` measurements. They describe an algorithm that leverges this relationship to identify high-impact events from traceroute datasets. The effectiveness of the approach is presented by utilizing publicly available RIPE Atlas `traceroute` datasets.

A number of research papers have also been published in the past that have used the legacy TTM measurement platform. For instance, C. J. Bovy *et al.* in [137] study distributions of end-to-end delay measurements between several pair of TTM boxes. They witnessed around 84% of these distributions were typical gamma shaped with a heavy tail. Artur Ziviani *et al.* in [138] show how a measurement-based service can be used to geographically locate Internet hosts. They use geographically distributed TTM boxes (equipped with GPS sensors) as landmarks to infer the location of the target by matching network delay patterns of the target to one of these known landmarks. Xiaoming Zhou *et al.* in [139] use TTM boxes to measure end-to-end packet reordering using UDP streams. They show that packet reordering is a frequent phenomenon, with a relatively small number of reordering events ocurring in an individual stream. They also observed that reordered stream ratios are fairly asymmetric. They go further in [63] to measure end-to-end IPv6 delays and hopcount between the TTM boxes. They observe how for a given source and destination pair, IPv6 paths show higher delay and variation when compared to IPv4 paths. They attribute the difference to the presence of badly configured tunnels in IPv6. Finally, with the decline of TTM service, Tony McGregor *et al.* in [119] announced the availability of a public data repository hosted by RIPE NCC. The dataset comprises of measurements conducted by RIPE NCC projects, National Laboratory for Applied Network Research (NLANR) project, and other external research institutions.

5.2 PERFSONAR

Performance Focused Service Oriented Network Monitoring Architecture (perfSONAR) is a collaborative initative by The Energy Sciences Network (ESnet), GEANT, Internet2, and Brazil's National Education and Research Network (RNP). perfSONAR is a network monitoring framework that seeks to solve end-to-end performance problems on paths crossing multi-domain networks. It is designed to support collaborative scientific experiments that rely on ubiquitous and high performing global network infrastructure. The support primarily involves identifying and isolating performance problems in network paths that underpin scientific data exchange. perfSONAR is a federation of measurement sites within these network paths. These sites are equipped with a set of measurement tools that can help localize the performance problems. Fig. 20 provides an overview of the architecture of the perfSONAR measurement platform.

5.2.1 *Scale, Coverage and Timeline*

perfSONAR started in 2004 and in 11 years they have deployed around 7.6K perfSONAR web services all around the globe as shown in Fig. 21. perfSONAR Performance Toolkit (perfSONAR-PS), a perfSONAR-based per-

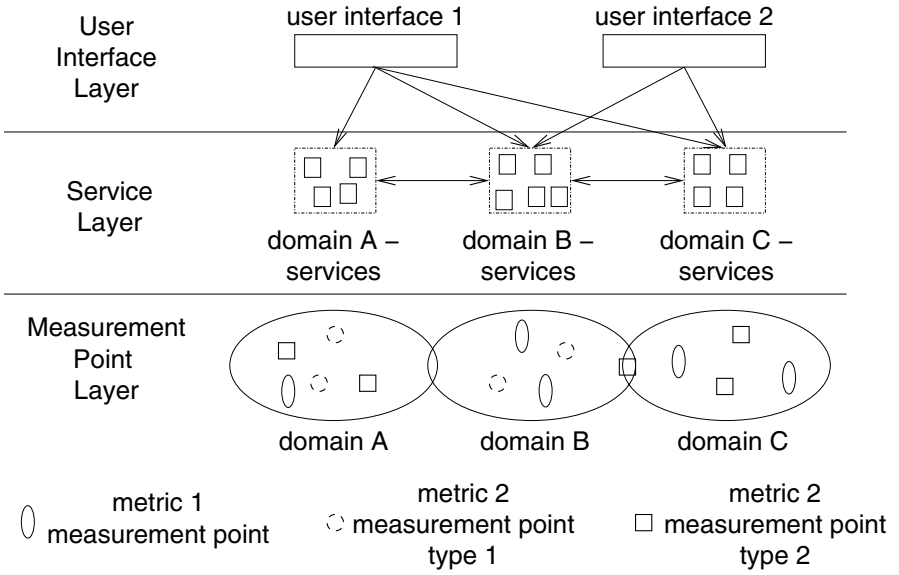


Figure 20: *An architecture of the perfSONAR measurement platform. The architecture is divided into three layers. The middleware layer is a network management web service layer. The bottom layer is a network measurement layer responsible for running active (or passive) measurement tests. The top layer interfaces with the user encompassing a number of visualization tools and methods to allow the user to trigger a measurement test [140].*

formance measurement toolkit developed by ESnet and Internet2, was first released as an open-source software in 2006. The US ATLAS project has been using this toolkit since 2008. US ATLAS is a subset of the ATLAS project. ATLAS is a particle physics experiment at the Large Hadron Collider (LHC). ATLAS itself is a subset of Worldwide LHC Computing Grid (WLCG), which is a grid computing infrastructure that aims to provide location-agnostic access to data incubating from LHC experiments. WLCG currently operates around 150 sites for exchange and analysis of scientific data. These sites are distributed all around the globe and are equipped with perfSONAR monitors as shown in Fig. 22. These monitors continously measure the performance of the multi-domain network path along which the scientific data is exchanged. perfSONAR Multi-Domain Monitoring (perfSONAR-MDM), a perfSONAR framework implementation by GEANT, was released in 2010. Since then, around 60 measurement points running the perfSONAR-MDM toolkit have been deployed around the globe as shown in Fig. 23. These measurement points are deployed at multiple European National Research and Education Network (NREN). perfSONAR-PS and perfSONAR-MDM are interoperable with one another since 2010.

5.2.2 *Hardware*

perfSONAR does not deploy dedicated hardware probes. The measurement software has been open-sourced and made freely available. There are two major software implementations available for the measurement framework: a) The perfSONAR-PS and b) The perfSONAR-MDM. The perfSONAR-PS toolkit is packaged as a CentOS bootable image (perfSONAR-PS tools were

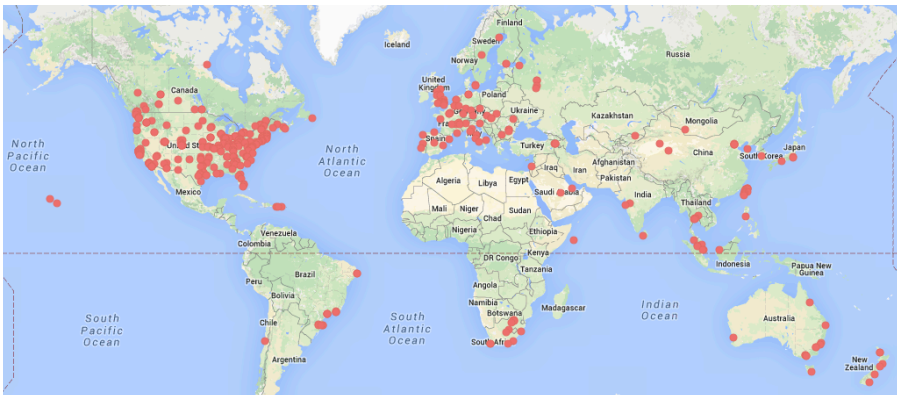


Figure 21: *The global coverage of the perfSONAR deployment as of Feb 2015 with around 7.6K operational web services:* `http://goo.gl/AhHuzr`

.

Figure 22: *The coverage of the perfSONAR-PS deployment within WLCG as of Feb 2015 with around 150 operational sites. The different shades of green (darker being better) indicate the current status of the monitoring sites as reported by ATLAS SSB and OSG GOC dashboards:* `https://goo.gl/lWevRy`.

earlier packaged together in a Knoppix-based bootable CD, called PS-NPToolkit). A perfSONAR measurement point can be made operational by running this image on a 1U server chassis. Running a perfSONAR measurement point from a desktop hardware is not recommended though. Detailed hardware requirements are made available online [141]. Instructions are also available on how to host a perfSONAR-PS measurement point in a virtualized environment, however, running the overlay on bare-metal servers is preferred. The perfSONAR-MDM toolkit on the other hand provides binary packages for Debian-like and RedHat-like distributions. Detailed hardware requirements are available online [142]. A dedicated hardware is recommended, however, some components (visualization and lookup service) can be virtualized. perfSONAR-MDM is also available in a USB-stick form factor (perfSONAR2Go). perfSONAR-PS has been implemented to allow a distributed support model, while perfSONAR-MDM implementation provides a more coordinated and centralized support model.

5.2.3 *Metrics and Tools*

perfSONAR supports both active and passive measurements. perfSONAR-PS is being used by the Brookhaven National Laboratory (BNL) and the Fermi National Accelerator Laboratory (FNAL), which serve as tier-1 facilities for the WLCG. The toolkit supports measuring network utilization, available bandwidth, end-to-end latency, packet loss, connection stability and forwarding path. These metrics are measured using specialized tools. For instance, perfSONAR-PS uses `bwctl` [143] to measure available bandwidth, `pingER` [144, 61] to measure end-to-end latency, end-to-end jitter and end-to-end packet loss, `OWAMP` [145] to measure one-way latency, one-way jitter and ony-way packet loss, `traceroute` to measure the forwarding path, NDT and Network Path and Application Diagnosis (NPAD) to generate network diagnostic reports for end-to-end and last-mile paths. A perfSONAR-BUOY service is used to configure a set of `OWAMP` and `bwctl` tests, archive their measurement results and provide a query interface for easy retrieval of measurement results. It also supports passive network monitoring such as `rrdtool` for network data polling using Simple Network Management Protocol (SNMP) and graphing using `cacti`. It also provides support for lookup and archival services to store SNMP, end-to-end and one-way latency and bandwidth measurements. The archives can be stored using either a Round-Robin Database (RRD) or an SQL instance. An `apache2` server and an `ntp` daemon is also packaged within the toolkit. perfSONAR-MDM on the other hand is used by the Port d'Informacio Científica (PIC) (tier-1), the Centro de Investigaciones Energéticas, Medioambientales y Tecnológicas (CIEMAT) (tier-2) and the Institut de Fisica d'Altes Energies (IFAE) (tier-2) which also are part of the WLCG. perfSONAR-MDM provides three software components: a) Hades Active Delay Evaluation System (HADES), b) Bandwidth Controller Measurement Point (BWCTL MP), and c) The Round Robin Database Measurement Archive (RRD MA). HADES is used to perform and store one-way delay, jitter, `traceroute`, and packet loss measurements. BWCTL MP is used to measure achievable bandwidth, RRD MA is used to measure link utilization, link capacity, input errors and output drops on a link. These tests can be initiated on-demand or in a scheduled fashion. A new weather map integration also provides the possibility to view live monitoring data in the dashboard interface. The metrics can also be visualized using the available iOS and Android mobile applications. A number of visualization tools have been developed to view the perfSONAR measurement archives. For instance, network-based maps are provided to the end-users using Customer Network Management (CNM) and Network Monitor System (Nemo) tools. CNM [146] is deployed within the DFN (Germany) network, while Nemo [147] is used within the UNINETT (Norway) network. Traceable network paths and diagnostics are provided to the staff members using the VisualperfSONAR [148] and perfSONARUI [149] tools. These tools are deployed by GEANT, Internet2 and ESnet.

5.2.4 *Architecture*

perfSONAR provides web-based services that perform measurements in a federated environment. These services are middlewares between measurement tools and visualization and diagnostic tools. perfSONAR implements a Service-Oriented Architecture (SOA) allowing network management functions to become services accessible over the Simple Object Access Proto-

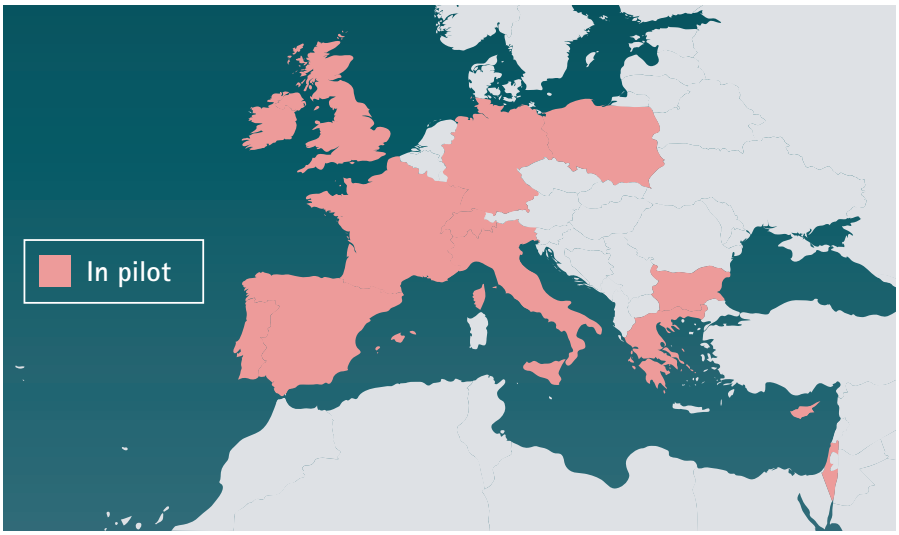


Figure 23: *The coverage of the perfSONAR-MDM deployment as of Feb 2015. Around 60 measurement points have been deployed in total (43 in GEANT service area, 8 in ESnet, 9 in Internet2). The measurement points within the GEANT are situated at multiple European NREN, such as, RedIRIS (es), DFN (de), PIONIER (pl), SWITCH (ch), HEAnet (ie), GARR (it), GRnet (gr), RENATER (fr), JANET (uk), FCCN (pt), BREN (bg), CYNET (cy), IUCC (il) and DANTE (for the GEANT backbone):* `http://services.geant.net/perfsonar/resources`.

col (SOAP). Each measurement probe can then be invoked as a web service to perform network diagnostic operations. The schema description of the network monitoring tasks are specified by the Open Grid Forum (OGF). The web services layer is broadly divided into two families: a) performance data services, and b) enabling services. The performance data services interact with elements that are associated with measurement data. They are further subdivded into three families: a) Measurement Points, b) Transformation services, and c) Measurement archives. Each family can have multiple instances. For instance, the measurement archives can either be stored as a RRD instance or as an SQL instance. Similarly a measurement point can be composed of instances of multiple disparate measurement tools. The enabling services provide authentication, authorization and information facilities. The Information Service (IS) services is used for registration, service and data discovery and network topology representation (The IS was formed by merging previously existing Lookup Service (LS) and Topology Service (TS) components). The IS services can be queried using XQuery. The authentication and authorization services have been federated across domains with the help of EduGAIN [150]. A dashboard framework is a centralized location to see the performance of the entire network at once. The dashboard also provides the capability of triggering alarms when a perfSONAR site detects a potential problem to allow rapid response to such events. There are multiple dashboard instances supporting individual networks. For instance, the Site Status Board (SSB) provides operational support through a dashboard interface to the ATLAS community, while Grid Operations Center (GOC) at the Indiana University is another instance that provides support to the Open Science Grid (OSG) community. The OSG is an initiative supported by the Department of Energy (DOE) and the National Science Foundation (NSF). The US contributes computing and storage resources to the WLCG through the OSG.

The status checks of the monitoring sites performed by perfSONAR-PS as viewed through these dashboards is shown in the Fig. 22. A dashboard on the status of the perfSONAR-PS monitors is available online [151].

5.2.5 *Research Impact*

Andreas Hanemann, *et al.* in [140] motivate the need for a network monitoring framework that can scale on multi-domain networks. They propose a SOA-based approach and describe the overall architecture of the perfSONAR framework. They describe how this framework will be used to facilitate the performance monitoring needs of the GEANT service area, associated NRENs and the Internet2 backbone. They go further in [152] and introduce a set of perfSONAR visualization tools and their feature sets. They reason how a variety of such tools have been developed to serve the needs of different use-cases such as end-users, research staff, operations staff and project managers. Jason Zurawski, *et al.* in [153] describe the data models and schemas used within the perfSONAR framework. They show how measurements are encoded in XML format and exchanged using SOAP. The base schemas are defined within OGF Network Measurement Working Group (NM-WG), while extensions are allowed using XML namespaces. They go further in [154] to describe a registration and discovery mechanism, the perfSONAR Lookup Service (perfSONAR LS), which can be used to locate available measurement services. They describe how LS instances are projected in LS rings, where leaders of each ring exchange summary information to help scale the LS across multi-domain networks. The leaders are chosen using an election algorithm. Brian Tierney, *et al.* in [155] describe the deployment of perfSONAR for the LHC community. The LHC generates 10TB of data per day, which is exchanged amongst 11 tier-1 LHC sites using dedicated 10Gbps links that are part of the LHC Optical Private Network (LHCOPN). Over 150 tier-2 institutions are connected to these tier-1 sites using a multipoint-to-multipoint network, called the LHC Open Network Environment (LHCONE). A large number of tier-3 institutes are connected to tier-2 institutes to form the entire grid infrastructure. In order to ensure consistent throughput, perfSONAR is used to create a persistent baseline of network performance across all segments of the paths traversed while exchanging this data. Prasad Calyam, *et al.* in [156, 157] present an ontology-based semantic priority scheduling algorithm for active measurements. The algorithm uses an inference engine to dynamically prioritise measurement requests, mitigate oversampling under high loads and is conflict-free. The evaluation performed using a perfSONAR-inspired simulation setup shows how generated schedules have low cycle times and high satisfaction ratios. Experiments on real-world perfSONAR traces show how the algorithm can mitigate oversampling under high loads. They go further in [158] to present OnTimeSecure, a secure middleware for perfSONAR. It provides user-to-service and service-to-service authentication and federated authorization based on hierarchical policies. It uses a REST-based approach and can also interface with the aforementioned meta-scheduler to handle prioritized measurement requests. Inder Monga, *et al.* in [159] describe their experiences in deploying and running the ESnet4 hybrid network. The hybrid network consists of a circuit-based core designed to carry large scientific data flows and an IP-based core to handle commodity traffic. The circuit-based core is controlled by the On Demand Secure Circuits and Reservation System (OSCARS), a network management system built on top of Multiprotocol Label Switching (MPLS). They describe

how perfSONAR has been deployed within ESnet and is planned to be integrated within OSCARS to monitor dynamic virtual circuits. Shawn McKee, *et al.* in [56] describe their experiences in deploying perfSONAR-PS at US ATLAS sites. They also introduce the monitoring dashboard that not only provides a centralized view of the performance of the entire network but also adds support for alarms. Arne Oslebo in [160] introduce perfSONAR NC, a Network Configuration (NETCONF)-based implementation of perfSONAR that uses the YANG data modeling language to specify schemas for each measurement archive. Julia Andreeva, *et al.* in [161] introduce the SSB, an implementation of the dashboard framework. The SSB provides an aggregated view of the real-time performance of distributed sites. They show how the SSB is integrated into the US Atlas operations and describe implementation aspects of deployed SSB sensors and alarm systems. Jason Zurawski, *et al.* in [162] describe how the Brown University Physics Department and the National Energy Research Scientific Computing Center (NERSC) are using perfSONAR to regularly monitor sites handling exchange of scientific data flows. Raphael A. Dourado, *et al.* in [163] present a software library that implements spatial composition of performance metrics [164]. They show how delay composition and delay variation composition can be done by running experiments against performance data collected by perfSONAR within the ESnet and GEANT networks. Partha Kanuparthy, *et al.* in [165, 166] introduce Pythia, a domain-knowledge based overlay that leverages active measurement infrastructures to detect, diagnose and localize performance problems using formally described pathology definitions. They use 11 such definitions and show how a deployment on perfSONAR monitors was able to detect congestion-related performance problems. Hao Yu *et al.* in [167] introduce CMon, an end-to-end multi-domain circuit monitoring system. It uses GEANT's perfSONAR-MDM and Automated Bandwidth Allocation across Heterogeneous Networks (AUTOBAHN) to provision circuits for high-volume data transfers. Prasad Calyam, *et al.* in [168] introduce a network topology-aware correlated anomaly detection and diagnosis scheme for perfSONAR deployments. They use the scheme to prioritize detected events by applying a set of filters. These filters can further be used to identify spatially and temporally critical network paths. They used the `traceroute` and one-way perfSONAR measurement data for validation.

6 STANDARDIZATION EFFORTS

Research findings from surveyed measurement studies have been a valuable input to the regulators in understanding how today's broadband services perform in practice. However, in order to not only allow the regulators to frame better broadband policies but also to allow the ISPs to manage networks on a finer granularity, the measurement activities need to scale up. This has been hard to achieve due to the sheer proprietary nature of the measurement efforts. Each involved organization uses its own dedicated measurement probes that not only need to be separately deployed but also the coordination with them is based on custom-designed mechanisms. This lack of interoperability makes it difficult for regulators to view measurement results from a macroscopic scale. Work is underway across multiple standardization bodies to describe use cases of interest and protocol requirements to pave way for a large-scale broadband measurement architecture. Such an architecture will make it possible to implement measurement capabilities directly in the CPE and give away the need to deploy dedicated measurement probes. The interaction with the CPE will be based on a standardized protocol to enable interoperability. A high-level interpretation of how each standardization body is trying to contribute (see Tables 1 and 2) is shown in Fig. 24. Trevor Burbridge gave a talk giving an overview of all these building blocks and how they fit together at the RIPE 66 meeting [169].

Contents

6.1 IETF LMAP

The IETF Large Scale Measurement of Access Network Performance (LMAP) working group is standardizing an overall framework for large-scale measurement platforms. This involves configuration and scheduling of measurements through a control protocol and reporting of measurement results through a report protocol. The abstract definitions of information carried by these protocols is being defined along with specific data models targeted to a specific protocol. Marcelo Bagnulo, *et al.* in [170, 171, 172] describe the motivation and provide an overview of the standardized architecture envisioned within LMAP.

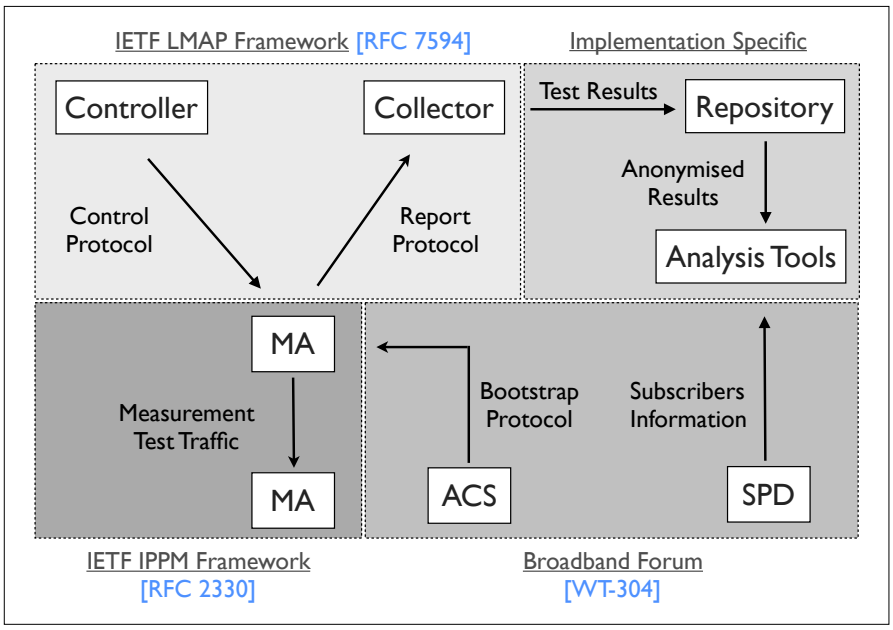


Figure 24: *A high-level overview of bodies involved in the standardization of large-scale measurement platforms. The IPPM working group defines standardized IP-based metrics that a MA uses to generate measurement test traffic directed towards a MP. The LMAP working group defines the architectural framework and the protocols involved in controlling the MA and reporting of measurement results. The BBF defines a bootstrap process to initialize a CPE. It supplies subscriber information to enrich measurement results. The query mechanism to retrieve measurement results and development of data analysis tools to mine the data are not standardized but are implementation-specific.*

6.1.1 *Background*

The Internet Architecture Board (IAB) in 2012 organized a plenary on *Challenges of Network Performance Measurement* at IETF 85 [173] to invite discussions on creating a standards-based network performance measurement architecture. In the plenary, Sam Crawford gave a talk describing the SamKnows measurement platform and he outlined the usefulness of performing end-to-end performance measurements. The data and operational challenges encountered in the process were also discussed. This was followed by Henning Schulzrinne describing the regulator's motivation towards developing a standardized network measurement and management infrastructure. The requirements to perform ISP diagnostics and planning, consumer diagnostics and public policy data collection were discussed. The plenary concluded with the attendees expressing interest towards the standardization effort. The plenary led to a LMAP Birds of a Feather (BOF) meeting at IETF 86 [174] where the scope and goals of the proposed working group were discussed. The LMAP BOF led to the formulation of the LMAP working group.

6.1.2 *LMAP Scope*

The LMAP working group has a charter [175] defining their milestones. The charter clarifies that a measurement system is assumed to be under the control of a single organization, whereby potential overlap amongst differ-

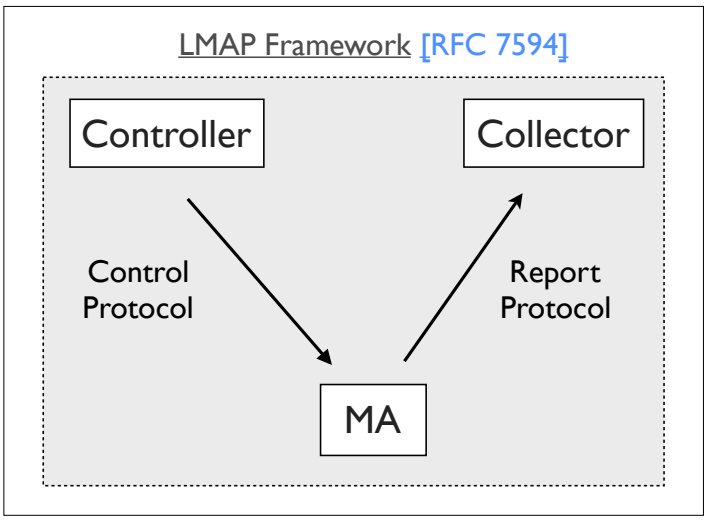


Figure 25: *A high-level reference architecture of the LMAP framework. A MA uses a control protocol to receive instructions from a controller. It uses these instructions to provision a schedule for measurement tests. The collected measurement results are later pushed to a collector using a report protocol.*

ent measurement systems can occur. A potential coordination within this overlapped region, however, is out of the scope of this work. A mechanism to bootstrap the Measurement Agent (MA) and discovery of service parameters is also out of the scope. Protection against malicious self-insertion of inaccuracies is also out of the scope. Both active and passive measurements are in scope and privacy is a critical requirement. The MA interaction with the controller and collectors must be based on simple transport protocols to facilitate a prototype implementation.

6.1.3 *LMAP Requirements and Use-Cases*

Mohamed Boucadair, *et al.* in [176] raise requirements and issues from a provider's perspective to help scope the problem. Marc Linsner *et al.* in [19] describe multiple use-cases of interest for broadband performance measurement. Scenarios around end-users, ISPs and third-party use-cases are described. Kenichi Nagami, *et al.* in [177] describe the LMAP use case from a measurement provider's perspective. A measurement provider measures the network performance from a user's vantage point, by deploying either hardware (or software) probes that run measurement tests against multiple content providers. They reason how this use-case directly complements the end-user's use case. Rachel Huang, *et al.* in [178] describe the LMAP use case for the service provider's network management systems. They propose measurement data collection in a common platform that can be used for variety of purposes such as network troubleshooting, performance evaluation and quality assessment.

6.1.4 *LMAP Framework*

Philip Eardley *et al.* in [179] describe the LMAP framework. The framework identifies key elements of an LMAP, and sketches a reference architecture

of such a platform. The definition of large-scale, scope and constraints of the LMAP work are also discussed along with a terminology to allow the efforts to converge into using a common language repertoire. The framework consists of a MA, a LMAP controller and a LMAP collector as shown in Fig. 25. An MA interacts with a controller to receive instructions on which measurement tasks are to be run, how to execute those measurements tasks using a measurement schedule, and how to report the collected measurement results. The interaction of the MA with a controller must be defined in a control protocol. The MA must periodically push the measurement results to a collector using a defined report protocol.

6.1.5 *LMAP Information Model*

The control and report protocol interaction requires a formal description of the exchanged information elements. The elements must be described at a level of abstraction that is agnostic to the device and used protocol implementation [180]. Trevor Burbridge, *et al.* in [181] describe such an information model. They enlist information elements (such as security credentials and controller server addresses) that must be pre-configured in a MA to allow initial communication with a controller. The configuration information subsequently pushed by the controller to provide additional contextual information to the MA is also described. The elements describing the instruction set sent by the controller and the elements of the measurement report sent to the collector are laid down alongside generic logging and status reporting information.

6.1.6 *LMAP Protocol and Data Model*

There has been a strong inclination in the IETF towards reusing protocols for the LMAP framework. The NETCONF [182] is one of the protocols that can be used by a LMAP controller to provision the MAs. Jürgen Schönwälder in [183] discusses some of the involved technical challenges such as a standardized call-home mechanism. Vaibhav Bajpai *et al.* in [184] deploy an optimized NETCONF server binary on a SamKnows probe to demonstrate the possibility of managing such MAs using the NETCONF protocol. NETCONF-based data models and protocol operations can be specified using the YANG data modeling language [185]. Jürgen Schönwälder *et al.* in [186] describe a YANG data model derived from the proposed LMAP information model that can be used to configure and schedule measurements. The YANG data model proposes to use a push-based design where the configurations are pushed from the LMAP controller to the MA. They take this further in [187] to describe how RESTCONF [188] can be used with such a YANG data model to configure MAs and report measurement results using stream notifications. Arne Oslebo in [189] adapts this YANG data model [186] to propose an alternative pull-based design. They propose the use of RESTCONF to pull configuration from a LMAP controller. In this model, a RESTCONF server needs to be deployed on the LMAP controller, while a RESTCONF client invokes Remote Procedure Call (RPC) calls to pull configuration according to a specific schedule. However, RESTCONF itself subsumes a push-based model in its design. It's unclear whether the protocol approach described in [189] can be deemed RESTCONF. The Internet Protocol Flow Information Export (IPFIX) [190] can also be used by the MA to report measurement results back to an LMAP collector. Marcelo Bagnulo, *et al.* in [191] discuss how an

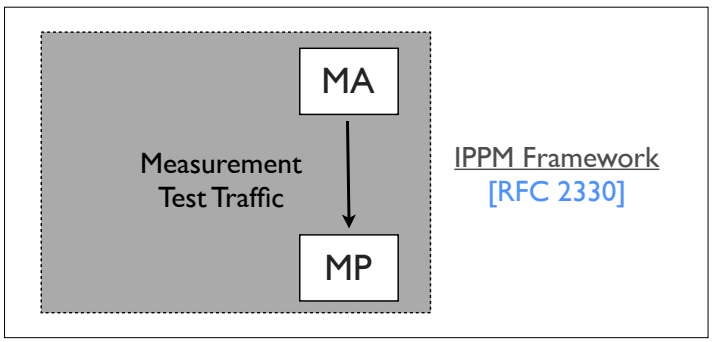


Figure 26: *A high-level reference architecture of the IPPM framework. A MA uses a standard IPPM metric to generate measurement test traffic directed towards an MP. The standardization of this model enables accurate and reproducible results which are relevant across different implementations.*

IPFIX reporting application will require a dedicated metering and exporting process on the MA and a collecting process on the collector. Application-Layer Traffic Optimization (ALTO) [192] is yet another protocol that can be used to perform queries on the LMAP measurement results repository. Jan Seedorf *et al.* in [193] discuss how ALTO provides the capability to define abstractions (network maps and cost maps) that can be used to tweak the aggregation-level of measurement results. The interaction is performed using a Representational State Transfer (REST) interface on top of HTTP while the carried data is encoded in JSON. David Goergen, *et al.* in [194] describe a methodology to derive the network topology from the FCC Measuring Broadband America dataset. The fabricated network and cost maps can then be used by an ALTO server. Marcelo Bagnulo, *et al.* in [195] use the information model to formulate a specific data model that describes the semantics of the information elements in a JSON encoded format. The data model can be used to exchange these information elements in a structured format using a REST architecture on top of HTTP. As such, HTTP can be used both as a control and report protocol in such a design. The Uniform Resource Identifier (URI) design of the proposed Application Programming Interface (API) is also discussed in detail. The proposal adheres to the charter requirement of a simple transport protocol to facilitate early prototype implementation. Vic Liu *et al.* in [196] provide an alternative proposal for a REST-based LMAP protocol. It utilises a push-based model (as opposed to a pull-based design as described in [195]) to configure and schedule measurements. At the state of this writeup, the LMAP working group is currently under discussion and a protocol selection is yet to be determined.

6.2 IETF IPPM

The IP Performance Metrics (IPPM) working group defines metrics that measure the quality, performance and reliability of protocols and services that operate on top of the IP. Vern Paxson, *et al.* in [197] describe the core IPPM framework that encompasses the terminology, metrics criteria, methodology and common issues associated with accurate measurements. The area of interest is scoped to particularly standardize the network path interaction and measurement test traffic of the measurement agents as shown in Fig. 26. The working group has produced several documents that define metrics to

accurately measure this network path. Fabien Michaut, *et al.* in [37] provide a detailed survey on IPPM-defined metrics and available measurement tools. CAIDA also maintains a taxonomy [198] along with a summary and webpage pointers to each measurement tool.

Jamshid Mahdavi, *et al.* in [199], define metrics for measuring connectivity between a pair of hosts. Metrics to measure uni-directional and bi-directional connectivity at a particular instant or over an interval of time are also described. Al Morton, *et al.* in [200] define a metric to measure whether the ordered delivery of packets is maintained in the network. It also provides sample metrics to measure the extent and frequency of reordering, and provides an assessment of effects on TCP. The tools `owping/owampd` and `QoSMet` can measure such packet reordering by analyzing packet sequence numbers. `sting` [201] can also measure reordering by evaluating the number of exchanges between pairs of test packets.

The asymmetry of network path, router queues and QoS provisioning procedures require that measurements be performed separately on a one-way path as opposed to a combined round-trip path. Guy Almes, *et al.* in [202] define a metric to measure the one-way delay in a network path. Carlo Demichelis, *et al.* take this further and in [203] define a metric to measure the variation in this one-way delay. Metrics to measure a single-shot observation and a sample covering a sequence of singleton tests are described. A number of statistics around the derived sample are also discussed. Guy Almes, *et al.* in [204] define a metric to measure one-way packet loss in a network path. Rajeev Koodli, *et al.* in [205] take this further and describe statistics around this packet loss pattern. These statistics can be used to calculate the average length of loss (or inter-loss) periods. Henk Uijterwaal in [206] defines a metric to measure one-way packet duplication in the network path. `owping/owampd` and `QoSMet` are the most popular tools to measure one-way delay, variation and packet loss. However, these tools require a server daemon installation on the remote end. Stefan Savage has overcome this limitation in [207] by introducing a non-cooperative tool, `sting` that measures one-way loss rate by observing TCP behavior.

On the other hand, measurements involving a round-trip path can leverage ICMP ECHO to subvert the requirement of a remote-end daemon installation. This ease of deployment coupled with the ease of result interpretation makes round-trip path metrics feasible. Guy Almes, *et al.* in [208] define a metric to measure the round-trip delay in a network path. They identify how the issue of synchronization of source and destination clocks has been reduced to an (easier) issue of self-synchronization on the source end. Al Morton in [209] defines a metric to measure the round-trip packet loss in a network path. `ping` is the most popular tool to measure round-trip delay and packet-loss.

Phil Chemento, *et al.* in [210] introduce a nomenclature to measure capacity and available bandwidth both over a link and over an end-to-end path. The variable packet size model and tailgating model are popular methodologies for measuring the per-hop link capacity. `pathchar`, `bing`, `clink`, `pchar`, and `nettimer` are popular per-hop capacity measurement tools. The end-to-end capacity can be measured using the per-hop capacity metrics, however a packet-pair dispersion methodology can be used to directly measure it. `bprobe`, `sprobe`, `pathrate`, and `nettimer` are popular end-to-end capacity measurement tools. There are three methodologies defined to measure available bandwidth of a link or an end-to-end path. `cprobe` is a popular tool that implements the packet train dispersion methodology. `pathload`, and `pathchirp`, implement the probe rate model methodology, while `IGI/PTR`,

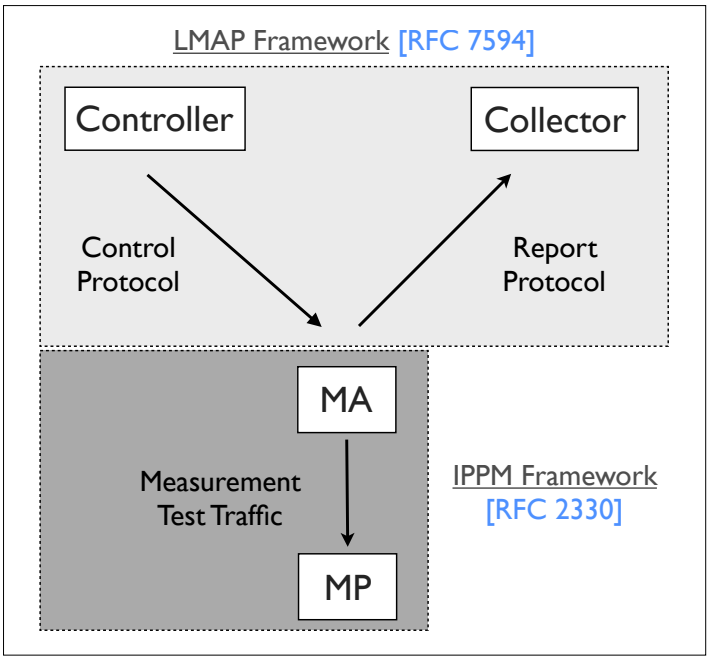


Figure 27: *A high-level interaction between LMAP and IPPM frameworks. The LMAP effort standardizes interaction of an MA with a controller and a collector. The IPPM effort standardizes metrics for measurement tests. A metrics registry acts as a glue to allow LMAP protocols to refer to IPPM-defined metrics.*

and `spruce` implement the probe gap model methodology. Ravi Prasad, *et al.* in [38] provide a detailed survey on available bandwidth estimation metrics, techniques and tools.

Matt Mathis, *et al.* in [211] propose a framework for defining Bulk Transfer Capacity (BTC) metrics. The BTC metric measures the *achievable* throughput of a TCP connection on an end-to-end path. `treno`, `cap`, `ttcp`, `netperf` and `iperf` are popular BTC measurement tools. Barry Constantine, *et al.* in [212] propose a framework to measure the achievable TCP throughput for business class services. This requires a phase of pre-determining the path MTU, bottleneck bandwidth and RTT before test initiation.

Matt Mathis, *et al.* in [213] define a metric to evaluate a network path's ability to carry bulk data. They propose TCP-based models that can be used to apply independent performance tests on smaller subpaths. The results from each subpath can then later be used to predict the end-to-end path's capability. This is made possible by opening up the TCP control loop. The model is designed to be independent of the measurement vantage point.

The IPPM working group has also designed communication protocols to enable interoperability amongst multi-vendor MA and Measurement Peer (MP). For instance, Stanislav Shalunov, *et al.* in [59] introduce the One-Way Active Measurement Protocol (OWAMP) to standardize a method for collection of one-way active measurements. This allows widespread deployment of open OWAMP servers and help one-way measurements become as common as the `ping` measurement tool. Similarly, Kaynam Hedayat, *et al.* in [214] introduce the Two-Way Active Measurement Protocol (TWAMP) to standardize two-way measurement capabilities. TWAMP in addition to the self-synchronization on the source end, also employs a timestamp at the remote end to facilitate greater accuracy. Saverio Niccolini, *et al.* in [215]

describe an information and a data model to store `traceroute` measurement results using XML. This is closely related to the `DISMAN-TRACEROUTE-MIB` module [216], which instead uses SNMP to access `traceroute` results. Al Morton in [217] defines a problem statement for conducting access rate measurements. It describes how the capability to test in two-directions with asymmetric size packets and asymmetric rates are critical functions needed in today's production network measurements.

The working group recently underwent a charter revision [218]. The focus now is to minimize defining newer metrics and measurement protocols, but instead reuse or improve developed standards. Efforts that introduce additional methods for metric calibration or describe the applicability and tradeoffs of current metrics will be encouraged. In this pursuit, Joachim Fabini, *et al.* in [219] have updated the IPPM framework to accomodate this evolution. Al Morton, *et al.* in [220] summarize two different formulations of delay variations used in wider context of active measurements: Inter-Packet Delay Variation (IPDV) and Packet Delay Variation (PDV). They provide recommendations on where each are applicable. Kostas Pentikousis, *et al.* in [221] propose to employ Internet Protocol Security (IPsec) to protect OWAMP and TWAMP protocols. This will not only secure the measurement traffic but also facilitate the applicability of these measurement protocols to current IPsec networks.

A MA is a common denominator within the LMAP and IPPM frameworks as shown in Fig. 27. A MA runs measurement tests that adhere to a standard metric defined within the IPPM working group. The decision on which measurement tests are to be run by a MA are dictated by the LMAP control protocol. The MA also later tags measurement results with the metric when pushing them using the LMAP report protocol. As such, these protocols need a mechanism to refer to an IPPM-defined metric. Marcelo Bagnulo, *et al.* in [222] describe a core registry for performance metrics and rules for metric assignments alongwith initial allocations. The LMAP control protocol can now refer to an IPPM-based metric through a URI scheme that hooks into the metrics registry. Marcelo Bagnulo, *et al.* in [223] take this further and define a reference path for LMAP by assigning a set of identifiable measurement points. The LMAP control protocol can now define a measurement path at a finer granularity using a set of defined measurement points. A reference path can also help complement the measurement results with additional information required for diagnostic and data analysis. Use cases mapping a particular network technology to a viewed reference path are also discussed.

6.3 IETF XRBLOCK

Henning Schulzrinne, *et al.* in [224] have defined the Real-time Transport Protocol (RTP) that facilitates applications in transmitting real-time audio and video data by providing an end-to-end network transport method. They have also defined a companion protocol, RTP Control Protocol (RTCP), that helps provide feedback on the quality of RTP data distribution by sending one or more reception report blocks as part of the sender (or receiver) reports. Kevin Almeroth, *et al.* in [225] have taken this further and defined RTCP Extended Reports (RTCP XR) that convey information beyond these reception report blocks. They have defined seven report block types that fall within three categories. The packet-by-packet block types report reception timestamps for each packet in addition to conveying encountered packet losses and duplicates. The reference time block types that convey receiver-end wallclock

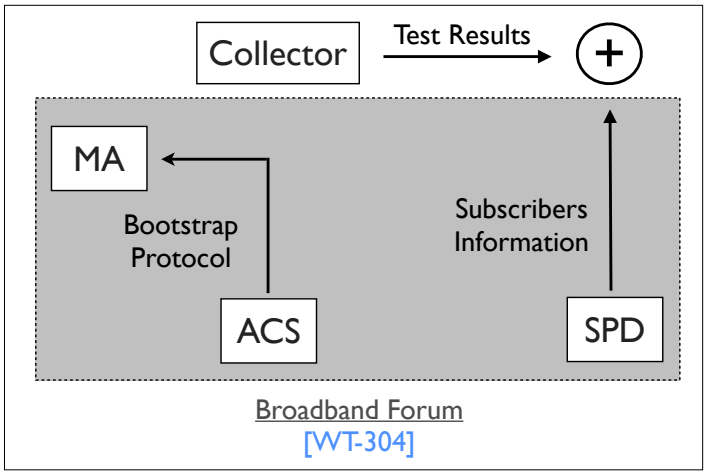


Figure 28: *A perceived BBF standardization contribution as seen from the LMAP and IPPM frameworks. The BBF can use TR-069 as a protocol to bootstrap the MA with preconfigured information to bring it within an LMAP ecoystem. The BBF can also supply subscriber information that can be later spliced into the measurement results for validation purposes.*

timestamps and the delay encountered in the reception of these blocks. Finally, summary metric block types convey summary statistics and metric to monitor VoIP calls. The authors also propose a framework which can be used to add additional block types in the future.

The Metric Blocks for use with RTCP's Extended Report Framework (xrblock) working group has been chartered to use this framework to invite proposals on new report blocks definitions. As a result, a number of documents describing newer performance metrics have emerged recently. Alan Clark, *et al.* in [226] define an RTCP XR block type that helps identify a measurement period to which other RTCP XR blocks may refer to indicate the span of the report. The receivers can use this information to verify the metric blocks. Alan Clark, *et al.* in [227] define an RTCP XR block type that allows statistical reporting of the network round-trip delay between RTP endpoints. The information can be used by the receivers for receive buffer sizing and selecting an appropriate playout delay. The information can also be used to troubleshoot the network path in general. Alan Clark, *et al.* in [228] define an RTCP XR block type that provides information on packet delay variation. The information can be used by the receivers to adapt the size of the jitter buffers to improve performance. Alan Clark, *et al.* in [229] define an RTCP XR block type that allows reporting of burst and gap loss metrics. The information is useful to applications that use jitter buffers but do not use stream repair means.

6.4 BROADBAND FORUM

The Broadband Forum (BBF) takes a unique position of being able to apply the standardization work incubating out of the IETF directly on vendor devices. This can be coupled with existing BBF protocols such as CPE WAN Management Protocol (TR-069) [230] or Data Over Cable Service Interface Specification (DOCSIS) [231] that can act as enablers to help expedite the adoption process. The Enabling Network Throughput Performance Tests and

Statistical Monitoring (TR-143) project [232] for instance, has been working on defining CPE data models to initiate performance throughput and latency tests and monitor CPEs using diagnostic mechanisms defined in TR-069. Both network-initiated and CPE diagnostics are in scope. The tests can be run either in an ongoing or on-demand fashion. Active monitoring of the broadband network will help base lining nominal service levels and validating QoS objectives. It also helps the service provider characterize the performance of end-to-end paths. Such an active monitoring using performance metrics will facilitate establishment of SLAs for guaranteed service offerings. The Broadband Access Service Attributes and Performance Metrics (WT-304) project [233] started in 2012, takes TR-143 further by developing additional performance tests such as packet loss, jitter, emulated streaming and browsing. The project intends to develop a framework to allow standards-based broadband performance testing and reporting. It plans to develop test methodologies that can segregate and measure a network segment. Tests metrics must be standardized to support multiple operator networks. Development of test schedule intervals and capability to trigger on-demand tests are in scope.

The LMAP information model [181] assumes that a number of configuration elements are pre-baked within a MA, even before the MA attempts a registration with the LMAP controller. These elements particularly include the MA security credentials and the Fully Qualified Domain Name (FQDN) of the controller that must be pushed to the MA during an initial bootstrap process. The MA must also perform an exchange to make the remote end learn about its capabilities. The possibility of triggering an on-demand test is also useful. These interactions can be done either using the TR-069 or DOCSIS protocol depending on the access technology used by the gateway. The service provider (part of the BBF) is also in a unique position to own the customer's subscription information. This subscriber parameter information, once spliced into the measurement results at the collector-end, can be used to validate the service offerings against the signed agreements as shown in Fig. 28. A TR-069-based data model using the IETF LMAP information model [181] was presented at a Leone workshop [234] on large-scale measurements co-located with the BBF meeting.

6.5 IEEE

The Institute of Electrical and Electronics Engineers (IEEE) 802.16 working group [235] on Broadband Wireless Access Standards develops standards to promote the growth of broadband Wireless Metropolitan Area Networks (MAN). The working group is currently developing the P802.16.3 project [236] on Mobile Broadband Network Performance Measurements, which is targeted to evaluate the performance of mobile broadband networks from a user's perspective. The architecture and requirements document however scopes the project only to mobile users. It introduces the concept of both private and public measurement peers, which can be used for conducting measurements. Private measurement peers can be useful in situations where the client wishes to perform measurements towards an exact location of interest. The model also introduces public and private data collectors. The data on public collector must be anonymized, however the data on private collector can be kept as is to facilitate more accurate data analysis.

Table 1: List of Surveyed Standardization Work (Part I)

	Document	Type	Date ↓	Status
IETF LMAP	Information Model for LMAP [181]	WG I-D	2015	Active
	A YANG Data Model for LMAP MA [186]	WG I-D	2015	Active
	Using RESTCONF with LMAP MA [187]	WG I-D	2015	Active
	A Framework for LMAP [179]	RFC 7594	2015	–
	LMAP Use Cases [19]	RFC 7536	2015	–
IETF xrblock	RTCP XR Block for MPEG-2 TS PSI Independent Decodability Statistics Metrics Reporting [237]	RFC 6990	2013	–
	RTCP XR Block for Burst/Gap Loss Metric Reporting [229]	RFC 6958	2013	–
	RTCP XR Block for Delay Metric Reporting [227]	RFC 6843	2013	–
	RTCP XR Block for Packet Delay Variation Metric Reporting [228]	RFC 6798	2013	–
	Measurement Identity and Information Reporting Using a SDES Item and an RTCP XR Block [226]	RFC 6776	2012	–

6.6 ITU-T

The ITU-T Joint Coordination Activity on Conformance and Interoperability Testing (JCA-CIT) within the International Telecommunication Union - Telecommunication Standardization Sector (ITU-T) are discussing methods of standardizing procedures to measure Internet access speeds. A meeting [239] was held to evaluate the proposals from various perspectives. Germany's Bundesnetzagentur, a telecommunications regulator, presented its comparative study on the received Internet access speeds to the service provider's advertised broadband speeds. The Central Research Telecommunication Institute (ZNIIS), Russia, a national testing laboratory, presented a proposal to establish a virtual laboratory that can be used to remotely access cutting-edge measurement tools. Arcatech, UK, a testing equipment manufacturer, gave a presentation on how to accurately perform QoS assessments. The project will run from 2013 to 2016.

Table 2: List of Surveyed Standardization Work (Part II)

	Document	Type	Date ↓	Status
IETF IPPM	Active and Passive Metrics and Methods [238]	WG I-D	2016	Active
	Registry for Performance Metrics [222]	WG I-D	2015	Active
	Model Based Bulk Performance Metrics [213]	WG I-D	2015	Active
	IKEv2-based Shared Secret Key for O/TWAMP [221]	RFC 7717	2015	–
	Rate Measurement Test Protocol Problem Statement and Requirements [217]	RFC 7497	2015	–
	A Reference Path and Measurement Points for LMAP [223]	RFC 7398	2015	–
	Advanced Stream and Sampling Framework for IP Performance Metrics (IPPM) [219]	RFC 7312	2014	–
	Round-trip Packet Loss Metrics [209]	RFC 6673	2012	–
	Framework for TCP Throughput Testing [212]	RFC 6349	2011	–
	A One-way Packet Duplication Metric [206]	RFC 5560	2009	–
	Packet Delay Variation Applicability Statement [220]	RFC 5481	2009	–
	Information Model and XML Data Model for Traceroute Measurements [215]	RFC 5388	2008	–
	A Two-Way Active Measurement Protocol (TWAMP) [214]	RFC 5357	2008	–
	Defining Network Capacity [210]	RFC 5136	2008	–
	Packet Reordering Metrics [200]	RFC 4737	2006	–
	A One-way Active Measurement Protocol (OWAMP) [59]	RFC 4656	2006	–
	IP Packet Delay Variation Metric for IPPM [203]	RFC 3393	2002	–
	One-way Loss Pattern Sample Metrics [205]	RFC 3357	2002	–
	A Framework for Defining Empirical Bulk Transfer Capacity Metric [211]	RFC 3148	2001	–
	A Round-trip Delay Metric for IPPM [208]	RFC 2681	1999	–
	A One-way Packet Loss Metric for IPPM [204]	RFC 2680	1999	–
	A One-way Delay Metric for IPPM [202]	RFC 2679	1999	–
	IPPM Metrics for Measuring Connectivity [199]	RFC 2678	1999	–
	Framework for IPPM [197]	RFC 2330	1998	–

7 SUMMARY

A number of measurement platforms have utilized datasets from more mature platforms to validate their experimental results during the early stages of their deployment as shown in Fig. 29. For instance, Enrico Gregori *et al.* in [114] use publicly available AS topology datasets collected by Archipelago and AS edges dataset collected by the DIMES measurement platform to validate AS-level topology graphs generated by Portolan. Adriano Faggiani *et al.* in [240] use the publicly available AS links datasets to validate the AS-level topology of Italian ISPs as revealed by Portolan.

Independent researchers have also made use of multiple measurement platforms to pursue a research question. For instance, Artur Ziviani *et al.* in [138] use RIPE TTM boxes as geographical landmarks to locate Internet hosts. They use probes deployed within the NIMI measurement platform as target hosts. Srikanth Sundaresan *et al.* in [28, 99, 22, 96] use the SamKnows/FCC data in conjunction with the dataset collected by the BISmark platform to study key broadband performance indicators within multiple ISPs in the US.

A number of platforms leverage one or more measurement facilitators to achieve geographical diversity as shown in Fig. 30. For instance, Srikanth Sundaresan *et al.* in [22] describe how SamKnows uses well-provisioned M-Lab servers as measurement targets to measure end-to-end latency, end-to-end packet loss and upstream and downstream throughput from SamKnows probes. Sarthak Grover *et al.* in [92] describe how BISmark uses strategically deployed M-Lab nodes as measurement servers that act as sources and sinks of measurement traffic for active measurement tools. A number of independent researchers have also used a combination of facilitators and measurement platforms to purse a research question. For instance, Massimo Rimondini *et al.* in [57] describe how they use the BGP data from RIPE RIS and RTT data from the RIPE Atlas platform to study effects of BGP routing changes on network delays.

A timeline of the evolution of the Internet performance measurement platforms according to the taxonomy described in this survey is shown in Fig. 31. SamKnows was established in 2008 to meet the growing need of the regulators to measure broadband performance across multiple service

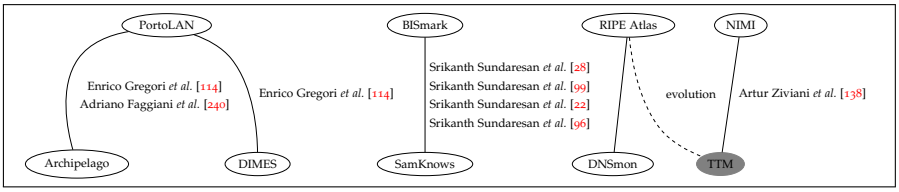


Figure 29: *A graph representing collaboration amongst Internet performance measurement platforms (in white). Greyed out measurement platforms have been decommissioned and superseded by their successors. Dotted lines indicate an evolution along with the research paper that describes this evolution marked with labelled edges. Straight lines connect one measurement platform with another, along with labelled edges that mark the research paper that describes how they utilized each other's dataset for validation purposes.*

Table 3: List of Internet Measurement Projects

	Projects	Description	Duration ↓	Website
EU	RIPE RIS	RIPE NCC Routing Information Service	2001–	http://ripe.net/ris
	RIPE DNSmon	RIPE NCC DNS Monitoring Service	2003–	http://ripe.net/dnsmon
	METRICS	Measurement for Europe: Training & Research for Internet Communications Science	2013–2017	http://metrics-itn.eu
	SMART	European Internet Traffic: Monitoring Tools and Analysis	2013–2015	http://internet-monitoring-study.eu
	RITE [241]	Reducing Internet Transport Latency	2012–2015	http://riteproject.eu
	M-Plane [242, 243]	An Intelligent Measurement Plane for Future Network & Application Management	2012–2015	http://ict-mplane.eu
	Leone [171]	From Global Measurements to Local Management	2012–2015	http://leone-project.eu
	DEMONS [244]	Decentralized, Cooperative, & Privacy-Preserving Monitoring for Trustworthiness	2010–2013	http://fp7-demons.eu
	PRISM [245]	Privacy-aware Secure Monitoring	2008–2010	http://fp7-prism.eu
	MOMENT	Monitoring and Measurement in the Next generation Technologies	2008–2010	http://www.fp7-moment.eu
APJ	ITZ [246]	University of Adelaide Internet Topology Zoo	2010–	http://topology-zoo.org
	MAWI [247]	Measurement and Analysis on the WIDE Internet	2002–	http://mawi.wide.ad.jp
	DIMES [15]	Distributed Internet Measurement and Simulation	2004–	http://netdimes.org
	WITS	Waikato Internet Traffic Storage Project	2003–2008	http://wand.net.nz
US	Science DMZ [248]	ESnet: A Network Design Pattern for Data-Intensive Science	2010–	http://fasterdata.es.net/science-dmz
	BGPmon [249]	A Real-Time, Scalable, Extensible Monitoring System	2008–	http://bgpmon.netsec.colostate.edu
	Ark [36]	CAIDA Archipelago Project	2007–	http://caida.org/projects/ark
	ATLAS	Arbor Networks: Active Threat Level Analysis System	2007–	https://atlas.arbor.net
	iPlane [16]	An Information Plane for Distributed Services	2006–	http://iplane.cs.washington.edu
	PeeringDB [250]	A Peering Database of Networks	2004–	http://peeringdb.com
	Network Telescope	UCSD/CAIDA Network Telescope Project	2002–	http://caida.org/projects
	E2Epi	Internet2 End-to-End Performance Initiative	2001–	http://e2epi.internet2.edu
	PCH IRTA	Packet Clearing House Internet Routing Topology Archive	1997–	https://pch.net
	PingER [61]	Ping End-to-End Reporting Project	1995–	http://www-iepm.slac.stanford.edu
	RouteViews	University of Oregon RouteViews Project	1995–	http://routeviews.org
	NAI [251]	NLANR Network Analysis Infrastructure	1995–2006	http://www.moat.nlanr.net
	IPMA	MERIT Internet Performance Measurement and Analysis	1997–2000	http://www.merit.edu/research/ipma

Table 4: Taxonomy of Internet Performance Measurement Platforms (Part I)

Class	Platform	Scale	Metrics	Tools	Hardware	Impact
	SamKnows	~ 70K	End-to-end latency, last-mile latency, latency-under-load, forwarding path, end-to-end packet loss, upstream and downstream throughput and goodput, end-to-end jitter, network availability, webpage download, VoIP, P2P, DNS resolution, email relays, FTP and video streaming performance.	`ping`, `mtr`, `cron`, `ntp` + custom-developed tools at SamKnows	OpenWrt-based TP-Link routers	[86, 87, 88, 89, 24, 90, 252, 91, 253, 184, 10, 9]
FIXED-LINE ACCESS	BISmark	~ 420	End-to-end latency, last-mile latency, latency under load, end-to-end packet loss, access-link capacity, upstream and downstream throughput, end-to-end jitter, webpage load time, uptime using special hearbeats, number of wired devices, number of devices associated on wireless link, number of wireless access points, packet and flow statistics, DNS responses and MAC addresses.	`d-itg`, `shaper-probe`, `iperf`, `mirage`, `paris-traceroute`, `cron`, `ntp`	OpenWrt-based Netgear routers	[96, 97, 22, 55, 98, 28, 99, 100, 92, 101, 60, 18, 102]
	Dasu	~ 100K	Number of per-torrent TCP resets, number of active torrents, number of active, failed and closed TCP connections, end-to-end latency, forwarding path, HTTP GET, DNS resolution, per-torrent, application-wide and system-wide upload and download throughputs.	`ping`, `tracer-oute`, `NDT`, `cron`, `ntp`, `netstat`	Vuze-based software plugin	[104, 87, 88, 58, 106, 107, 105]

Table 5: Taxonomy of Internet Performance Measurement Platforms (Part II)

Class	Platform	Scale	Metrics	Tools	Hardware	Impact
	Netradar	~ 5K	Signal strength quality, operating system, device type, radio type, positioning information, handovers using base station ID, vendor information, latency, TCP goodput using upload and download speed tests, TCP statistics, Internet connectivity.	custom-developed tools at Aalto University	Android, iOS, Meego, Symbian, and Windows mobile platforms	[109, 110, 111]
MOBILE ACCESS	Portolan	~ 300	Latency, IP and AS forwarding path, achievable bandwidth, available wireless networks, signal strength, cell coverage, traffic shaping detection.	`smartprobe`, `MDA-traceroute`	Android	[62, 114, 115, 117, 113, 112]

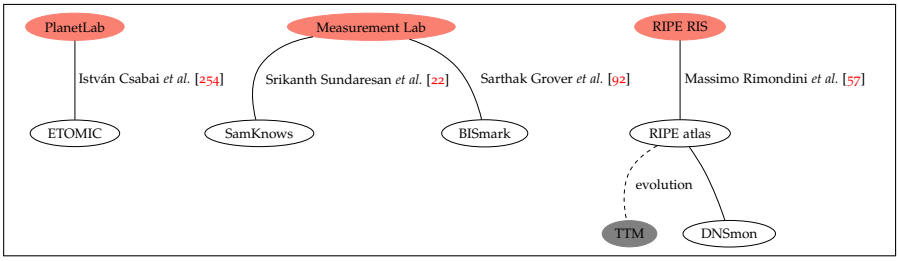


Figure 30: *A graph representing facilitators (in salmon) used by Internet performance measurement platforms (in white). A number of platforms utilize more than one facilitator. Greyed out measurement platforms have been decommissioned and superseded by their successors. Dotted lines indicate an evolution of the platform, along with the research paper that describes this evolution marked in labelled edges. Straight lines connect a measurement platform with a facilitator, along with labelled edges that mark the research paper that describes how they use it.*

providers. An academic interest to perform accurate measurements from the edge led to the development of Dasu and BISmark platforms in this area. The broadband performance measurement community has long been preceded by topology measurement platforms (not shown in the figure) and measurement platforms designed to provide operational support. RIPE TTM started in 1997 and has evolved into the RIPE Atlas measurement platform that provides support to network operators. perfSONAR was started in 2004 to support the scientific community. The mobile measurement space is starting to take shape with the developments within the Portolan and Netradar measurement platforms since 2012. The IETF IPPM and xrblock working group have been involved in standardizing measurement metrics for

Table 6: Taxonomy of Internet Performance Measurement Platforms (Part III)

Class	Platform	Scale	Metrics	Tools	Hardware	Impact
	RIPE Atlas	~ 12K + ~ 100	Latency, forwarding path, HTTP GET, and SSL queries to preconfigured destinations. Latency to first and second hop, DNS queries to DNS root servers. All built-in measurements run both over IPv4 and IPv6. Periodic local uptime, total uptime, uptime history and current network configuration measurements.	`perd`, `eperd`, `evping`, `evtraceroute`, `evtdig`, `evhttpget` `sslgetcert`, `eooqd`	OpenWrt-based TP-Link routers (previously Lantronix XPort Pro modules) + Soekris-based anchors (previously Dell PowerEdge-based units)	[129, 57, 130, 132, 133, 135, 136] + http://atlas.ripe.net/results/analyses
OPERATIONAL SUPPORT	RIPE TTM	~ 100	One-way latency, packet loss, jitter, root-nameserver reachability, routing statistics, GPS satellite conditions and Path Maximum Transmission Unit (PMTU) discovery.	`traceroute`	A PC and a GPS antenna	[137, 138, 139, 63, 119]
	perfSONAR	~ 7.6K	Network utilization, available bandwidth, achievable bandwidth, one-way latency, one-way jitter, end-to-end latency, end-to-end jitter, end-to-end packet loss, connection stability, forwarding path, end-to-end and last-mile network diagnostics, link utilization, link capacity, link input and output errors.	`hades`, `bwctl`, `pingER`, `NDT`, `NPAD`, `OWAMP`, `traceroute`, `rrdtool`, `cacti`, `apache2`, `ntp`	perfSONAR-P CentOS bootable image, perfSONAR-M RedHat and Debian packages and perfSONAR2C USB stick	[140, 152, 153, 154, 155, 156, 157, 158, 159, 56, 161, 162, 163, 165, 168]

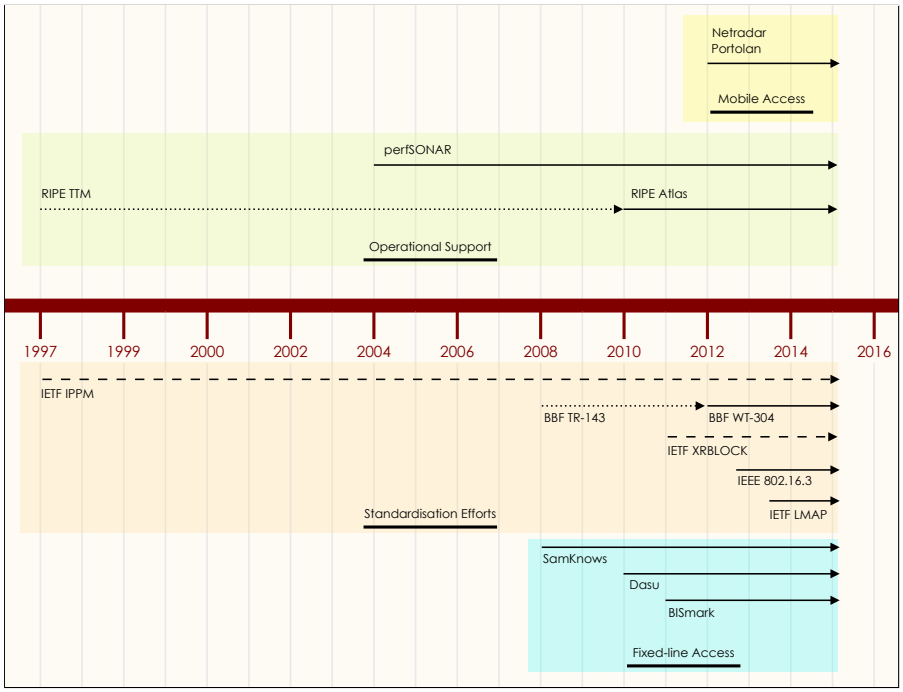


Figure 31: *A timeline of Internet performance measurement platforms. Fixed-line access measurement platforms started with SamKnows in 2008 and the area has been further developed by Dasu and BISmark. They have been preceded by platforms that measure topology discovery (not shown) and provide operational support. The mobile access measurement platforms have more recently emerged since 2012. The relevant but less specific metrics standardization activities (in dashed lines) within the IETF have been active for a while. Work on designing a measurement framework within the BBF and the IETF has picked up only recently. The dotted lines indicate an evolution.*

quite a while. However, the activities within the BBF and the IETF to design a standardize framework for large-scale measurements have only started recently.

A number of measurement-based research projects also utilize these measurement platforms for measurement research. The Leone project for instance, builds new metrics and measurement tools to study the QoE of home users using the SamKnows measurement platform. The M-Plane project on the other hand aims to build a measurement plane that can incorporate measurements from multiple measurement platforms. A large-scale data analysis of these measurement results can allow a reasoner to perform root-cause analysis of issues in the network. The RITE project studies network conditions that contribute towards Internet latency. The aim is to develop and implement novel methods in end-systems that can help reduce latency at the transport layer. Table 3 provides a listing of such measurement-based projects. We also include in this list well-known topology measurement and deprecated performance measurement platforms that did not fall within the scope of this survey.

We also witnessed split preferences on the use of software/hardware probes. SamKnows, BISmark, and RIPE Atlas tend to deploy dedicated hardware-based probes, while Dasu, Netradar, Portolan and perfSONAR provide software installations for compatible hardware devices. In hindsight, performance measurement tools running on hardware probes are also soft-

ware. The advantage of dedicated hardware probes comes instead from the ability to be able to gather round-the-clock measurements. The software measurements that can be installed directly on host devices are more susceptible to resource contention from other applications. The software-suite can also be installed on large variation of hardware devices that makes the measurements harder to calibrate. The software-based solution on the other hand has lower distribution costs. This not only provides low-barrier to entry; but also allows the measurement compaign to quickly span larger demographics. The standardization efforts eventually aim towards facilitating service providers to provide measurement-capable CPEs that will eliminate the need to deploy dedicated probes. As such the conundrum on the choice of a hardware/software probe deployment model may fade away in near future.

7.1 CONCLUSION

We have presented a taxonomy of Internet measurement platforms as: topology discovery and performance measurement platforms. We further classified the performance measurement platforms based on their deployment use-case: fixed-line access measurements, mobile access measurements and operational support. We described the performance measurement platforms in detail by exploring their scale, coverage, timeline, deployed metrics and measurement tools, architecture and overall research impact. Tables 4, 5 and 6 provide a summary of this survey. We also presented common set of measurement tools shared by these performance measurement platforms along with the level of collaboration amongst them through the usage of publicly available datasets. We also showed how platforms have been using measurement facilitators to conglomerate data from multiple sources to pursue a particular research question. We concluded the survey by describing recent standardization efforts to make large-scale performance measurement platforms interoperable.

Part III

MEASURING IPV6 PERFORMANCE

We measure IPv6 performance from 80 SamKnows probes deployed behind dual-stacked networks across the globe.

At Layer 4, we measure TCP connection establishment times to ALEXA top 100 dual-stacked websites. We show that CDN clusters serving popular websites are different for IPv4 and IPv6. Furthermore, we observe that CDN caches are largely absent over IPv6. TCP connect times observed over 3 years long dataset show that TCP connect times to popular websites over IPv6 have improved over time. We go further and measure the effects of the Happy Eyeballs (HE) algorithm. We show that HE makes IPv6 connections towards 99% of websites to be preferred more than 98% of the time. However, HE with a 300 ms timer value prefers slower IPv6 connections 90% of the time. We show that lowering the HE timer value to 150 ms gives us a margin benefit of 10% while retaining same preference levels over IPv6.

At Layer 5, we measure IPv6 performance towards YouTube. We show that TCP connect times to YouTube media servers makes HE prefer a connection over IPv6 even when the measured throughput over IPv4 is better. This results in lower bit rates and lower resolutions when streaming a video than can be achieved if streamed over IPv4. We show how this is also due to the disparity in the availability of YouTube content caches which are largely absent over IPv6. We also compare the similarity of webpages delivered by HTTP over IPv4 and IPv6. We show that 14% of these dual-stacked websites exhibit a dissimilarity in the number of fetched webpage elements, with 94% of them exhibiting a dissimilarity in their size. We show that 27% of the dual-stacked websites have some fraction of webpage elements that fail over IPv6, with 9% of the websites having more than 50% webpage elements that fail over IPv6. We perform a causality analysis and also identify sources for these failing elements.

In Chapter 8 we measure TCP connection establishment times over IPv4 and IPv6 to popular dual-stacked websites. In Chapter 9 we go further and utilise the TCP connect times to measure the effects of the HE algorithm. In Chapter 10 we measure the performance of YouTube streaming over IPv6. In Chapter 11 we compare the similarity of webpages delivered over IPv4 and IPv6.

8 MEASURING TCP CONNECT TIMES

We compare IPv4 and IPv6 connectivity of dual-stacked hosts using a metric that measures TCP connection establishment time to 100 popular dual-stacked websites. We have deployed an implementation of this metric on 20 SamKnows probes connected to dual-stacked networks that are part of 18 different AS. Using a year-long dataset gathered from these vantage points, we show how most of these websites centralise around CDN deployments and consequently show similar performance. We show that these CDN clusters are different for IPv4 and IPv6. Furthermore, some of these websites tend to be served by CDN caches deployed within service provider networks. We show how these CDN caches are largely absent over IPv6. The distributions of TCP connect times show how clusters serving popular websites over IPv6 have improved over time. We also illustrate cases where network policies inhibit hosts from connecting to websites over IPv6.

Contents

8.1 INTRODUCTION

Research and corporate networks have had the capability to carry IPv6 traffic for a long time. However, due to the lack of IPv6 enabled content, the available infrastructure has rarely been used to access services outside of the internal network. With the World IPv6 day in 2011 [255], this is starting to change with several notable web service providers enabling dual-stack mode to provide content over both IPv4 and IPv6. This has pushed network operators to develop deployment plans to bring IPv6 to residential customers. However, many network operators are still in a very early stage of deployment. As a consequence, early adopters that do not yet receive native IPv6 connectivity rely on tunnels to reach content over IPv6. Even the residential customers that do receive native IPv6 connectivity may experience performance and reliability issues, because the IPv6 deployed infrastructure may not be as mature as that of IPv4.

With the World IPv6 Launch day in 2012, several notable web service providers started providing content services over IPv4 and IPv6. In two years since then, a number of large IPv6 broadband roll-outs have happened

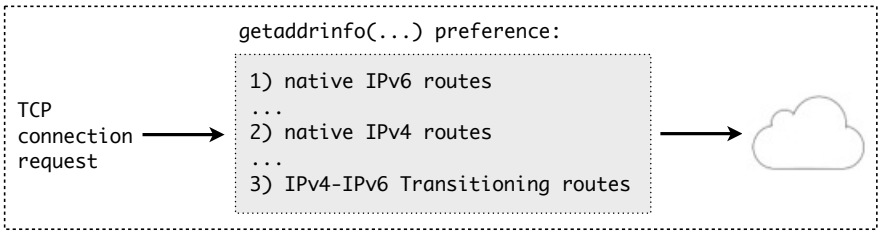


Figure 32: *`getaddrinfo(...)` behavior as dictated by the default destination address selection algorithm [39]. The algorithm makes applications iterate over endpoints in an order that prefers an IPv6-upgrade path.*

[256]. For instance, Comcast, Deutsche Telekom AG and AT&T have demonstrated increased penetration of IPv6 in the fixed-line space, with Verizon Wireless and T-Mobile USA showing similar trends in the cellular space. In fact, Comcast recently [257] completed transition of their entire broadband network infrastructure to be 100% IPv6 ready. These efforts have eventually led to an increased global adoption of IPv6 to 5%, with Belgium (~28.7%), Germany (~11.9%) and USA (~11.7%) leading the adoption rates as seen by Google's IPv6 adoption statistics [27] as of November 2014. These numbers demonstrate that IPv6 adoption is finally happening. Jakub Czyz *et al.* in [7] (2014) provide a high-level view of the current state of IPv6 adoption. They study the deployment from two lenses: a) prerequisite IP functions (addressing, naming, routing and end-to-end reachability), and b) operational characteristics (usage profile and performance). However, they measure IPv6 performance using an approximation of 10— and 20—hop RTTs over a sample of dual-stacked nodes. In fact, they concede that a measure of actual client-to-service network performance would be a more ideal metric. In this study, we plug this gap by using a year-long dataset to measure IPv6 performance of operational dual-stacked websites from 20 dual-stacked vantage points.

A dual-stacked host with native IPv6 connectivity establishing a TCP connection to a dual-stacked website will prefer IPv6. Fig. 32 shows how the function, `getaddrinfo(...)` adheres to the default address selection policy [39] by resolving a service name to a list of endpoints in an order that prioritizes an IPv6-upgrade path. As a result, any application using `getaddrinfo(...)` to resolve service names will tend to prefer connections made over IPv6. We want to know whether users experience benefit (or an added penalty) when connecting to websites over IPv6.

In order to achieve this, we introduce a metric that measures TCP connection establishment times. We deploy an implementation of this metric on 20 SamKnows [4] probes connected behind dual-stacked networks. We ran measurements to a selectively chosen list of top 100 dual-stacked websites from these vantage points and collected measurement data for a year. We show insights uncovered by analyzing this year-long dataset. We explore raw TCP connection establishment times and uncover techniques to cluster websites around CDN deployments. We show how these clusters are different for the IPv4 and the IPv6 network infrastructure. These clusters also reveal which websites are currently being served by content caches deployed inside the service provider network. We show how these content caches are largely absent over IPv6. The gathered trends have allowed us to identify special cases where network policies have resulted in inhibiting IPv6 for certain websites for some hosts. We describe these special cases.

Our measurement study provides four main contributions: a) An active metric (and a corresponding implementation) to measure TCP connection establishment times alongwith a list of top 100 dual-stacked websites processed from Amazon 1M Alexa entries. We release these to the measurement community. b) Identification of CDN deployments and content-caches in service provider networks using BGP-based clusters processed from IP endpoints seen from globally distributed SamKnows vantage points. A quantification of disparity in IPv4 and IPv6 clusters is also made available. c) Distributions of TCP connection establishment times over an year-long dataset to compare IPv4 and IPv6 performance over each CDN cluster and d) A study of special cases such as `www.bing.com` globally stopping IPv6 services in 2013, and Google CDN blacklisting resolvers that inhibit some hosts from receiving their services over IPv6.

The chapter is organized as follows. In Section 8.2 we survey studies measuring IPv6 performance. In Section 8.3 we introduce our measurement methodology, we describe our metric and related design choices, the measurement setup and current deployment. We capture our data analysis insights in Section 8.4 and conclude in Section 8.5.

8.2 RELATED WORK

A number of studies have been conducted to measure the amount of IPv6 adoption on the Internet. Lorenzo Colitti *et al.* in [258] (2009) measure IPv6 adoption from the perspective of the Google web services provider. They witnessed how (in 2009) IPv6 deployments were prevalent in academic networks, but largely absent in consumer networks. Sebastian Zander *et al.* in [259] propose a web-based technique using Google ads and custom Javscript snippets to measure the IPv6 capability of a wider dataset of clients. They witnessed that around 2% of the total connections used IPv6 in a dual-stacked environment, where a sample re-weighting technique reduced multiple biases to show a 20% increase in clients using happy eyeballs in their applications. The authors use this metric in [260] to further investigate Teredo capability of internet clients. They show that around 16% of total connections would be able to reach IPv6 services if Teredo capabilities in Windows were not reduced (Teredo in Windows cannot resolve service names to IPv6 endpoints). They also witnessed significantly higher latencies when using Teredo over native IPv4 or IPv6 connections. The metric based on Google ads is again used by Manish Karir *et al.* [261] in an extended seven-month long study to understand the amount and nature of IPv6 population on the Internet. They observed around 14M unique IPv6 addresses with native IPv6, Teredo, and 6to4 connectivity, and utilized the information embedded in IPv6 addresses to infer their geographical location, ISP, type of transition and NAT technology used. Amogh Dhamdhere *et al.* in [6] perform a thorough study of the IPv6 internet topology evolution as compared to that of IPv4. They use the publicly available BGP data to show that IPv6 and IPv4 performances are comparable when the forward AS-level paths are the same, but are much worse when they differ. Google has been collecting overall and country-based IPv6 adoption statistics [27] for a few years. The statistics reveal that IPv6 adoption is increasing with a decrease in Teredo [41] and 6to4 clients. HE also maintains metrics [262] of global IPv6 deployment on the Internet with statistics such as registered domains with `AAAA` records or networks with IPv6 support. Jakub Czyz *et al.* in [7] (2014) provide a survey of studies measuring IPv6 adoption on the Internet.

Kenjiro Cho *et al.* in [263] (2004) passively monitor DNS records for 3 months from within the WIDE research network to extract a destination list of ~4K dual-stacked nodes. They study IPv6 performance by comparing RTT and AS-level forward paths using a day-long dataset of `ping` and `traceroute` measurements collected from 3 vantage points. They witnessed 16% unreachable destinations; while only a small proportion (among the rest) exhibited larger RTT over IPv6. Lorenzo Colitti *et al.* in [258] (2009) study IPv6 performance by measuring latency using HTTP requests to two experimental Google web service hostnames using a small fraction of Google users. They show how performance of native IPv6 (although small in 2009) is comparable to that of IPv4, but transitioning technologies add considerable latency. They also show how operating systems (and browsers) by default tend to favor connections over IPv6. These studies however are dated. We therefore defer our methodology comparison in favor of more recent studies discussed next.

Mehdi Nikkhah *et al.* in [8] (2011) study IPv6 performance by measuring average download speeds (95% confidence interval within 10% of mean) towards dual-stacked webpages within Alexa top 1M websites (also used by us) from 6 vantage points (as opposed to 20 vantage points used by us). They measure object size of the downloaded root page (without downloading embedded objects) and filter out websites where these sizes are not within 6% (over IPv4 and IPv6) of each other. They separate websites served by same (and different) origin AS over IPv4 and IPv6 and use AS paths (derived from BGP route tables) to further separate them over same (and different) paths. We currently do not capture AS paths, but we do extend this technique by using origin AS to cluster websites by CDN deployments and CDN caches in service provider networks. They [8] analyse performance by studying controlled averages. We instead show distributions of TCP connect times over an year-long dataset. Amogh Dhamdhere *et al.* in [6] (2012) study the deployment of IPv6 from three lenses: a) topology, b) routing dynamics and c) performance. The performance test extends on [8] in two ways: a) It downloads the smallest object (including embedded objects) that is atleast 10KB in size to overcome TCP slow start and b) It measures AS paths using TCP `traceroute` (instead of BGP routing tables). They [6] measure the time to fetch the page object towards a dual-stacked websites list generated from Alexa top 1M websites (also used by us). The performance measurements were conducted from 5 vantage points (as opposed to 20 vantage points used by us). Both studies [8, 6] show how IPv6 performance is comparable to IPv4 when forward AS-level paths are same, but much worse when they differ. They [6] reason how page fetch times (due to small size of typical pages) are more dominated by delay rather than available bandwidth. This is why we measure TCP connection establishment times since it allows us to capture this end-to-end delay at the transport layer. Hussein A. Alzoubi *et al.* in [264] (2013) study the performance implications of unilateral enabling of services over IPv6. They witnessed no performance penalty in disabling the opt-in service. Google used to impose such an opt-in policy to allow hosts to receive Google services over IPv6. However, we show how Google has recently changed the policy [253]. Ari Keranen *et al.* in [265] (2013) discuss preliminary results from measurements conducted during the World IPv6 day in 2011 [255]. They noticed that around 300 within the top 10K Alexa Top Sites (ATS) web services were dual-stacked that day. This includes around 30 within the top 100 ATS web services. The measurement trials and results obtained however are different from our study in three ways: a) The

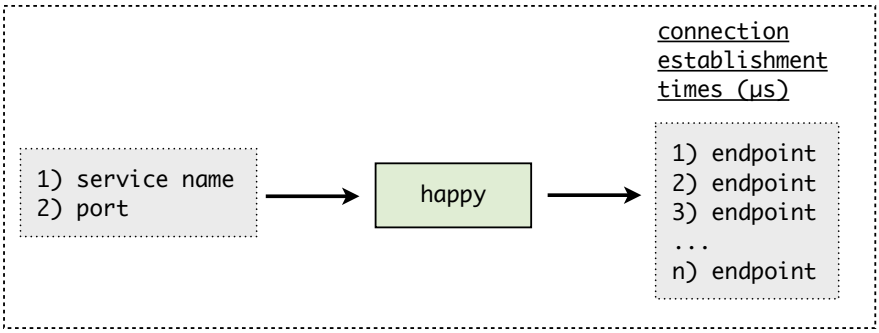


Figure 33: *happy: A tool to measure TCP connection establishment times. The input parameter is a tuple (service name, port number) and the output is the connection establishment time for each endpoint (measured in microseconds). The tool has been open-sourced and is available at: http://happy.vaibhavbajpai.com*

.

measurements were conducted from May 25, 2011—July 11, 2011 using 3 MAs deployed in Finland, Sweden and Canada. Our measurement study is newer and the measurements are conducted from a wider deployed vantage point. At present 14 MAs are deployed across Europe. We have also performed a more detailed TCP connection setup delay study since we take the happy eyeballs algorithm's effects into account. We also measure the routing path differences over IPv4 and IPv6 to analyze the reason for the delay differences, b) As opposed to the study, we witnessed significantly higher TCP connection setup delay differences between IPv4 and IPv6. Generally, services were slower over IPv6, with multiple services being twice as slow over IPv6, and c) We witnessed significantly lower TCP connection setup failure rates. We observed less than 1% service failure rates, as opposed to 20% failure rates. This could be because the authors employ a metric that measures against only the first DNS response, while our metric takes all the endpoints returned by the DNS response into account.

8.3 METHODOLOGY

In this section, we describe our methodology. We introduce our metric and a corresponding implementation. We describe our rationale in selecting a list of dual-stacked websites and illustrate the overall measurement setup that utilizes SamKnows probes. We show the scope and lifetime of our measurement trial by presenting the global vantage point distribution.

8.3.1 *Metric, Implementation and Features*

We have defined a metric that measures the time taken to establish a TCP connection to a given endpoint. The input parameter of the metric is a tuple (service name, port number) and the output is the TCP connection establishment time for all endpoints the service name resolves to, typically measured in microseconds, as shown in Fig. 33.

The `happy` tool, is an implementation of our metric. The tool can read one or more service names at once and apply `getaddrinfo(...)` to resolve their DNS entries to `A` and `AAAA` resource records. The list of service names can either be supplied as command-line arguments or as a separate file. It

then uses non-blocking TCP `connect(...)` calls to concurrently establish connections to all endpoints seen in the resource records of each service name. It calculates the time it takes for the TCP `connect(...)` call to complete as a measure of the elapsed time. In order to allow delineating connection timeouts it also keeps a flag as an indication on whether the connection got established. This indication is made once a socket in a `select(...)` call becomes writeable with no pending socket errors. We do not account the DNS resolution time in the measured connection establishment time. This is done to avoid slow resolvers from biasing our connection establishment time results. The tool enforces a small delay (25ms by default) between concurrent TCP `connect(...)` calls to avoid generating bursty TCP `SYN` traffic. This delay, however, does not come in the way of pending TCP `connect(...)` calls. As such the measured times are not skewed by this feature. We also added the capability to lock the output stream to allow multiple processes to coordinate writes to the same output stream. This is useful when multiple `happy` instances try to append results to a single regular file from a resource-constrained device. The output can be either generated in a human-readable format, or in a Comma Separated Value (CSV) format feasible for consumption by other programs. We have cross-compiled `happy` for the OpenWrt [266] platform, so that the tool can be deployed on SamKnows [4] probes. A manpage describing the tool with possible options is also available.

8.3.2 *Selection of Websites*

We wanted to measure a representative list of popular dual-stacked websites. A large list will allow us to capture the perspective of dual-stacked hosts that frequently visit popular regional websites. The top websites within that list when combined with a widely distributed vantage point will additionally allow us to also capture the perspective of dual-stack hosts from a global standpoint.

We investigated sources that can reveal this information. For instance, Alexa ranks and maintains listings of the most popular websites on the Internet. The public REST API, however, provides the capability to retrieve only the top 100 website names. This is not enough, since only a fraction of these top 100 website names are dual-stacked today. Hurricane Electric, a major IPv6 tunnel-broker based in the US, maintains a list of top 100 dual-stacked website names [262]. The backend uses the top 1M website names list made available by Amazon. However, we noticed that some of the popular websites (e.g. Wikipedia) are missing from this list even though they are dual-stacked. It appears some websites provide `AAAA` records only for domain names starting with `www`. For example, `www.bing.com` does have a `AAAA` record while `bing.com` does not (In this particular case, a request to fetch the latter leads to a redirect to the former). Since, HE does not follow `CNAME`s, they miss some dual-stacked websites in their top dual-stacked website list calculation.

We decided to use Amazon's top 1M website names list [267] used by HE as input to prepare a top 100 dual-stacked website names list using our own custom script. Our script prepends each website name with the label `www` to make an additional DNS request and it also explicitly follows `CNAME`s. This way, we do not miss any of the popular dual-stacked websites like `wikipedia.org`. It is also important to note that we only measure websites in

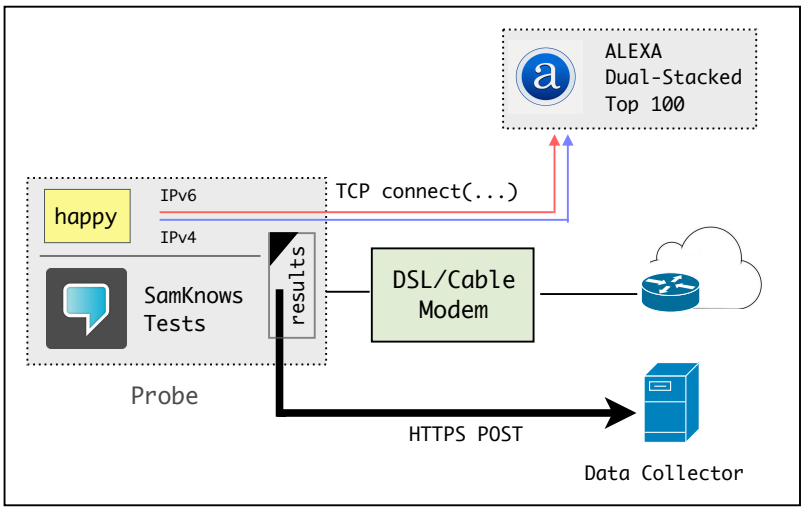


Figure 34: *A measurement setup on top of the SamKnows platform. A dual-stacked probe in addition to the standard SamKnows tests, executes a happy test. The happy test runs every hour and measures TCP connect times to 100 dual-stacked websites both over IPv4 and IPv6. The locally collected measurement results are pushed every hour to a data collector using HTTP.*

this work. As such the connection establishment times and their comparison over IPv4 and IPv6 reflect the performance as seen over TCP port 80.

8.3.3 *Measurement Setup*

We cross-compiled `happy` for the OpenWrt platform and deployed it on SamKnows probes. These probes, in addition to the happy test, also perform standard SamKnows IPv4 measurements. The test is executed on the top 100 dual-stacked websites list and the measurement runs every hour. Due to the inherent storage limitation of the probes, the locally collected measurement results are pushed every hour to our data collector using a REST based architectural style on top of HTTP as shown in Fig. 34.

8.3.4 *Measurement Trials*

We wanted to measure from different locations of the Internet and wanted to ensure that access to certain websites is not blocked administratively. As such, we strategically deployed SamKnows probes to cover a diverse range of origin-ASes. Fig. 35 shows the current deployment status of the SamKnows probes that are part of our measurement trial. An associated table shows the origin AS (both over IPv4 and IPv6) of each vantage point along with its geographic location. The probes have different flavors of IPv4 and IPv6 connectivity ranging from native IPv4, native IPv6, IPv6 tunnel broker endpoints [268], Teredo [41] and tunnelled IPv4. Most of these probes are deployed behind residential networks and receive native IPv6 connectivity. Some probes are also deployed in NREN. We have been collecting this data since March 10, 2013. This has allowed us to collect time series of TCP connect times that may be representative enough to provide us with insights on how IPv6 connectivity to websites compares to IPv4 connectivity.

8.4 DATA ANALYSIS INSIGHTS

We performed a pre-processing run on the dataset to reduce the volume of raw measurements. In this work, we do not look at TCP connection failure rates. As such we pruned out entries where the test reported a TCP connection timeout event. We also removed entries where the test failed in situations where it ran out of socket descriptors (a rare but plausible occurance). We investigated time scales where the variation in TCP connection establishment times is small enough to allow statistically meaningful aggregation. Since applications usually honor the order of endpoints returned by `getaddrinfo(...)` when establishing a TCP connection, we decided to pick the first endpoints returned in each measurement over a day for both

TYPE	IPv4 AS	IPv6 AS	LOCATION	PROVIDER	PROBE ID
RESIDENTIAL	AS31334	AS31334	BREMEN	KABELDEUTSCHLAND	#02
RESIDENTIAL	AS3320	AS3320	BREMEN	DEUTSCHE TELEKOM	#04
RESIDENTIAL	AS50989	AS1257	STOCKHOLM	SITAB	#11
RESIDENTIAL	AS4685	AS4718	FUKUOKA	ASAHI NET	#12
RESIDENTIAL	AS12715	AS12715	MADRID	JAZZ TELECOM	#13
RESIDENTIAL	AS9031	AS9031	ALLEUR	EDPNET	#17
RESIDENTIAL	AS3320	AS3320	BREMEN	DEUTSCHE TELEKOM	#19
RESIDENTIAL	AS2518	AS2516	SHIZUOKA	BIGLOBE NEC	#20
RESEARCH	AS513	AS513	CERN	CERN	#16
NREN	AS680	AS680	BREMEN	DFN	#01
NREN	AS2614	AS2614	TIMISOARA	ROEDUNET	#08
NREN	AS2611	AS2611	LOUVAIN	BELNET	#15
NREN	AS680	AS680	BREMEN	DFN	#18
LAB	AS5607	AS5607	LONDON	BSKYB-BROADBAND	#05
LAB	AS3269	AS3269	TORINO	TELECOM ITALIA	#06
LAB	AS8903	AS8903	MADRID	BT ESPANA	#07
LAB	AS2856	AS5400	IPSWICH	BT UK	#10
IXP	AS18070	AS18070	NIIGATA	NDAC	#14
BUSINESS	AS24956	AS24956	BRAUNSCHWEIG	GAERTNER DATENSYSTEME	#03
BUSINESS	AS13030	AS13030	OLTEN	INIT SEVEN	#09

Figure 35: *Deployment status of our measurement trial as of July 2014. Each vantage point is a SamKnows probe which is part of a larger SamKnows measurement platform. Most of these probes are deployed behind residential networks and receive native IPv6 connectivity from their service provider. A part of these probes are also connected within NREN.*

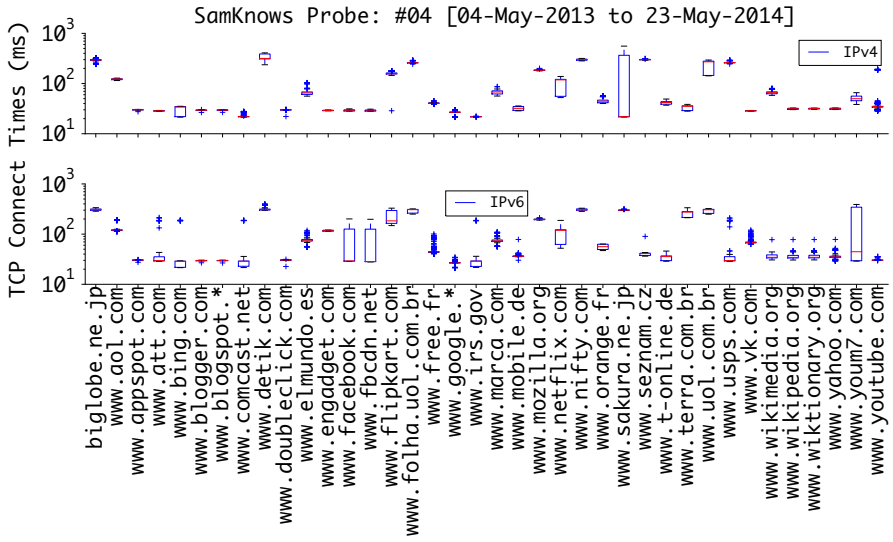


Figure 36: *Box plots showing distributions (in log-scale) of TCP connection establishment times to 100 dual-stacked websites. The SamKnows probe is connected at a premium Deutsche Telekom customer. It has native IPv4 and IPv6 connectivity via DTAG - Deutsche Telekom AG [AS3320].*

address families, and aggregated their TCP connect times centered around the median. The calculated Interquartile Range (IQR) ranges around the median are low. As such, each data point in the subsequent analysis refers to the median of TCP connection establishment times seen by IPv4 and IPv6 endpoints over a day.

8.4.1 *Measuring Raw TCP Connect Times*

Fig. 36 shows box plots of raw TCP connection establishment times to 100 dual-stacked websites from one of the SamKnows probes over the entire year-long duration. This probe is connected behind a residential network in Bremen. The host is subscribed to a premium triple-play service from Deutsche Telekom and as a result receives native IPv4 and IPv6 connectivity at home. It can be seen how several websites appear to show similar performance over IPv4 and IPv6. However, there are also websites such as `www.facebook.com`, `www.fbcdn.net` (served by Facebook CDN) and `www.youm7.com` (served by Cloudflare CDN) where the probe reports substantially higher variance over IPv6. In fact observing the time-series of TCP connection establishment times for `www.facebook.com` for this probe show how TCP connection establishment times have tangibly improved over time as shown in Fig. 39. Additionally websites like `www.att.com`, `comcast.net` and `www.irs.gov` appear significantly faster over IPv4 than IPv6. This is discussed in the following sections.

8.4.2 *Website Clusters*

WHOIS-based: It can be seen from Fig. 36 that several related websites, such as `www.google.*` within each address family show very similar behavior. In fact, the median TCP connection establishment times and the IQR values of many disparate websites within the same address family are also comparable.

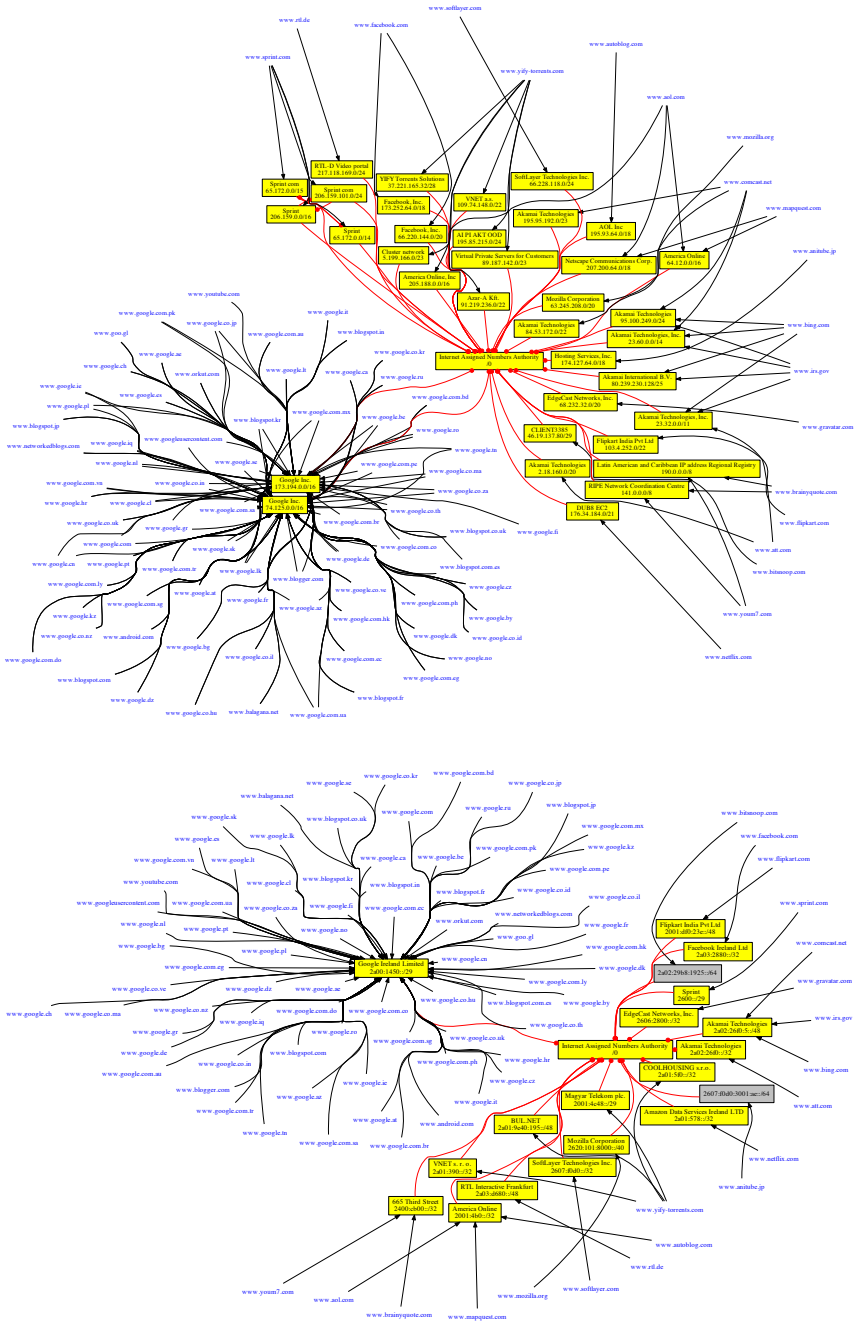


Figure 37: *An IPv4 (above) and IPv6 (below) WHOIS-based aggregation of websites as seen by this (above) probe depicts how most of the websites centralize on core CDNs and major cloud platforms. (The plots are vector graphics and hence zoomable.)*

For instance, `www.att.com` (a DSL network provider), `www.comcast.com` (a cable network provider), and `www.irs.gov` (the US tax collection agency) show very similar performance. One possible explanation is that these websites are provided via common CDNs. Looking at the collected IP endpoints, we found that these websites either resolve to the same endpoint or a set of endpoints that belong to the same allocated address block. Digging through the `WHOIS` information for each of the endpoints (obtained via programmatic

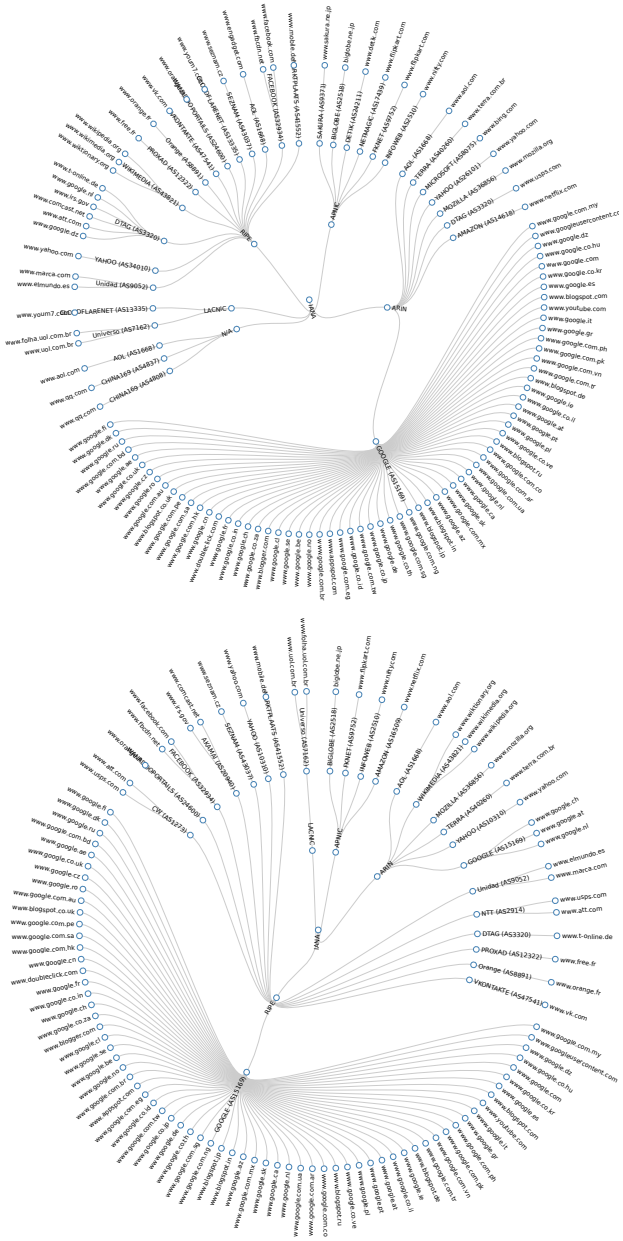


Figure 38: *An IPv4 (above) and IPv6 (below) BGP-based aggregation of websites as seen by one vantage point. The endpoints are aggregated to the announced BGP prefixes as seen by RIPE RIS route collectors. The leaves represent individual websites. The level-2 nodes represent the AS announcing the BGP prefix and its holder name. Finally, the level-1 nodes represent the RIR that allocated the address block to the AS. The SamKnows probe is connected at a premium Deutsche Telekom customer. It has native IPv4 and IPv6 connectivity via DTAG [AS3320].*

APIs from the RIRs) seems to indicate that major portions of the websites map to address blocks owned by organizations such as Google and Akamai as shown in Fig. 37.

BGP-based: The WHOIS-based aggregated clusters are coarse-grained. This is due to the fact that an LIR can decide to split an allocated address block into multiple smaller chunks. The LIR can then decide to announce these smaller

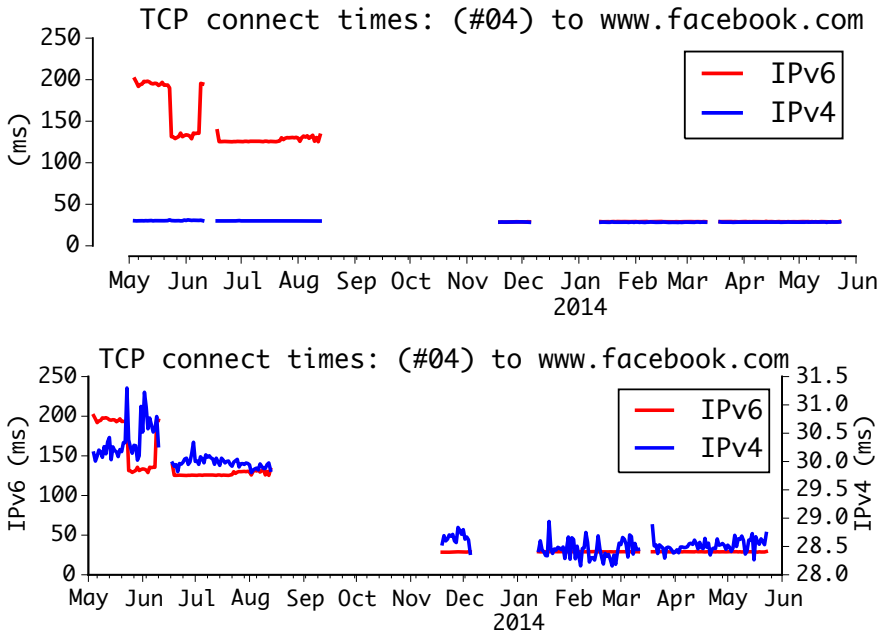
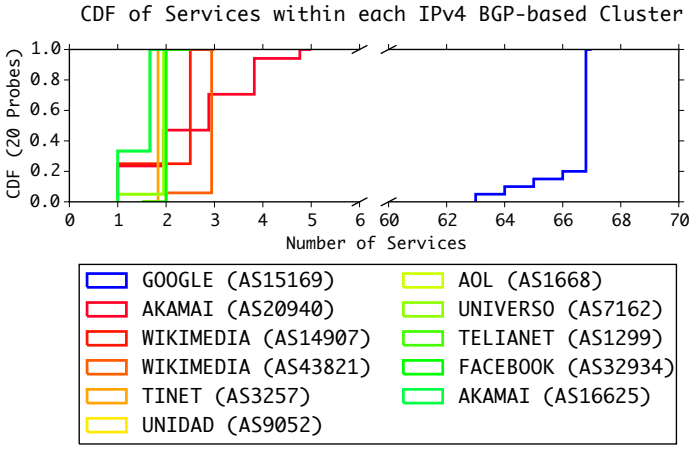


Figure 39: *Time-series to www.facebook.com from SamKnows Probe #04 from May 2013 to April 2014. It can be seen that this probe witnessed significantly improved TCP connect times over IPv6 since November 2013. The bottom plot (in two separate scales) shows that TCP connect times over IPv4 also improved at the same time, but on a much smaller scale.*

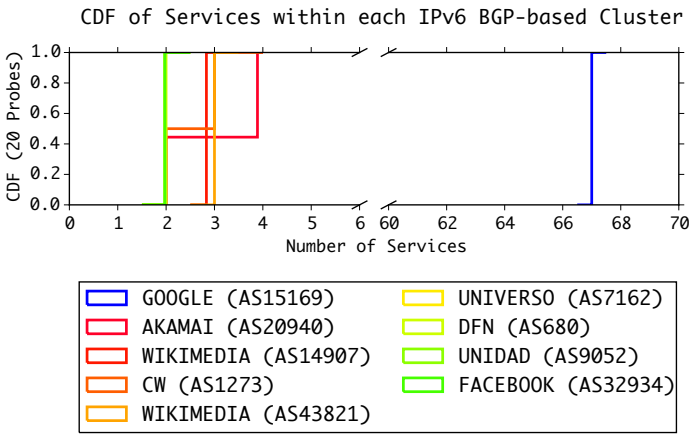
chunks from different ASes. Therefore, we decided to map the collected IP endpoints to announced BGP prefixes as seen by RIPE RIS [269] route collectors. We capture the AS, its holder name, and the RIR that allocated the address block for each announced BGP prefix as an additional metadata in our dataset. Fig. 38 for instance, shows an equivalent BGP-based cluster of websites as seen from the vantage point of this SamKnows probe. It can be seen how aforementioned websites like `www.att.com`, `www.comcast.net` and `www.irs.gov` get clustered behind Deutsche Telekom AG (DTAG) for IPv4, but are dissassociated behind separate clusters for IPv6. These websites are being served over IPv4 by Akamai content caches deployed directly within the DTAG service provider network. However, these caches appear to be missing over IPv6. This correlates with the relative difference between TCP connection establishment times seen over IPv4 and IPv6 for these websites. The BGP-based clusters shown in Fig. 38 are specific to this vantage point. Fig. 40 shows the distribution of number of websites as seen across all probes, both over IPv4 and IPv6. The variation most likely is due to some of the websites getting pushed into service provider networks as content caches. An associated table lists all the clusters in descending order of aggregated number of websites centered around the median. Going forward we use these clusters to perform the rest of the analysis.

8.4.3 *Distribution of TCP Connect Times*

In our pursuit to cover all vantage points, we narrowed down the list to clusters that were seen in *both* address families and by *all* probes. The resultant clusters: Google, Akamai, Facebook and Wikimedia are used in



IPv4 Cluster	#(↓)
GOOGLE (AS15169)	67
SEABONE (AS6762)	05
NTT (AS2914)	03
JAZZNET (AS12715)	03
AKAMAI (AS20940)	03
WIKIMEDIA (AS43821)	03
COMHEM (AS39651)	03
WIKIMEDIA (AS14907)	03
ECHIGO (AS55895)	02
AOL (AS1668)	02
UNIDAD (AS9052)	02
TINET (AS3257)	02
EDPNET (AS9031)	02
TELIANET (AS1299)	02
UNIVERSO (AS7162)	02
AKAMAI (AS16625)	02
FACEBOOK (AS32934)	02



IPv6 Cluster	#(↓)
GOOGLE (AS15169)	67
AKAMAI (AS20940)	04
WIKIMEDIA (AS43821)	03
CW (AS1273)	03
WIKIMEDIA (AS14907)	03
UNIDAD (AS9052)	02
TINET (AS3257)	02
ECHIGO (AS55895)	02
DFN (AS680)	02
ACONET (AS1853)	02
UNIVERSO (AS7162)	02
NTT (AS2914)	02
FBDC (AS10013)	02
FACEBOOK (AS32934)	02

Figure 40: *CDF showing the distribution of number of services within each cluster as seen by all probes. A complementary table shows the number of services within each cluster (across all probes) centered around the median.*

the analysis going forward. Fig. 41 and Fig. 42 show the distribution of TCP connection establishment times as seen by each probe. Fig. 43 on the other hand shows box plots of observed TCP connection establishment times for each probe and a CDF as seen by all probes combined. It can be seen how probes deployed in Japan (#12, #14, and #20) do not appear in Wikipedia-EU CDN measurements, but in fact measure against Wikipedia CDN (not shown). It can also be seen how probes connected behind DTAG networks (#04 and #19) do not reach out to websites served by Akamai CDN over IPv4, but instead are directly served by Akamai content caches deployed from within the DTAG network. It can also be seen how such content caches are largely absent over the IPv6 network. A probe connected to BELNET (the Belgian NREN) (#15) shows consistent behaviour across address families. A probe connected to the DFN (the German NREN) (#01) shows similar

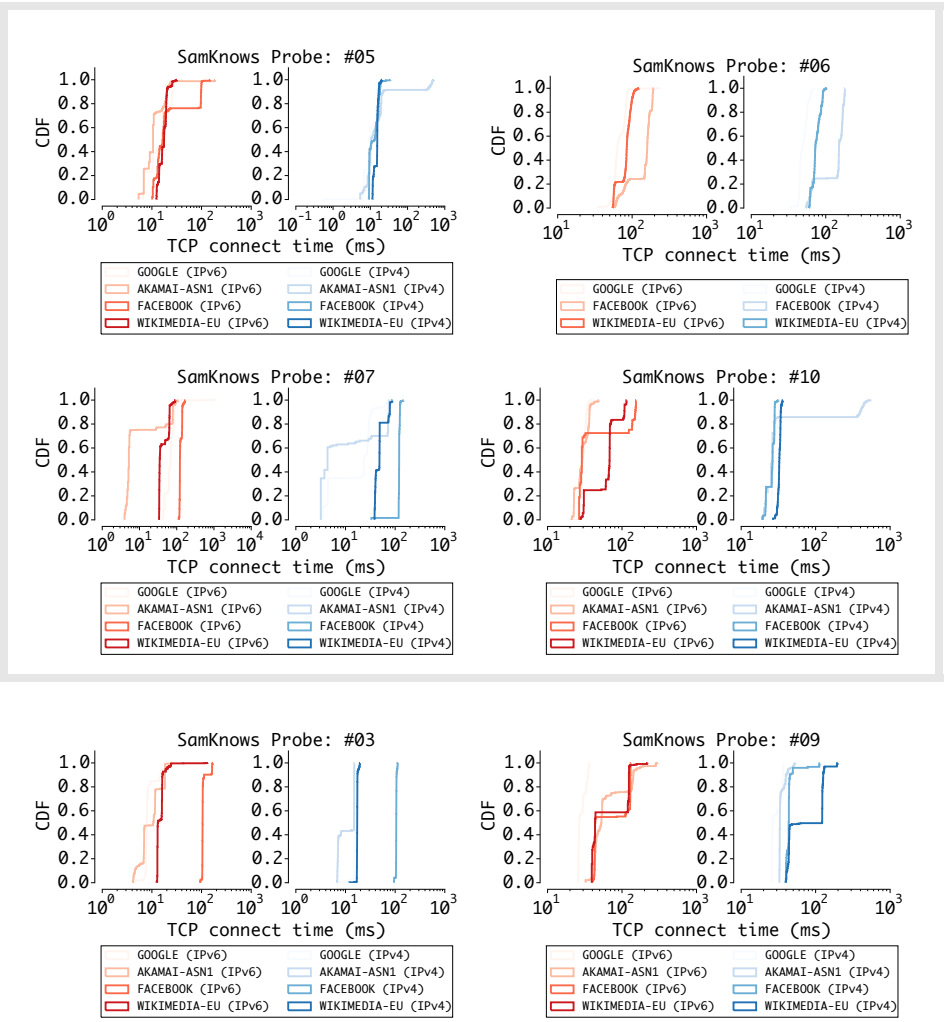


Figure 41: *Distribution of TCP connect times (in log scale) over IPv4 (blue) and IPv6 (red) as seen by probes wired behind an operator's lab (boxed) and business network (unboxed) for 4 CDN deployments: Google, Akamai-ASN1, Facebook and Wikimedia-EU. The list of origin AS (IPv4 and IPv6) of each SamKnows probe is available in Fig. 35.*

medians over the address families, however, the variation for the Facebook CDN over IPv6 is much higher. The probe connected to Kabel Deutschland (#02) shows very similar behaviour with a certain delay offset. This offset is likely due to the different access technology (cable). In general, it seems that IPv6 access to the Facebook CDN shows much higher variation compared to IPv4. Some of the probes occasionally also see very slow connect times (For instance, #13 connected to Jazz Telecom in Spain for all four CDNs and #02 connected to Kabel Deutschland for all except the Facebook CDN). It is not clear what causes this but at least these effects do not seem to be address family specific. A probe connected to ROEDUNET (the Romanian NREN) (#08) does not perform any IPv6 measurements due to a routing issue in the upstream network.

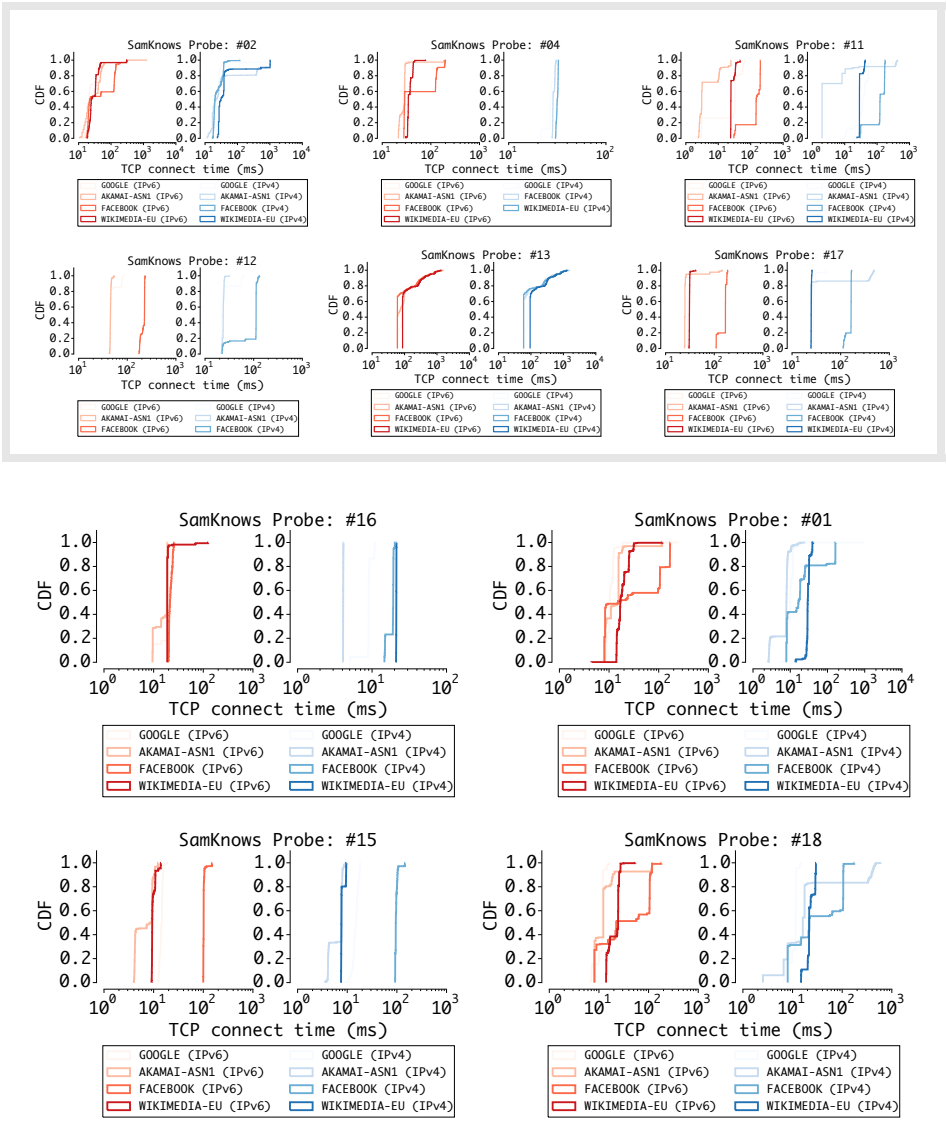


Figure 42: *Distribution of TCP connect times (in log scale) over IPv4 (blue) and IPv6 (red) as seen by probes wired behind a residential gateway (boxed) and research network (unboxed) for 4 CDN deployments: Google, Akamai-ASN1, Facebook and Wikimedia-EU. The list of origin AS (IPv4 and IPv6) of each SamKnows probe is available in Fig. 35.*

8.4.4 *Special Cases*

Our dataset from a distributed set of vantage points has allowed us to identify global events that have affected dual-stacked hosts. In this section, we discuss these events:

Bing: The website `www.bing.com` used to be dual-stacked. However, we witnessed how all of our SamKnows probes stopped performing measurements to `www.bing.com` over IPv6 starting September 2013. Fig. 44 shows the time series of TCP connection establishment times over IPv4 and IPv6 as seen from all and individual vantage points towards this website. There appears to be an abrupt cut-off of IPv6 hinting towards a network policy decision. We investigated the DNS entries returned for `www.bing.com` and

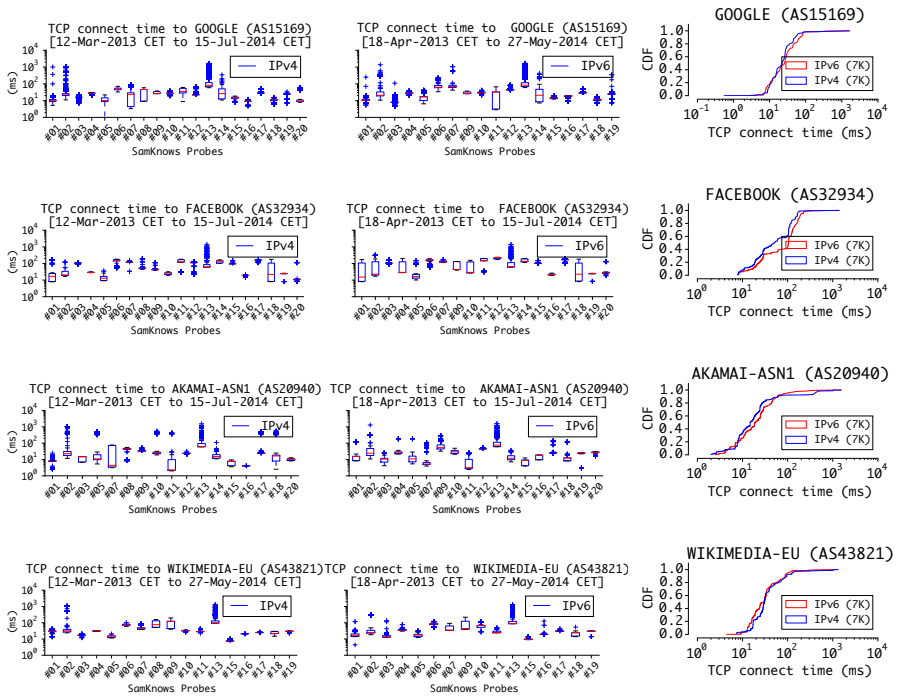


Figure 43: *Box plots of TCP connection establishment times (in log scale) over IPv4 (left) and IPv6 (right) for 4 CDN deployments: Google, Akamai-ASN1, Facebook and Wikimedia-EU as seen by all vantage points. An associated CDF plot shows the distribution of TCP connection establishment times (in log scale) over IPv4 (blue) and IPv6 (red) aggregated over all SamKnows probes.*

found that the upstream resolvers have stopped providing AAAA entries for this website.

Google: On another SamKnows probe (deployed in Japan) we noticed how there were no measurements being performed to any of the google websites. Fig. 45 shows BGP-based clusters formed from endpoints seen by this vantage point both over IPv4 and IPv6. The measurements appear to be active to Google CDNs over IPv4, but are completely absent for IPv6. The probe itself is also successfully able to measure against other websites over IPv6. We investigated the issue and found that this happens to be a network policy decision made by these content providers.

For instance, Google used to perform AAAA prefix whitelisting to prevent users with broken IPv6 connectivity from requesting services over IPv6. Only the whitelisted DNS resolvers received AAAA records for Google services. This was an opt-in process, where an ISP explicitly signed up to receive Google services over IPv6. This helped ensure users had reliable IPv6 connectivity before trying to reach Google services over IPv6 [270]. Since the World IPv6 Launch Day in 2012 [25], Google has changed their policy. The whitelist has been replaced by a blacklist [271]. This eliminates the opt-in process and increases the chance of a dual-stacked host reaching Google services over IPv6. However, if a host is behind a resolver from a blacklisted prefix, it will not receive Google services over IPv6 even though the host may enjoy perfect IPv6 connectivity from the network provider.

The pie chart in Fig 46 shows a country-based distribution of the blacklisted prefixes. The geolocation of the prefix is fetched from the GeoLite data created by MaxMind [272] and is derived from the announcements received from

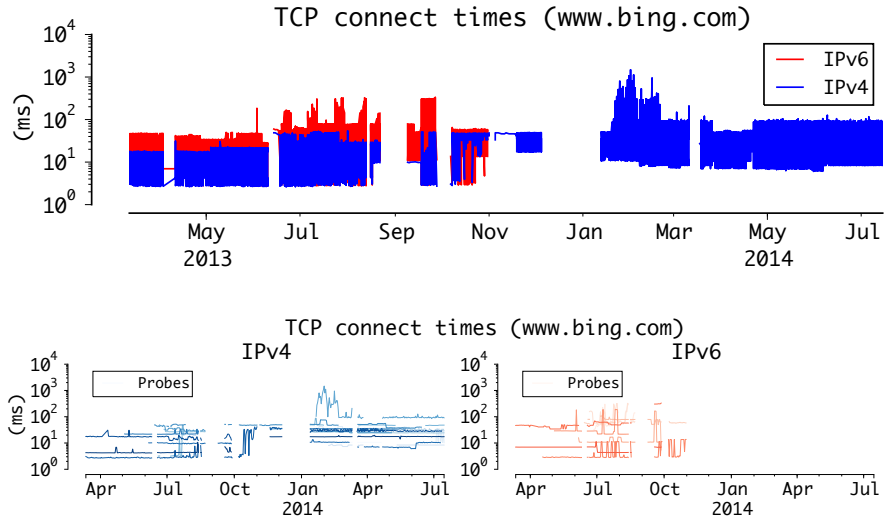


Figure 44: *Time series of TCP connect times to www.bing.com over IPv4 (blue) and IPv6 (red) as seen from all (above) and each (below) vantage point. The measurements over IPv6 stopped for all probes starting Sep 2013.*

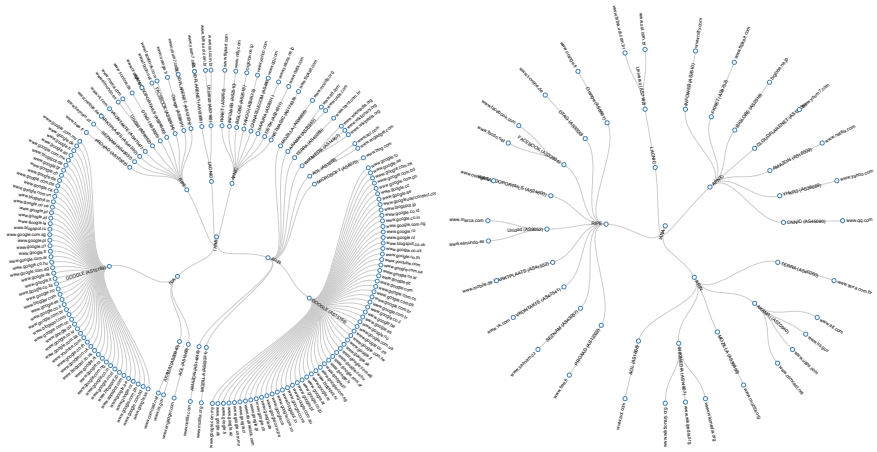


Figure 45: *An IPv4 (left) and IPv6 (right) BGP-based aggregation of websites as seen by a SamKnows probe deployed in Japan connected via BIGLOBE NEC [AS2518, AS2516]. The probe does measurements to Google websites over IPv4, but not over IPv6. Its IPv6 connectivity is not broken, since it does perform measurements to rest of the websites over IPv6.*

within the BGP routing system. The BGP routing data used is made publicly available by RIPE NCC's RIS. It is possible that a prefix may be used from locations encompassing multiple countries. In such a case, the prefix is made to fall in a country with the highest coverage. Ideally, each location of the prefix should be accounted for to make the distribution more accurate. It is important to note that the information on the number of hosts behind the blacklisted resolver prefixes is not avaiable and is not depicted in the distribution.

A google map in Fig 47 shows the location of the blacklisted prefixes from where they are announced into the BGP routing system. A large number of blacklisted prefixes appear to originate from Japan. These are ISPs whose DNS resolvers explicitly started filtering `AAAA` records after World IPv6 launch

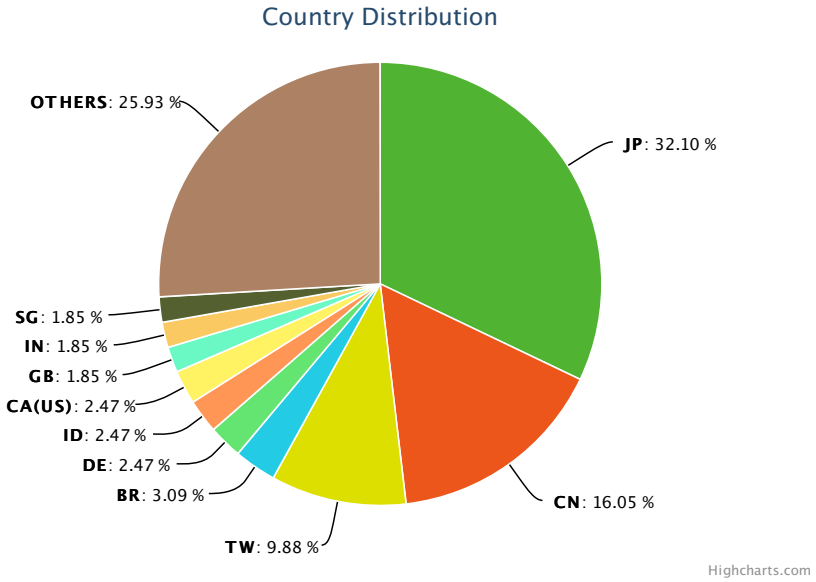


Figure 46: *A distribution of prefixes blacklisted by Google over IPv6. A large number of resolvers in Japan appear to be blacklisted.*

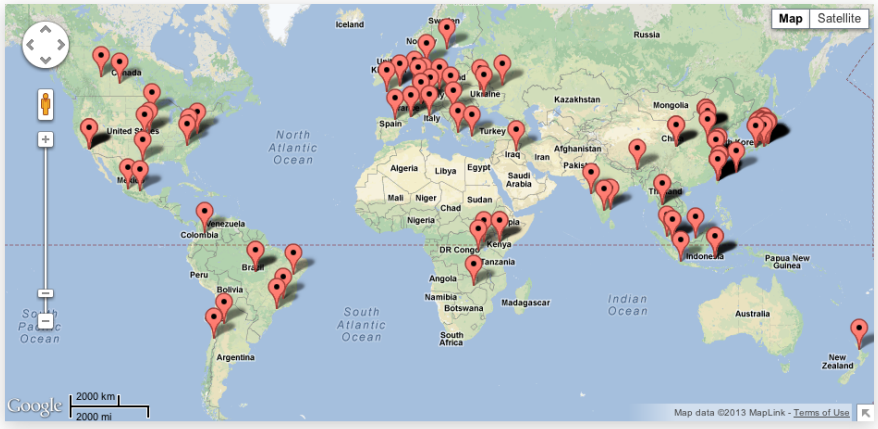


Figure 47: *The geolocation of announced prefixes blacklisted by Google over IPv6.*

day, and are now blacklisted. We checked and our probe appears to be behind such a blacklisted resolver. Google's blacklist is dynamically changing. As a result, a backend scheduled job is provisioned to periodically update the raw data and regenerate the plots. The periodicity is currently set to a month. A webpage (http://googleipv6.vaibhavbajpai.com) has been created to keep the plots updated and allow further interactivity.

8.5 CONCLUSION

We have performed a study using a metric that measures TCP connection establishment times to 100 dual-stacked websites from SamKnows probes connected behind both residential and NREN. Using a year-long dataset derived from these measurements we showed how popular websites cluster

around CDN deployments. We showed how multiple websites are served from CDN caches deployed within access networks. We also witnessed cases where these CDN caches were present for IPv4, but were largely absent for IPv6 leading to relatively higher TCP connection establishment times. We also showed how CDN clusters and number of websites within each cluster vary depending on the used address family. The distributions of connection setup times revealed how IPv6 connectivity to popular CDN deployments have improved over time. We showed how `www.bing.com` stopped providing websites over IPv6 since Sep 2013 and how Google employ blacklists to block some hosts from receiving their services over IPv6.

9 MEASURING EFFECTS OF HAPPY EYEBALLS

The IETF has developed protocols that promote a healthy IPv4 and IPv6 co-existence. The HE algorithm, for instance, prevents bad user experience in situations where IPv6 connectivity is broken. Using an active test (`happy`) that measures TCP connection establishment times, we evaluate the effects of the HE algorithm. The `happy` test measures against ALEXA top 100 dual-stacked websites from 80 SamKnows probes connected behind dual-stacked networks. Using a 3-years long (2013 - 2016) dataset, we show that TCP connect times to popular websites over IPv6 have considerably improved over time. As of Jan 2016, 5% of these websites are faster over IPv6 with 90% being atmost 1 ms slower. The historical trend shows that only around 1% of the TCP connect times over IPv6 were ever above the HE timer value (300 ms), which leaves around 2% chance for IPv4 to win a HE race towards these websites. As such, 99% of these websites prefer IPv6 connections more than 98% of the time. We show that although absolute TCP connect times (in ms) are not that far apart in both address families, HE with a 300 ms timer value tends to prefer slower IPv6 connections in around 90% of the cases. We show that lowering the HE timer value to 150 ms gives us a margin benefit of 10% while retaining same preference levels over IPv6.

Contents

9.1 INTRODUCTION

The HE algorithm [40] (2012) provides recommendations to application developers to help prevent bad user experience in situations where IPv6 connectivity is broken. The algorithm when combined with the default address selection policy [39] (2012), gives a noticeable advantage (300 ms) to connections made over IPv6. The HE timer value was chosen during a time when IPv6 brokeness was quite prevalent, which made applications stall for several seconds before attempting a connection over IPv4. For instance, Savolainen *et al.* in [273] (2011) reported browser connection timeouts to be in the order of 20 seconds. A 300 ms HE timer value allowed applications to fast fallback to IPv4 in such situations. The IPv6 brokeness has been largely attributed to failures caused by Teredo [41] and 6to4 relays [42]. Studies [274, 5] have shown that even in situations where relays work, Teredo/6to4 add noticeable latency when compared to native IPv4 and IPv6. With considerable efforts made by the IPv6 operations community, these transition mechanisms appear

to steadily decline over the last 5 years. For instance, Christopher Palmer in [275] (2013) announced that Microsoft will stop Teredo on Windows and deactivate its public Teredo servers in 2014. The 6to4 anycast prefix recently has been obsoleted [276] (2015) and future products are recommended to not use 6to4 anycast anymore. Geoff Huston [277] (2016) recently showed that as a consequence, failure rates over IPv6 have dropped from 40% (2011) to 3.5% (2015). In fact unicast IPv6 failure rates have also gone down from 5.3% (2011) to 2% (2015).

Today, IPv6 adoption has reached 10.2% (native) with Teredo/6to4 at around 0.01% according to Google IPv6 adoption statistics [27] (as of Feb 2016). The Google over IPv6 (whitelist) program no longer exists, but has been replaced by an IPv6 blacklist [9]. In fact, today Google will not return AAAA entries to DNS resolvers where latency over IPv6 is consistently 100 ms or more slower [271] than IPv4. In such a changed landscape, the effect of the HE timer value (300 ms) on the overall experience of a dual-stacked user remains largely unclear. We want to know: a) What are the percentage of cases where HE makes a bad decision of choosing IPv6 when it's slower. Furthermore, in such situations what is the amount of imposition (in terms of latency impact) a dual-stacked user has to pay as a result of the high HE timer value. This is critical since applications on top of TCP not only apply HE in scenarios where IPv6 connectivity is broken, but also in scenarious where IPv6 connectivity is comparable.

The fragmentation of the algorithm due to the high HE timer value is visible in implementations today. For instance, Firefox (since v15) [43] and Opera (since v12.10) [44] by default use parallel TCP connections over IPv4 and IPv6. Firefox also provides a parameter to disable the fast fallback option, after which it prefers IPv6 using a 250 ms timer value. Apple (since OS X 10.11 and iOS 9) [46] uses a considerably smaller 25 ms timer value in favor of IPv6 connections. Google Chrome (since v11) [45] is the only browser that sticks to the 300 ms timer value. Note, these values are arbitrarily chosen. We want to empirically determine the right HE timer value that provides the same preference levels over IPv6 as is today but also reduces the performance penalty in situations where IPv6 is considerably slower.

Towards this pursuit, we have developed an active test (`happy`) [9] that measures TCP connection establishment times. We deploy this test on 80 geographically distributed SamKnows [4] probes connected behind dual-stacked networks (see Fig. 48) to provide diversity of network origins. The test measures against ALEXA top 100 dual-stacked websites. Using a 3-years long (2013 - 2016) dataset of TCP connection establishment times obtained from our metric, we are able to calculate decisions a HE enabled application would have taken. We are also able to experiment with variations of the HE algorithm and propose changes to it.

Our **contributions** – a) We show that TCP connect times to popular websites over IPv6 have considerably improved over time. As of Jan 2016, 5% of these websites are faster over IPv6 with 90% being atmost 1 ms slower. b) Only around 1% of the TCP connect times over IPv6 were ever above the HE timer value (300 ms), which leaves around 2% chance for IPv4 to win a HE race towards these websites. As such, 99% of these websites prefer IPv6 connections more than 98% of the time and c) Although absolute TCP connect times (in ms) are not that far apart in both address families, HE with 300 ms timer value tends to prefer slower IPv6 connections in around 90% of the cases. A lowering of the HE timer value to 150 ms gives us a margin benefit of 10% while retaining same preference levels over IPv6.

NETWORK TYPE	#
RESIDENTIAL	55
NREN / RESEARCH	11
BUSINESS / DATACENTER	09
OPERATOR LAB	04
IXP	01

RIR	#
RIPE	42
ARIN	29
APNIC	07
AFRINIC	01
LACNIC	01

Figure 48: *Measurement trial of 80 dual-stacked SamKnows probes as of Feb 2016. The entire metadata for each probe is available online: http://goo.gl/PwD4yN*

The chapter is organized as follows. In Section 9.2 we discuss background on the HE algorithm, associated browser implementations and related work. Insights derived from the data analysis are presented in Section 9.3 with high-level conclusions in Section 9.4.

9.2 BACKGROUND

The function, `getaddrinfo(...)` resolves a dual-stacked website to a list of endpoints in an order that prefers IPv6 endpoints [39] (2012). The dictated order can dramatically reduce the application's responsiveness in situations where IPv6 connectivity is broken. In fact, an attempt to connect over an IPv4 endpoint will only take place when the IPv6 connection attempt has timed out, which can be in the order of several seconds. The HE algorithm recommends that a host, after resolving the DNS name of a dual-stacked website, tries a TCP `connect(...)` to the first endpoint (usually IPv6). However, instead of waiting for a timeout, which is typically in the order of seconds, it only waits for 300 ms, after which it must initiate another TCP `connect(...)` to an endpoint with a different address family and start a competition to pick the one that completes first.

The HE algorithm biases its path selection in favor of IPv6 by design. The connection establishment race has been handicapped to: a) prefer IPv6 paths and reduce contention towards the critical IPv4 address space in CGN deployments, b) move IPv4 traffic (usually billed) to IPv6 networks and reduce costs, and c) reduce load on load balancers and peering links on the

IPv4 paths. The HE algorithm honors this IPv6 upgrade policy. It is therefore designed to not encourage aggressive connection requests over IPv4 and IPv6, but instead to satisfy the following goals: a) The connection requests must honor the destination-address selection policy [39], unless overriden by user or network configuration. The client must prefer IPv6 over IPv4 whenever the policy is not known, b) The connection initiation must quickly fallback to IPv4 to reduce the wait times for a dual-stack host in situations where the IPv6 path is broken, and c) The network path and destination servers must not be thrashed by mere doubling of traffic by making simultaneous connection requests over IPv4 and IPv6. The connection requests over IPv6 must be given a fair chance to succeed to reduce load on IPv4, before a connection over IPv4 is attempted.

9.2.1 *Browser Implementations*

Google Chrome has an implementation of the HE algorithm since v11.0.696.71 [45], which was released in 2011. It uses a 300 ms timer, which is fired after the first TCP SYN request has been sent. Once the timer expires the browser switches to a different address family and starts a competition between IPv4 and IPv6 connection requests.

Mozilla Firefox released its first HE implementation with v7.0. The implementation received multiple bug reports leading to a stable implementation by v15.0 [43]. Firefox by default, unlike Chrome follows a more aggressive approach by starting parallel TCP connections to the first endpoints of each address family. However, once one of the connections has been successfully established, the second connection request is not closed by sending a TCP RST, instead the connection request is allowed to continue until exhaustion. Opera, since v12.10 [44] has an implementation similar to that of Mozilla Firefox. It tries simultaneous TCP connections to the first endpoint of each address family and chooses whichever completes first. It remains unclear whether parallel connection attempts can be deemed as a flavor of HE, since the algorithm is designed to honor the IPv6 upgrade policy and therefore does not encourage aggressive connection requests over IPv4 and IPv6. As such, Firefox also allows to set a parameter, `network.http.fast-fallback-to-IPv4` to `false`, after which the browser starts preferring IPv6 connection requests with a 250 ms timer value in favour of IPv6.

Apple Safari prior to OS X 10.11 (since OS X 10.7) [278] used a more hybrid approach. The OS X networking APIs maintained a history of the previously witnessed latencies to each destination along with a combined mean for each address family. Safari instead of using `getaddrinfo(...)` used these higher level APIs to prefer the fastest connection. Moreover, Safari did not switch to a different address family if no response was received from the first endpoint, instead it tried a TCP connection with the next endpoint in the same address family. This took a long time for an address family switch-over. Apple with OS X 10.11 and iOS 9 has a new simplified HE implementation [46] which uses a 25 ms timer value.

9.2.2 *Related Work*

Jakub Czyz *et al.* in [7] (2014) provide a survey of studies measuring IPv6 adoption on the Internet. We in [9] (2015) have recently provided a short survey on studies measuring IPv6 performance. In this work, we therefore scope our survey to studies measuring HE.

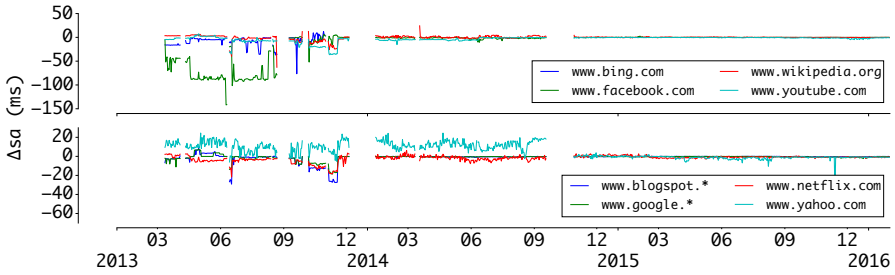


Figure 49: *Time series (gaps represent missing data) of absolute difference in TCP connect times to dual-stacked websites. TCP connect times to popular websites over IPv6 have improved over time.*

Studies [279, 280, 281] (2011) in the past have analysed HE implementations in Firefox 7 and 8, Google Chrome 11, Opera 11 and Apple Safari on OS X 10.7. It was witnessed that Google Chrome (with a 300 ms timer) helps reduce the degraded user experience in situations where a dual-stacked host's IPv6 connectivity is broken. Firefox (with the fast fallback parameter disabled) HE behaviour is similar to Google Chrome. Apple Safari on OS X 10.7 tends to prefer the fastest connection, but in the process also prefers legacy IPv4 connectivity even where IPv6 connectivity is relatively similar, a situation referred to as *hampering eyeballs*, since it tends to delay the transition to IPv6. These studies however are dated. Fred Baker in [282] (2012) describes HE metrics and testbed configurations in a controlled setting to measure how quickly an application can reliably establish connections from a dual-stacked environment. Sebastian Zander *et al.* in [283] (2012) showed that 20% of the hosts had a HE implementation, out of which 75% of the connection attempts preferred IPv6. We show that this preference (due to decreased latencies over IPv6) has increased to 98% today. They observed that HE was used by hosts running Chrome (9% of connections), Safari (4%) and Firefox (1%). We recently showed [91] (2013) that HE (with a 300 ms timer value) never prefers IPv6 using Teredo except in situations where IPv4 reachability of the destination endpoint is broken. We further showed [10] (2015) that HE (with a 300 ms timer value) prefers a connection over IPv6 to YouTube media servers even when the measured throughput over IPv4 is better. This results in lower bit rates and lower resolutions when streaming a video than can be achieved if streamed over IPv4.

9.3 DATA ANALYSIS INSIGHTS

We performed a pre-processing run on the dataset to reduce the volume of raw measurements. In this work, we do not look at TCP connection failure rates. We further investigated time scales where the variation in TCP connect times is small enough to allow statistically meaningful aggregation. Since applications usually honor the order of endpoints returned by `getaddrinfo(...)` when establishing a TCP connection, we decided to pick the first endpoints returned in each measurement over a day for both address families, and aggregated their TCP connect times centered around the median. Each data point in subsequent analysis refers to the median of TCP connect times as seen by IPv4 and IPv6 endpoints over a day.

Let u denote a website identified by a URL. We call the time taken to establish a TCP connection towards a website u as $t(u)$. Since we study the impact of accessing websites using different IP protocols, we denote the

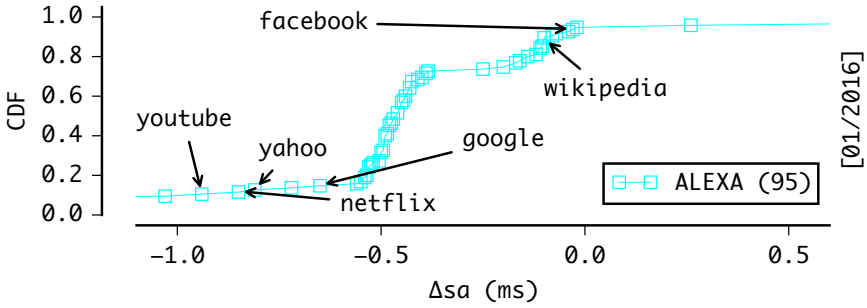


Figure 50: *CDF of absolute difference of TCP connect times between IPv4 and IPv6 as of Jan 2016. 5% of the top 100 dual-stacked ALEXA websites are faster over IPv6 today, although 90% are atmost 1 ms slower.*

TCP connect time of u accessed over IP version v as $t_v(u)$. We use *slowness* to adjudicate the performance difference over IPv4 and IPv6. We use both absolute slowness (s_a) and relative slowness, (s_r) as described in Eq. 9.1. Absolute slowness (s_a) is the difference between TCP connect times over IPv4 and IPv6, while relative slowness (s_r) is the fraction of absolute slowness over the observed TCP connect times using IPv4.

$$\begin{aligned} \Delta s_a(u) &= t_4(u) - t_6(u) \\ \Delta s_r(u) &= \frac{\Delta s_a(u)}{t_4(u)} \end{aligned} \tag{9.1}$$

We use $\hat{\Delta}s_a(u)$ and $\hat{\Delta}s_r(u)$ to represent the median of the sample of $\Delta s_a(u)$ and $\Delta s_r(u)$ values across all probes respectively. The median is taken to ensure measured performance does not get biased by a specific vantage point. This terminology will be used in the rest of the data analysis.

9.3.1 *Trends*

Fig. 49 shows timeseries of absolute slowness, $\hat{\Delta}s_a(u)$ towards popular dual-stacked websites. Note, observations from all google and blogspot websites are clubbed together as `www.google.*` and `www.blogspot.*` since they are served by the same CDN [9] and therefore tend to offer similar performance. It can be seen that TCP connect times to popular websites over IPv6 appear to have considerably improved over time. It can also be noticed that `www.bing.com` permanently stopped (even though `www.microsoft.com` and `www.office.com` are still IPv6 enabled) providing IPv6 services in Sep 2013. The time series however, does not reveal whether IPv6 is faster (or slower) today. It can be seen that there is marginal variation in 2016. As such, we aggregated the absolute slowness over Jan 2016. Fig. 50 shows the absolute slowness, $\hat{\Delta}s_a(u)$ for ALEXA top 100 dual-stacked websites as of Jan 2016. It can be seen that 5% of the websites are faster over IPv6 today, although 90% are atmost 1 ms slower. Around 6% of the websites are atleast 10 ms slower. `www.flipkart.com` is around 32ms slower with `www.qq.com` being around 364 ms slower (not shown) over IPv6. Facebook recently showed [284] (2015) that their news feeds load 30% faster over IPv6 from a US mobile service provider (undisclosed). Our analysis using more diverse vantage points reveals that `www.facebook.com` is as fast over IPv6.

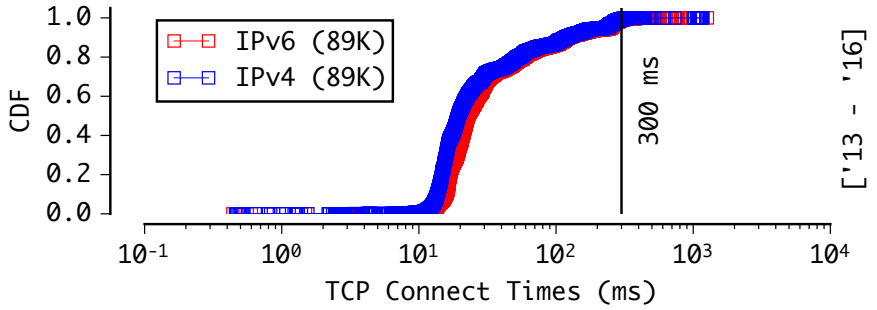


Figure 51: *CDF of TCP connect times over IPv4 and IPv6 over entire 3 years long duration. Only around 1% of the samples over IPv6 exhibit TCP connect times above 300 ms.*

9.3.2 *Measuring Preference*

Fig. 51 shows the distribution of TCP connect times over IPv4 and IPv6 over the entire 3 years long duration. As can be seen, only around 1% of the samples over IPv6 exhibit TCP connect times above the HE timer (300 ms) value. In fact 90% of the samples over IPv6 are below 100ms with 82% of the samples below 50ms. Similarly, 86% of samples over IPv4 are below 50 ms with 75% below 30 ms. Fig. 52 shows the preference calculated over a 3 year long dataset using the HE timer (300 ms) value. It can be seen that during the last 3 years, all probes (sources) preferred IPv6 atleast 93% of the time with 99% of probes preferring it more than 98% of the time. Similarly TCP connections over IPv6 to 99% of websites (destinations) were preferred more than 98% of the time. Note, the probe CDF is invariant of the websites (out of 100 samples), while the website CDF is invariant of the probes (out of 77 samples) The only probe with less than 98% (93.5%) IPv6 preference is a probe behind a TWC subscriber. The subscriber has a Motorola SB6183 cable modem which is known [285] to drop TCP segments over IPv6 when the TCP timestamp option is set (set by default in Linux). As such each TCP SYN packet lost can add to a second delay thereby perturbing the IPv6 preference calculation from this vantage point. This is the reason why we prefer to take median aggregation across all probes to remove bias introduced by issues closer to the vantage point. The only website with less than 98% (28.7%) IPv6 preference is `www.qq.com`, where we witnessed that all TCP connect times over IPv6 are more than 200ms with 63% of the values being more than 300 ms. On the other hand, 88% of TCP connect times over IPv4 are less than 50 ms. This makes about half of the probes to not prefer connecting over IPv6 to this website with 80% of probes having less than 50% preference over IPv6. We can conclude that with a HE 300 ms advantage, a dual-stack host tends to use IPv4 connections only around 2% of the time.

9.3.3 *Measuring Slowness*

Fig. 53 shows relative slowness $\hat{\Delta}s_r(u)$ for situations where HE prefers IPv6 using the 300 ms timer value. Note, this only includes cases where HE prefers connections over IPv6. The positive values on x-axis represent samples where IPv6 is faster which is around 10% of the total samples. IPv6 is more than 10% faster in around 3% of the samples. On the other hand, IPv6 is more than 2% slower in half of the samples with being more than 20% slower in

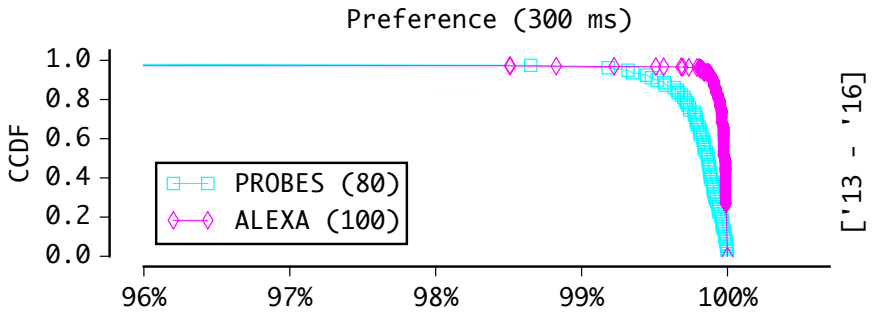


Figure 52: *Complementary CDF of TCP connection establishment preference over IPv6 both from source (probes) to destinations (websites). A 300 ms timer value leaves around 2% chance for IPv4 to win a HE race to popular dual-stacked ALEXA websites.*

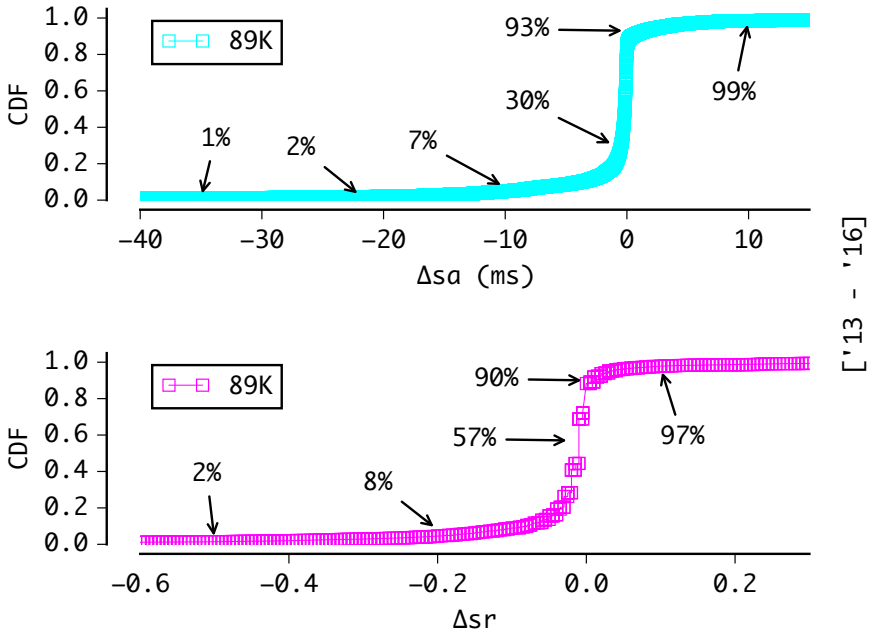


Figure 53: *CDF of absolute (above) and relative (below) difference of TCP connection establishment times over IPv4 and IPv6 for situations where HE prefers IPv6 using 300 ms timer value. HE tends to prefer slower IPv6 connection in around 90% of the samples, but absolute TCP connect times are not that far apart from IPv4.*

8% of the samples. Worse, it is more than 50% slower in 2% of the samples. Fig. 53 also shows the corresponding absolute slowness, $\hat{\Delta}s_a(u)$. It can be seen that around 7% of the samples exhibit TCP connect times that are atleast 1ms faster over IPv6 with around 1% samples that are atleast 10ms faster. On the other hand, around 30% of the samples are atleast 1ms slower with 7% of samples that are atleast 10ms slower. In fact only 2% of the samples are atleast 22 ms slower with 1% samples being atleast 35 ms slower over IPv6. As such, IPv6 may be slower in 90% of the cases where HE prefers it, but the TCP connect times are not that far apart from IPv4. We know that a 300 ms timer value leaves around 2% chance for IPv4 to win a HE race (see Fig. 52). In 90% of these cases, HE tends to prefer slower IPv6 connection. This shows that the timer value (300 ms) used by the HE algorithm has past its time and

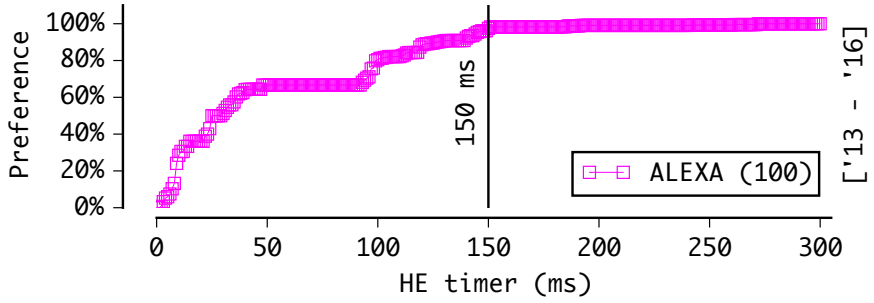


Figure 54: *TCP connection establishment preference over IPv6 towards ALEXA websites by varying the HE timer value. A HE timer value of 150 ms allows same 99% of the websites to still prefer connections at least 98.5% of the time.*

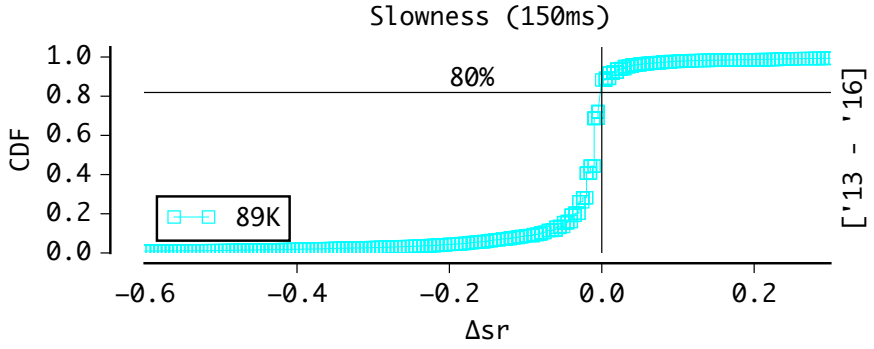


Figure 55: *CDF of relative difference of raw TCP connection establishment times over IPv4 and IPv6 for situations where HE prefers IPv6 using 150 ms timer value. By lowering the timer, HE tends to save 10% (8.9K) of the samples from prefering slower IPv6 connections.*

is not suitable in today's landscape. Perhaps a lower HE timer value can give the same (99%) preference to IPv6 (see Fig. 52) but not penalise IPv4 in rare cases where IPv6 is (such as `www.qq.com`) slower.

9.3.4 *HE Timer by Preference*

We experimented by lowering the HE timer advantage. We know that by using 300 ms HE timer, IPv6 connections to 99% of ALEXA websites are preferred more than 98.6% of the time (see Fig. 52). The idea towards finding a better HE timer is to control these two parameters (99% websites prefer IPv6 connections 98.6% of the time) and lower the HE timer value to see until when this precedence remains true. This is important because the timer value cannot be lowered to zero (parallel connections over IPv4 and IPv6), since HE must still adhere to the IPv6 upgrade policy (see Sec. 2.1) to prefer IPv6 paths. As such, the timer value by design should give IPv6 a fair chance to succeed to reduce load on IPv4, but at the same time reduce wait time for a dual-stack host in situations where IPv6 is considerably (such as `www.qq.com`) slower. Fig. 54 shows TCP connection establishment preference over IPv6 towards ALEXA websites by varying the HE timer value. Each data point is the 1^{th} percentile preference towards dual-stacked websites. As can be seen, a HE timer value of 150 ms allows same 99% of the websites to still prefer connections at least 98.5% of the time. Fig. 55 shows that lowering

HE timer to 150 ms gives us a margin benefit of 10%. A 300 ms timer value preferred 90% of the connections where IPv6 was slow (see Fig. 53) which has been reduced to 80% with a 150 ms timer value. This means 10% (around 9K connections with a daily aggregate) of the samples where $t_6(u)$ is at least 150 + $t_4(u)$ ms but less than 300 + $t_4(u)$ ms (because HE timer with 300 ms was preferring IPv6 in these cases) now prefer IPv4 because the timer cuts it early. These may be cases where content over IPv6 is served from a different continent. The new HE timer value is ideal because it comes with no IPv6 preference penalty to observed dual-stacked websites.

9.4 CONCLUSION

We measured the effects of the HE algorithm. Using a 3-years long trend, we showed that TCP connect times to popular dual-stacked websites over IPv6 have improved over time. As of Jan 2016, 5% of the top 100 dual-stacked ALEXA websites are faster over IPv6 and 90% are atmost 1 ms slower. A 300 ms timer value therefore leaves only around 2% chance for IPv4 to win a HE race to these websites. In 90% of these cases, HE tends to prefer slower IPv6 connection, although the TCP connect times are not that far apart from IPv4. We showed that a HE timer value of 150 ms provides a margin benefit of 10% while retaining similar IPv6 preference levels for 99% of the dual-stacked websites.

10 MEASURING YOUTUBE

We measure the performance of YouTube over IPv6 from 80 SamKnows probes connected to dual-stacked networks representing 58 different ASes. Using a 21-months long (Aug 2014 - Apr 2016) dataset, we show that success rates of streaming a stall-free version of a video over IPv6 have improved over time. We show that a HE race during initial TCP connection establishment leads to a strong (more than 97%) preference of IPv6. However, even though clients prefer streaming videos over IPv6, we show that the observed performance over IPv6 is worse than IPv4. We witness consistently higher TCP connection establishment and startup delays (100 ms or more) over IPv6. We also observe consistently lower achieved throughput both for audio and video over IPv6. We observe less than 1% stall rates over both address families and reduced stall durations over the years. Due to lower stall rates, bitrates that can be reliably streamed over both address families are comparable. However, in situations, where a stall does occur, 80% of the samples experience higher stall durations that are at least 1s longer over IPv6 when compared to IPv4. We also witness disparity in the availability of content caches, whereby content caches over IPv6 are largely absent.

Contents

10.1 INTRODUCTION

The Internet is rapidly exhausting IPv4 address space [26], which has prompted global initiatives (such as the World IPv6 Launch day [25] in 2012) to promote the deployment and adoption of IPv6. Within a span of 4 years since the initiative, global adoption of IPv6 [7] has increased to around 11.25% (as of April 2016) according to Google IPv6 adoption statistics [27] with Belgium (42.5%), Switzerland (27.3%), US (25.4%) and Germany (24.2%) leading IPv6 adoption rates. This has largely been possible due to spear-headed IPv6 deployment by service providers both in the fixed-line (such as Telenet, Belgacom, VOO in Belgium, Swisscom in Switzerland, Comcast in US, Deutsche Telekom and Kabel Deutschland in Germany) and cellular (such as AT&T, Verizon Wireless and T-mobile USA) space.

Nadi Sarrar *et al.* in [47] (2012) show that IPv6 traffic after the World IPv6 Day in 2011 is largely dominated by services running over HTTP and that YouTube is the primary service over HTTP that contributes heavily to large volumes of IPv6 traffic. Today, AMS-IX on a daily basis witnesses up to

NETWORK TYPE	#
RESIDENTIAL	55
NREN / RESEARCH	11
BUSINESS / DATACENTER	09
OPERATOR LAB	04
IXP	01

RIR	#
RIPE	42
ARIN	29
APNIC	07
AFRINIC	01
LACNIC	01

Figure 56: *Measurement trial of 80 dual-stacked SamKnows probes as of Apr 2016. The entire metadata for each probe is available online: http://goo.gl/PwD4yN*

57.8 Gbps / 4.1 Tbps of IPv6 traffic (as of May 2016) with timing of peaks aligned over both address families [286]. Fixed-line service providers such as Comcast and Swisscom estimate IPv6 traffic within their network to be around 25% of the total traffic [287]. In terms of traffic volume this is more than 1 Tbps of native IPv6 traffic (as of July 2014) as witnessed by Comcast. Furthermore, Swisscom reports (as of October 2014) that 60% of their IPv6 traffic is served by YouTube (with 5% by Facebook) alone [287]. As can be seen, YouTube is the single largest source of IPv6 traffic. This suggests that measuring the performance of YouTube content delivery over IPv6 is not only feasible but necessary today. We want to know: Do users experience benefit (or suffer) from streaming YouTube videos over IPv6? Towards this pursuit, we developed an active test (`youtube`) [10] (2015) that compares YouTube performance over IPv4 and IPv6. We deployed this test on 80 geographically distributed SamKnows [4] probes (see Fig. 56) to provide diversity of network origins. These probes receive native IPv6 connectivity and belong to different ISPs covering 58 different IPv4 and IPv6 ASes. In this work, we perform analysis using a 21-months long (August 2014 - April 2016) dataset collected from these dual-stacked probes.

Our **contributions** — a) We show that success rates (see Section 10.4) of streaming a stall-free version of a video over IPv6 have improved over time. b) We show that a HE race during initial TCP connection establishment leads to a strong (more than 97%) preference (see Section 10.5) to stream audio and video content over IPv6. c) Even though clients prefer streaming videos over IPv6, we show that the observed performance over IPv6 is worse. We witness consistently higher TCP connection establishment and startup delays

(100 ms or more) (see Section 10.6) over IPv6. d) Furthermore, we observe consistently lower achieved throughput (see Section 10.7) both for audio and video streams over IPv6, although the throughput difference has improved over time. e) We observe less than 1% stall rates (see Section 10.8) over both address families and stall durations tend to have reduced over the years. Due to lower stall rates, bitrates that can be reliably streamed over both address families are comparable. However in situations where a stall does occur, 80% of the samples experience stall durations that are at least 1s longer over IPv6. f) We also witness that 97% of our probes receive content delivery through a content cache (see Section 10.9) over IPv4 while only 5% receive it from a content cache over IPv6.

10.2 RELATED WORK

A number of studies have focussed on characterization [288, 289] (2007) of YouTube videos to profile workload patterns, observe trends of popular videos, and impact of content duplication on system characteristics. These studies have been followed by a number of passive measurement efforts [290, 291] (2010-2011) to study traffic dynamics, load-balancing strategies and device / location-based user access patterns. We do not discuss them in detail, but we refer the reader to a survey [292] (2016) that discusses these related studies. We instead focus on active measurement studies. For instance, Vijay Kumar Adhikari *et al.* in [293] (2012) use PlanetLab vantage points to crawl a finite subset of YouTube videos to explore the logical organization of the YouTube infrastructure. Parikshit Juluri *et al.* in [294] (2013) use `Pytomo` [295], a Python client, to measure YouTube experience from within three ISP networks. They witnessed noticeable difference in experienced quality across ISPs. They reason that the selection mechanisms largely vary depending on the delivery policies and individual ISP agreements. Hyunwoo Nam, *et al.* in [296] (2016) introduce YouSlow, a browser-based plugin that can detect and report startup delay, rebuffering and bitrate change events during live playback of a YouTube video. They show that these are good metrics to quantify abandonment rates for short videos on YouTube. These studies however measure YouTube performance over IPv4 only. Studies measuring IPv6 performance [8, 6, 9, 297] (2011-2016) on the other hand have largely focussed on websites.

To the best of our knowledge, this is the first study to measure YouTube performance over IPv6. The study is a continuation of our previous work [10] (2015), where we presented preliminary results from a 20-days (Sep 2014) long dataset collected from a smaller sample of 21 probes deployed within the EU. This chapter presents results from probes that cover a much larger geographical area over a longer trial period of last 21 months.

10.3 METHODOLOGY

We have developed a `youtube` test [10] (2015) that downloads and mimics playout of YouTube videos. It measures TCP connect times, startup delay, achievable throughput, bitrate, number of stalls and stall durations as indicators of performance when streaming a YouTube video. The test takes a YouTube URL as input and scrapes the fetched HTML page to extract the list of container formats, available resolutions and URL locations of media servers. The test then establishes two concurrent HTTP sessions to fetch audio and video streams in the desired format and resolution. The client

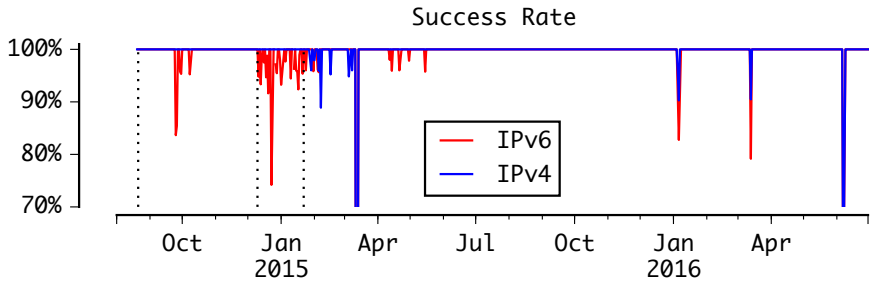


Figure 57: *Time series of success rates to YouTube. Success rates over IPv6 have improved over time.*

ensures temporal synchronization between the audio and video streams. The test does not at any time render content, but it only reads the container format to extract frame timestamps. The payload is subsequently discarded.

We deployed the `youtube` test on 80 SamKnows probes (see Fig. 56) connected in dual-stacked networks representing 58 different origin ASes. To put numbers into perspective, this is more than the number of CAIDA Archipelago (Ark) [14] probes (62 as of May 2016) with native IPv6 connectivity. The YouTube test runs twice, once for IPv4 and subsequently for IPv6 and repeats every hour. We use the YouTube Data API [298] to measure performance of globally popular videos. The popularity list is generated on the SamKnows backend and is refreshed every 12 hours. Probes pull this list on a daily basis. This allows us to measure the same video for the entire day, which enables temporal analysis, while cycling videos on a daily basis allows larger coverage of videos (around 458) with different characteristics.

We further refer the reader to our previous work [10] (2015) for a more detailed description of our methodology. The rest of the chapter presents analysis using a 21-months long (Aug 2014 - Apr 2016) dataset collected from these probes.

10.4 SUCCESS RATE

We start by comparing the success rate of execution of the test over both address families. We define *success rate* as the number of successful iterations to the total number of iterations of the test. The test is deemed successful when it successfully downloads a stall-free version of the video. When a stall occurs, the test reports an error and restarts by stepping down to the same video of a lower bitrate. Fig. 57 shows the timeseries of median success rates over IPv4 and IPv6 across all probes on each day. Vertical markers indicate a rollout of a test update. We apply a median aggregate, to ensure success rates do not get biased by a specific vantage point. The spikes in the timeseries are not due to outages but an indication that the test experiences a stall and steps down to a lower resolution. It can be seen that success rates in 2014 and 2015 over IPv6 were worse than IPv4 but they appear to have considerably improved over time. We further investigate the distribution of success rates by removing cases where an error is reported due to a stall event. Fig. 58 shows the distribution of success rate (without stall events) over both address families as seen by all probes. The numbers in the legend represent the number of samples in the distribution. It can be seen that probes relatively achieve a slightly lower success rate over IPv6. For instance,

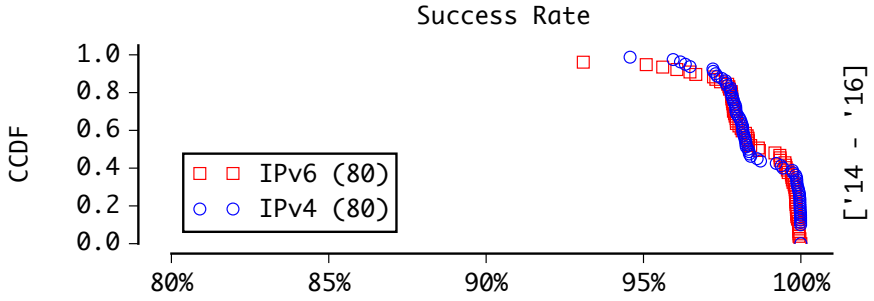


Figure 58: *CCDF of success rates over both address families. The probes successfully execute the test slightly more often over IPv4 than over IPv6.*

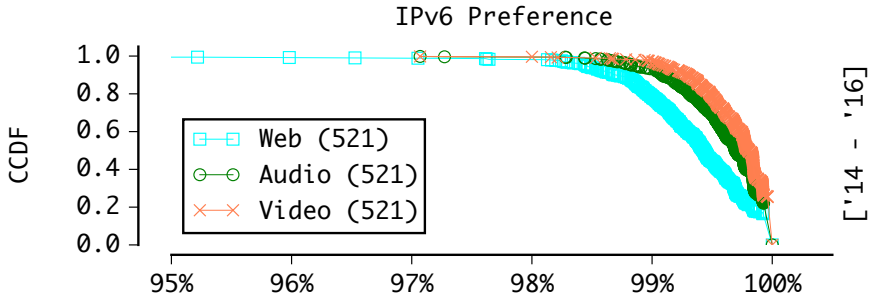


Figure 59: *CCDF of TCP connection establishment preference over IPv6. TCP connections over IPv6 to all audio and video streams are preferred at least 97% of the time.*

99% of the probes achieve success rate of more than 90% over IPv4, while 97% of probes achieve the same success rate over IPv6. We investigated the distribution of error codes reported during these failures. The slightly lower success rates over IPv6 are largely due to issues (such as network error, TCP timeouts or DNS resolution error) encountered closer to the vantage point. Going forward we perform analysis on the subset of results where the test reports success over both address families.

10.5 IPV6 PREFERENCE

We measure TCP connect times to the YouTube website as well as to media servers hosting audio and video streams. The test captures this by recording the time it takes for the `connect()` system call to complete. The DNS resolution time is not taken into account in this measure. This is important to measure because applications running on dual-stacked hosts will prefer connections made over IPv6. This is mandated by the destination address selection policy [39], which makes `getaddrinfo()` resolve DNS names in an order that prefers an IPv6 upgrade path. However, the Happy Eyeballs (HE) algorithm [40] allows these applications to switch to IPv4 in situations where IPv6 connectivity is bad. The connectivity is considered bad when connections made over IPv4 can tolerate the 300 ms advantage imparted to IPv6 and still complete the TCP connection establishment in less time. Fig. 59 shows the effects of the HE algorithm. It can be seen that TCP connections over IPv6 to all audio and video streams are preferred at least 97% of the time.

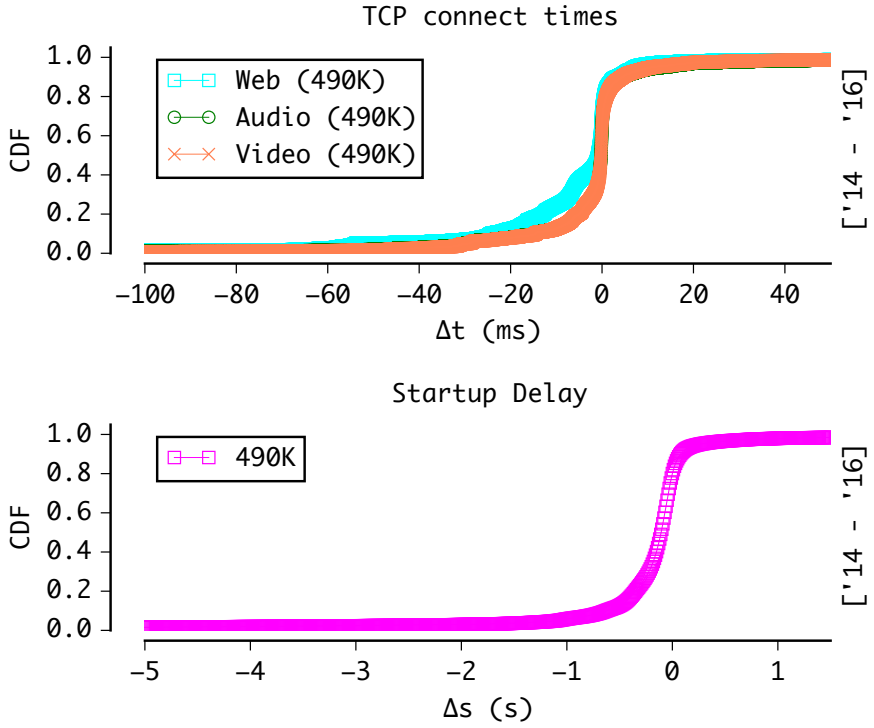


Figure 60: *CDF of difference of TCP connect times (above) and startup delay (below) between IPv4 and IPv6. 60% of the audio and video streams (and 73% of web connections) exhibit higher TCP connect times over IPv6 with 15% of them being at least 10 ms slower. 80% of the streams exhibit higher startup delay over IPv6 with 50% being at least 100 ms slower.*

10.6 STARTUP DELAY

We have seen that in situations where the test succeeds over both address families, clients strongly prefer streaming videos over IPv6. We now investigate how the observed performance over IPv6 compares to IPv4. We begin by defining a terminology. Let y denote a YouTube video identified by a URL. We call the time taken to establish a TCP connection towards y as $\mathrm{tc}(y)$. Since we study the impact of accessing YouTube using different IP protocols, we denote the TCP connect time of y accessed over IP version v as $\mathrm{tc}_v(y)$. Similarly, we denote prebuffering duration and startup delay of y accessed over IP version v as $\mathrm{pd}_v(y)$ and $\mathrm{sd}_v(y)$ respectively. We define *prebuffering duration* as the time it takes to fetch 2s of playable video from media servers. This timer is only triggered once the client has retrieved media server hostnames. As such, prebuferring duration exclusively captures the latency experienced while interacting with the media servers alone. We further define *startup delay* as the time measured from the start of the test until the end of prebuffering. This also involves the initial time it takes for the test to contact the YouTube web server, scrape the HTML page to extract hostnames of media servers and the aforementioned prebuffering duration. As such, startup delay captures the overall latency experienced for the video to start playing on the screen. DNS resolution times and TCP connect times are accounted in both prebuffering duration and startup delay.

Lower latency achieved using a combined effect of lower TCP connect times and lower startup delay is desirable for a good user experience. We use

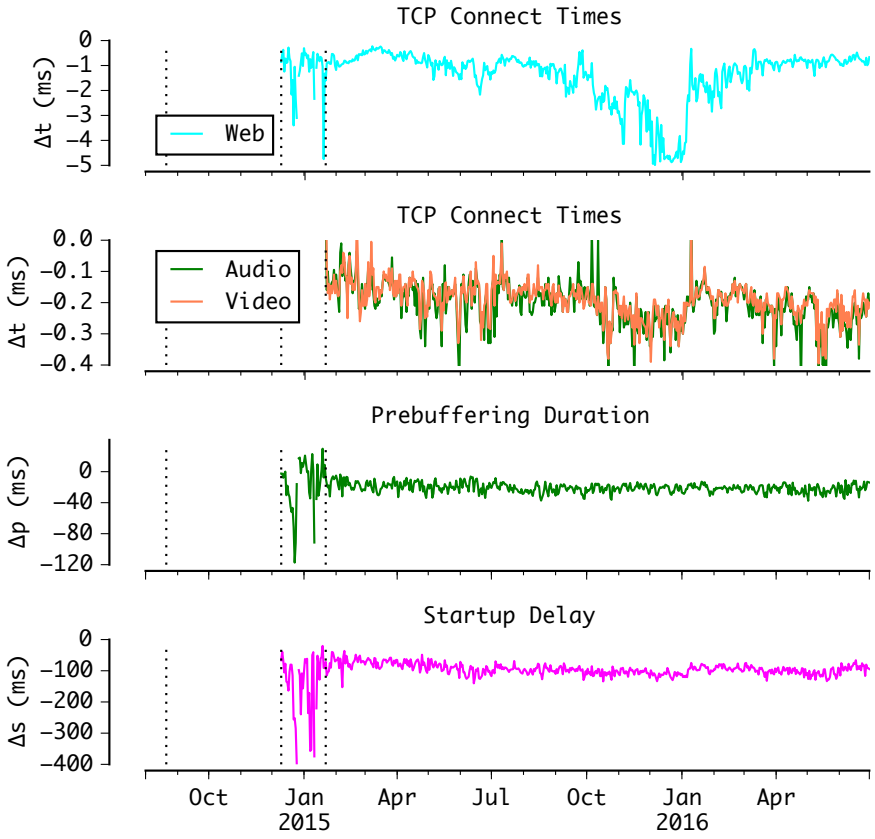


Figure 61: *Time series of difference in TCP connect times, prebuffering durations and startup delay over IPv4 and IPv6 to YouTube. Latency is consistently higher over IPv6 and has not improved over time. Higher prebuffering durations (25 ms or more) and higher startup delays (100 ms or more) are experienced over IPv6.*

Eq. 10.1 to calculate the latency difference over IPv4 and IPv6, where $\Delta t(y)$, $\Delta p(y)$ and $\Delta s(y)$ are the differences between TCP connect times, prebuffering durations and startup delay respectively,

$$\begin{aligned} \Delta t(y) &= tc_4(y) - tc_6(y) \\ \Delta p(y) &= pd_4(y) - pd_6(y) \\ \Delta s(y) &= sd_4(y) - sd_6(y) \end{aligned} \tag{10.1}$$

Fig. 60 shows the distribution of difference in TCP connect times $\Delta t(y)$ and difference in startup delay $\Delta s(y)$ using the entire 21 months long dataset. Values on the positive scale indicate that IPv6 is faster. The comparison of TCP connect times shows that 60% of the audio and video streams (and 73% of the web connections) are slower over IPv6 with 15% of them being at least 10 ms slower. The comparison of startup delay shows that 80% of the samples are slower over IPv6 with half of the samples being at least 100 ms slower.

We further apply a median aggregate on the TCP connect times, prebuffering duration and startup delay across all probes over each day. Fig. 61 shows the timeseries of median TCP connect times, prebuffering duration and startup delay over IPv4 and IPv6 across all probes. Vertical markers indicate a rollout of a test update. The values on the positive scale indicate that IPv6 is faster. Each of the sub figure is on a different y-scale. It can be seen that

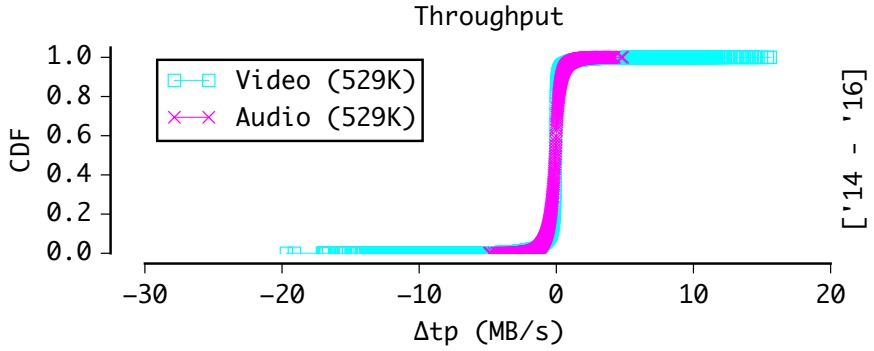


Figure 62: *CDF of difference of throughput between IPv4 and IPv6. 80% of the video samples and 65% of the audio samples achieve lower throughput over IPv6.*

TCP connect times tend to be consistently higher over IPv6 and have not improved over time. The TCP connect times towards the webpage appear much worse over IPv6 than towards media servers. Even though TCP connect times to fetch audio and video streams are only less than 1 ms slower over IPv6, they play a vital role since it's at this stage where the HE algorithm [40] chooses which address family should be preferred for streaming the video. As a result of a smaller difference in TCP connect times, HE prefers a TCP connection over IPv6. However, once the TCP connection is established, longer startup delays (100 ms or more) are experienced over IPv6. Since the prebuffering durations are not that far off (25 ms or more) over IPv6 compared to that of startup delay, it shows that it's the initial interaction with the web server that makes the startup delay (100 ms or more) worse over IPv6. Our initial observation of TCP connect times also revealed that web connect times over IPv6 are worse than TCP connect times to media servers. As such, even though the media content delivery is almost congruent over both address families, the web server interaction still needs to be optimised to reduce the increased startup delay experienced over IPv6.

10.7 THROUGHPUT

We have seen that clients strongly prefer streaming videos over IPv6, but they suffer from consistently higher TCP connect times, prebuferring durations (25 ms or more) and startup delays (100 ms or more) when compared to IPv4. We now investigate how the achieved throughput compares over both address families. The test measures throughput over a single TCP connection separately (and combined) over both audio and video streams. We denote the throughput of y accessed over IP version v as $tp_v(y)$. We use Eq. 10.2 to calculate the difference in achieved throughput over IPv4 and IPv6.

$$\Delta tp(y) = tp_6(y) - tp_4(y) \tag{10.2}$$

Fig. 62 shows the distribution of difference in achieved throughput $\Delta tp(y)$ for both audio and video streams using the entire 21 months long dataset. It can be seen that 80% of the video and 65% of the audio samples achieve lower throughput over IPv6. The test steps down to a lower resolution video once a stall event is triggered, which subsequently lowers the achieved throughput, since the test then chooses the next highest bit rate and begins the download from the beginning. This enables the test to produce a more user oriented

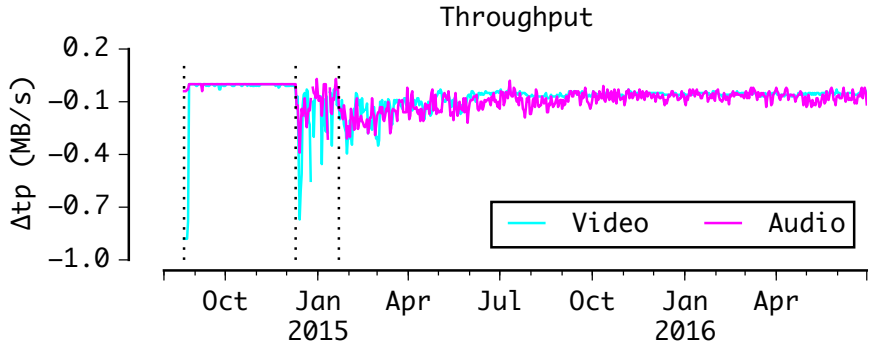


Figure 63: *Time series of difference in achieved throughput over IPv4 and IPv6. The achieved throughput is consistently lower over IPv6, but it has improved over time.*

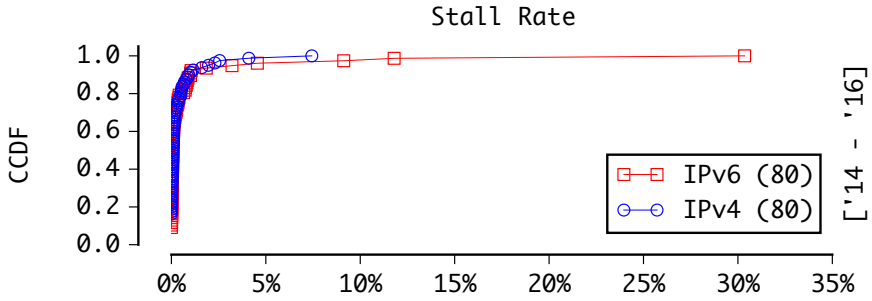


Figure 64: *CDF of stall rates over IPv4 and IPv6. 90% of the probes experience less than 1% stall rate both over IPv4 and IPv6.*

result in the form of the highest resolution that the client can play out without disruptions over a particular connection. The test is designed to pace the media streams to maintain a playout buffer of 40s (which means, the buffer can only store 40s of playable video) and must wait for the buffer to empty before requesting more frames. This behavior limits the achieved throughput for tests, making it depend largely on the video selected and network conditions.

We further apply a median aggregate on the throughput difference across all probes over each day. Fig. 63 shows the timeseries of median throughput difference over IPv4 and IPv6 across all probes. The values on the positive scale indicate that higher throughput is achieved over IPv6. It can be seen that achieved throughput both for audio and video streams tend to be consistently lower over IPv6, although the difference has reduced over time.

10.8 STALL EVENTS

We have seen that clients prefer streaming videos over IPv6, but the observed performance (both in terms of latency and throughput) over IPv6 is worse. We further compare the number of stall events and stall durations over both address families. We define a *stall* as an event that triggers during playback in situations when a frame is not received before its playout time. Stall events occur due to throughput constraints caused by a bottleneck at any point on the path between the media server and the client. To avoid unnecessary stalling we use results from SamKnows speed tests [10] to limit the maximum

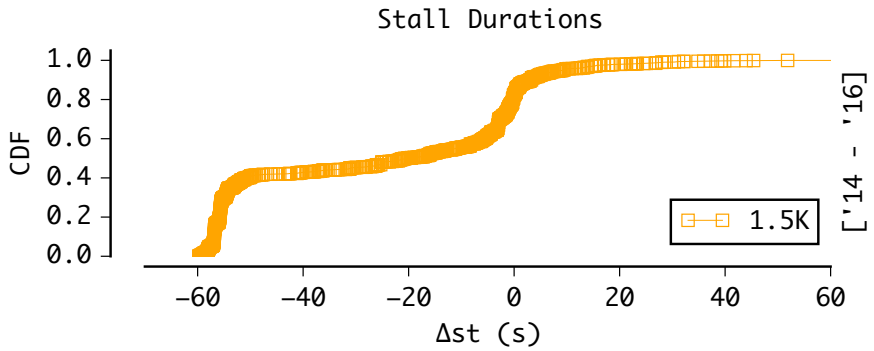


Figure 65: *CDF of difference of stall duration between IPv4 and IPv6. 80% of the samples experience stall durations that are at least 1s longer over IPv6.*

bit rate that the client will attempt to download. The test uses a playout buffer of 40s. In case a stall occurs, 1s of media rebuffering is done before resuming the playout timer. The media download is paced by downloading both streams in chunks instead of requesting the entire video. As such, this behavior is inline with a browser operation and prevents either of the streams from consuming too much bandwidth and causing unnecessary stall events. Fig. 64 shows the distribution of stall rates over IPv4 and IPv6 as seen by all probes. It can be seen that stall rates are comparable over both address families. 90% of the probes witness less than 1% stall rate over both address families. In order to analyse the effects of stalls on achieved bitrate, we utilise a metric, *bitrate reliably streamed* which [299] defines as the highest available bit rate that the test is able to download without experiencing stall events. Since the test cycles through different popular videos each day (which themselves may support different set of available resolutions), we further normalise this metric by taking the ratio of bitrate reliably streamed to the maximum available bit rate of the video. The ratio (br) lies between 0 and 1 where 1 is reported in situations when the test can successfully stream the highest available resolution without experiencing any stall events. We observe that 5.7% of the samples over IPv4, while a slightly larger 6.6% of the samples over IPv6 report a br value of less than 1. We further observe that 3% of the samples report a higher br value over IPv4, while a slightly lower 2% of the samples report a higher br value over IPv6. As such, since the stall rates are fairly low, the bitrate reliably streamed is also comparable over both address families.

In situations where a stall does occur, we further measure the durations of the stall. We use Eq. 10.3 to calculate the difference in stall duration over both address families, where $st_v(y)$ is the stall duration witnessed for video y accessed over IP version v.

$$\Delta st(y) = st_4(y) - st_6(y) \tag{10.3}$$

Fig. 65 shows the distribution of difference in stall duration $\Delta st(y)$ using the entire 21 months long dataset. The values on the positive scale indicate that stall durations are lower over IPv6. It can be seen that 80% of the samples experience stall durations that are at least 1s longer over IPv6 with half of them being at least 50s longer. We also apply a median aggregate on the stall durations across all probes over each day. Fig. 66 shows the median stall durations over IPv4 and IPv6 across all probes. It can be seen that stall

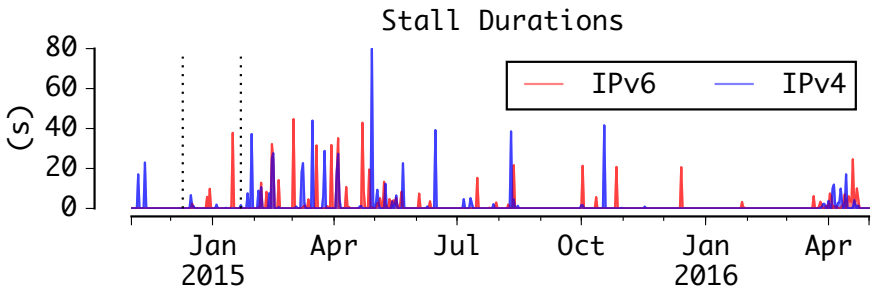


Figure 66: *Time series of stall durations over IPv4 and IPv6. Stall durations have reduced over time.*

durations in early 2015 over IPv6 were much higher but they appear to have reduced over time.

10.9 CONTENT CACHES

We have seen that clients prefer streaming videos over IPv6, but the observed performance over IPv6 is worse. Furthermore, in situations where a stall occurs, stall durations over IPv6 are also higher. We further investigate the reason for worse performance over IPv6. In order to improve content delivery, operators can deploy servers to host content caches within their networks. These caches form GGC [300] and help bring the content closer to the users, thereby improving performance and minimizing transit bandwidth. In our dataset, we identified GGC by looking up reverse DNS entries of media server IP endpoints. We searched for popular keywords in reverse DNS entries and filtered expressions such as `*-ggc.*.sky*` or `*.cache.google*.com` or `ggc*.plus.net` to flag endpoints as GGC nodes. We observe that 97% of probes over IPv4 receive content delivery through a GGC node while only 5% receive it over IPv6. We further flag an IP endpoint as a non-GGC cache (such as an Akamai / Cloudflare cache) if its reverse DNS entry does not match the GGC expressions and the IP endpoint belongs to the origin AS of the probe. This heuristic provides an indication that the content is served from within the service provider's network. In situations where the content is not served by a content cache, we mapped the IP prefixes to ASNs and used PeeringDB [250] to select ASNs that classify as content providers. This revealed that 96% of the probes do not get content served from a content cache over IPv6 but instead have to reach out to the Google CDN to fetch media streams.

10.10 CONCLUSION

We measured YouTube performance over IPv6. Using a 21-months long dataset we showed that success rates of streaming a stall-free version of a video over IPv6 were lower compared to that of IPv4 but they tend to have improved over time. In situations where the test succeeds over both address families, we witnessed that HE strongly prefers (more than 97%) connections made over IPv6 for streaming media content. This preference to IPv6 brings worse performance in comparison with IPv4, since we observed consistently higher TCP connect times and startup delays (100 ms or more) over IPv6. Furthermore, throughput achieved was also consistently lower over IPv6 for

both audio and video streams. Although we witnessed low stall rates over both address families and reduced stall durations over the years, in situations where a stall occurred, the stall durations were relatively higher (1s or more) over IPv6.

11 MEASURING WEB SIMILARITY

We compare the similarity of webpages delivered over IPv4 and IPv6. Using the SamKnows web performance (`webget`) test as the baseline, we implemented an extension (`simweb`), that allows us to measure the similarity of webpages. The similarity is calculated using well-known metrics that measure the content and service complexity of a webpage. The `simweb` test measures against ALEXA top 100 dual-stacked websites from 77 SamKnows probes connected behind dual-stacked networks. Using a two months-long dataset we show how 14% of these dual-stacked websites exhibit a dissimilarity in the number *of fetched webpage elements, with 94% of them exhibiting a dissimilarity in their* size*. We further show how 6% of these dual-stacked websites announce AAAA entries in the DNS but no content is delivered over IPv6 when an HTTP request is made. We also noticed several cases where* not all *webpage elements (such as images, javascript and CSS) of a dual-stacked website were available over IPv6. We show how 27% of the dual-stacked websites have some fraction of webpage elements that fail over IPv6, with 9% of the websites having more than 50% webpage elements that fail over IPv6. We perform a causality analysis and also identify sources for these failing elements. These failures tend to cripple experience for users behind an IPv6-only network and a quantification of failure cases may help improve IPv6 adoption on the Internet.*

Contents

11.1 INTRODUCTION

The IANA in 2011 allocated the last available IPv4 address block to RIRs thus depleting its pool of available IPv4 address space [26]. Since then, the RIRs are rapidly depleting their pool of IPv4 address space. For instance, APNIC (in Apr 2011), RIPE (in Sep 2012), LACNIC (in Jun 2014), and ARIN (in Sep 2015) have already exhausted their available pool; consequently LIRs now receive allocations from within the last available IPv4 /8 address block. The World IPv6 Launch day in 2012 [25] helped shift gears where a number of significant content and service providers joined efforts to expedite IPv6 [301] adoption. Within a span of 3 years since then, large IPv6 broadband rollouts have happened both in the fixed-line (such as Comcast and Deutsche Telekom) and cellular space (such as AT&T, Verizon Wireless and T-mobile USA). This has led to an increased global adoption of IPv6 to around 8% (as of Oct 2015) according to Google IPv6 adoption statistics [27] with Belgium (36.16%), Switzerland (23.25%) and the US (21.52%) leading IPv6 adoption.

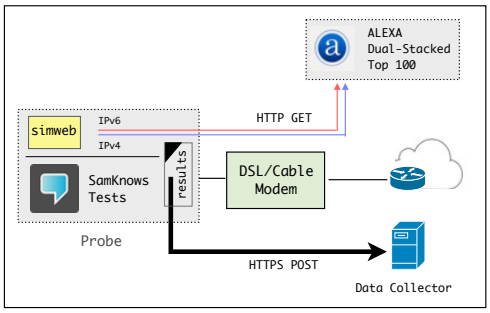
Jakub Czyz *et al.* in [7] (2014) provide a nice overview (particularly in address allocation, prefix announcements, DNS, AS-level connectivity, reachability, usage-profile and end-to-end latency) on the state of IPv6 adoption on the Internet. They [7] have shown how 3.5% (350) of ALEXA top 10K websites announce AAAA in DNS, with 3.2% (320) of these being reachable over IPv6. Recent studies [9], [8], [6] have compared performance of these dual-stacked websites over IPv4 and IPv6. However, there has been no study comparing the similarity of webpage content delivered over IPv4 and IPv6. This is important since applications running on top of TCP will prefer fetching webpages over IPv6 due to the default address selection policy [39] which prefers IPv6. As such, we want to know: *a) How similar are the webpages accessed over IPv6 to their IPv4 counterparts?* and *b) What factors contribute to the dissimilarity over IPv4 and IPv6?* A subjective study [302] recently compared dual-stacked webpage content from a single vantage point. The study revealed few cases where CSS webpage elements or flash advertisements were not available over IPv6. We build upon this observation by developing an active test (`simweb`) that uses well-known content and service complexity metrics [48] to quantify the level of webpage dissimilarity. We deploy this test on 77 geographically distributed SamKnows [4] probes connected behind dual-stacked networks to provide diversity of network origins. The test measures against ALEXA top 100 dual-stacked websites. Using a two-months long dataset we quantify the dissimilarity of dual-stacked webpages. In situations where there is a dissimilarity we also perform a causal analysis and identify sources responsible for the difference.

In this work we provide four main research contributions: a) `simweb`: A tool for measuring webpage similarity over IPv4 and IPv6. The tool is written in C and open-sourced for the measurement community. b) 14% of the ALEXA top 100 dual-stacked websites exhibit dissimilarity in the *number* of fetched webpage elements with 6% showing more than 50% difference. 94% of dual-stacked websites exhibit dissimilarity in *size* with 8% showing atleast 50% difference. This dissimilarity in number and size of elements negatively impacts webpages fetched over IPv6. c) 27% of dual-stacked websites have some fraction of webpage elements that fail over IPv6 with 9% of the websites having more than 50% webpage elements that fail over IPv6. Worse, 6% announce AAAA entries in the DNS but no content is delivered over IPv6 when an HTTP request is made. d) Failure rates are largely affected by DNS resolution error on images, javascript and CSS content delivered from both same-origin and cross-origin sources.

To the best of our knowledge this is the first study to measure webpage content similarity over IPv4 and IPv6. This is also the first study to investigate IPv6 adoption that goes beyond the root page of a dual-stacked website. The rest of the chapter is organized as follows. In Section 11.2 we discuss related work. The methodology describing our test, measurement setup, trial deployment and dataset are described in Section 11.3. The insights derived from the collected dataset are discussed in Section 11.4.

11.2 RELATED WORK

Mehdi Nikkhah *et al.* in [8] (2011) measure webpage performance within ALEXA top 1M websites (also used by us) over IPv4 and IPv6 from 6 vantage points (as opposed to 77 vantage points used by us). They measure the object size of the downloaded root page (without downloading embedded objects) and filter out websites where these sizes are not within 6% (over IPv4 and



NETWORK TYPE	# (↓)
RESIDENTIAL	53
NREN / RESEARCH	10
BUSINESS / DATA CENTER	09
OPERATOR LAB	04
IXP	01

Figure 67: *A measurement setup (left) on top of the SamKnows platform. A dual-stacked probe in addition to the standard SamKnows tests, executes a* `simweb` *test. The* `simweb` *test runs every hour against 100 dual-stacked websites both over IPv4 and IPv6. The locally collected measurement results are pushed every hour to a data collector using HTTP. A table on the right shows the distribution of these probes by network type. Most of these probes are deployed behind residential and research networks. All the probes receive native IPv6 connectivity.*

IPv6) of each other. Going forward, they measure average download speeds of the rest of the identifical pages. Amogh Dhamdhere *et al.* in [6] (2012) take this forward to measure webpage performance from 5 vantage points by downloading the smallest webpage element which is atleast 10K bytes in size. They also filter out websites where these sizes are not within 1% (over IPv4 and IPv6) of each other. However, in both works no further analysis on content dissimilarity is performed. In this study, we plug this gap by quantifying the amount of dissimilar dual-stacked websites with a causal analysis to identify potential areas for improvements. Troy Johnson *et al.* in [303] (2014) use a 2-year long `httparchive.org` dataset to compare similarity of desktop and mobile version of webpages. They show higher rates in the number of webpage requests for desktop versions but average bytes per request appear to show close proximity with mobile versions. They show how images, javascripts and flash content contributes heavily to webpage sizes for both versions.

11.3 METHODOLOGY

Metric / Implementation: We want to compare the similarity of webpages delivered over IPv4 and IPv6. A headless browser engine (such as `phantomjs`) would be an ideal candidate for such a study. However, due to storage limitations, it's currently not possible to port `phantomjs` to the OpenWrt platform (used by SamKnows probes). Therefore, we use the SamKnows web performance test (`webget`) as the baseline, to implement an extension (`simweb`), that we use to measure this similarity. `webget` (also called Mirage within the BISmark [18] platform) [28] is written in C and is part of the SamKnows measurement test suite. For a given website, `webget` downloads the root webpage and all its referenced webpage elements over IPv4. In the process it calculates the DNS lookup time, time to first byte, HTTP request time, total size and download speed to fetch all webpage elements of a given website. While `webget` provides an aggregated statistics report across all webpage elements of a website, we are interested in the individual statistical report of each webpage element. As such, `simweb` builds upon this test to report the content type, content size, resource URL, and IP endpoint used to fetch each webpage element. These properties are reported both over

RIR	# (↓)
RIPE	42
ARIN	26
APNIC	07
AFRINIC	01
LACNIC	01

Figure 68: *Geographical distribution (left) of our measurement trial comprising of 77 dual-stacked probes as of October 2015. Each vantage point is a SamKnows probe which is part of a larger SamKnows measurement platform. A table on the right shows the distribution of these probes by RIR region. Most of these probes are connected behind the RIPE and ARIN region. The entire probe metadata is released at: http://goo.gl/laik85*

IPv4 and IPv6. Given a hostname can point to multiple IP endpoints in DNS, `simweb` picks up the first IP endpoint returned by `getaddrinfo(...)` to establish the TCP connections both over IPv4 and IPv6. In addition HTTP status codes and underlying `curl` response codes are also used to identify the network level status of each request. The similarity is calculated in the data analysis phase using well-known metrics [48] that measure the content and service complexity of a webpage. We use the number and content sizes of fetched webpage elements to quantify the content complexity of a website. The service complexity of a website is quantified by classifying webpage elements to belong to same and cross-origin sources using hostnames derived from resource URLs.

Measurement Setup: We cross-compiled `simweb` for the OpenWrt platform and deployed it on SamKnows probes. These probes, in addition to the `simweb` test, also perform standard SamKnows IPv4 measurements. The `simweb` test runs twice, once for IPv4 and subsequently for IPv6 and repeats every hour. This is to ensure that the first HTTP request to fetch the root webpage and all subsequent HTTP requests to fetch the webpage elements are made over one specific address family only. The test measures against ALEXA top 100 (generated in 2013) dual-stacked websites [9]. It uses the user-agent string `Mozilla/4.0` when establishing HTTP session with the servers. Due to the inherent storage limitation of the probes, the locally collected measurement results are pushed every hour to our data collector as shown in Fig. 67.

Measurement Trial / Dataset: We investigated potential measurement platforms that we could use for this study. RIPE Atlas [4] with around 15K probes is ideal, but it currently does not support HTTP measurements to custom targets. PlanetLab [50] would be another choice, but then the vantage points are restricted to only research networks. We are interested in measuring from different types of networks. BISmark [18] probes are similar to SamKnows probes but it's currently unknown how many probes are deployed behind native IPv6 lines. As such, we strategically deployed SamKnows probes behind native IPv6 lines to cover a diverse range of origin-ASes. Fig. 68 shows the current deployment status of 77 SamKnows probes that are part of our measurement trial. To put numbers into perspective, this

is more than the number of CAIDA Ark [14] probes (53 as of Oct 2015) with native IPv6 connectivity. An associated table shows the number of probes within each RIR region. As can be seen most of these probes are connected within the RIPE and ARIN region. An associated table in Fig. 67 shows the number of probes behind each network type. It can be seen how most of these probes are deployed behind residential and research networks and receive native IPv6 connectivity from their service provider. The dataset consists of `simweb` measurements collected for 65 days between April 2015 and June 2015. This includes around 207M data points captured from 77 SamKnows probes.

11.4 DATA ANALYSIS

Let u denote a website identified by a URL. We call the HTML page returned by fetching the URL u as the root page of u denoted by $r(u)$. The root HTML page contains a set of embedded objects (such as images, CSS or javascripts) which we denote by $O(u)$. Furthermore, the root HTML page usually has a set of embedded links denoted as $L(u)$. Since we study the impact of accessing websites using different network protocols, we denote the root page of u accessed over IP version v as $r_v(u)$. We refer to the set of embedded objects in $r_v(u)$ as $O_v(u)$ and the set of links in $r_v(u)$ as $L_v(u)$. Given $r_v(u)$ for IP version v, the set of objects and links that we are able to retrieve successfully using IP version v is given by $O'_v(u)$ and $L'_v(u)$ respectively. This terminology will be used in the rest of the data analysis.

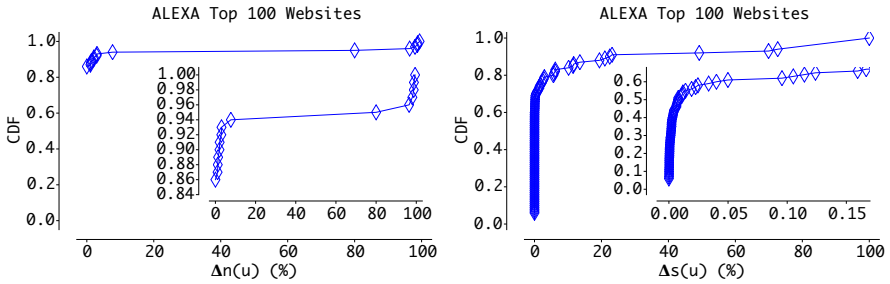
11.4.1 *Comparing Content Similarity*

Is there a difference in the number of fetched webpage elements? – In order to estimate the difference in the number of objects fetched for a website u over IPv4 and IPv6 we used Eq. 11.1. For a dual-stacked website u, it calculates the fraction of difference between number of fetched objects using IPv4 and IPv6, over total number of objects fetched using IPv4 where $\hat{n}_v(u)$ represents the median of the sample of $n'_v(u)$ values across all measurements from all probes.

$$\Delta n(u) = \frac{\hat{n}_4(u) - \hat{n}_6(u)}{\hat{n}_4(u)} \times 100\% \tag{11.1}$$
$$n'_v(u) = |O'_v(u)| + |L'_v(u)|$$

Fig. 69 (a) shows the distribution of $\Delta n(u)$ across ALEXA top 100 dual-stacked websites. It can be seen how 14% of websites exhibit dissimilarity in the number of fetched webpage elements with 6% showing more than 50% difference. This dissimilarity in number of elements negatively impacts webpages fetched over IPv6.

Is there a difference in the size of fetched webpage elements? – In order to estimate the difference in the size of objects fetched for a website u we used Eq. 11.2. For a dual-stacked website u, it calculates the fraction of difference between size of objects fetched using IPv4 and IPv6, over total size of objects fetched using IPv4 where $\hat{s}_v(u)$ represents the median of the sample of $s'_v(u)$ values across all measurements from all probes. Note, the reported content size is the size of the payload (excluding the header). In situations where the response is HTTP chunked encoded, the payload is the sum of the size of all chunks (excluding the chunked metadata). In situations where the response



(a) (b)

Figure 69: *Distribution of fractional difference in total number (see left) and total size (see right) of successfully fetched objects over IPv4 and IPv6 for ALEXA top 100 dual-stacked websites. 14% of websites exhibit dissimilarity in the number of fetched webpage elements with 6% showing more than 50% difference. 94% of dual-stacked websites exhibit dissimilarity in size with 8% showing atleast 50% difference. The raw values are available at: http://goo.gl/qdtLGl*

is compressed, the content size reports the payload size before the receiver decompresses the data.

$$\Delta s(u) = \frac{\hat{s}_4(u) - \hat{s}_6(u)}{\hat{s}_4(u)} \times 100\% \tag{11.2}$$
$$s'_v(u) = s(O'_v(u)) + s(L'_v(u))$$

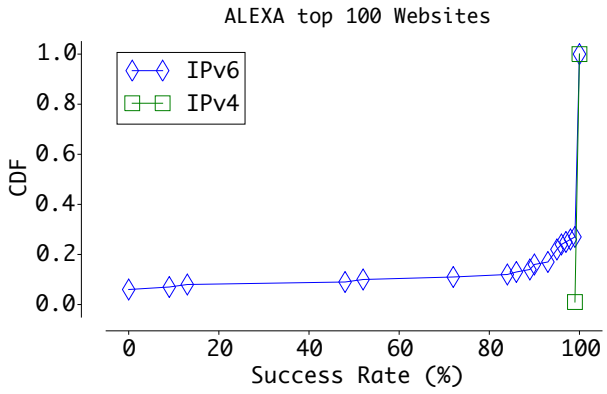
Fig. 69 (b) shows the distribution of $\Delta s(u)$ across ALEXA top 100 dual-stacked websites. It can be seen how 94% of dual-stacked websites exhibit dissimilarity in size with 8% showing atleast 50% difference. This dissimilarity in size of elements also negatively impacts webpages fetched over IPv6.

11.4.2 *Comparing Success Rates*

Can all webpage elements be successfully fetched over IPv6? – In order to make this estimation we measure success rate. For a website u, we define success rate as the fraction of number of successfully fetched webpage objects over the total number of objects and links embedded in the root page. as shown in Eq. 11.3.

$$p_v(u) = \frac{n'_v(u)}{n_v(u)} \times 100\% \tag{11.3}$$

We consider $\hat{p}_v(u)$, the median of the sample of success rate $p_v(u)$ values across all measurements from all probes to a website u over IP version v as the representative success rate value for that website over IP version v. A website u with a x% value of $\hat{p}_v(u)$ means that only x% of the total objects embedded in the root page can be fetched over IP version v. Fig. 70 shows the distribution of $\hat{p}_v(u)$ across ALEXA top 100 dual-stacked websites. It can be seen how over IPv4, all webpages except `www.flipkart.com` have a median success rate value of 100% over the entire measurement duration from all probes. However, we see how 27% of websites show some rate of failure over IPv6, with 9% of websites exhibiting more than 50% failures



AF	PERCENTILE	SUCCESS RATE (↓)
IPv6	0.06	0%
	0.07	9%
	0.08	13%
	0.09	48%
	0.10	52%
	0.11	72%
	0.12	84%
	0.13	86%
	0.14	89%
	0.16	90%
	0.17	93%
	0.22	95%
	0.24	96%
	0.25	97%
	0.26	98%
	0.27	99%
	1.00	100%
IPv4	0.01	99%
	...	...
	1.00	100%

Webpage	Success Rate (%)	
	IPv6(↓)	IPv4
www.bing.com	0	100
www.detik.com	0	100
www.engadget.com	0	100
www.nifty.com	0	100
www.qq.com	0	100
www.sakura.ne.jp	0	100
www.flipkart.com	09	99
www.folha.uol.com.br	13	100
www.aol.com	48	100
www.comcast.net	52	100
www.yahoo.com	72	100
www.mozilla.org	84	100
www.orange.fr	86	100
www.seznam.cz	89	100
www.mobile.de	90	100
www.wikimedia.org	90	100
www.t-online.de	93	100
www.free.fr	95	100
www.usps.com	95	100
www.vk.com	95	100
www.wikipedia.org	95	100
www.wiktionary.org	95	100
www.elmundo.es	96	100
www.uol.com.br	96	100
www.marca.com	97	100
www.terra.com.br	98	100
www.youm7.com	99	100
biglobe.ne.jp	100	100
www.appspot.com	100	100
www.att.com	100	100
. . .		

Figure 70: *Distribution (see top-left) of success rates towards ALEXA top 100 dual-stacked websites. Each data point is a median success rate of a webpage as measured from all probes. It can be seen how 27% of the dual-stacked websites have some fraction of webpage elements that fail over IPv6 (see bottom-left) with 9% having more than 50% webpage elements that fail over IPv6. Worse 6% of dual-stacked websites exhibit complete failure over IPv6. The table on the right shows top 30 dual-stacked websites ordered by ascending order of their success rates over IPv6. The entire list with individual success rate values is released at: http://goo.gl/bHlxLa*

over IPv6. Worse 6% of websites shows complete failure (0% success) over IPv6. An associated table shows the success rate of the first 30 websites arranged in ascending order of their success rate over IPv6. The special case of `www.bing.com`, which has globally stopped providing IPv6 services in 2013, was recently identified in [9]. Apart from `www.bing.com` we further identified 5 more such websites. These websites were accessible over IPv6 in the past, but have stopped providing AAAA entries and therefore exhibit 0% success rate over IPv6.

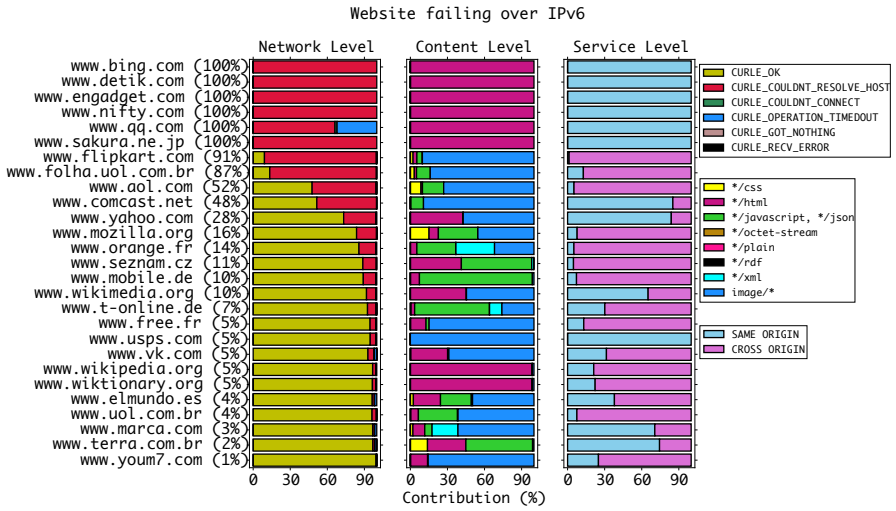


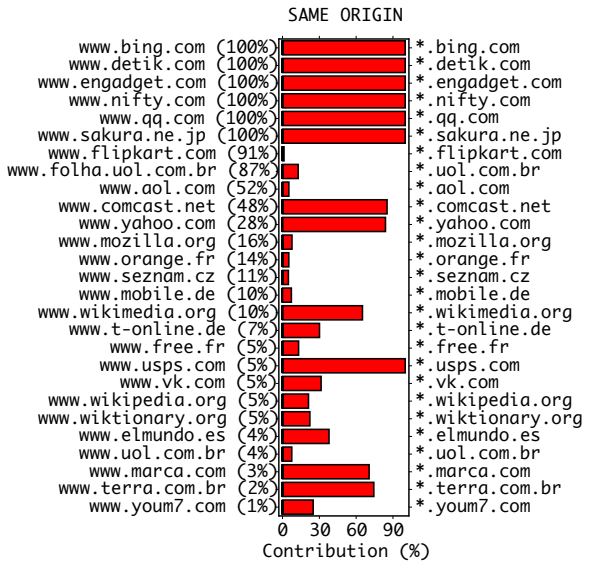
Figure 71: *The causal analysis of failing websites over IPv6 at the network (left), content (middle) and service level (right). The percentage next to each website is its failure rate. The network level analysis reveals that most of webpage elements fail due to a DNS resolution error. The content level analysis reveals that images, javascripts and CSS contribute to the majority of the failure. The raw contributions of each error code and MIME type are available here: http://goo.gl/WJDK6t. The service level analysis reveals that both same and cross origin sources are responsible for the objects that fail over IPv6.*

11.4.3 *Causality Analysis*

We further performed a causal analysis on the failure of 27% of ALEXA top 100 dual-stacked websites to investigate the network, content and service level source of the issue as discussed below:

Where in the network does the failure occur? – We investigated the spectrum of `libcurl` error codes reported by `simweb` for each object of a failing website u. We grouped objects by their error codes for each measurement run to a website u from a probe. We subsequently calculated the median contribution to each website for the entire measurement duration across all probes. Fig. 71 (left) shows the percentage contribution of error codes to each failing website u. The numbers next to each failing website are the failure rates, $100\% - p_6(u)$ flipped over from Fig. 70. The error code `CURLE_OK` contributes to the success rate, with rest of the error codes contributing to the failure rate of each website u over IPv6. It can be seen how `CURLE_COULDNT_RESOLVE_HOST` is the major contributor to failure rates. This goes to show how most of the webpage elements of an IPv6-capable website fail due to a DNS resolution error. This is usually caused due to missing AAAA entries for these webpage elements in the DNS.

Which type of objects fail more than others? – We also investigated MIME types reported by `simweb` for each object of a failing website u. Note that, `simweb` would not be able to return MIME types of objects that fail to be fetched. We therefore ran a post-processing function to fetch the response headers (and consequently the MIME type) of these objects over IPv4. We grouped objects by their MIME types for each measurement run to a website u from a probe. We subsequently calculated the median contribution for the entire measurement duration across all probes. Fig. 71 (middle) shows the percentage contribution of MIME types to each failing website u. It can be



Webpage	Same Origin
www.bing.com	100%
www.detik.com	100%
www.engadget.com	100%
www.nifty.com	100%
www.usps.com	100%
www.qq.com	100%
www.sakura.ne.jp	100%
www.comcast.net	85%
www.yahoo.com	83%
www.terra.com.br	74%
www.marca.com	70%
www.wikimedia.org	65%
www.elmundo.es	37%
www.vk.com	31%
www.t-online.de	30%
www.youm7.com	24%
www.wiktionary.org	22%
www.wikipedia.org	21%
www.free.fr	13%
www.folha.uol.com.br	12%
www.mozilla.org	7%
www.uol.com.br	7%
www.mobile.de	7%
www.aol.com	5%
www.orange.fr	5%
www.seznam.cz	4%
www.flipkart.com	1%

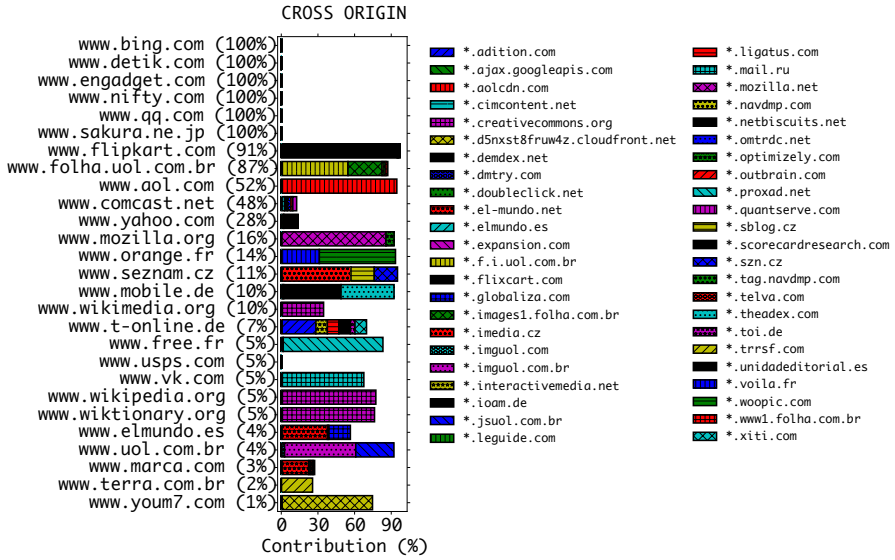
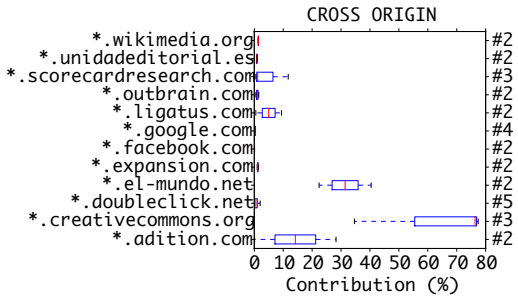


Figure 72: *Classification of webpage elements by same and cross origin sources. 12% of websites have more than 50% webpage elements that belong to the same origin source and fail over IPv6.*

seen how for websites which have AAAA entries (websites with less than 100% failure rate) – images, javascripts, and CSS content contribute to the majority of the failure over IPv6.

Where do the failing objects originate from? – We further investigated the URLs reported by `simweb` for each object of a website u failing over IPv6. We used the URLs to identify the hostnames of these failing objects. We used these hostnames to classify (see Fig. 72) objects into *same origin* and *cross origin* sources. We classify objects of a website u, to belong to a cross



CROSS ORIGIN	MEDIAN
*.creativecommons.org	76.33%
*.el-mundo.net	31.41%
*.adition.com	14.20%
*.ligatus.com	4.98%
*.wikimedia.org	1.40%
*.expansion.com	1.21%
*.scorecardresearch.com	1.19%
*.outbrain.com	1.06%
*.unidadeditorial.es	0.94%
*.doubleclick.net	0.54%
*.google.com	0.31%
*.facebook.com	0.06%

Figure 73: *Boxplot distribution (left) of cross-origin sources that contribute to the failure of more than 1 dual-stacked website. The right y-axis shows the number of websites spanned by each of the cross-origin sources. An associated table (right) shows the median contribution of each cross-origin source to all spanned websites. doubleclick.net has the highest span across 5 websites, with creativecommons.org having highest median contribution (76%) to the failure rate of 3 websites.*

origin source whenever their hostnames do not match the hostname of the website u. The rest of the objects (including subdomains) belong to the same origin source. We grouped the number of objects by their origin source for each measurement run to a website u from a probe. We subsequently calculated the median contribution for the entire measurement duration across all probes. Fig. 71 (right) shows the contribution of same and cross origin sources to webpage elements that failed over IPv6. It can be seen how all failing (27% of dual-stacked) websites have some fraction of webpage elements that belong to the same origin source and fail over IPv6. Worse, Fig. 72 (top), shows how 12% of dual-stacked websites have more than 50% webpage elements that belong to the same origin source and fail over IPv6. Fig. 72 (bottom) shows the contribution of webpage elements that belong to cross origin sources. Note, we only show cross origin sources with more than 1% contribution to the failure of a website over IPv6. It can be seen how for websites which have AAAA entries (websites with less than 100% failure rate) – both same and cross origin sources contribute to the failure of webpage elements over IPv6.

Which cross-origin sources span across multiple failing websites? – Given some of the cross-origin sources contribute to the failure of multiple websites, we tried to identify sources that would help benefit more websites if their content was available over IPv6. Fig. 73 shows the distribution of cross-origin sources that contribute to the failure of more than 1 dual-stacked website. It can be seen that the cross-origin source `doubleclick.net` has the highest span across 5 websites, with a 0.54% median contribution to failure rates. The cross-origin source `creativecommons.org` on the other hand has 76% median contribution to the failure rate of 3 websites. The number of spanned websites show how many can be benefited with the median contribution exhibiting how much the failure rate can be reduced by enabling IPv6 content delivery from these cross-origin sources.

11.5 CONCLUSION

We showed how websites which used to be IPv6 capable once did not remain as such forever. We witnessed a few cases where a dual-stacked website stopped announcing AAAA entries in DNS over time. Metrics that measure IPv6 adoption should account for such changes. We also showed how metrics that limit only to the root webpage of a dual-stacked website can lead to an overestimation of IPv6 adoption numbers on the Internet. We witnessed several cases where images, javascript and CSS content of a dual-stacked website did not have AAAA entries in the DNS. It remains unclear whether such websites can be deemed IPv6 ready. We also identified cross-origin sources that would help improve IPv6 web experience of a number of dual-stacked websites once they enable content delivery over IPv6.

Part IV

MEASURING ACCESS NETWORK PERFORMANCE

We measure access network performance using residential 696 RIPE Atlas and 1245 SamKnows probes.

We begin by presenting a methodology to select vantage points deployed in home networks. The methodology can be used to automate the discovery of RIPE Atlas tags associated with home probes. We apply this methodology to select RIPE Atlas and SamKnows residential probes to measure last-mile latency characteristics from multiple network service providers across the globe. We show that latencies within the home network provide a tangible contribution and must not be accounted when measuring last-mile links. We show that DSL deployments employ multiple interleaving depth levels that change over time. We show that last-mile latency is considerably stable over time and not affected by diurnal load patterns. We also show that last-mile latencies vary by subscriber location and by broadband product.

In Chapter 12 we present a methodology to select RIPE Atlas vantage points for broadband measurement studies. We utilise this methodology in Chapter 13 to measure last-mile latency of home broadband networks.

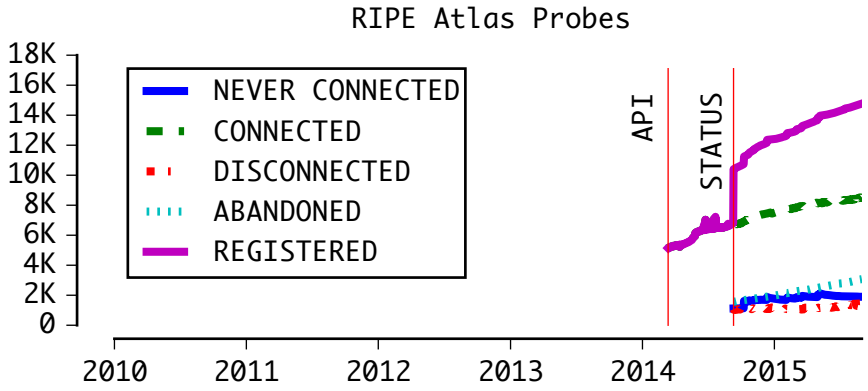
12 RIPE ATLAS VANTAGE POINT SELECTION

RIPE Atlas consists of 15.7K probes (as of October 2015) deployed in core, access and home networks. Recently (July 2014) RIPE Atlas introduced a tagging mechanism for fine-grained vantage point selection of probes. These tags are subdivided into system and user tags. System tags being automatically assigned and frequently updated are stable and accurate. We show an application of system tags by performing vantage point selection of dual-stacked probes. This exploration reveals how with around 2K dual-stacked probes, RIPE Atlas provides the richest source of vantage points for IPv6 measurement studies. User tags on the other hand are based on a manual process which is largely dependent on proactive participation of probe hosts. We extend this effort and show how probes deployed within home networks can be isolated and tags associated with them can be automatically applied. We validate our findings against the ground truth obtained from the user tags.

Contents

12.1 INTRODUCTION

RIPE Atlas [17, 4] has deployed around 15.7K (as of October 2015) dedicated hardware probes all around the globe as shown in Fig. 74. These probes perform active measurements (see Table 7) to ascertain the network performance of the global Internet. A majority of these probes are running measurements either from the core or from within access networks. A discernible number of probes are also hosted by volunteers within their home network. RIPE Atlas provides a public API [127] (starting February 2013) to programmatically provision measurements on these probes. However the probe selection (until recently) was limited to either geographic-based (using latitude and longitude) or network origin-based (using network prefixes) filters. In order to cope with this limitation, RIPE Atlas introduced (starting July 2014) a tagging mechanism that allows tags to be applied on individual probes. These tags are subdivided into system and user tags. The *system* tags are tags automatically applied by RIPE Atlas based on results collected from built-in (see Table 7) measurements. In addition to system tags, hosts can also voluntarily tag their own probes using *user* tags. A capability to filter vantage point selection based on these tags was recently (starting October 2014) made available. The system tags being directly derived from measurements and being frequently updated (every 4 hours) are fairly stable and accurate. The accuracy of user tags on the other hand is largely dependent on the proactive participation of hosts to not only tag, but also update their tags as and when network environments around the probe change. This may therefore lead to



NEVER CONNECTED	2239	14.24%
CONNECTED	8835	56.20%
DISCONNECTED	1375	8.75%
ABANDONED	3271	20.81%
REGISTERED	15720	100.0%

Figure 74: *Evolution of RIPE Atlas probes by their connection status. The plot is generated using the probe archive API: goo.gl/pMHs9Q which provides probe metadata since March 2014. The API was updated to also report the status information of each probe starting Sep 2014. An associated table shows the number (and fraction) of probes as of October 2015. Around 8.8K probes are connected out of the overall 15.7K deployed probes.*

stale user tags that do not reflect the current network situation of the probe. In this chapter we provide two main contributions: a) We show that system tags have improved the vantage point selection process by exhibiting a case study on selecting dual-stacked probes for IPv6 measurement studies and b) We extend the tagging effort to allow automated tagging of popular user tags. This will eliminate the need for probe hosts to manually tag their probes. We validate our results against the ground truth obtained from user-tagged probes.

The chapter is organized as follows. In Section 12.2 we describe the current state of RIPE Atlas system tags. In Section 12.3 and 12.4, we show an application of system tags by performing a case study on vantage point selection of dual-stacked probes and using these probes to measure IPv6 performance. In Section 12.5 we extend the tagging effort by automating a set of popular user tags. We validate our results using user tags as the ground truth and conclude in Section 12.7.

12.2 SYSTEM TAGS

RIPE Atlas recently (starting July 2014) introduced [304] a feature that allowed tags to be applied to probes. These tags are subdivided into system and user tags. System tags are automated tags generated by the RIPE Atlas system. Fig. 75 shows the distribution of these system tags across all probes. These system tags highlight the state of DNS (such as `system-resolves-a-correctly` *et al.*) and IP connectivity (such as `system-ipv6-works` *et al.*) of the vantage point and are based on insights derived from

Table 7: A list of built-in measurements performed by probes by default as of October 2015. (*) in the target fields indicate multiple servers within the domain.

MEASUREMENT	TARGET
ping, ping6	first hop, second hop (derived from traceroute measurements), *.root-servers.net, *.atlas.ripe.net
traceroute, traceroute6	*.root-servers.net, *.atlas.ripe.net, labs.ripe.net
dns, dns6	*.root-servers.net: TCP (SOA), UDP (SOA, version.bind, hostname.bind, id.server, version.server)
sslcert, sslcert6	www.ripe.net, atlas.ripe.net
http, http6	www.ripe.net/favicon.ico, ip-echo.ripe.net

continuous built-in measurements (see Table 7) performed by the probes. For instance, `system-resolver-mangles-case` is applied on probes whose resolver implements case mangling of DNS requests [305] to provide increased protection against spoofing attacks. Similarly, the system tag `system-`

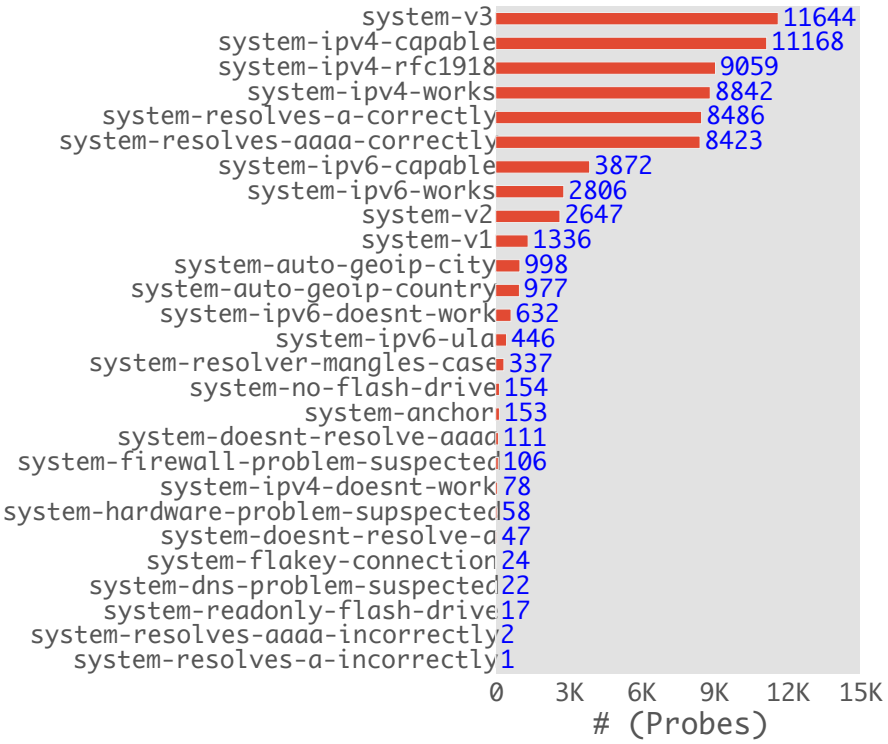


Figure 75: *Distribution of probes based on tags automatically assigned by the RIPE Atlas system as of October 2015. The raw dataset is available at: http://goo.gl/gXTEZQ.*

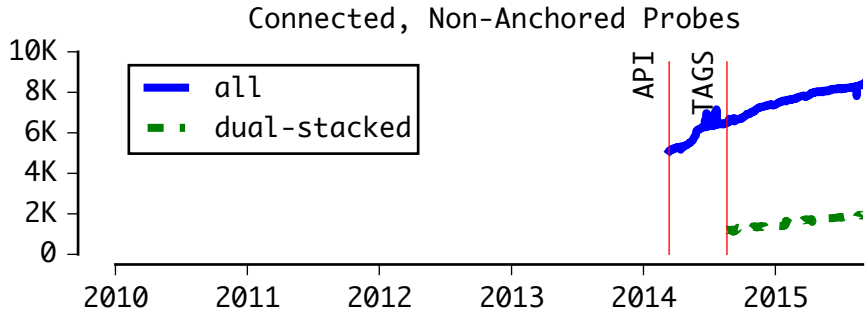


Figure 76: *Evolution of dual-stacked probes by time. The plot is generated using the probe archive API: goo.gl/pMHs9Q which provides probe metadata since March 2014. The API was updated to also report associated tags of each probe starting August 2014. Around 23.6% (2049 / 8685) of all connected non-anchored probes are dual-stacked as of October 2015.*

`firewall-problem-suspected` is set when only DNS activity is visible with `system-dns-problem-suspected` when only IP-level connectivity (with no DNS activity) is observed. Given the RIPE Atlas platform consists of three versions (v1, v2, and v3) of hardware probes and dedicated anchors, system tags (such as `system-v1` *et al.*) are also provided to allow hardware-based calibration of the probes. Using such a calibration, we recently discovered [306] (2015) how older versions of the probes experience load issues due to their hardware limitations.

John P. Rula *et al.* in [307] (2015) recently performed a factor analysis of the stratified sampling process used in the SamKnows / FCC Broadband America study. They motivated towards a principled approach that takes network and region based diversity into account to maintain the integrity of the sampling process. In this pursuit, using tag assisted vantage point selection we explore the region and network based diversity of connected dual-stacked probes within the RIPE Atlas platform. Fig. 76 shows the evolution of dual-stacked probes using these system tags. We define dual-stacked probes as probes with the same ASN over IPv4 and IPv6. This condition allows us to filter out hosts that use a 6in4 (such as Hurricane Electric) tunnel for IPv6 connectivity. This is useful to ensure only probes with native IPv4 and IPv6 connectivity are used for studies such as comparing IPv4 and IPv6 latencies to services over the Internet. We further only consider probes dual-stacked when they are tagged with `system-ipv4-works` and `system-ipv6-works` tags. The system evaluates each probe every 4 hours for all system tags by inspecting results obtained from built-in (see Table 7) measurements. For instance, Stéphane Bortzmeyer in [308] (2013) has shown how from amongst a sample of 1K RIPE Atlas probes, 10% of the probes believe to have IPv6 connectivity but fail when IPv6 measurments are provisioned on them. This study was one of the triggers that resulted in the introduction of `system-ipvX-works` tags. By using `*-works` instead of `*-capable`, such measurements tend to have more useful results. As such, the presence of these tags allow us to ensure selected dual-stacked probes are in fact able to reach out to services over both IPv4 and IPv6 on the Internet. As can be seen around 23.6% (2049 / 8685) of all connected and non-anchored probes are dual-stacked as of October 2015. To put numbers into perspective, this is way more than the number of CAIDA Ark [14] dual-stacked probes (53 as of Oct 2015) with native IPv6

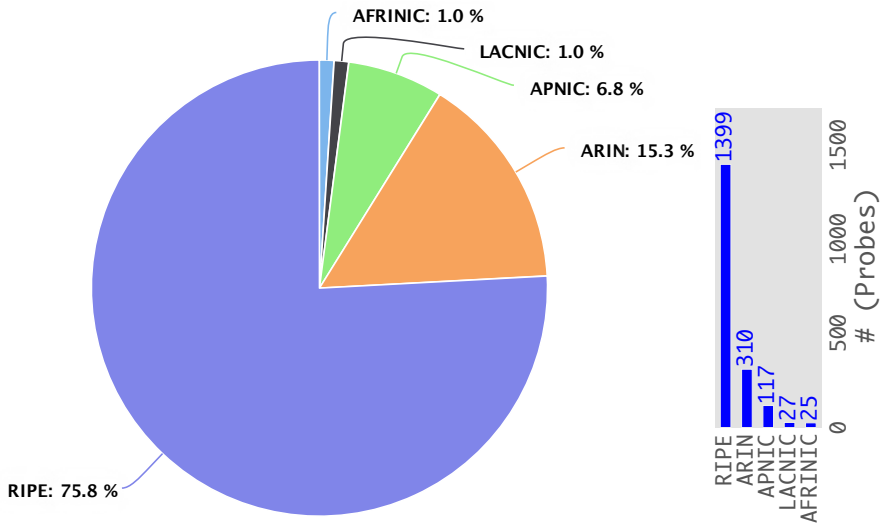


Figure 77: *RIR-based distribution of dual-stacked probes. The plot is generated using the RIPE Atlas Probe API: https://goo.gl/76VG1P and RIPE Data API: https://goo.gl/kh1kZx. Around 91% of dual-stacked probes are connected within the RIPE and ARIN region.*

connectivity. We use this definition of dual-stacked probes in the rest of the chapter.

12.3 IPV6 PROBES BY REGION

We use the RIPE Data API to map the IP endpoint used by each dual-stacked probe (derived from the RIPE Atlas Probe API) to the RIR that allocated the encompassing prefix of the IP endpoint resource. The registration information is derived from each RIR's WHOIS service. Using this mapping we cluster the probes by RIR region. Fig. 77 shows this RIR-based distribution of dual-stacked probes. It can be seen how more than 90% of the dual-stacked probes are connected within the RIPE and ARIN region. We further used the RIPE Atlas Probe API to split the RIR region by country. This country information is provided by probe hosts during initial registration. The system also uses geolocation services in case the user does not provide this information. For instance, the `system-auto-geoip-country` and `system-auto-geoip-city` system tags are used specifically for this purpose. These system tags are overidden when a user manually geolocates the probe. Fig. 78 shows this country-based distribution of dual-stacked probes. As can be seen the probes exhibit significant coverage with 91 spanned countries. Although, few countries with a large IPv6 userbase do serve only a small fraction of dual-stacked probes. For instance, Fig. 79 shows the correlation of percentage of dual-stacked probes against the percentage of IPv6 user population using the APNIC dataset. An associated table shows the top 10 countries with a large IPv6 userbase that have a small fraction of dual-stacked probes. For instance, it can be seen how JP with around 17% IPv6 usage ratio and around 20M IPv6 users serve only 1.4% (29/2049) dual-stacked probes. We hope this analysis will help improve the deployment of probes in such underrepresented countries with a large IPv6 userbase.

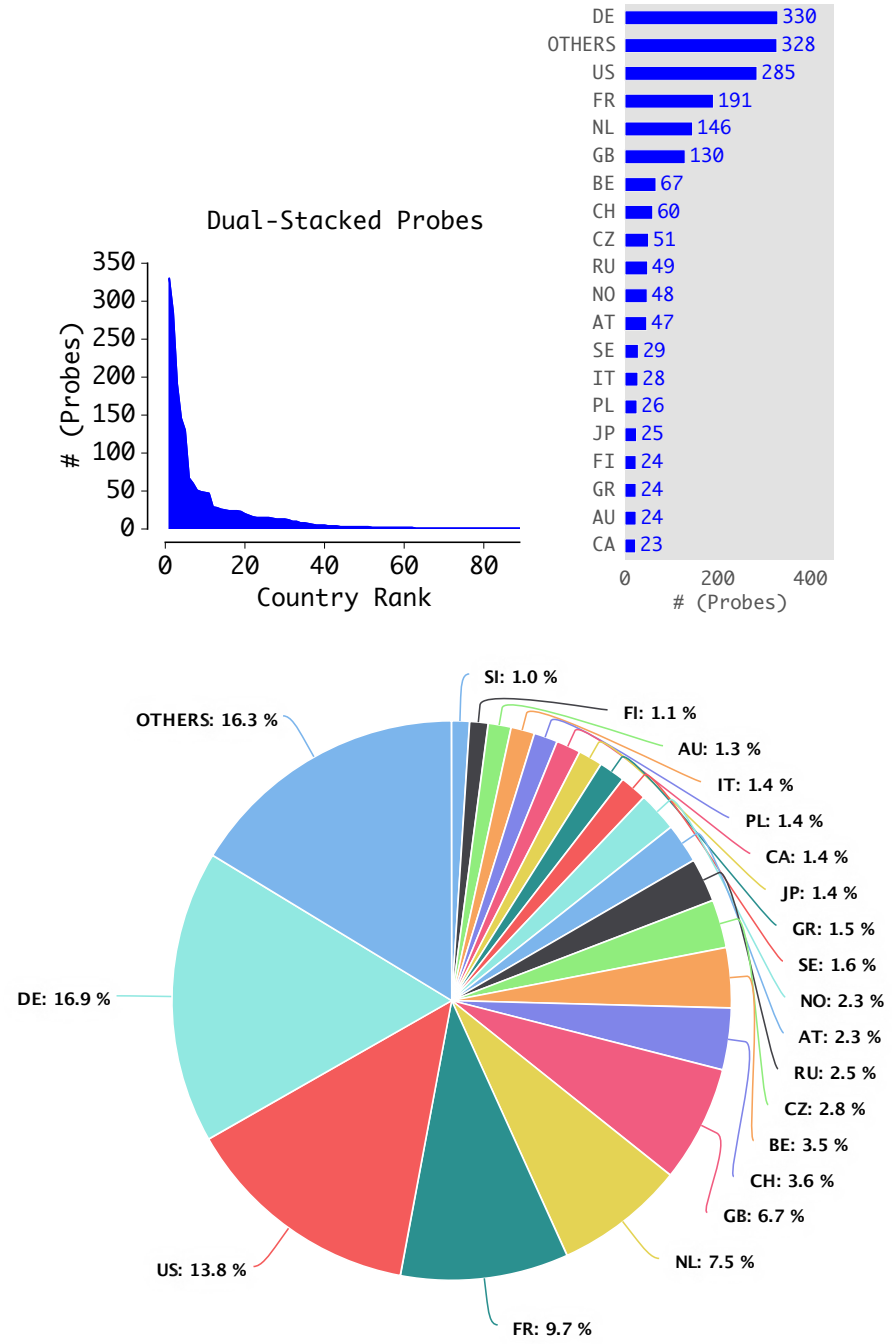
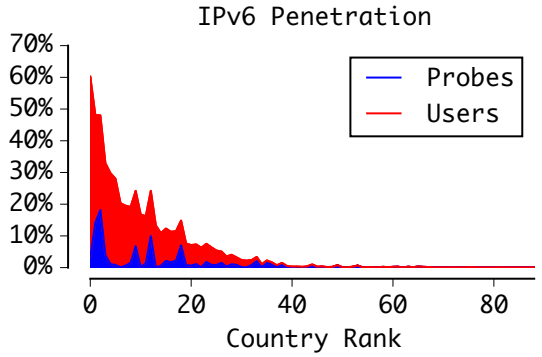


Figure 78: *Country-based distribution of dual-stacked probes. The plot is generated using the RIPE Atlas Probe API: https://goo.gl/76VG1P. The countries are ranked by the number of deployed probes. 91 countries are covered by dual-stacked probes. The entire list is made available at: http://goo.gl/UdEe1Q*

12.4 IPV6 PROBES BY NETWORK

We further used the RIPE Atlas Probe API to cluster the dual-stacked probes by their origin AS. Fig. 80 shows this AS-based distribution of dual-stacked probes. Using this information with the country-based distribution, it can be seen which service providers contribute to the large fraction of probes within the top countries. For instance, dual-stacked probes within DE are



	USERS	PROBES
BE	48.5%	3.3%
CH	28.4%	3.5%
PT	20.3%	0.9%
EE	18.5%	0.1%
GR	19.3%	1.5%
LU	18.5%	0.8%
JP	17.3%	1.4%
US	27.4%	13.7%
DE	29.0%	17.1%
CZ	13.7%	2.7%
MY	10.7%	0.2%

Figure 79: *Correlation (left) of percentage of IPv6 users against dual-stacked RIPE Atlas probes by country. The countries are ranked by the percentage of IPv6 users as of October 2015. The estimation of number of IPv6 users is available from APNIC dataset: http://labs.apnic.net/dists/v6dcc.html. A delta comparison (right) reveals the top 10 countries with a large IPv6 userbase that would benefit from more deployment of dual-stacked probes.*

connected behind DTAG (AS 3320) and KabelDeutschland (AS 31334), US behind COMCAST (AS7922), FR behind PROXAD (AS12322) and NL behind XS4ALL (AS3265) service provider networks.

Filtering ISPs: It must also be noted that not all probes are deployed behind service provider networks. From the perspective of vantage point selection, it is essential to be able to select probes deployed behind a specific type of a network (such as service provider networks) that spans multiple ASes and countries. We therefore, searched the literature for techniques that can classify ASes. Xenofontas Dimitropoulos *et al.* in [309] apply machine learning techniques to classify ASes into six categories: a) large ISPs, b) small ISPs, c) customer networks, d) universities, e) IXPs, and f) NICs. They use data from CAIDA Ark [14], RouteViews, and Internet Routing Registries (IRR). This study however is dated. Therefore we used PeeringDB to map ASes hosting dual-stacked probes by their network type information. PeeringDB is a database holding peering information of participating networks. Aemen Lodhi *et al.* in [250] show how the information maintained within this database is reasonably representative of network operator peering and is also up-to-date. Not all ASes hosting dual-stacked probes could be mapped to a network type due to missing AS information encompassing 23.01% (472 / 2049) dual-stacked probes (as of October 2015) in the PeeringDB database. Fig. 81 shows the evolution of dual-stacked probes by network type. It can be seen how 79% (1243 out of 1577) of the dual-stacked probes are deployed behind service provider networks. As a result, the RIPE Atlas platform is a potential platform for measuring native IPv6 performance delivered by service provider networks.

Filtering Residential Probes: Not all of these dual-stacked probes that mapped to a service provider network are particularly deployed within a home network. From the perspective of vantage point selection, it is essential to be able to delineate residential probes from probes hosted deep within

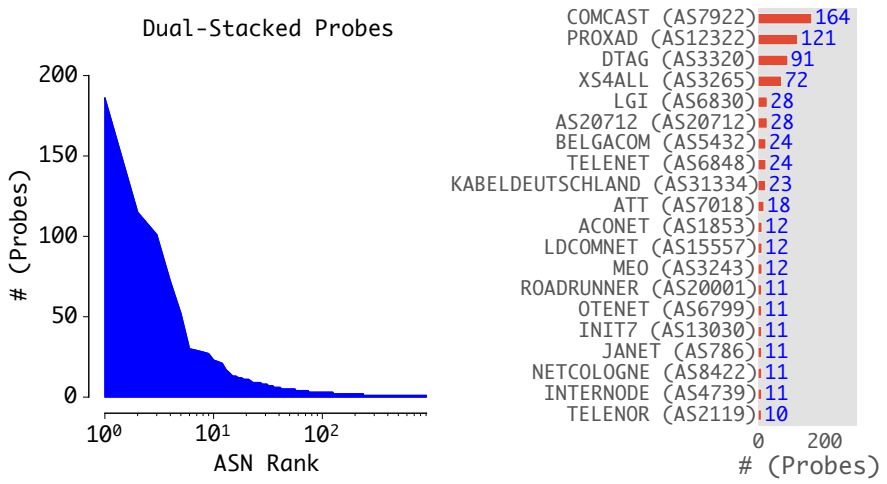
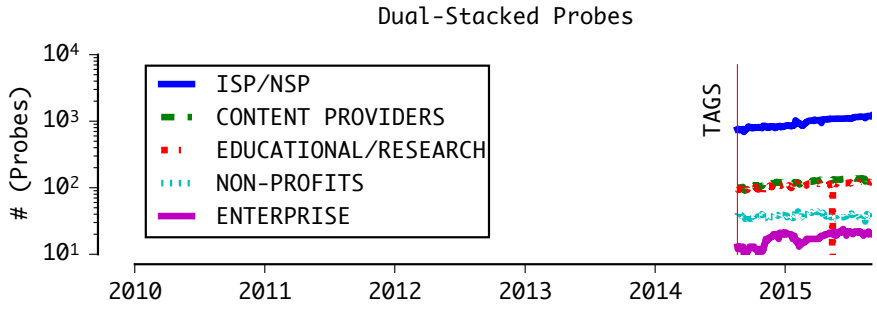


Figure 80: *AS-based distribution of dual-stacked probes. The plot is generated using the RIPE Atlas Probe API: https://goo.gl/76VG1P. The ASNs are ranked by the number of deployed probes. A large number (949) of ASNs are covered by dual-stacked probes. The entire list is made available at: http://goo.gl/bR5JEd.*



ISP/NSP	1243	79.0%
CONTENT PROVIDERS	143	9.1%
EDUCATION/RESEARCH	127	8.0%
NON-PROFITS	41	2.6%
ENTERPRISE	23	1.3%

Figure 81: *Evolution of dual-stacked probes by network type as mapped by PeeringDB. The plot is generated using the probe archive API: goo.gl/pMHs9Q which provides probe metadata since March 2014. The API was updated to also report associated tags of each probe starting August 2014. An associated table shows the number (and fraction) of dual-stacked probes within each network type as of October 2015. More than three quarter portion of dual-stacked probes are connected behind service provider networks.*

access or backbone network of a service provider. In order to identify residential probes, we used the RIPE Atlas measurement creation API [310], to provision one-off `traceroute` measurements towards RIPE Atlas anchors. We created separate measurements for each ISP in order to cycle through all available target anchors. This allowed us to evenly distribute the measurement load inside the platform. Measurements were performed using the

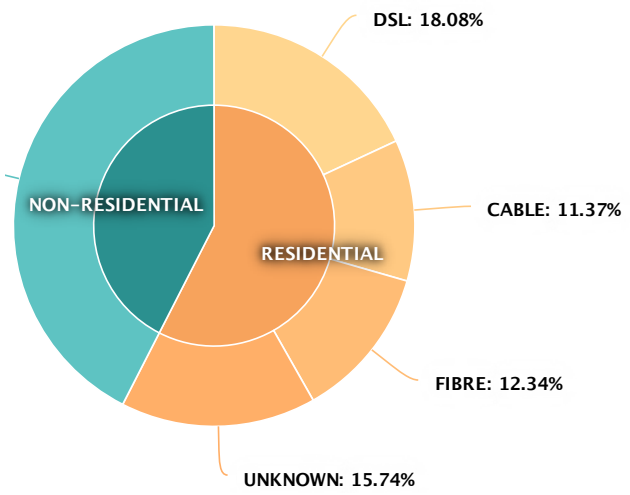


Figure 82: *Distribution of dual-stacked probes deployed behind service provider networks. 57.53% (592) of probes are wired behind a home gateway. Amongst residential probes, 18.08% (186) are behind a DSL, 11.37% (117) are behind cable while 12.34% (127) are behind FIBRE networks. 15.74% (162) residential probes did not have any user tags to delineate their access type.*

ICMP Paris probing method [64] implemented in the `evtraceroute` busybox applet within the platform. We define residential probes as probes that are directly wired behind the home gateway. In order to achieve this, we searched for probes whose first-hop was in a private IPv4 address space [311], but their second hop was in a public IPv4 address space. This criteria eliminates the situation where the service provider uses a private address space within the access network unless a probe is situated at the edge of last-mile. This also ensures that we do not incorrectly classify a probe behind business lines (which likely crosses multiple hops of private addresses before reaching out through the main router) as a residential probe.

We assume that in our one-off `traceroute` measurements ICMP responses are generated from the ingress interface [312] of each router on the forwarding path. Zachary S. Bischof *el al.* in [313], however, have shown how some home routers send ICMP responses using their egress interface. In such a situation, the first hop of a residential probe will appear public, and will not satisfy our aforementioned criteria. As such probes where the home gateway responds using the egress interface are automatically filtered out and are not part of our study. Fig. 82 shows the fraction (57.53%) of residential dual-stacked (592) probes deployed behind service provider networks.

Categorizing Residential Probes by Access Technology: We further tried to classify the residential dual-stacked probes into DSL, cable and fibre service providers. In this pursuit, we searched literature for techniques to identify the access technology used by the home gateway. For instance, Lucas DiCioccio *et al.* [314] use `netalyzr` [315] to send UPnP discovery messages to home gateways. They show how responses from these queries can reveal access technology used on the WAN interface. The measurements were performed on 120K homes in 2012, but only 35% of the gateways were found UPnP enabled. 10% of the gateways were connected further to a modem device, while 3% of the homes had more than one UPnP gateway. Even more, UPnP responses are not always accurate. In any case, since RIPE Atlas probes currently do not support a measurement that can perform UPnP queries, we

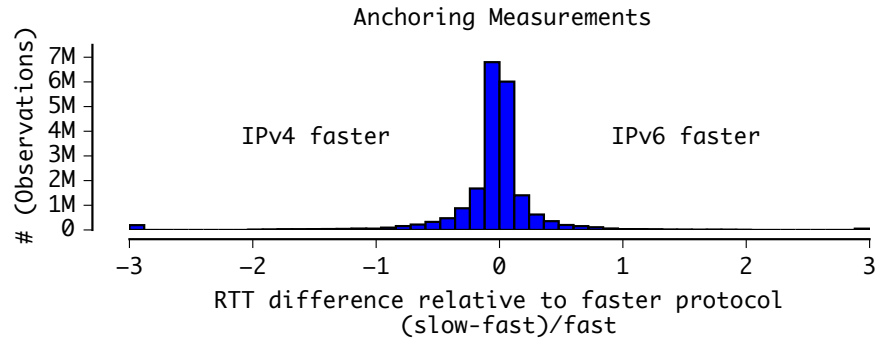


Figure 83: *5th percentile comparison of latencies over IPv4 and IPv6 from 2941 RIPE Atlas probes to 149 RIPE Atlas anchors using a month-long dataset consisting of around 20M data points. The dataset was collected in Sep-Oct 2015. The latencies appear comparable although IPv4 tends to show marginally better performance. The raw dataset is available at: http://goo.gl/dOJL5Q*

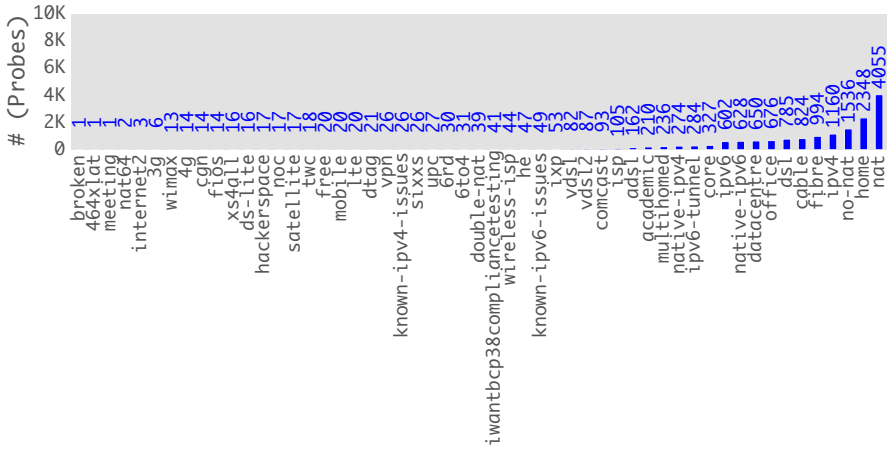


Figure 84: *Distribution of probes based on tags manually assigned by probe hosts as of October 2015. The raw dataset is available at: http://goo.gl/WTwL9X.*

relied on *user* tags (described in the next section) to categorize residential dual-stacked probes by the access technology used by the home gateway. Fig. 82 shows the split distribution of residential dual-stacked probes by access technology. It can be seen how this being an even split of dual-stacked probes across access technology is a good sample for IPv6 measurement studies from home networks.

Example: Measuring IPv6 Performance: A practical application of using these dual-stacked probes is to determine performance of IPv6 relative to IPv4. We use these dual-stacked probes to measure IPv6 performance towards RIPE Atlas anchors. We used a month-long dataset of `ping` measurements provisioned from 2941 probes to 149 anchors. Fig. 83 shows the 5^{th} percentile latency comparison between IPv4 and IPv6. The 5^{th} percentile was used to illustrate the best case scenario. It can be seen how IPv4 and IPv6 latencies between RIPE Atlas probes and RIPE Atlas anchors are comparable, although relative performance in IPv4 still seems marginally better.

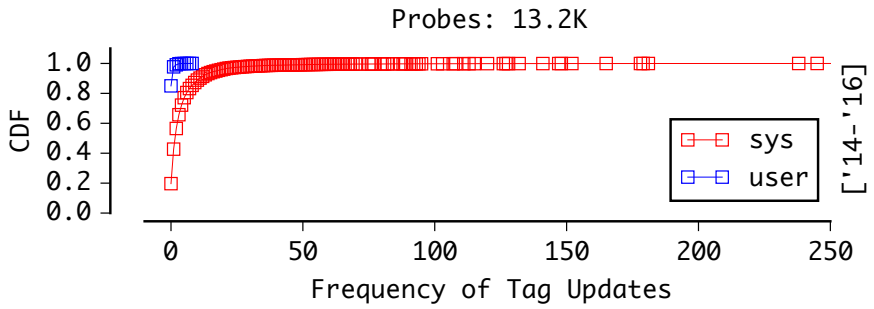


Figure 85: *The distribution of frequency of tag changes over time. Around 1% of probe hosts ever update their user tags. On the other hand more than half of the probes (54.64%) received at least one update on system tags with 9.51% of probes receiving atleast 10 updates. Whenever user tags are changed, more number of tags are added / deleted (upto 7 tags changed at once) when compared to system tags. The plot was calculated using the probe archive API: goo.gl/pMHs9Q which provides probe metadata since March 2014.*

12.5 USER TAGS

In addition to system tags, RIPE Atlas also allows probe hosts to tag their *own* probes with additional tags. Given the sample space of words that can be used for user tags is large, the visibility of user tags is set to private by default. This allows the system to not automatically offer the tag words to other users. The RIPE Atlas team periodically checks newly entered user tags and approves the ones that seem to be of general use. The approved user tags are then made available to other users. RIPE Atlas also periodically sanitizes the word space by merging similar tags. For instance, administrators can merge `v6-tunnel`, `ipv6-tunnel` and `tuneled-ipv6` into one user tag. This ultimately helps achieve sane vantage point selection for the large number of probes supported by the system. Fig. 84 shows the distribution of these user tags across all probes. It is worth noting that a large number of probes did benefit in the beginning when some of these user tags (such as `nat`) were automatically applied to probes to initially seed the system. Although system tags being generated directly by the RIPE Atlas platform are stable, the accuracy of user tags is largely dependent on the proactiveness of the host. Even though this is not expected to happen often, the host needs to update probe tags as and when network conditions change. For instance, in situations where a host forgets to change a tag due to change in either service subscription or even worse moving the probe to a new location, vantage point selection based barely on user tags would lead to entirely different measurement results. Fig. 85 compares the frequency of tag (user and system) changes over time. It can be seen how around only 1% of probes received any updates on their user tags. As such, we introduce the notion that user tags tend to stale over time. Given, the RIPE Atlas platform does not associate a tag creation timestamp, it is currently not possible to perform a predictive weighting of user tag accuracy. We instead extend the tagging effort by cherry-picking popular user tags to automate their discovery process. As can be seen from Fig. 84 most of the popular tags are centered around probes deployed behind residential deployments. As a result, we utilize the multidimensional data derived from RIPE Atlas APIs, PeeringDB and one-off (measurements that run only once) `traceroute` measurements (as described in previous sections) to automatically isolate these residential probes.

Table 8: The validation of our automated probe classification approach against the ground truth (user tags). We achieve higher coverage across tags with high specificity.

	tag	auto	sensitivity	specificity
nat	3270	4340	90.0%	56.1%
no-nat	1214	2115	95.1%	81.7%
academic	146	208	39.0%	97.6%
home	2672	2060	28.6%	87.9%
dsl	664	1224	57.4%	85.4%
cable	550	1052	62.9%	88.0%

We use the user tag dataset as the ground truth to validate our results. Table 8 provides a summary of investigated tags. It can be seen how we increased the coverage of probes within each popular user tag using the automated approach. A comparison against our ground truth reveals how high specificity [316] is observed across all studied tags. However, we do notice that for some tags (such as `home` and `academic`) we experience low sensitivity [316]. This can be due to multiple reasons. For once, PeeringDB currently does not reveal the network type information for 23.01% of probes. A comparison of probe coverage distribution across all ASes sampled using the automated approach and that using user tags reveals how more coverage can be achieved using the automated approach within ASes hosting a large number of probes. On the other hand, user tags help improve the probe coverage within AS with smaller number of probes since we miss their classification due to aforementioned reasons.

12.6 LIMITATIONS

We assume that probes that are assigned an endpoint from the private address space [311] and whose second hop is a publicly routable address are deployed behind a residential network. It is possible that there may be home probes that are behind multiple layers of NAT. It's also possible that some (although a smaller fraction) home probes may not be behind any NAT. Our methodology to automatically tag home probes will filter out these situations. Moreover, since the methodology relies on running measurements from within the probe, it can only be applied to connected probes. Although, both limitations will largely affect the coverage and less likely the accuracy of automatically inferred probes, we believe a combined approach (automated and manual) would be an accepted tradeoff that utilizes the best of both worlds.

12.7 CONCLUSION

We showed the utility of RIPE Atlas system tags by performing a region-based and network-based vantage point selection of dual-stacked probes. Although some regions and networks with a large number of probes can produce a sampling bias, the exploration revealed how RIPE Atlas to-date

provides the richest source of vantage points (around 2K dual-stacked probes) for IPv6 measurement studies. This exploration also helped us identify underrepresented regions (such as JP) with a large IPv6 userbase that can benefit from increased deployment of probes. We further identified how combining the use of multi-dimensional data and active measurements can be used to automatically tag home probes. This will eliminate the need for probe hosts to manually tag their probes. We validated our results against the ground truth obtained from the user tagged dataset and verified results with higher coverage and higher specificity. A combined approach (automated and manual) utilising the best of both worlds to tag these probes will further help improve the vantage point selection within the RIPE Atlas measurement platform.

13 REVISITING LAST-MILE LATENCY

Recent research has shown that last-mile latency is a key network performance indicator that contributes heavily to DNS lookup and page load times. Using a month-long dataset collected from 696 residential RIPE Atlas probes and 1245 residential SamKnows probes, we measure last-mile latencies from 19 (RIPE Atlas) network service providers across the globe and 9 (SamKnows) network service providers in the UK. We show that latencies within the home network provide a tangible contribution and must not be accounted when measuring last-mile links. We show that DSL deployments not only tend to enable interleaving on the last-mile, but also employ multiple depth levels that change over time and can be delineated from latency distributions. We also witness that last-mile latency is considerably stable over time and not affected by diurnal load patterns. Unlike observations from prior studies, we show that cable providers in the US do not generally exhibit lower last-mile latencies when compared to that of DSL. We instead identify that last-mile latencies vary by subscriber location and show that last-mile latencies of cable providers in the US are considerably different across the east and west US coast. We further show how last-mile latencies also vary depending on the broadband product and the access technology used by the DSL modem in the subscriber's network.

Contents

13.1 INTRODUCTION

Srikanth Sundaresan *et al.* in [28] (2013), using the BISmark [4, 18] platform, have shown that latency becomes a critical factor impacting quality of experience in networks where downstream throughput exceeds 16Mb/s. The effects of this observation are visible today with continuous efforts that attempt to move popular content as close [31, 32] to the edge as possible. Yi-Ching Chiu *et al.* in [317] (2015) recently showed that popular paths to CDNs serving high volume client networks tend to be shorter than paths to other networks. This is taken even further by some large content providers

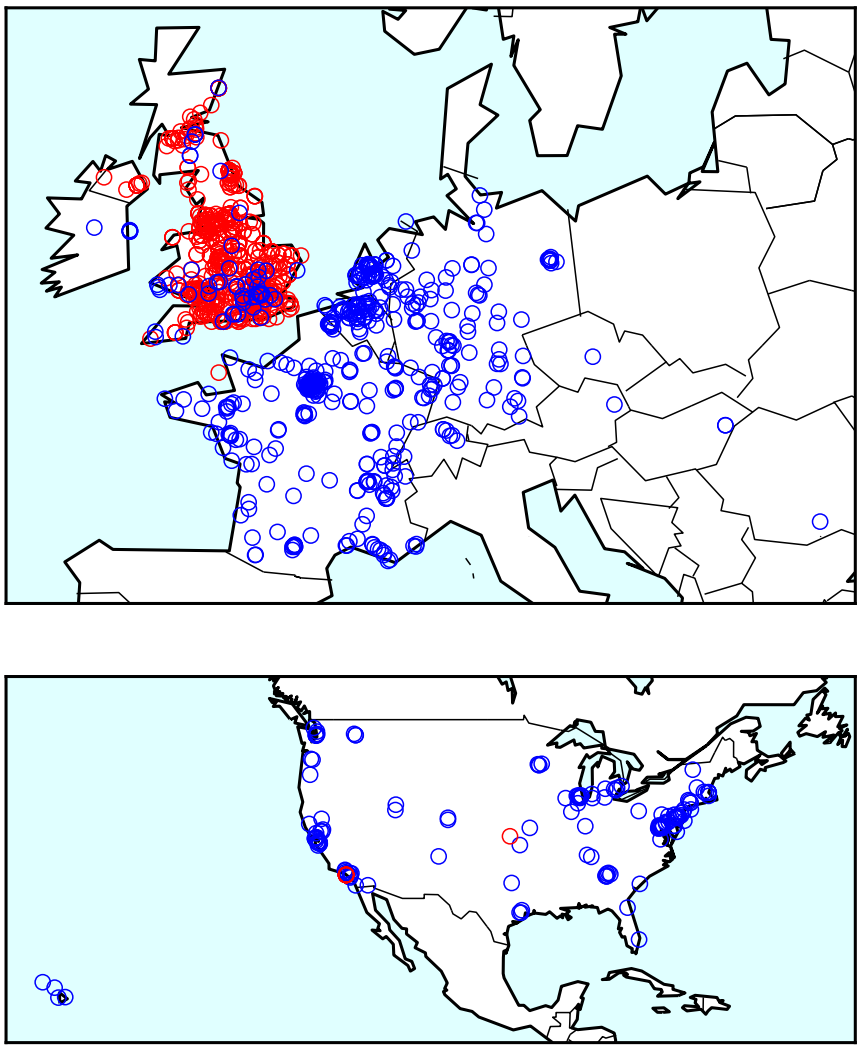


Figure 86: *Distribution of 696 RIPE Atlas v3 (blue) and 1245 SamKnows (red) residential probes. Each data point is a probe location as registered by the host. RIPE Atlas probes span the EU (521 probes) and the US (161 probes). SamKnows probes span the UK (1233 probes) and the US (11 probes).*

that deploy content caches [29, 30] directly in service provider networks. Furthermore, new standards such as Hypertext Transfer Protocol Version 2 (HTTP/2) [33] (2015) have been defined with a goal to improve webpage load times. Ongoing efforts such as QUIC [34] (2015) and TLS 1.3 [35] (2015) take this further to target operation on a much reduced latency (known as 0-RTT mode) overhead. In efforts to highlight confounding factors responsible for degraded webpage performance, Srikanth Sundaresan *et al.* in [28] (2013) recently showed that last-mile latency is major contributor to end-to-end latency and it contributes heavily to DNS lookup and page load times. Last-mile latency is becoming a key broadband network performance indicator and factors affecting last-mile latency need further investigation.

Early studies [12, 13] (2011, 2007) to investigate last-mile latency behaviour have shown that cable users experience lower last-mile latencies than DSL users (due to interleaving) using a dataset collected in the US. Using a dataset spanning the US and the EU, we show that *not all* cable deployments show

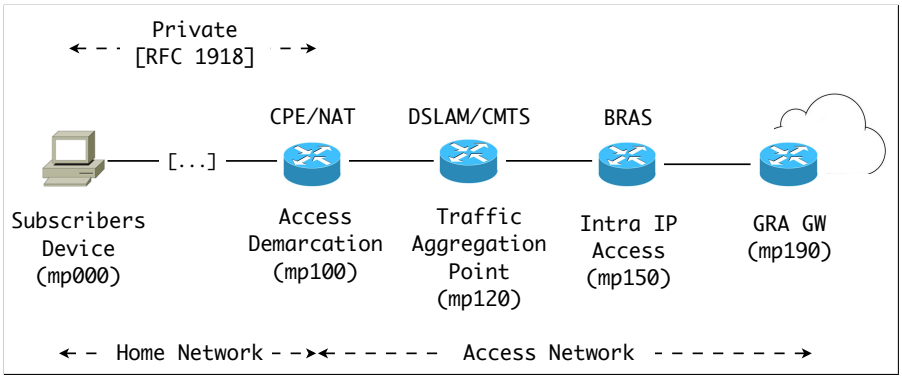


Figure 87: *The IETF IPPM reference path description of access network components. A CPE delineates the boundary of the home network where a private address space is used. Traffic aggregation points can either be layer-2 or layer-3 depending on the access network.*

last-mile latencies lower than DSL. We instead identify that last-mile latencies vary by subscriber location. Although, last-mile latencies of cable providers within the EU are generally lower than that of DSL, observations contrary to prior studies were witnessed within the US. We show that last-mile latencies of some cable providers in the US are considerably different across the US east and west coast. Subscribers of those cable providers around the US east coast experience last-mile latencies similar to that of DSL.

For our study we use two datasets as shown in Fig. 86. The first dataset has been obtained using probes that are part of the RIPE Atlas [17, 4] platform. The RIPE Atlas platform consists of thousands of probes (15.9K as of November 2015) deployed in core networks, access networks and residential networks. We utilise RIPE Atlas probes deployed within residential networks (1.5K as of August 2014). We also discovered [306] (2015) that older versions of the probes (around 43.1% of all probes) experience load issues due to their hardware limitations. Recently, it has been further confirmed [318] (2015) how these delays are more pronounced in situations where older version of probes are loaded with concurrent measurements. We therefore base our measurements on the most recent hardware version (v3), yielding a set of 696 RIPE Atlas probes that we have used to measure last-mile latency from 19 different network service providers in the US and the EU. The second dataset has been obtained using 1245 SamKnows [4] probes deployed behind 9 network service providers in the UK. The metadata available for this dataset also reveals the broadband product subscription.

Last-mile latencies are latencies to the first IP hop within the ISP's network. They are measured by capturing `traceroute` responses to TTL expiry and consequently include latencies within the home network. We show that latencies within the home network have a discernible impact and must not be included when measuring last-mile latency. As a result, we redefine last-mile to exclude the home network and we align the definition of the last-mile with the IETF IPPM reference path [223] (2015) description as shown in Fig. 87. We also show that some CPE rate limit ICMP responses to TTL expiry and therefore latencies towards these CPEs should *not* be used for baseline measurements.

It's suspected that DSL deployments enable interleaving on the last-mile to trade latency with lower packet loss rates. We extend this state of the art and show that DSL deployments *not* only enable interleaving, but also implement

multiple interleaving depth levels and vary them over time. We show that these depth levels can be delineated in last-mile latency distributions. DSL technology has also evolved over the years. For instance, ADSL2 provides multichannel transmission capability that allows different latency characteristics to be applied to each channel over the last-mile. ADSL2+ also uses higher frequencies to double bandwidth capacities. Using the SamKnows dataset, we show that last-mile latency is *not* the same for all subscribers of a DSL service provider, but they differ by broadband product and access technology used by the DSL modem.

To the best of our knowledge, this is the first study that measures last-mile latency characteristics on multiple perspectives covering serveral service providers in the US and the EU. This is the first study to show interleaving depth levels, last-mile latency behaviour by time of day, last-mile latency by subscriber location and last-mile latency characteristics based on the access technology used by the DSL modem. Overall, in this work, we provide seven main contributions: a) The home network latency makes a discernible contribution and therefore should not be accounted when measuring last-mile links. b) Some CPEs rate limit ICMP responses to TTL expiry. Latencies towards these CPEs should not be accounted for baseline measurements. c) DSL service providers not only enable interleaving, but also dynamically adapt the depth levels with time. d) Last-mile latency is considerably stable over time and not affected by diurnal load patterns. e) Last-mile latencies for DSL center at around 16 ms, with cable at around 8 ms, and fibre deployments at around 4 ms. f) Subscribers of some US cable providers experience considerably different last-mile latencies across the US east (centered at around 32 ms) and west coast (centered at around 8 ms) and g) Last-mile latencies decrease with increase in broadband product. VDSL deployments show last-mile latencies lower than ADSL2/ADSL2+.

The chapter is organized as follows. In Section 13.2 we describe prior work on last-mile latency measurements. In Section 13.3, we provide a precise definition of last-mile that adheres to the IETF IPPM reference path [223] description. Our measurement methodology and datasets are described in Section 13.4. Insights derived from the data analysis are presented in Section 13.5 with conclusions in Section 13.6.

13.2 RELATED WORK

Marcel Dischinger *et al.* in [13] (2007) inject packet trains and use responses received from home gateways to infer broadband link characteristics. They show that last-mile latencies are mostly affected by large modem queues and are higher for DSL when compared with cable networks. The inference is made by aggregating the distribution of last-mile latencies over all service providers. We note that not all service providers implement the same cable/DSL access technology. Therefore, we study the distribution of last-mile latencies separately for each service provider. In addition, using our SamKnows dataset we study last-mile latencies separately for four DSL broadband products in the UK. They define last-mile latency as the difference of minimum latencies between the second and first hop. The majority of measurements were performed within ISPs in North America.

Aaron Schulman *et al.* in [68] (2011) use PlanetLab [50] vantage points to send ICMP echo request packets to broadband hosts. They describe how physical factors (snow, wind, rain) affect the reliability of last-mile links. Srikanth Sundaresan *et al.* in [12] (2011) use the SamKnows platform to

show that cable users within the US experience lower last-mile latencies when compared to DSL. The comparison is performed using averages over a month-long dataset consisting of 4200 users. We on the other hand, show distribution of last-mile latencies experienced by users within several (19 using RIPE Atlas and 9 using SamKnows) service providers in the US and the EU. Our distribution shows that last-mile latencies for cable are *not* generally lower than DSL users within the US. Srikanth Sundaresan *et al.* define last-mile latency as latency to the first public IP hop and consequently include latencies within the home network. Recent studies [313, 319, 102] (2012-2015) however, have shown how much the home network delay contributes towards the last-mile latencies. We confirm this finding and show that latencies within the home network have a discernible impact and must not be included when measuring last-mile links.

Zachary S. Bischof *el al.* in [313] (2012) run `traceroute` measurements from within a BitTorrent plugin to measure last-mile latencies. They define last-mile latency as the difference of median latencies between the last-private hop to the first-public hop in a `traceroute` result: $median(h2) - median(h1)$. We employ a similar approach but instead calculate median of individual last-mile latencies: $median(h2 - h1)$. We also take it further to eliminate two situations where a) probes cross a wireless-link within the private network and b) service providers use the private address space [311] within their access network. Zachary S. Bischof *et al.* performed measurements directly from the end host and from within one ISP network: AT&T. We use dedicated RIPE Atlas and SamKnows probes that are directly connected to the home gateway and we cover 19 different service providers for RIPE Atlas and 9 for SamKnows.

Igor Canadi *et al.* in [320] (2012) show that end-to-end latencies to servers hosting `speedtest.net` experienced by DSL users are higher with more variance in US markets. They show that the geographical distance to these servers impacts latency but they do not take this factor into account when comparing results from DSL and cable networks. Swati Roy *et al.* in [321] (2013) use the BISmark platform to measure end-to-end latencies to M-Lab [52] servers and Google's anycast DNS service. They propose an algorithm that can correlate latency increases to a subset of the path responsible for the anomaly. They observed less number of last-mile latency issues. Daniel Genin *et al.* in [90] (2013) measure effects of congestion on access networks. They show that DSL links are mostly congested on the last-mile, while cable links usually experience congestion beyond the last-mile and show higher variability of such congestion events. Srikanth Sundaresan *et al.* in [28] (2013) show that last-mile latency is a bottleneck in high-throughput networks. They propose methods to perform DNS prefetching and TCP connection caching on the residential gateway to mitigate last-mile latency bottlenecks. John P. Rula *et al.* in [307] (2015) use the SamKnows FCC 2012 dataset to investigate packet loss and latencies observed towards fixed (such as M-Lab servers) landmarks. They show that these metrics are largely affected by access technology, geographical location and subscription rate of vantage points.

13.3 DEFINING LAST-MILE

Srikanth Sundaresan *et al.* in [12] (2011) define last-mile as the physical connection between the home gateway and the DSL Access Multiplexer (DSLAM) or Cable Modem Termination System (CMTS) depending on the

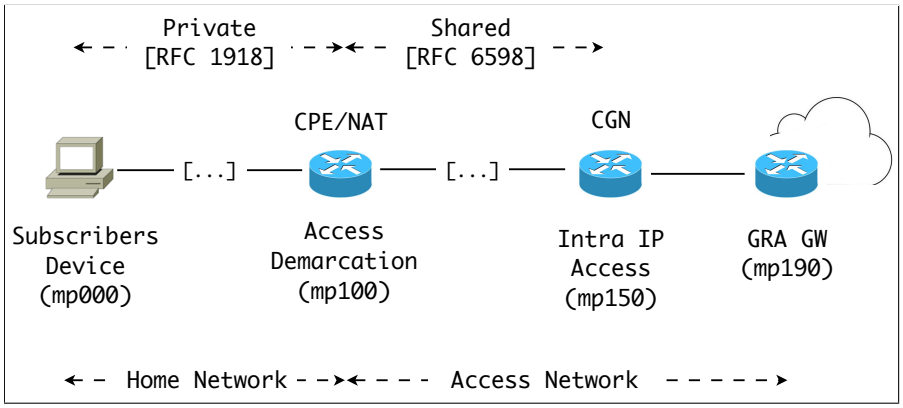


Figure 88: *The IETF IPPM reference path description of access network components for CGN deployments. A CGN delineates the boundary within the access network where a shared address space is use. The egress side of the CGN is the first interface with a public IP endpoint that is globally routable.*

deployed access technology. The IETF IPPM reference path [223] (2015) describes DSLAM/CMTS as traffic aggregation points (`mp120`) within the access networks as shown in Fig. 87. However, these traffic aggregation points may not be layer-3 devices. Usually the subscriber's traffic goes over a Point-to-Point Protocol over Ethernet (PPPoE)/Point-to-Point Protocol over ATM (PPPoA) tunnel that ends at the Broadband Remote Access Server (BRAS) (`mp150`). The BRAS assigns IP network parameters, and is usually the first IP hop from the subscriber to the Internet. In fact they acknowledge this limitation of tools that measure at layer-3 and reason that the possibility of measuring slightly further than the traffic aggregation point may not materially affect the results. They expect that the latency between hops inside an ISP is typically much smaller than the latency to the traffic aggregation point. We also honor the existence of a number of layer-2 devices within the last-mile. We also do not rule out existence of MPLS [322] tunnels within the last-mile. Recent access network deployments also allow the DSLAM to end the tunnel at the DSLAM itself, making it IP-capable. As such, tools searching for the first public IP hop will either terminate at `mp120` or at `mp150`.

A number of service providers have adopted an incremental approach to IPv6 migration by deploying Carrier-grade NAT (CGN)s [323] in the access network. We would like to identify last-mile for such access networks, since a number of probes used in our study may run measurements behind a CGN deployment. The reference path [223] (2015) describes how a CGN deployed in the access network would exist somewhere between `mp100` and `mp190`. They [223] portray how the egress side of a CGN with a public IP endpoint will typically be designated `mp150` as shown in Fig. 88. CGNs typically allocate endpoints from within a shared address space [324] inside the access network. This address space is private and not globally routable. Last-mile measurements must ensure that such an address space range is not treated as private, otherwise tools will inaccurately consider the region further beyond `mp150` as the last-mile, which is far deep within the access network.

Due to discernable latency contributions (described later in more details) within the home network, we also do not consider the `mp000 - mp100` to be part of the last-mile. The definition of last-mile therefore varies by context

and can range from either `mp100 - mp120` in situations where the traffic aggregation point is IP-capable or from `mp100 - mp150` otherwise.

13.4 METHODOLOGY / DATASETS

RIPE Atlas has deployed around 15.9K and SamKnows has deployed around 70K [4] (2015) dedicated hardware probes (as of November 2015) all around the globe. RIPE Atlas probes perform active measurements to ascertain network connectivity and reachability of the global Internet. A majority of RIPE Atlas probes are running measurements either from the core or from within access networks. A discernible number of probes are also hosted by volunteers within their home networks. SamKnows probes also perform active measurements, but with a primary goal to ascertain broadband performance. As a result, a considerable number of SamKnows probes are directly connected to the residential gateways. We describe how we select for these probes in the next section.

RIPE Atlas being an open platform makes all probes available for measurement research. These probes in addition to built-in measurements [4] (2015) can also run User Defined Measurements (UDM). A UDM allows any user registered on RIPE Atlas to provision measurements supported by the platform on probes with tailor-made measurement parameters. A registered user spends credits by provisioning a UDM on probes. Credits can be gathered by hosting a probe. Using credits gathered by hosting probes for multiple years, we were able to provision measurements on large sample of residential RIPE Atlas probes continuously for a month-long duration. In order to complement this study with another platform, we collaborated with SamKnows to provision measurements on a large sample of their probes within the UK. We further describe how we provisioned month-long `traceroute` measurements from both these platforms.

13.4.1 *Filtering residential probes*

RIPE Atlas currently runs measurements from three (v1, v2 and v3) different probe hardware versions. v1 and v2 probes are made of a custom hardware built around a Lantronix XPort Pro module, while v3 probes are off-the-shelf TP-Link wireless routers flashed with OpenWrt. SamKnows (similar to RIPE) also uses off-the-shelf TP-Link routers that are flashed with a custom OpenWrt firmware. The probes however are procured from a higher-end of the hardware spectrum. We initially calibrated the probes [306] (2015) to segregate them by their hardware family. This allowed us to identify (described later in the section) whether different hardware versions have any effect on measurement results. We also used calibration to rule out RIPE Atlas Anchors. RIPE Atlas periodically schedules measurements using a batch of several hundred probes against anchors to measure connectivity and reachability of a region. Anchors are dedicated servers that are designed to act as sinks of measurement traffic and are not relevant for this study.

For the RIPE Atlas probes, we used the RIPE Atlas probe API [325] to capture a list of connected probes. We used the AS Number (ASN) revealed by the probe API to cluster these probes by their AS. For each SamKnows probe, we extracted the public IP revealed by the Session Traversal Utilities for NAT (STUN) request. We later used the RIPE stat data API [326] to map this public IP to its first-level less-specific prefix entry. We used this prefix to get the corresponding ASN announcing the prefix as seen by RIS collectors.

We ranked ASNs by sorting them by the number of deployed probes separately for both RIPE Atlas and SamKnows. We pruned out ASNs with less than 10 probes since they may not make a representative sample. Not all ASes are network service providers. We searched the literature for techniques that can classify ASes. Xenofontas Dimitropoulos *et al.* in [309] (2006) apply machine learning techniques to classify ASes into six categories: a) large ISPs, b) small ISPs, c) customer networks, d) universities, e) IXPs, and f) Network Information Center (NIC)s. They use data from CAIDA Ark [14], RouteViews, and IRR. This study, however, is dated. Therefore, we used PeeringDB to map ASes hosting probes by their network type information. PeeringDB is a database holding peering information of participating networks. Aemen Lodhi *et al.* in [250] (2014) show that the information maintained within this database is reasonably representative of network operator peering and is also up-to-date. Not all ASes hosting probes could be mapped to a network type due to missing AS information in the PeeringDB database. For the unmapped AS, we decided to apply a semi-automatic approach. In the first pass, we used the RIPEstat Data API to capture the holder name for each AS as revealed in the Classless Inter-Domain Routing (CIDR) report. We used the holder name as a starting premise to filter service provider networks. In the second pass, we used the RIPE Atlas network coverage map to view the geographical distribution of probes deployed behind AS we deem as service provider networks to cross-confirm our premise.

In our pursuit to identify methods to further classify our list into DSL and cable service providers, we searched the literature for techniques to identify the access technology used by the home gateway. For instance, Lucas DiCioccio *et al.* [314] (2012) use `netalyzr` [315] to send UPnP discovery messages to home gateways. They show that responses from these queries can reveal access technology used on the WAN interface. The measurements were performed on 120K homes in 2012, but only 35% of the gateways were found UPnP enabled. 10% of the gateways were connected further to a modem device, while 3% of the homes had more than one UPnP gateway. Furthermore, UPnP responses are not always accurate. In any case, since RIPE Atlas and SamKnows probes currently do not support a measurement that can perform UPnP queries, we used reverse DNS entries derived from the public IP endpoint revealed by the STUN request, to identify probes deployed behind DSL, cable and fibre deployments. We also manually searched for service offers on each ISP's official website to cross-confirm our premise.

A large fraction of the probes identified so far are hosted by service providers and are running measurements from within access or backbone networks. In order to further filter down residential probes, we provisioned one-off `traceroute` measurements. The destinations were randomly chosen to evenly distribute the measurement load inside the platform. For RIPE Atlas, measurements were performed using the `evtraceroute` busybox applet, while for SamKnows measurements were performed using `mtr`. Both the measurements used the ICMP `traceroute` probing method.

We define *residential probes* as probes that are directly wired to the home gateway. This helps ensure that our last-mile latency measurements do not get skewed by probes that cross any wireless links within the home network. Probes within both the SamKnows and the RIPE Atlas platform do not associate to a wireless access point. In order to achieve this, we searched for probes whose hop1 was in a private IPv4 address space [311], but their hop2 was in a public IPv4 address space. This criteria also eliminates the situation where the service provider uses a private address space [311] within

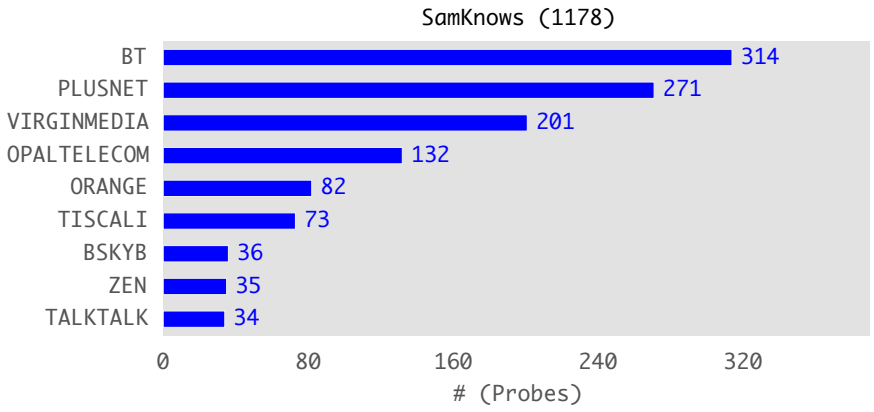


Figure 89: *Distribution of residential probes (1178) by service provider in the SamKnows dataset as of August 2014.*

the access network unless a probe is situated at the edge of last-mile. This also ensures we do not incorrectly classify a probe behind business lines (which likely crosses multiple hops of private addresses before reaching out through the main router) as a residential probe. It is worth noting that the `100.64.0.0/10` address block within the shared address space [324] is considered public. As a result, we do not cross the last-mile in situations where an ISP has a CGN deployment within the access network.

We assume that in our one-off `traceroute` measurements ICMP responses are generated from the ingress interface [312] (2013) of each router on the forwarding path. Zachary S. Bischof *el al.* in [313] (2012), however, have shown how some home routers send ICMP responses using their egress interface. In such a situation, the first hop of a residential probe will appear public, and will not satisfy our aforementioned criteria. As such, probes where the home gateway responds using the egress interface are automatically filtered out and are not part of our study. Going forward, we use the term probes to refer to residential probes.

13.4.2 *Running traceroute from residential probes*

We used the RIPE Atlas measurement creation API [310] to provision month-long `traceroute` measurements to randomly distributed RIPE Atlas anchors. In the process, we were hit by RIPE Atlas rate limits [306] (2015), which were lifted on our user accounts by proposing the measurement study on the `atlas` mailing list. The destination anchors were chosen outside the country where the ISP provides broadband services. This was to ensure we cross any deployed MPLS tunnels [327] within the last-mile. Measurements were performed every 4 hours using the ICMP probing method implemented in `evtraceroute` as of August 2014. Similarly for SamKnows, we provisioned month-long `traceroute` measurements using `mtr` to SamKnows servers using the same frequency and time period.

Both RIPE Atlas and SamKnows `traceroute` measurements send 3 ICMP queries per hop and repeat themselves every 4 hours. We extracted latencies measured to hop1 and hop2 for each probe over this month-long duration. We discard queries where no ICMP response (or an ICMP response with the source IP endpoint marked with `*`) is received. We also discard `traceroute` measurements where none of the queries on hop1 or hop2 generated

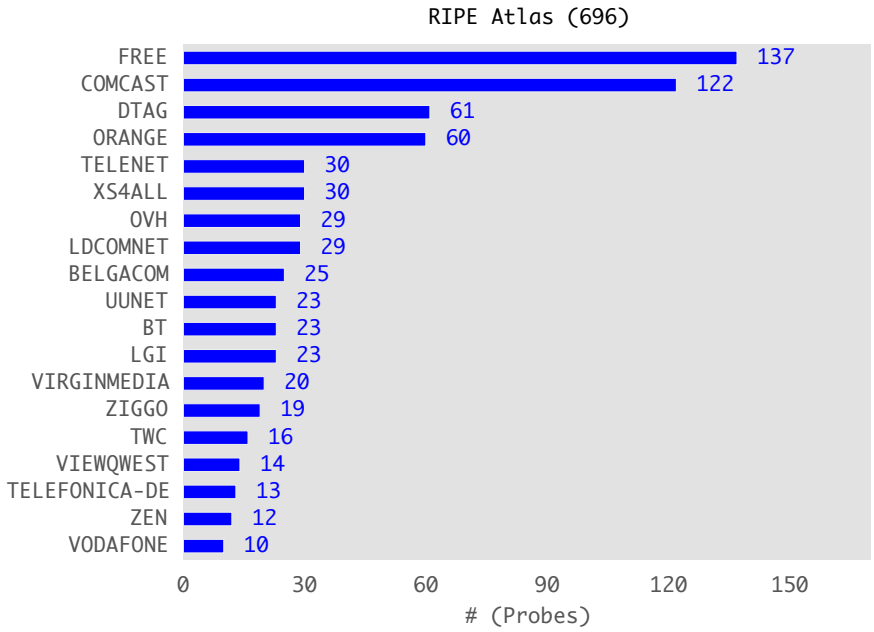


Figure 90: *Distribution of v3 residential probes (696) by service provider in the RIPE Atlas dataset as of August 2014.*

any ICMP response (or an ICMP response with the source IP endpoint marked with *). We noticed several cases where at a particular point in time, responses were being received from different IP endpoints on hop2. We assume the router responding on hop1 load balances these requests over time to either different interfaces of the same hop2 router or different hop2 routers within the access network. We discard such probes where responses have different source IP endpoints within a single hop at an instance. We filter them out because measured line characteristics may be different.

In this process, we also identifed load issues (causing timestamping delays) [306] (2015) in older (v1 and v2) hardware versions of RIPE Atlas probes. Recently, it has been further confirmed [318] (2015) how these delays are more pronounced in situations where older version of probes are loaded with concurrent measurements. We therefore decided to prune the older versions of the probe out of our RIPE Atlas dataset.

13.4.3 *Datasets*

The `traceroute` measurements were conducted every 4 hours (6 times a day) over 35 days in (July-August) 2014. Our dataset consists of 135K last-mile latency data points captured from 696 residential v3 RIPE Atlas probes and 440K last-mile latency data points captured from 1245 residential SamKnows probes. Fig. 86 shows the geographical distribution of these probes. Each data point is the location where the probe is hosted. The location is *not* derived using an IP geolocation service, but is the geographical location registered by the probe host. Fig. 89 shows the distribution of residential SamKnows probes separated by service provider. Fig. 90 shows the distribution of residential v3 RIPE Atlas probes separated by service provider.

The RIPE Atlas dataset contains 429 DSL probes, 225 cable probes and 36 fibre probes. The DSL probes span 11 service providers, cable probes

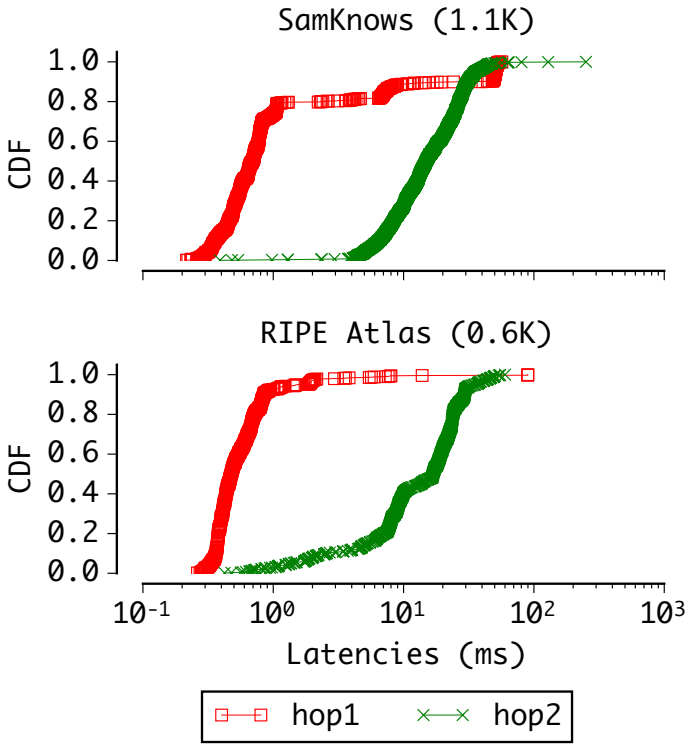


Figure 91: *Distribution of hop1 and hop2 latencies in log scale from SamKnows (above) and RIPE Atlas (below) probes. Hop1 latencies appear to cluster in three intervals. A major portion of probes show expected hop1 latencies of upto 1.5ms. A discernible number of probes show higher hop1 latencies upto 30ms. Few probes also show hop1 latencies more than that of hop2.*

span 6 service providers while fibre probes span 2 service providers. The SamKnows dataset on the other hand contains 994 DSL probes and 195 cable probes. The DSL probes span 8 service providers and cable probes span 1 service provider in the UK. The SamKnows dataset even provides broadband product subscription of each vantage point as metadata information. We utilize this metadata to further classify probes depending on the access technology used by the DSL/cable modem.

The RIPE Atlas dataset covers latencies measured by each ICMP query. This is in contrast to our SamKnows dataset, where latencies from multiple ICMP queries are averaged over a single hop by default.

13.5 DATA ANALYSIS INSIGHTS

13.5.1 *Latency contributions of home network*

We investigated the latency contributed by the home network (hop1) to that of the first hop in the service provider's network (hop2). Fig. 91 shows the distribution of absolute hop1 and hop2 latencies in RIPE Atlas and SamKnows platforms. Each data point is a median of latencies observed by a probe over the entire measurement duration: $\mathtt{median}(\mathtt{h1}), \mathtt{median}(\mathtt{h2})$. Latencies within the home network appear to cluster in three latency intervals. Table 9 shows the fraction of probes experiencing hop1 latencies within each

Table 9: The hop1 latencies divide into 3 clusters for both RIPE Atlas and SamKnows probes. Probes within the last cluster appear to witness rate limited ICMP responses from CPE.

	CLUSTERS	PROBES
RIPE Atlas	[0, 1] ms	92.47% (639/691)
	(1, 20] ms	7.09% (49/691)
	(20, _) ms	0.43% (3/691)
SamKnows	[0, 1.5] ms	79.68% (992/1245)
	(1.5, 30] ms	10.20% (127/1245)
	(30, _) ms	10.12% (126/1245)

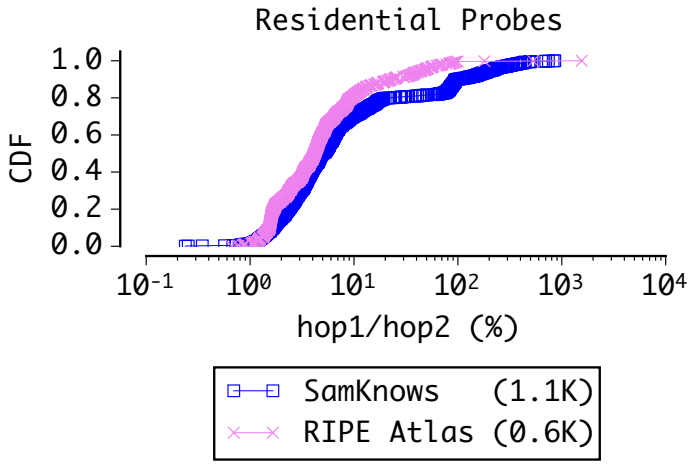


Figure 92: *Distribution of relative contribution of hop1 latencies to that hop2. 19.2% of RIPE Atlas probes and 29.7% of SamKnows probes show hop1 contributing to > 10% of hop2 latency. The home network shows a discernible contribution and should not be accounted when measuring last-mile latency.*

such interval. As can be seen, a major portion of RIPE Atlas (92.47%) and SamKnows (79.68%) probes show expected hop1 latencies of less than 1.5ms. A small fraction of probes (discussed in the next section) show hop1 latencies more than that of hop2. A discernible number of probes also show more than expected hop1 latency. For these probes, we studied the relative contribution of hop1 latency to that of hop2 as shown in Fig. 92. Each data point is a median of hop1 contribution to hop2 latency observed by a probe over the entire measurement duration: $\mathtt{median}(h1/h2)$. It can be seen how 19.2% (133/696) of RIPE Atlas probes and 29.7% (370/1245) of SamKnows probes witness hop1 latency contributing to 10% or more of hop2 latency. The home network latency appears to show a discernible contribution and therefore should not be accounted when measuring last-mile latency.

13.5.2 *Rate Limited ICMP responses*

Fig. 92 shows how 0.4% (3/691) of RIPE Atlas probes and 9.95% (124/1245) of SamKnows probes show hop1/hop2 contribution of more than 100%.

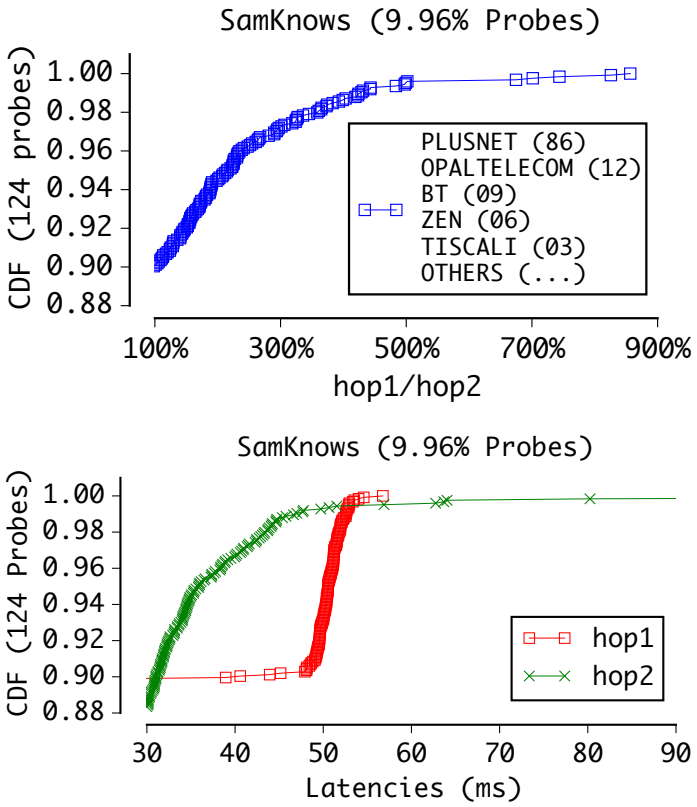


Figure 93: *9.95% (124/1245) of SamKnows probes show hop1/hop2 contribution of more than 100%. These probes are behind CPEs that may prefer to rate limit ICMP responses. 69% of SamKnows probes (above) behind these CPEs are connected to PLUSNET. The hop1 latency experienced by these probes (below) is around 50ms and is fairly stable.*

These 3 RIPE Atlas probes are connected behind different service provider networks. We further investigated these SamKnows probes and clustered them by ASN. Fig. 93 shows the origin ASNs of these probes. For instance, 69% (86/124) of these probes are connected behind PLUSNET. We also further separated probes by broadband product, but they were spread out across from 8Mbps to 80Mbps products. We also split the contribution to see absolute hop1 and hop2 latencies witnessed by these probes. Fig. 93 shows that hop1 latencies for these probes appear to be around 50ms. We suspect that these probes are behind CPEs that prefer to rate limit ICMP responses to TTL expiry and therefore have higher `traceroute` response times. We pruned out these probes and do not consider them as part of our last-mile latency measurement dataset.

13.5.3 *Interleaving depths in DSL networks*

It is suspected that DSL networks exhibit higher last-mile latencies because operators use interleaving on the last-mile to trade latency with lower packet loss rates [241]. An interleaving channel intersperses the payload between DSL frames to provide Impulse Noise Protection (INP) on the last-mile. This is usually implemented along with the Reed–Solomon (RS) Forward Error Correction (FEC) technique to make the channel more resilient to packet loss.

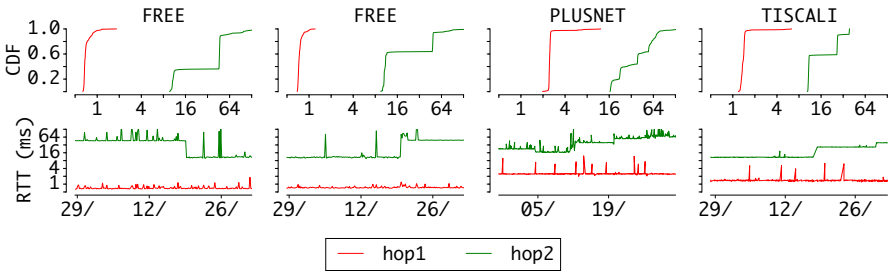


Figure 94: *CDF (top) of latencies to hop1 and hop2 from four probes connected behind DSL networks. Interleaving depth level changes can be seen through a step-wise functional change. Hop2 latency transitions tend to correlate with corresponding timeseries (bottom).*

The number of RS codewords accumulated before transmitting the frame determines the depth of the interleaving channel. DSL deployments employ the Dynamic Line Management (DLM) technique to remotely monitor line characteristics such as the amount of packet loss encountered on the last-mile. They use this information to dynamically adapt interleaving depth levels. An increase in depth level increases latency. An increase in latency can directly impact applications leveraging congestion aware transport protocols such as TCP. An interleaving depth level of 1 is known as *fastpath* which is more sensitive to real-time communication applications but only appropriate for links with low error rates. DSL operators tend to support both fastpath and higher depths, although not all operators allow fastpath on the last-mile. It is also unlikely that a deployment will only support fastpath.

In our pursuit to identify interleaving depths, we separately investigated latencies observed by probes deployed behind DSL networks. A change in the interleaving depth level changes the hop2 latency by around 5ms [241]. A step-wise transition on the CDF derived from hop2 latencies indicates a switch between such depth levels. Fig. 94 shows example probes that witnessed depth-level changes. These probes portray hop2 latencies distributed as step-wise functions. It can be seen how multiple depth level transitions occurred over a span of a month. The corresponding timeseries tends to correlate with the depth changes showing how DSL networks tend to vary interleaving depths over time. SamKnows probes perform measurements only in the absence of cross-traffic, as a result the second-hop transitions cannot be attributed to bufferbloat [23] on the home gateway. Each datapoint in the timeseries is an average of three queries, as a result, some spikes are also visible. In order to automate the discovery of probes experiencing such a behavior, we extracted relative maximas from the Kernel Density Estimation (KDE) derived from hop2 latencies witnessed by each probe. We used a sample threshold on the frequency of occurrence for each local maxima to ensure hop2 latencies remained stable for an extended period. We tagged probes with a depth-level transition in situations where the local maximas were atleast 5ms apart from each other. Fig. 95 shows the distribution of probes that experienced 2-levels and 3-levels of interleaving depth level changes.

13.5.4 *Last-mile latencies by time of day*

In order to circumvent effects of latencies induced within a home network, we calculate *last-mile latency* as the difference between the hop2 and hop1

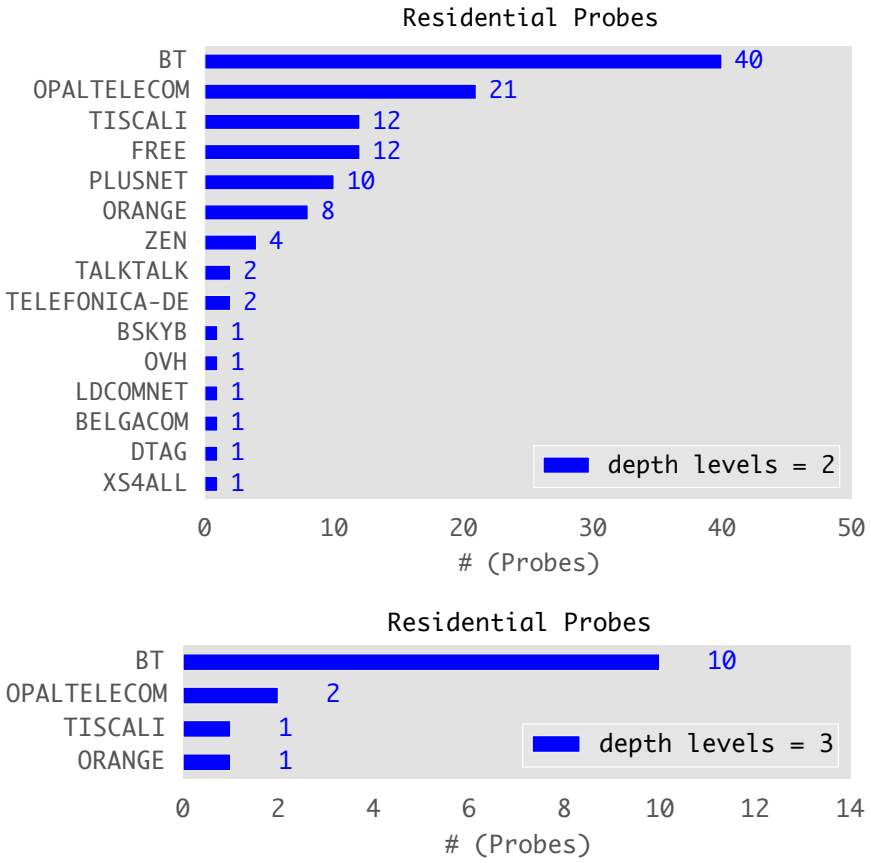


Figure 95: *Distribution of probes that witnessed 2-levels (above) and 3-levels (below) of interleaving depth changes over the span of a month across DSL service providers.*

latency. Last-mile latencies described beyond this point reflect this definition. We describe last-mile latency observed by an individual probe as the median of last-mile latencies observed by that probe over the entire measurement duration: $median(h2 - h1)$.

We investigated the distribution of last-mile latencies over 24 hour cycles for DSL, cable and fibre deployments. Fig. 96 shows boxplots of last-mile latencies observed over each hour. Note, our measurements were taken every 4 hours over a 35 days period. Since SamKnows tends to distribute probes within the frequency interval, measurements spread over each hour of the day. RIPE Atlas only recently (since Nov 2015) introduced this feature of controlling the spread [328]. Given our dataset spans Aug 2014, RIPE Atlas measurements strictly occur on the 4 hour boundary. Since, SamKnows (unlike RIPE Atlas) probes do not perform measurements in presence of cross-traffic [306], the number of SamKnows probes running measurements change every hour unlike that of RIPE Atlas where all probes participate in the measurement. It can be seen that the last-mile latency is stable over time and is not affected by diurnal load patterns. Note that our measurement method has been designed to eliminate queuing delays such as delays caused by home gateways with bloated buffers [23] in front of an overloaded access line. As a such, this observation is in line with expectation. A DSL line is not shared with other customers (except indirectly via crosstalk impacting signal quality) and hence load should not affect DSL line behaviour in significant ways. For cable access networks, the situation is slightly different but it

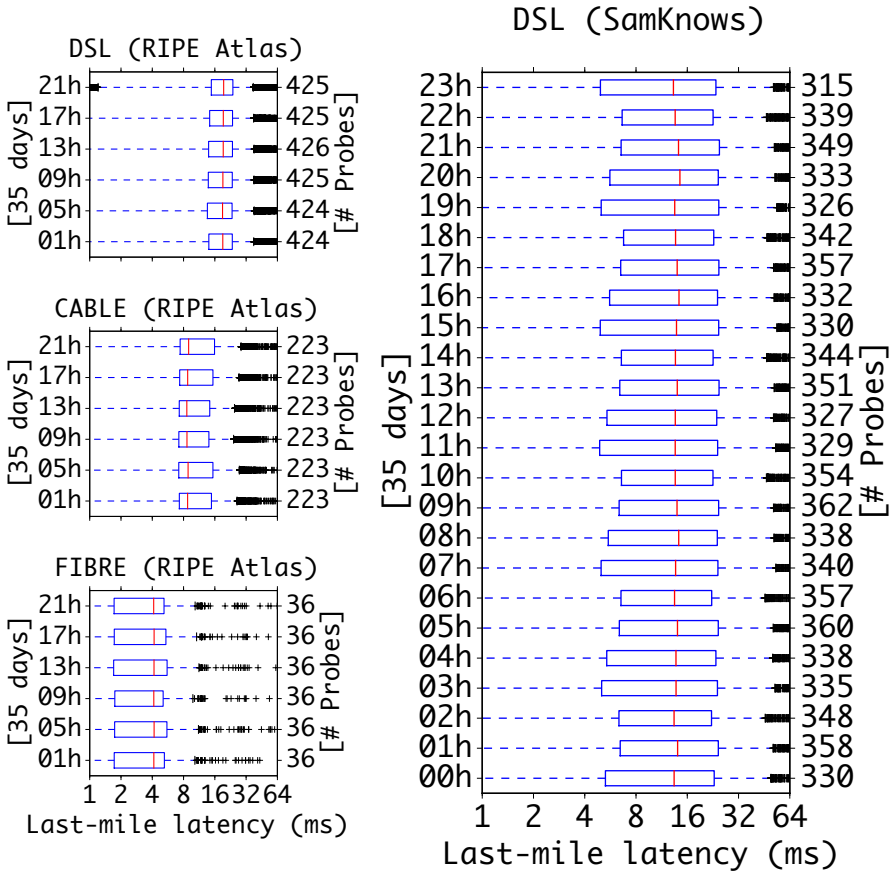


Figure 96: *Last-mile latencies by time (UTC) of day for RIPE Atlas and SamKnows. The last-mile latencies remain considerably stable by time of day irrespective of the used access technology.*

seems that deployments have enough capacity to sustain load such that the time-slotted approach makes them behave in a reasonably robust way.

13.5.5 *Last-mile latencies by service provider*

We further grouped last-mile latencies by each service provider network. Fig. 97 shows the distribution of last-mile latencies observed by RIPE Atlas probes within DSL (429 probes), cable (225 probes) and fibre (36 probes) service provider networks. Each data point is last mile latency observed by a probe. We witness how last-mile latencies exhibited by DSL providers in the EU (centered around 16 ms) are higher when compared to cable providers (centered around 8 ms). The last-mile latencies within fibre deployments are relatively lower than that of both DSL and cable deployments. The last mile latencies for fibre appear to be less than 2 ms for VIEWQUEST, while probes appear to cluster into 2 groups (centered around 4 ms and 7 ms) for UUNET fibre-only providers. The distribution shows higher variation in DSL networks due to the multiple levels of interleaving depths enabled depending on the line characteristics and geographical location of the subscriber. On the contrary, cable providers in the US appear to show significantly different results. Probes behind COMCAST and TWC for instance appear to cluster together in two groups. One of the clusters exhibit last-mile latencies similar to EU cable providers (centered around 8 ms), while the other cluster exhibits

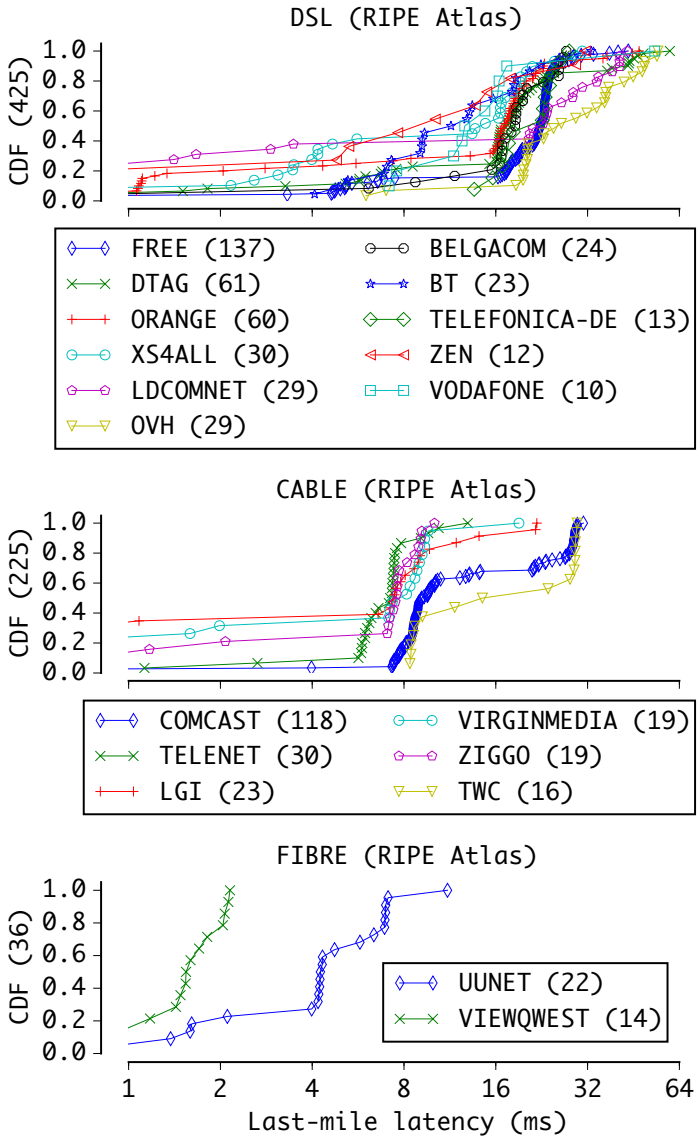


Figure 97: *CDF of last-mile latencies for DSL, Cable and Fibre service providers (RIPE Atlas). Last-mile latencies are ordered as DSL > Cable > Fibre for multiple service providers. The last-mile latencies for Cable providers in the US are clustered into 2 groups. One of the clusters shows last-mile latencies similar to that of DSL.*

last-mile latencies similar to EU DSL providers (centered around 32 ms). We suspect that cable providers in the US enable interleaving on the last-mile for some (discussed in the next section) of their subscribers. Fig. 98 shows corresponding box plots of last-mile latencies separated by service provider in each access technology. It can be seen how the 75^{th} percentile for DSL is between [16-40] ms, for cable is between [6-10] ms and for fibre is between [2-8] ms for EU service providers. The 75^{th} percentile for cable providers in the US is similar to DSL and is between [28-32] ms. Fig. 99 shows box plots of last-mile latencies observed by SamKnows probes within DSL (994 probes) and cable (195 probes) deployments in the UK. It can be seen how the 75^{th} percentile for DSL is between [22-40] ms, while that for cable is around 10 ms which is similar to observations witnessed from the RIPE Atlas dataset. Furthermore, with higher density of probes in the SamKnows

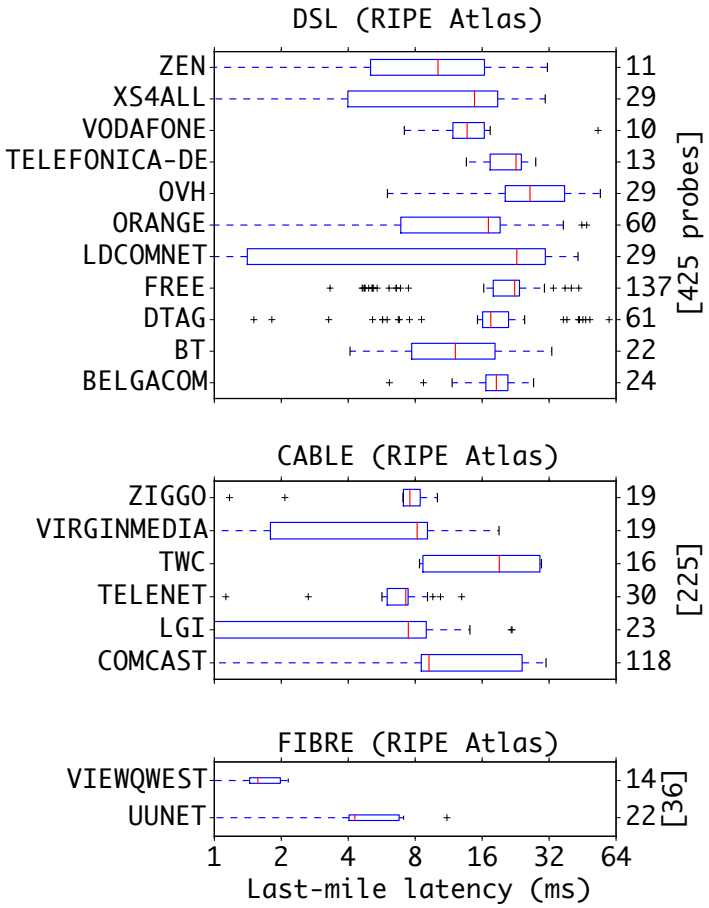


Figure 98: *Box plots of last-mile latencies for DSL, Cable and Fibre deployments (RIPE Atlas). The 75th percentile of last-mile latencies for DSL > Cable > Fibre within EU. The 75th percentile for Cable in the US is similar to DSL.*

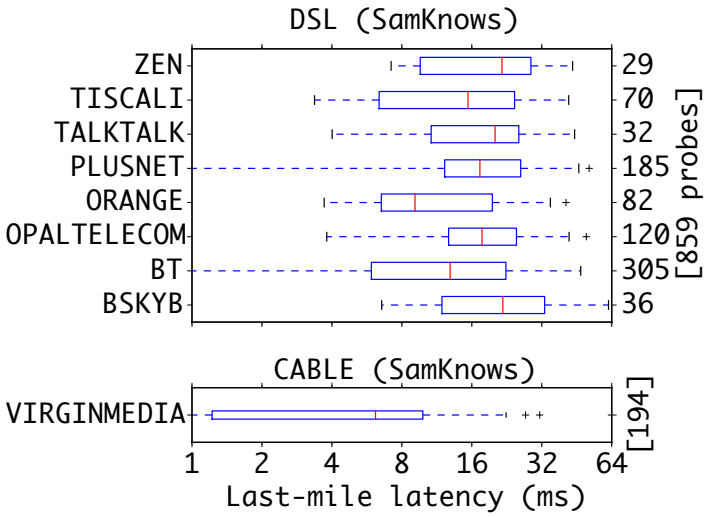


Figure 99: *Last-mile latencies for DSL and Cable deployments in the UK (SamKnows). The 75th percentile of last-mile latencies for DSL > Cable.*

dataset within each service provider network, we were able to check the

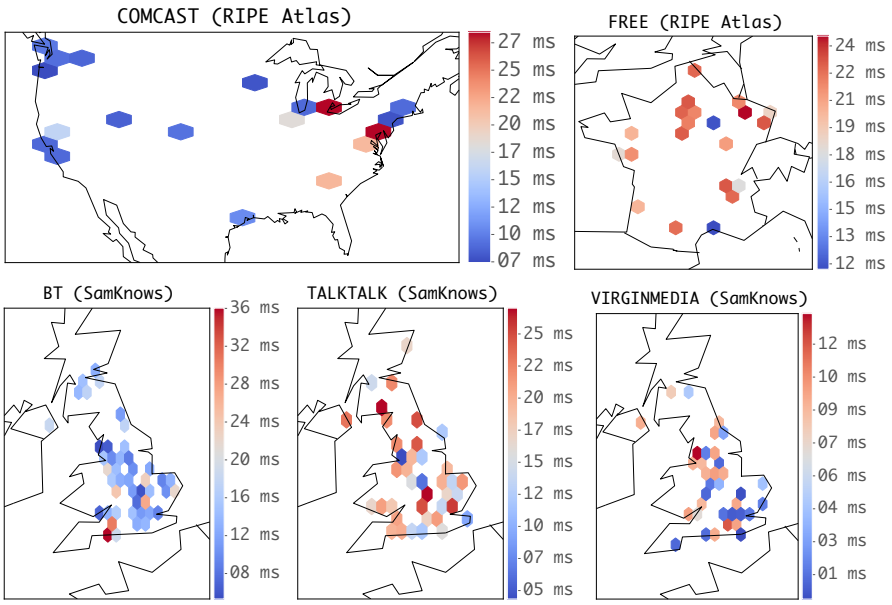


Figure 100: *Last-mile latencies of service providers segregated by location. Users experience different last-mile latencies depending on the location of service provider subscription.*

consistency of our results obtained from the RIPE Atlas dataset spanning more service providers with relatively less density of probes.

13.5.6 *Last-mile latencies by subscriber location*

Fig. 97 shows that probes behind COMCAST, TWC and LGI appear to cluster together in two groups. One of the clusters exhibit last-mile latencies centered around 8 ms, while the other cluster exhibit last-mile latencies centered around 24 ms. We further investigated the last mile latencies by clustering probes of a service provider by their subscriber location. Given the RIPE Atlas dataset consists of probes located in both EU and US regions, the probes are located in different timezones. We use timezones since they provide a good granular separation by location (countries are too coarse grained, cities are too fine grained given the number of probes within each service provider). Fig. 100 shows the distribution of last-mile latencies grouped by timezones for selected service providers where we have a higher sample (more than 100) of probes. This division reveals the reason for 2 clusters witnessed in the CDF (see Fig. 97) plot. Fig. 101 shows that COMCAST with last mile latencies centered around 8 ms are exhibited by probes in the Los Angeles region, while last mile centered around 24 ms is exhibited by probes in the NYC region. Similar results are observed for TWC and LGI-UPC service providers. Although, for LGI, the difference is small but for TWC and COMCAST the difference of the medians is very significant.

13.5.7 *Last-mile latencies by broadband product*

Fig. 102 shows last-mile latencies observed by DSL SamKnows probes separated by 4 broadband products: a) ADSL 8 Mbps (133 probes), b) ADSL2+ 20 Mbps (420 probes), c) VDSL 40 Mbps (128 probes) and d) VDSL 80 Mbps (246 probes). The lastmile latencies observed by probes behind ADSL and

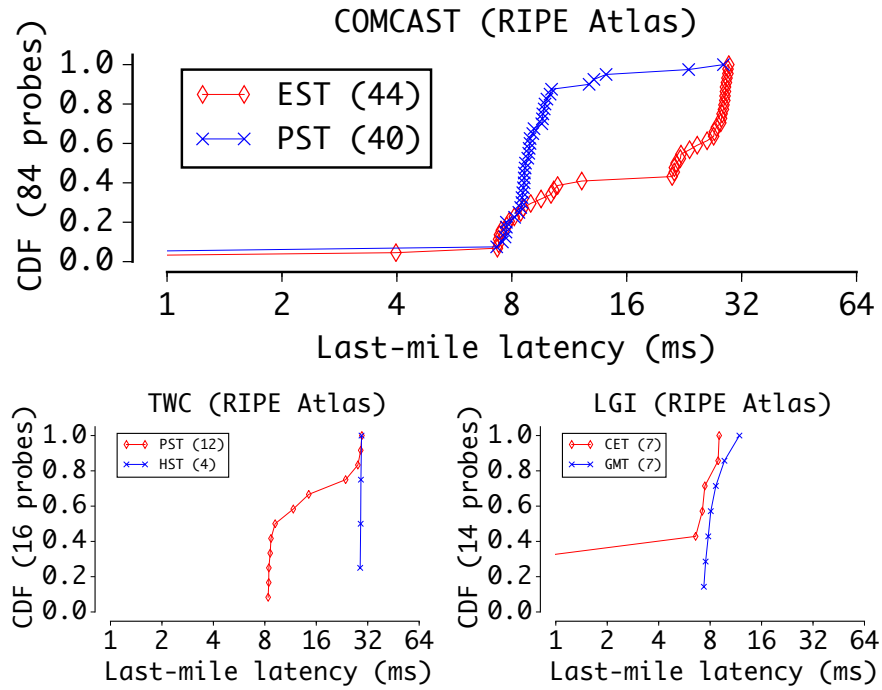


Figure 101: *Last-mile latencies of service providers separated by timezone. COMCAST and TWC subscribers experience considerably different last-mile latencies across the east and west US coast.*

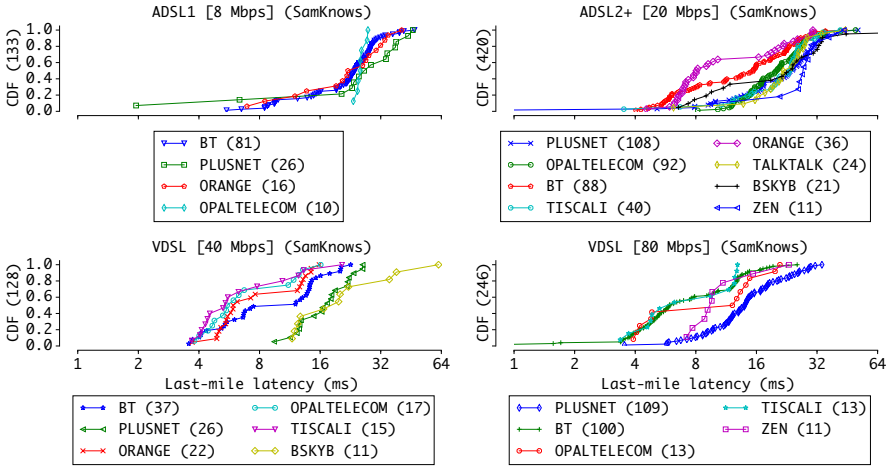


Figure 102: *CDF of last-mile latencies for DSL service providers seperated by broadband product. The last-mile latencies for ADSL/ADSL2+ > VDSL deployments.*

ADSL2+ products are similar and fall within [16-32] ms range. Although a cluster of probes behind ADSL2+ lines also center around 8ms and show last-mile latencies lower than ADSL. On the other hand last-mile latencies for VDSL products tend to show considerably lower (centered around either 6ms or 12-22ms) last-mile latencies when compared to ADSL and ADSL2+ products. Fig. 103 shows corresponding box plots of last-mile latencies within DSL deployments separated by broadband product. It can be seen how the 75^{th} percentile for ADSL is between [25-35] ms, for ADSL2+ is between [22-35] ms, for VDSL (40 Mbps) is between [7-30] ms while for VDSL (80

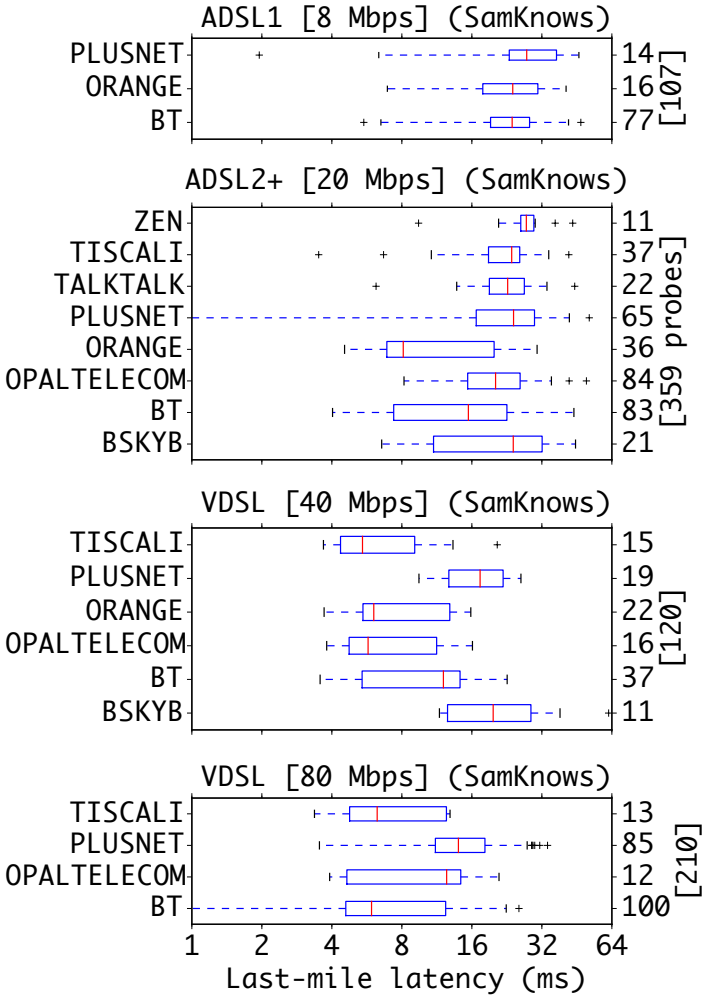


Figure 103: *Box plots of last-mile latencies for DSL service providers separated by broadband product (SamKnows). The 75th percentile of last-mile latencies for VDSL < ADSL/ADSL2+*

Mbps) is between [4-20] ms. The 75th percentile of last-mile latencies for VDSL products is lower than ADSL and ADSL2+ products.

Fig. 104 shows service providers where there is a tangible decrease of last-mile latency with increase in broadband product. In DSL deployments, higher bandwidth capacities are made possible by using higher range frequencies on the physical link. These frequencies tend to dissipate over shorter distances. Therefore, ADSL2+ and VDSL deployments tend to be closer to the traffic aggregation points. Although, a reduction in copper length does not have significant effects on last-mile latency. Furthermore, with an increase in line speeds, ADSL2+ and VDSL deployments allow frames to be transmitted faster. Higher transmission rates help reduce interleaving delays, which can significantly reduce latencies experienced on the last-mile.

13.6 CONCLUSION

We leveraged the RIPE Atlas and SamKnows platform to measure last-mile latency. This is the first study that has measured last-mile latencies on such a

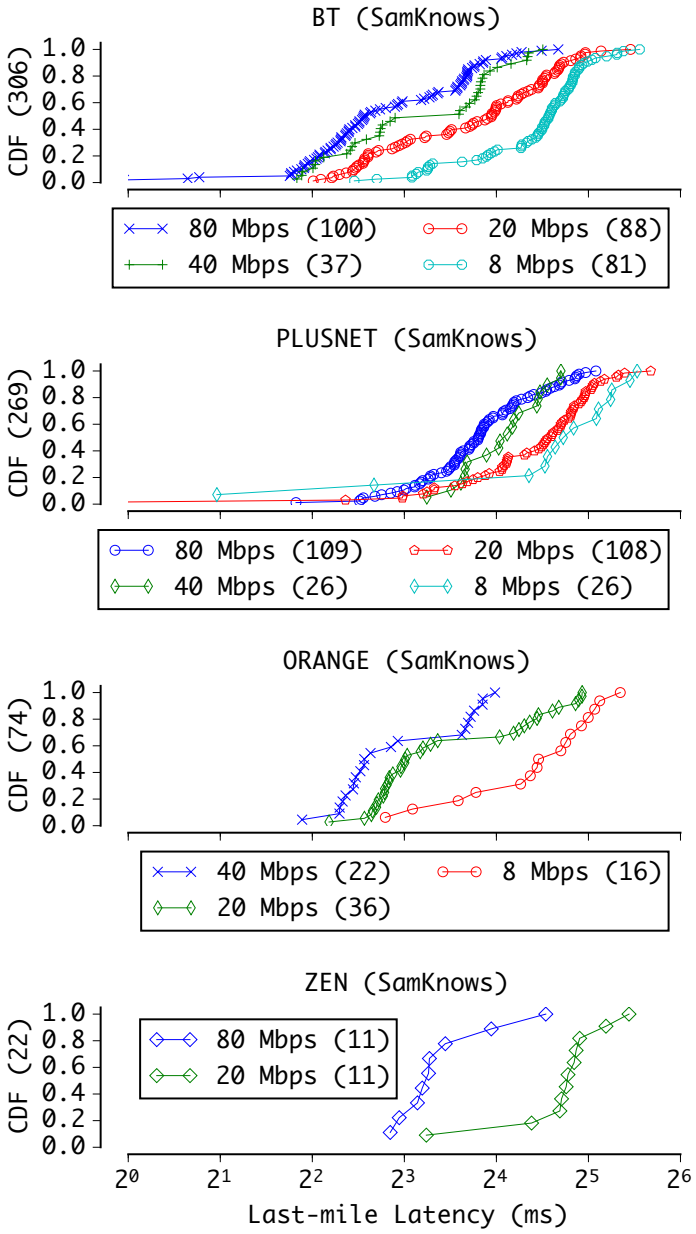


Figure 104: *Four service providers exhibiting a tangible decrease in last-mile latency with increase in broadband product subscription. Higher transmission rates with ADSL2+ and VDSL deployments help reduce last-mile latencies.*

global scale from within multiple service providers networks in the US and the EU. We conclude with some high-level takeaway lessons:

Lesson 1: *The home network latency can make a discernible contribution and therefore should not be accounted when measuring last-mile latency.* – We therefore define last-mile latency as the latency between the home router and the first IP hop in the access network.

Lesson 2: *Some home routers rate limit ICMP responses to TTL expiry that makes them unsuitable for baseline measurements.* – Measurement points impacted by such home routers should not be included for baseline latency measurements.

Lesson 3: *DSL service providers enable interleaving and some providers dynamically adapt interleaving depth levels depending on the line characteristics*

and geographic location of the subscriber – For some measurement points, we observed depth level changes occuring on a weekly time scale.

Lesson 4: *Once the effects of queuing delay caused by bufferbloat have been eliminated, access networks tend to exhibit robust last-mile latency* – We witnessed that last-mile latency is considerably stable over time and not affected by diurnal load patterns.

Lesson 5: *Last-mile latency for DSL deployments is centered around 16 ms. Cable networks show a last-mile latency centered around 8 ms and fibre to the home networks show a last-mile latency centered around 4 ms.* – This observation will allow simulation studies to appropriately model DSL, cable and fibre access links in future research.

Lesson 6: *Last mile latencies of a service provider can depend on the geographic location of a subscriber.* – We observed significant last-mile latency differences for US cable service providers across the east (centered at around 32ms) and west (centered at around 8ms) coast.

Lesson 7: *Last-mile latencies of DSL deployments vary with the the broadband product subscription.* – While the last-mile latencies for products based on ADSL2+ and VDSL can be significantly lower compared to the latency of ADSL1 products, we also observed an increase in latency variation across our measurement points for ADSL2+ and VDSL products.

We have shown that, with careful vantage point selection [306], an open measurement platform such as RIPE Atlas can be used to study last-mile behavior. We have validated the results obtained using the RIPE Atlas platform against data obtained from the SamKnows measurement platform, which was specifically designed for measuring broadband network performance.

Part V

LESSONS LEARNED / FUTURE OUTLOOK

We share our experiences in using an open platform, RIPE Atlas and compare it with SamKnows that is dedicated for measuring broadband performance. For instance, we discuss the significance of probe calibration and and show how we leverage it to identify load issues in older hardware versions of RIPE Atlas probes. We demonstrate example use-cases how performance measurement platforms can benefit from each other's experience. We further stress towards the importance of probe metadata and inherent sampling bias embedded in probe-based measurement platforms.

In Chapter 14 we describe lessons learned from using RIPE Atlas and SamKnows platforms. We provide high-level conclusive summary and directions for future research in Chapter 15.

14 LESSONS LEARNED FROM USING RIPE ATLAS

We reflect upon our experience in using the RIPE Atlas platform for measurement-based research. We show how in addition to credits, control checks using rate limits are in place to ensure that the platform does not get overloaded with measurements. We show how the AS-based distribution of RIPE Atlas probes is heavily skewed which limits possibilities of measurements sourced from a specific origin-AS. We discuss the significance of probe calibration and how we leverage it to identify load issues in older hardware versions (38.6% overall as of Sep 2014) of probes. We show how performance measurement platforms (such as RIPE Atlas, SamKnows, BISmark and Dasu) can benefit from each other by demonstrating two example use-cases. We also open discussion on how RIPE Atlas deployment can be made more useful by relaying more probe metadata information back to the scientific community and by strategically deploying probes to reduce the inherent sampling bias embedded in probe-based measurement platforms.

Contents

14.1 INTRODUCTION

RIPE Atlas [4] has deployed around 12.8K dedicated hardware probes and around 109 anchors (as of Feb 2015) all around the globe as shown in Fig. 105. Probes perform active measurements to ascertain network connectivity and reachability of the global Internet, while anchors are dedicated servers that can act as sources and sinks of measurement traffic. RIPE Atlas periodically schedules measurements using a batch of several hundred probes against anchors to measure region-based connectivity and reachability. A majority of these probes are running measurements either from the core or from within access networks. A discernible number of probes are also hosted by volunteers within their home networks. Table 10 provides a list of built-in measurements performed by probes by default. All hosted probes are made publicly available for measurement research. These probes in addition to built-in measurements can also run UDMs. A UDM allows any user registered (around 19K as of Feb 2015) on RIPE Atlas to provision measurements supported by the platform (see Table 10) on probes with tailor-made measurement parameters. A registered user spends credits by provisioning a UDM on probes. Credits can be gathered by either hosting a probe (for no purchase cost) or an anchor (for a purchase cost). RIPE Atlas also released (on Feb 2013) a public API that allows one to programmatically provision

Figure 105: *Coverage of the RIPE Atlas measurement platform as of Feb 2015. Around 12.8K probes (top) and 109 anchors (bottom) have been deployed in total: atlas.ripe.net/results/maps. The green, red and grey areas (above) represent connected, disconnected and abandoned probes respectively.*

UDMs. Using these public APIs and credits gathered by hosting probes for multiple years, we were able to provision UDMs on a large sample of probes. We share our experiences and lessons learned from using the RIPE Atlas platform for measurement research.

14.2 RATE LIMITS

RIPE Atlas uses credits as a virtual currency to regulate UDM usage within the platform. Millions of credits can be accumulated by hosting probes for multiple years. Given the credit consumption of individual built-in measurement is fairly low (see Table 11), it provides an impression that given ample credits, large number of measurements can be provisioned on the platform. However, the platform also imposes four daily rate limit thresholds on each user account: a) No more than 100 simultaneous measurements, b) No more than 500 probes/measurement, c) No more than 1M credits may be used each day and d) No more than 10 ongoing and 10 one-off measurements of

MEASUREMENT	TARGET
ping, ping6	first hop, second hop, ns.ripe.net, *.root-servers.net, *.atlas.ripe.net
traceroute, traceroute6	*.root-servers.net, *.atlas.ripe.net, labs.ripe.net
dns, dns6	*.root-servers.net: TCP (SOA), UDP (SOA, version.bind, hostname.bind, id.server, version.server)
sslcert, sslcert6	www.ripe.net, atlas.ripe.net
http, http6	www.ripe.net/favicon.ico, ip-echo.ripe.net

Table 10: *A list of built-in measurements performed by probes by default as of Feb 2015. (*) in the target fields indicate multiple servers within the domain.*

MEASUREMENT	CREDITS/RESULT ↓
traceroute, traceroute6	30
dns, dns6 (TCP)	20
dns, dns6 (UDP)	10
sslcert, sslcert6	10
ping, ping6	3

Table 11: *Credit cost consumption of built-in measurements as of Feb 2015: atlas.ripe.net/docs/credits. These are credits consumed by measurements using default parameters. These costs can increase (or decrease) if default measurement parameters are tweaked.*

the same type against same target at any time. These rate limits, although documented [329] may not be well-known to the research community. These limits may coerse one to design experiments that span multiple days. As such a request to lift these limits can be made by proposing and gathering support for the measurement study on the `atlas` mailing list.

14.3 HEAVY-TAILED PROBE DISTRIBUTION

The geographical distribution of the probes (see Fig. 105) provides a decent high-level overview of the coverage of the platform. Although the network coverage map [330] provides a facility to filter probes by ASN, the overall distribution of probes across ASes and density of probes within each AS is not well known. Measurements sourced from a specific AS require high probe density to mantain a representative sample, while measurements destined towards a specific AS require diversity of network origins. As such, we performed an experiment to better understand the AS-based distribution of these probes.

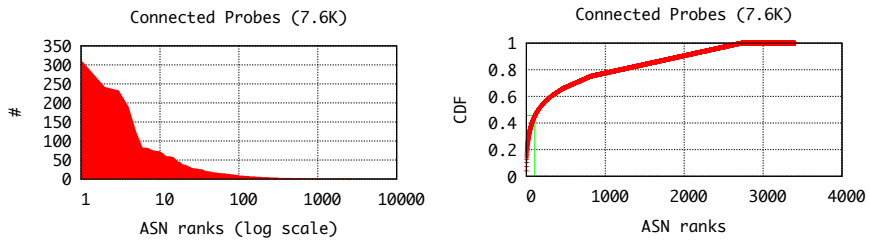


Figure 106: *Distribution of a subset of connected and non-anchored probes (7672) sorted by AS rank as of Feb 2015. ASes are ranked by number of probes. 44.59% (3421) of connected probes fall within AS ranks <= 101. Rest of ASes contain < 10 probes. The dataset is available at: goo.gl/kmIydP*

AS RANK	AS (ASN)	#(PROBES) ↓
01	COMCAST (AS7922)	313
02	PROXAD (AS12322)	242
03	LGI-UPC (AS6830)	233
04	DTAG (AS3320)	190
05	ORANGE (AS3215)	124
06	ZIGGO (AS9143)	83
07	XS4ALL (AS3265)	82
08	BT (AS2856)	76
09	UUNET (AS701)	74
10	VIRGINMEDIA (AS5089)	73

Table 12: *Distribution of a subset of connected and non-anchored probes (7672) sorted by AS rank as of Feb 2015. ASes are ranked by number of connected probes. The entire dataset is available at: goo.gl/kmIydP*

Clustering probes by ASN

We use the RIPE Atlas probe API [325] to capture a list of connected probes in order to later cluster them by their origin AS. The API, however, does not reveal the ASN for all probes. For instance, some probes (2037, 15.9% of all registered probes as of Feb 2015) did not expose either their public IP or their origin-AS. We grabbed the probe IDs of these probes and provisioned a one-off (measurement that runs only once) `traceroute` measurement. The measurement was scheduled only on a few probes (43 out of 2037) while the rest were deemed disconnected by the scheduler. We identified the origin AS of these probes, and pruned the rest of the disconnected probes out of the list. We also used the mapping in Fig. 108 (described later in the chapter) to rule out anchors (109 as of Feb 2015). Going forward, we use the term probe to refer to the connected and non-anchored subset (7672) of all RIPE Atlas probes (12790).

Ranking ASNs by number of probes

We ranked ASNs by sorting them by the number of deployed probes. Table 12 provides a list of top 10 ASes containing the highest number of probes. For instance, Comcast (AS7922) has 313 (out of 7672) probes which contributes to

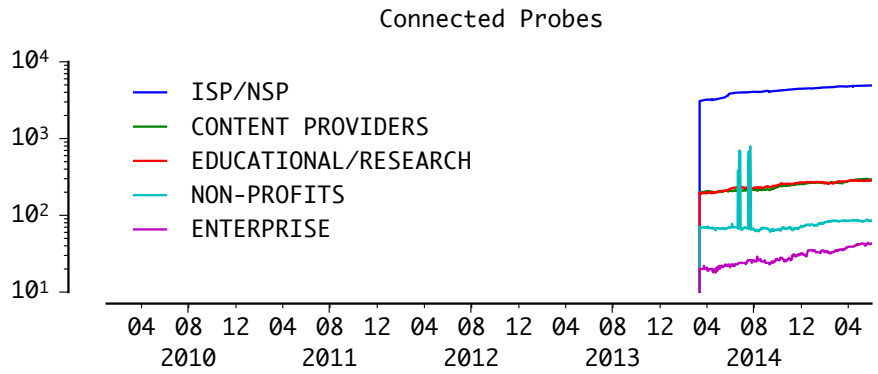


Figure 107: *Evolution of probes by network type as mapped by PeeringDB. The plot is generated using the probe archive API: goo.gl/pMHs9Q which provides probe metadata since March 2014. Majority of probes are deployed behind service provider networks.*

4% of all probes. The cumulative probes within top 10 AS ranks contribute to 18% of all probes as of Feb 2015. Fig. 106 shows the long-tail probe distribution sorted by AS ranks. A corresponding CDF of this long-tail, shows how probes deployed within AS ranks > 101 have less than even 10 probes. To bring numbers into perspective, if we were to consider 10+ probes as a representative sample within each AS, the number of probes falling within AS ranks <= 101 would contribute 44.59% (3421 out of 7672) which is less than half of the entire population of probes.

Clustering ASNs by network type

Using PeeringDB, we further mapped ASes hosting the connected probes (7672 as of Feb 2015) by their network type information. PeeringDB [331] is a database holding peering information of participating networks. Aemen Lodhi *et al.* in [250] show how the information maintained within this database is reasonably representative of network operator peering and is also up-to-date. Fig. 107 shows the evolution of probes by network type over a year. Few spikes occur in the non-profit network type due to a large fraction of probes (with a series of consecutive probe IDs) coming online for a day (or few days) from within the RIPE NCC network. Not all ASes hosting connected probes could be mapped to a network type due to missing AS information (encompassing 33.5% probes as of Feb 2015) in the PeeringDB database. Nevertheless, this mapping provides an indication on which type of networks hold major portion of connected probes. As such, RIPE Atlas is a potential platform for performing active measurements from within service provider networks.

Skewed distribution of probes

The RIPE Atlas platform ostensibly appears to have a large number of deployed (12.8K registered as of Feb 2015) probes. However, it turns out that the number of probes available for a measurement study sourced from a specific origin-AS is small. This is due to the skewed distribution of probes which considerably reduces the density of probes behind each AS. In all fairness, the platform was initially designed to measure connectivity and reachability. As such, there has been an inclination to deploy probes to

PROBE ID	HARDWARE VERSION	HARDWARE	RAM	WEBPAGE
[1, 1521]	probev1	Lantronix XPort Pro	8 MB	probev1.ripe.net
(2000, 5000)	probev2	Lantronix XPort Pro	16 MB	probev2.ripe.net
(10000, +∞)	probev3	TP-Link TL-MR3020	32 MB	probev3.ripe.net
(6000, 6018)	anchorv1	Dell PowerEdge	-	-
[6018, 7000)	anchorv2	Soekris Net6501-70	-	anchorv2.ripe.net

Figure 108: *Family of hardware probes deployed by the RIPE Atlas platform as of Sep 2014. v3 probes are more capable than v1/v2 probes in hardware specifications. Anchors are dedicated servers that act as sources and sinks of measurement traffic. The probeID can be used to identify the hardware version. Firmwares are kept in sync across hardware versions. The probe ID to hardware mappings were generated from: goo.gl/qABo1w.*

increase coverage (than density) by biasing distribution in favor of under-served ASNs. As a result, the platform is more suitable for performing measurements targeted to a specific destination as it provides diversity of network origins.

14.4 LOAD ISSUES IN OLDER PROBES

RIPE Atlas currently runs measurements from three (v1, v2, v3) different probe hardware versions as shown in Fig. 108. In order to have the same capabilities available, the platform tries to keep firmware versions in sync across hardware versions. In our pursuit to understand whether running the same firmware release on all hardware versions makes any impact on measurement results, we performed firmware and hardware calibration of the probes. We show how such a calibration allowed us to identify load issues in older (v1 and v2) hardware versions of the probes.

Probe calibration

RFC 3432 [332] defines calibration as the process of determining the systematic (constant bias in measured values) and random error generated by the instruments themselves in as much detail as possible. In this work we focus on calibration to adjudicate the systematic error in probes.

Firmware variants: The firmware release running on the probes is one such parameter that can create a systematic error in measured values. Each firmware release brings with it, codebase changes either as bug fixes or as new feature updates that can have an impact on measurement results. Fig. 109 for instance shows that RIPE Atlas firmware release cycles have become more frequent since 2013. As a result, chances of a measurement campaign crossing these firmware release boundaries have also become more pertinent. Even if a measurement compaign does not cross a firmware boundary, it's generally useful to be able to track back to the firmware codebase in situations where

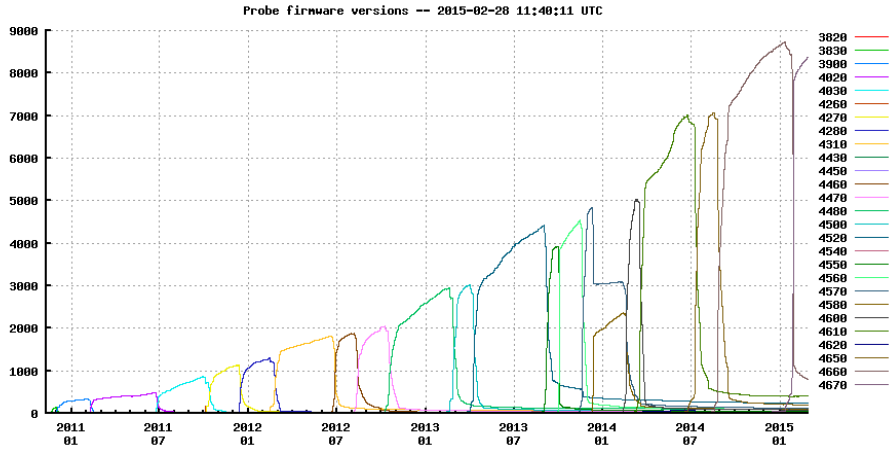


Figure 109: *Firmware release cycles since 2011 (as of Feb 2015): atlas.ripe.net/results/graphs*

```
{
"prb_id": 10305,
"type": "traceroute"
"fw": 4660,
...
}
```

Listing 1: A snippet of a `traceroute` measurement result from a probe (as of Sep 2014).

an unexpected measurement result is observed. In order to allow firmware calibration, the platform inherently tags (see Listing 1) the firmware release for each measurement result to allow one to later trace back to the source code.

Hardware variants: While RIPE Atlas attaches each UDM with the firmware version of the probe, hardware versions are not tagged and therefore not reported. The platform runs measurements from three probe (v1, v2 and v3) hardware versions. v1 and v2 probes are made of a custom hardware built around a Lantronix XPort Pro module, while v3 probes are off-the-shelf TP-Link wireless routers flashed with OpenWrt [266]. As a result, v3 probes are more capable (in terms of hardware specifications) than older v1 and v2 probes. In addition, measurements can also be provisioned on anchors (dedicated servers), further adding to the hardware variability. Therefore, we asked on the `atlas` mailing list and identified how the probe ID itself can reveal hardware versions of the probes. Fig. 108 describes the mapping of a probe ID to its hardware version.

Segregating measurements by hardware

In our pursuit to study whether different hardware versions have effects on measurement results, we performed an experiment on probes deployed in a residential network. We specifically used probes that were directly wired behind the home gateway. This helps ensure that our measurements do not get skewed by probes that cross any wireless links (not wireless bridges)

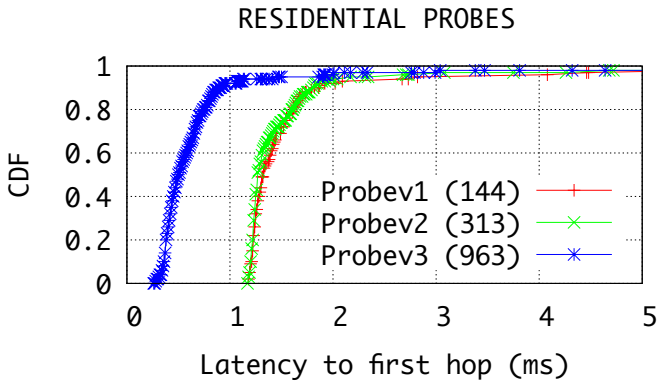


Figure 110: *CDF of latencies to first hop observed over a day-long traceroute measurement for v1, v2 and v3 hardware probes wired behind residential gateways as of Sep 2014. v3 probes (in blue) show expected <1ms latencies, while v1 probes (red) and v2 probes (green) show higher latencies to the home gateway. Probes were running firmware version: 4650 and 4660. The x-axis of the plot is cut off at 5ms. The entire raw dataset is publicly released at: goo.gl/NRPxb7.*

within the home network. The probe itself cannot associate to a wireless access point because RIPE Atlas has stripped all wireless capabilities out of the firmware. In order to filter for this sample, we searched for probes whose first-hop was in a private IPv4 address space [311], but their second hop was in a public IPv4 address space. Using this sample of residential probes, we provisioned IPv4 `traceroute` measurements once every 15 minutes for a day. In order to study effects of hardware (see Fig. 108), we further separated measurement results by each hardware version.

Fig. 110 shows the latency measured to the first hop (home gateway) observed over a day from all three (v1, v2, v3) probe hardware versions. A probe directly connected to the residential gateway should not show first-hop latencies of more than 1ms. We see how a significant number of v3 probes show such a behavior, however almost all v1/v2 probes show higher first-hop latencies.

Since the platform tags the firmware release in each measurement result, we were able to trace back to the source code of the firmware running these measurements to better understand the source of the issue. The source code reveals how the entire measurement framework is built around busybox [333]. Each measurement test has been adapted to run in an event-driven manner using `libevent`. As a consequence, whenever a UDM request is initiated, tests that run the measurement are not spawned as new processes, but are invoked as separate function calls. There is a single process that handles a single event loop for all incoming measurement requests. The source code has been designed in this way to help circumvent the unavailability of a MMU in v1 and v2 probes and to avoid allocating memory for multiple stacks (such as one would do in a multithreaded implementation). The latest family of v3 probes do have a MMU and significantly more memory (see Fig. 108), but in order to keep firmwares in synchronization across hardware versions, this implementation strategy has also been carried forward in v3 probes.

```
static struct trtbase *traceroute_base_new (
  struct event_base *event_base
) {
...
event_assign(&base->event4, base->event_base,
  base->v4icmp_rcv, EV_READ | EV_PERSIST,
  ready_callback4, base);
}

static void ready_callback4 (
  int __attribute((unused)) unused,
  const short __attribute((unused)) event,
  void *s
) {
...
struct timeval now;
gettimeofday(&now, NULL);
ms=(now.tv_sec-state->xmit_time.tv_sec)*1000 +
   (now.tv_usec-state->xmit_time.tv_usec)/1e3;
}
```

Listing 2: A traceroute code snippet from 4570 running on v1/v2 probes as of November 2013. The source code is available at: atlas.ripe.net/get-involved/source-code

Listing 2 shows a sample snippet from the `traceroute` source code of the firmware release running these measurements. The function `traceroute_base_new(...)` is invoked when a `traceroute` measurement is requested, where it registers a callback. As can be seen, the RTT time stamping of a response to an ICMP query is performed in the event callback function `ready_callback4(...)` in user space. This means that if a probe is *loaded* with multiple measurements, the user-space time stamping will be delayed. These delays will be more pronounced on constrained hardware such as v1/v2 probes (3961 of 10260 registered probes as of Sep 2014). As such v1/v2 probes (38.6% as of Sep 2014) experience load issues whenever a number of UDMs are provisioned on them.

RIPE Atlas has recently acknowledged our findings [334]. They confirm how adding more code has not had much effect on reducing load issues in v1/v2 probes. They add, in situations where measured first-hop latencies get upto 6ms (also witnessed by us) is when these slower probes are busy performing an Address Resolution Protocol (ARP) request to update their cache entries. In all fairness, the contribution factor of these older hardware versions will slowly fade away (31% as of Feb 2015) as shown in Fig. 111, since the RIPE Atlas platform now dispatches only v3 probes for new volunteers. RIPE Atlas also recently (starting October 2014) introduced the capability to filter probes by their hardware version using tags (such as `system-v1` *et al.*). Using this feature, older versions of the probes can be filtered out when running performance-based (such as latency) measurements. In hindsight, even though v3 probes reduce the impact of user-space timestamping, the platform would also benefit from using kernel-based timestamping using the `SO_TIMESTAMP` socket option on the packets's reception path.

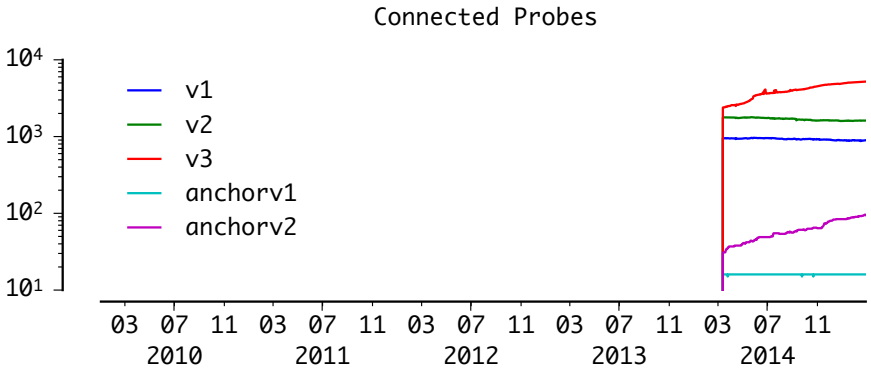


Figure 111: *Evolution of probes by hardware family using probe ID to hardware mapping described in Fig. 108. The plot is generated using the probe archive API: goo.gl/pMHs9Q which provides probe metadata since March 2014. The contribution factor of older hardware version of the probes is fading away.*

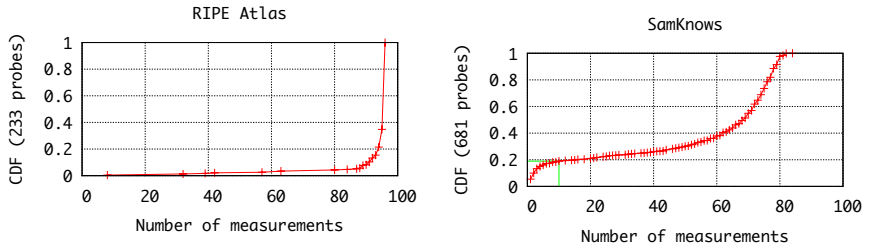


Figure 112: *CDF of number of measurements performed by probes. Around 90% of RIPE Atlas probes (being agnostic to cross-traffic) performed most of the provisioned measurements (more than 90 out of 96) as of Nov 2013. 20% of SamKnows probes (due to cross-traffic detection) performed only few of the provisioned measurements (less than 10 out of 84) as of Feb 2014.*

14.5 CROSS-TRAFFIC AGNOSTIC PROBES

The RIPE Atlas platform (unlike other performance measurement platforms) does not take cross-traffic detection into account when performing measurements. BISmark [4, 18] probes, for instance, read byte counters from `/proc/net/dev` to record passive traffic volume. SamKnows [4] probes use a threshold service to monitor both inbound/outbound traffic on the probe's Wide Area Network (WAN) interface to detect wired cross-traffic. They also record traffic volume exchanged on the user's wireless Service Set Identifier (SSID) to detect wireless cross-traffic. The test runs are delayed once cross-traffic is detected and re-tried with a back-off timer. The entire test cycle is abandoned if the threshold is crossed more than five times in a row. Dasu probes [105] follow a similar approach, but rely on UPnP to query traffic counters on the WAN interface of the residential gateway. SamKnows probes also utilize this out-of-band technique in situations where hosts are not wired behind the probe, but are directly connected to the home gateway.

We performed an experiment to compare the behavior of RIPE Atlas and SamKnows probes in presence of cross-traffic. We requested `traceroute` measurements from both RIPE Atlas (96 samples) and SamKnows probes (84 samples). Fig. 112 shows the distribution of the number of measurements

performed by probes within each platform. It can be seen how 20% of the SamKnows probes provided less than 10% samples due to cross-traffic detection during multiple measurement runs, while 90% of the RIPE Atlas probes being agnostic to cross-traffic contributed to more than 90% of all measurement samples.

In all fairness, the RIPE Atlas platform does not perform cross-traffic detection out of principle. The probes strictly perform active measurements only and no form of passive monitoring (even for cross-traffic detection) is performed in practise. Therefore, studies using RIPE Atlas for performance-based measurements should be aware that their measurements can possibly run in presence of cross-traffic.

14.6 PER-HOP LATENCY AGGREGATIONS

RIPE Atlas probes use `evtraceroute`, a modified version of `traceroute` available in `busybox`. SamKnows probes on the other hand use `mtr`. Whenever a `traceroute` measurement request is initiated on these platforms; three ICMP queries are dispatched per hop by default. While RIPE Atlas probes separately report latencies measured by each ICMP query; SamKnows probes average latencies from multiple ICMP queries over each hop.

We investigated effects of averaging latencies from multiple ICMP queries over a single hop. Fig. 113 shows how averaging latencies over each hop can significantly vary observed results. It can be seen how effects of averaging latencies becomes more pronounced towards the second hop as the latency distribution starts to become more skewed. A mere difference between the averaged second and first hop latencies will now lead to negative results. The aggregation (if necessary) must be done by taking a median of latencies that can better tolerate outliers. We (in collaboration with SamKnows) have updated the `mtr` implementation used by SamKnows to expose each query result separately without any aggregation.

14.7 METADATA IS (CHANGING) DATA

Proper interpretation of measurement results requires metadata to be treated as important as raw measurement data. RIPE Atlas does reveal the geographical location and origin AS of the probe deployment as a metadata entry. However, more metadata is needed to be able to perform specific measurement studies. For instance, the type of network where the probe is deployed, the connection speed and the WAN type of the upstream connection are details that facilitate data analysis. In fact, it requires tremendous manual effort to infer these connection properties through active measurements. Even though possible, these inferences are only heuristics and do not guarantee correct metadata, which only the probe host can accurately supply during the initial registration process. In fact, the current registration procedure [335] does allow a host to provide some details on its connection profile. However, this information is not currently relayed back through the public API. The platform should expose this metadata information alongwith the metadata history so that one can track changes. This would make it easier to isolate probes for a specific measurement study.

RIPE Atlas currently prefers not to report broadband subscription information because of two reasons: a) not all probe hosts record it correctly and b) subscription information tends to stale over time and it takes a major effort to track record changes in subscription switches.

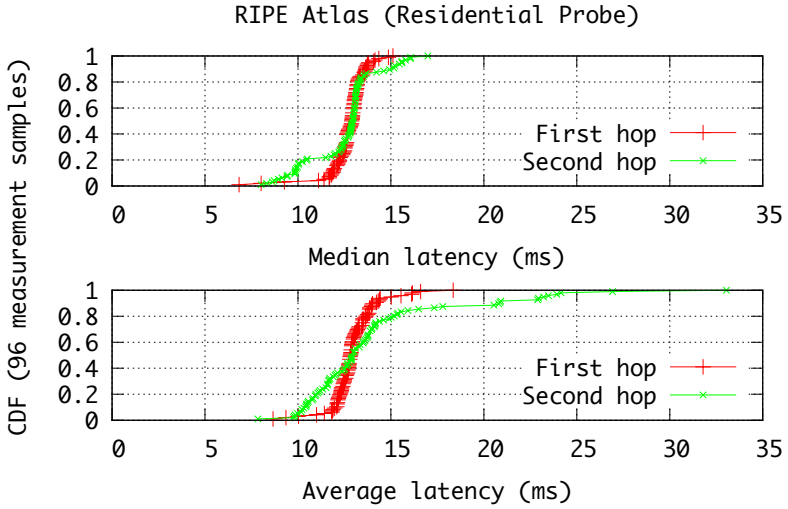


Figure 113: *CDF of latency to the first and second hop from a RIPE Atlas probe as of November 2013. The effect of averaging (below) three queries becomes more pronounced over the second hop when compared to median (above) of three queries. A difference between the averaged latency to second (in green) and first (in red) hop will now lead to negative values.*

14.8 INHERENT SAMPLING BIAS

The deployment of RIPE Atlas probes is biased towards technically-inclined volunteers. A majority of volunteers are network enthusiasts or tend to have close degrees of connections with one. Volunteers hosting such probes tend to have a more complex home network than usual. Since the probe metadata available is currently bleak; the amount of this bias cannot be quantified. Nevertheless, it is important to state that measurements from such vantage points cannot be generalised, particularly in situations where the sample population is low. BISmark [92] and Dasu [105] measurement platforms acknowledge such a biasing limitation in their recent measurement research work.

14.9 CONCLUSION

The RIPE Atlas measurement platform was initially designed to measure connectivity and reachability of the Internet. With the deployment of 12.8K probes, the trend is shifting more towards using this platform for performance-based measurements. In this work, we identified how from among three hardware versions of probes, v3 probes are more suitable for performance (such as latency) measurements than older versions (38.6% of all probes as of Sep 2014) that suffer load issues. Studies using RIPE Atlas to measure latencies therefore need to take the hardware version into account because older versions can produce less accurate results. Given the platform dispatches only v3 probes for new hosts, the contribution impact of older versions (31% as of Feb 2015) is slowly fading away. Although older versions are still useful for measuring reachability and even latency if high precision accuracy is not

the desired goal. We also demonstrated how measurement-based studies that require higher coverage of network origins would benefit more from the platform than those that require high probe density within each network. We also discussed two use-cases where measurement platforms can benefit from one another: a) SamKnows probes are cross-traffic aware (unlike RIPE Atlas probes) and b) RIPE Atlas probes do not aggregate latencies over each `traceroute` hop (unlike SamKnows probes) both of which when disabled can heavily impact measurement results.

15 CONCLUSIONS AND FUTURE WORK

We conclude by summarizing the contributions of the thesis and providing suggestions for future research.

Contents

15.1 CONCLUSIONS

a) **Internet Performance Measurement Platforms**

> RQ – 1 : *What is the state-of-the-art in Internet performance measurement platforms? What is the coverage, scale, lifetime, deployed metrics and measurement tools, architecture and overall research impact of such performance measurement platforms? What standardization efforts are currently being pursued in this space?*

In Part II, we presented a taxonomy of Internet measurement platforms. We subdivided them into topology discovery and performance measurement platforms and further classified the performance measurement platforms based on their deployment use-case: fixed-line access measurements, mobile access measurements and operational support. We described the performance measurement platforms in detail by exploring their scale, coverage, timeline, deployed metrics and measurement tools, architecture and overall research impact. We also presented common set of measurement tools shared by these performance measurement platforms along with the level of collaboration amongst them through the usage of publicly available datasets. We also showed how platforms have been using measurement facilitators to conglomerate data from multiple sources to pursue a particular research question. We concluded the survey by describing recent standardization efforts to make large-scale performance measurement platforms interoperable.

b) **Measuring IPv6 Performance**

> RQ – 2 : *Do users experience benefit (or an added penalty) when connecting to popular dual-stacked websites over IPv6?*

In Chapter 8 we measured TCP connect times to 100 dual-stacked websites from SamKnows probes. The distributions of TCP connect times over a year long dataset (2013-2014) revealed that IPv6 connectivity to popular CDN deployments have improved over time. We revisited this question in Chapter 9 and showed that as of Jan 2016, 5% of these websites are faster over IPv6 with 90% being atmost 1 ms slower. We showed that `www.bing.com` stopped providing services over IPv6 since Sep 2013 and Google now employs blacklists to block hosts behind resolvers from receiving their services over IPv6 in situations where latency over IPv6 is considerable worse than IPv4.

RQ – 3 : *How do websites centralize over CDN infrastructure for IPv4 and IPv6 content delivery? Is there disparity in the availability of CDN caches over IPv4 and IPv6?*

In Chapter 8, we showed that popular websites centralise around CDN deployments and consequently show similar performance, although these CDN clusters are different for IPv4 and IPv6. We showed that some of the popular websites are even served from CDN caches deployed directly within access networks, although we witnessed cases where these CDN caches were present for IPv4, but were largely absent for IPv6. This lead to relatively higher TCP connection establishment times over IPv6.

RQ – 4 : *What are the percentage of cases where HE makes a bad decision of choosing IPv6 when it's slower. Furthermore, in such situations what is the amount of imposition (in terms of latency impact) a dual-stacked user has to pay as a result of the high HE timer value.*

In Chapter 9 we measured the effects of the HE algorithm. We showed that only around 1% of the TCP connect times over IPv6 (2013 - 2016) were ever above the HE timer value (300 ms), which leaves around 2% chance for IPv4 to win a HE race towards these websites. As such, IPv6 connections to 99% of these websites were preferred more than 98% of the time. We showed that HE with a 300 ms timer value preferred slower IPv6 connections in around 90% of the cases, although the TCP connect times are not that far apart from IPv4.

RQ – 5 : *What is the right HE timer value that provides the same preference levels over IPv6 as is today but also reduces the performance penalty in situations where IPv6 is considerably slower.*

In Chapter 9, we showed that that a HE timer value of 150 ms provides a margin benefit of 10% while retaining similar IPv6 preference levels for 99% of the dual-stacked websites.

RQ – 6 : *Do users experience benefit (or an added penalty) when streaming YouTube videos over IPv6? How do failure rates compare over IPv4 and IPv6? What factors contribute towards the performance difference? Is there disparity in the availability of GGC over IPv4 and IPv6?*

In Chapter 10, we measured YouTube performance over IPv6. Using a 21-months long dataset we showed that success rates of streaming a stall-free version of the video over IPv6 were lower compared to that of IPv4 but they tend to have improved over time. In situations where the test succeeds over both address families, we witnessed that HE strongly prefers (more than 97%) connections made over IPv6 for streaming media content. This preference to IPv6 brings worse performance in comparison with IPv4, since we observed consistently higher TCP connect times and startup delays (100 ms or more) over IPv6. Furthermore, throughput achieved was also consistently lower over IPv6 for both audio and video streams. Although we witnessed low stall rates over both address families and reduced stall durations over the years, in situations where a stall occurred, the stall durations were relatively higher (1s or more) over IPv6.

RQ – 7 : *How similar are the webpages accessed over IPv6 to their IPv4 counterparts? Is it that most of the content providers provide a AAAA entry but only serve a landing page when a request is made over IPv6, or is the content delivery over both routes the same for all the services?*

In Chapter 11 we measured similarity of dual-stacked webpages. We witnessed that 14% of the ALEXA top 100 dual-stacked websites exhibit dissimilarity in the *number* of fetched webpage elements with 6% showing more than 50% difference. 94% of dual-stacked websites exhibit dissimilarity in *size* with 8% showing atleast 50% difference. We further observed that 27% of dual-stacked websites have some fraction of webpage elements that fail over IPv6 with 9% of the websites having more than 50% webpage elements that fail over IPv6. Worse, 6% announce AAAA entries in the DNS but no content is delivered over IPv6 when an HTTP request is made.

> RQ − 8 : *In situations where the content is dissimilar over IPv4 and IPv6, what factors contribute to the dissimilarity?*

In Chapter 11, we show that failure rates are largely affected by DNS resolution errors on images, javascript and CSS content delivered from both same-origin and cross-origin sources.

c) **Measuring Access Network Performance**

> RQ − 9 : *Should last-mile latency measurements include latencies within the home network? How to account for queuing delay caused by bufferbloat on home routers when measuring last-mile latencies?*

In Chapter 13 we measured last-mile latency using residential 696 RIPE Atlas and 1245 SamKnows probes. We witnessed 19.2% (133/696) of RIPE Atlas probes and 29.7% (370/1245) of SamKnows probes exhibit hop1 latency contributing to 10% or more of hop2 latency. We conclude that home network latency can make a discernible contribution and therefore should not be accounted when measuring last-mile latency. We also witnessed 9.95% (124/1245) of SamKnows probes show hop1/hop2 contribution of more than 100% where hop1 latencies for these probes appear considerably stable at around 50ms. These probes are behind home routers that rate limit ICMP responses to TTL expiry.

> RQ − 10 : *What characteristic value of last-mile latency can be used by simulation studies to model DSL, cable and fibre access links?*

In Chapter 13, we witnessed that last-mile latency for DSL deployments is centered around 16 ms. Cable networks show a last-mile latency centered around 8 ms and fibre to the home networks show a last-mile latency centered around 4 ms.

> RQ − 11 : *Do service providers employ multiple interleaving depth levels? Do these depth levels vary over time?*

In Chapter 13 we showed that DSL service providers enable interleaving and some providers dynamically adapt interleaving depth levels depending on the line characteristics and geographic location of the subscriber. For some measurement points, we observed depth level changes occuring on a weekly time scale.

> RQ − 12 : *Do last-mile latencies vary by time of day? Do they vary by subscriber location? Do they vary by broadband product subscription and the access technology used by the DSL modem?*

In Chapter 13 we showed that once the effects of queuing delay caused by bufferbloat have been eliminated, access networks tend to exhibit robust last-mile latency. We witnessed that last-mile latency is considerably stable over time and not affected by diurnal load patterns. We showed that last-mile latencies of a service provider can depend on the geographic location of a subscriber. We observed significant last-mile latency differences for US cable service providers across the east (centered at around 32ms) and west (centered at around 8ms) coast. We showed that last-mile latencies of DSL deployments vary with the the broadband product subscription. While the last-mile latencies for products based on ADSL2+ and VDSL can be significantly lower compared to the latency of ADSL1 products, we also observed an increase in latency variation across our measurement points for ADSL2+ and VDSL products.

15.2 FUTURE DIRECTIONS

1. In Chapter 8 we measured TCP connect times to popular dual-stacked websites. However, the performance of the TCP connection, after the connection establishment, when data is exchanged between the client and the server has not yet been measured. It would be nice to know how does the raw throughput performance of a TCP connection compare over IPv6 to that of IPv4. This requires measuring the BTC (since we observe TCP) rather than the end-to-end capacity or available bandwidth of the path. BTC measurement tools require access at both ends of the measured path. As such, a study that compares BTC over IPv4 and IPv6 towards operational dual-stacked websites, will require collaborative support from large CDN providers.

2. In Chapter 9 we showed how TCP connect times over IPv6 to popular dual-stacked websites have considerably improved over time. However, it is unclear whether this is due to IPv6 content moving closer to the client (similar to how it is in IPv4). Moreover, in situations where there is considerable disparity in TCP connect times to the same website, it remains unclear whether this is due to dissimilarity of paths traversed over IPv4 and IPv6. It would be nice to measure the similarity of paths traversed for each website. This requires running `traceroute` to capture the forwarding path information over IPv4 and IPv6. Edmond W. W. Chan *et al.* in [336] use Jaccard distance to measure the similarity of IP-level and AS-level routes over IPv4 over time. Levenschtein distance [337], an extension to this metric takes the reordering of IP (or AS) elements in the path into account as well.

3. In Chapter 9 we showed how lowering the HE timer value to 150 ms (from 300 ms) provides a margin benefit of 10% while retaining similar IPv6 preference levels for 99% of the dual-stacked websites. Another approach is to make clients adaptively change the HE timer value based on the previously witnessed history of TCP connection establishment times over both IPv4 and IPv6 routes. Clients can apply a weighted combination of past witnessed history with the 150 ms timer value [40], where it can begin with a 150 ms advantage, but gradually increase the weighting towards the past witnessed history if the variation in the TCP connection establishment times are high over one address family.

4. In Chapter 13 we showed that last mile latencies of a service provider can depend on the geographic location of a subscriber. We observed

significant last-mile latency differences for US cable service providers across the east (centered at around 32ms) and west (centered at around 8ms) coast. However, the causes of this observed effect remain unclear. Analyzing this further requires collaboration with network service providers to understand the underlying last-mile infrastructure differences by subscriber location of each service provider.

BIBLIOGRAPHY

[1] B. Donnet, "Internet Topology Discovery," in *Data Traffic Monitoring and Analysis*, ser. Lecture Notes in Computer Science, E. Biersack, C. Callegari, and M. Matijasevic, Eds. Springer Berlin Heidelberg, 2013, vol. 7754, pp. 44–81. [Online]. Available: http://dx.doi.org/10.1007/978-3-642-36784-7_3

[2] H. Haddadi, M. Rio, G. Iannaccone, A. Moore, and R. Mortier, "Network Topologies: Inference, Modeling, and Generation," *Commun. Surveys Tuts.*, vol. 10, no. 2, pp. 48–69, Apr. 2008. [Online]. Available: http://dx.doi.org/10.1109/COMST.2008.4564479

[3] B. Donnet and T. Friedman, "Internet Topology Discovery: A Survey," *Commun. Surveys Tuts.*, vol. 9, no. 4, pp. 56–69, Oct. 2007. [Online]. Available: http://dx.doi.org/10.1109/COMST.2007.4444750

[4] V. Bajpai and J. Schönwälder, "A Survey on Internet Performance Measurement Platforms and Related Standardization Efforts," ser. COMST '15, vol. 17, no. 3, 2015, pp. 1313–1341. [Online]. Available: http://dx.doi.org/10.1109/COMST.2015.2418435

[5] L. Colitti, S. H. Gunderson, E. Kline, and T. Refice, "Evaluating IPv6 Adoption in the Internet," ser. PAM '10, 2010, pp. 141–150. [Online]. Available: http://dx.doi.org/10.1007/978-3-642-12334-4_15

[6] A. Dhamdhere, M. Luckie, B. Huffaker, k. claffy, A. Elmokashfi, and E. Aben, "Measuring the Deployment of IPv6: Topology, Routing and Performance," in *Proceedings of the 2012 ACM Conference on Internet Measurement Conference*, ser. IMC '12. New York, NY, USA: ACM, 2012, pp. 537–550. [Online]. Available: http://doi.acm.org/10.1145/2398776.2398832

[7] J. Czyz, M. Allman, J. Zhang, S. Iekel-Johnson, E. Osterweil, and M. Bailey, "Measuring IPv6 adoption," ser. ACM SIGCOMM '14, pp. 87–98. [Online]. Available: http://doi.acm.org/10.1145/2619239.2626295

[8] M. Nikkhah, R. Guérin, Y. Lee, and R. Woundy, "Assessing IPv6 Through Web Access a Measurement Study and Its Findings," in *Proceedings of the Seventh COnference on Emerging Networking EXperiments and Technologies*, ser. CoNEXT '11. New York, NY, USA: ACM, 2011, pp. 26:1–26:12. [Online]. Available: http://doi.acm.org/10.1145/2079296.2079322

[9] V. Bajpai and J. Schönwälder, "IPv4 versus IPv6 - who connects faster?" in *Proceedings of the 14th IFIP Networking Conference, Networking 2015, Toulouse, France, 20-22 May, 2015*, 2015, pp. 1–9. [Online]. Available: http://dx.doi.org/10.1109/IFIPNetworking.2015.7145323

[10] S. Ahsan, V. Bajpai, J. Ott, and J. Schönwälder, "Measuring YouTube from Dual-Stacked Hosts," in *Passive and Active Measurement - 16th International Conference, PAM 2015, New York, NY, USA, March 19-20, 2015, Proceedings*, 2015, pp. 249–261. [Online]. Available: http://dx.doi.org/10.1007/978-3-319-15509-8_19

[11] V. Bajpai and J. Schönwälder, "Measuring the Effects of Happy Eyeballs," in *Proceedings of the ACM, IRTF & ISOC Applied Networking Research Workshop, Berlin, Germany, 16 July*, 2016. [Online]. Available: http://dx.doi.org/10.1145/2959424.2959429

[12] S. Sundaresan, W. de Donato, N. Feamster, R. Teixeira, S. Crawford, and A. Pescapè, "Broadband Internet Performance: A View from the Gateway," in *Proceedings of the ACM SIGCOMM 2011 Conference*, ser. SIGCOMM '11. New York, NY, USA: ACM, 2011, pp. 134–145. [Online]. Available: http://doi.acm.org/10.1145/2018436.2018452

[13] M. Dischinger, A. Haeberlen, K. P. Gummadi, and S. Saroiu, "Characterizing Residential Broadband Networks," in *Proceedings of the 7th ACM SIGCOMM Conference on Internet Measurement*, ser. IMC '07. New York, NY, USA: ACM, 2007, pp. 43–56. [Online]. Available: http://doi.acm.org/10.1145/1298306.1298313

[14] kc claffy, "The 7th Workshop on Active Internet Measurements (AIMS7) Report," *Computer Communication Review*, vol. 46, no. 1, pp. 50–57, 2016. [Online]. Available: http://doi.acm.org/10.1145/2875951.2875960

[15] Y. Shavitt and E. Shir, "DIMES: let the internet measure itself," *SIGCOMM Comput. Commun. Rev.*, vol. 35, no. 5, pp. 71–74, Oct. 2005. [Online]. Available: http://doi.acm.org/10.1145/1096536.1096546

[16] H. V. Madhyastha, T. Isdal, M. Piatek, C. Dixon, T. Anderson, A. Krishnamurthy, and A. Venkataramani, "iPlane: An Information Plane for Distributed Services," in *Proceedings of the 7th Symposium on Operating Systems Design and Implementation*, ser. OSDI '06. Berkeley, CA, USA: USENIX Association, 2006, pp. 367–380. [Online]. Available: http://dl.acm.org/citation.cfm?id=1298455.1298490

[17] RIPE NCC Staff, "RIPE Atlas: A Global Internet Measurement Network," *Internet Protocol Journal*, Sep. 2015. [Online]. Available: http://ipj.dreamhosters.com/wp-content/uploads/2015/10/ipj18.3.pdf

[18] S. Sundaresan, S. Burnett, N. Feamster, and W. De Donato, "BISmark: A Testbed for Deploying Measurements and Applications in Broadband Access Networks," in *Proceedings of the 2014 USENIX Conference on USENIX Annual Technical Conference*, ser. USENIX ATC'14. Berkeley, CA, USA: USENIX Association, 2014, pp. 383–394. [Online]. Available: http://dl.acm.org/citation.cfm?id=2643634.2643673

[19] M. Linsner, P. Eardley, T. Burbridge, and F. Sorensen, "Large-Scale Broadband Measurement Use Cases," RFC 7536 (Informational), Internet Engineering Task Force, May 2015. [Online]. Available: http://www.ietf.org/rfc/rfc7536.txt

[20] "Measuring Broadband America - Federal Communications Commission," https://www.fcc.gov/general/measuring-broadband-america, [Online; accessed 18-January-2016].

[21] "Fixed Broadband Map - Ofcom," http://maps.ofcom.org.uk/broadband, [Online; accessed 25-February-2016].

[22] S. Sundaresan, W. de Donato, N. Feamster, R. Teixeira, S. Crawford, and A. Pescapè, "Broadband internet performance: a view from the gateway," in *Proceedings of the ACM SIGCOMM 2011 conference*, ser. SIGCOMM

'11. New York, NY, USA: ACM, 2011, pp. 134–145. [Online]. Available: http://doi.acm.org/10.1145/2018436.2018452

[23] J. Gettys and K. Nichols, "Bufferbloat: Dark Buffers in the Internet," *Commun. ACM*, vol. 55, no. 1, pp. 57–65, Jan. 2012. [Online]. Available: http://doi.acm.org/10.1145/2063176.2063196

[24] I. Canadi, P. Barford, and J. Sommers, "Revisiting broadband performance," in *Proceedings of the 2012 ACM conference on Internet measurement conference*, ser. IMC '12. New York, NY, USA: ACM, 2012, pp. 273–286. [Online]. Available: http://doi.acm.org/10.1145/2398776.2398805

[25] The Internet Society, "World IPv6 Launch," http://www.worldipv6launch.org, [Online; accessed 11-January-2016].

[26] P. Richter, M. Allman, R. Bush, and V. Paxson, "A Primer on IPv4 Scarcity," *Computer Communication Review*, vol. 45, no. 2, pp. 21–31, 2015. [Online]. Available: http://doi.acm.org/10.1145/2766330.2766335

[27] "Google IPv6 Adoption Statistics," http://www.google.com/intl/en/ipv6/statistics.html, [Online; accessed 11-January-2016].

[28] S. Sundaresan, N. Feamster, R. Teixeira, and N. Magharei, "Measuring and Mitigating Web Performance Bottlenecks in Broadband Access Networks," in *Proceedings of the 2013 Conference on Internet Measurement Conference*, ser. IMC '13. New York, NY, USA: ACM, 2013, pp. 213–226. [Online]. Available: http://doi.acm.org/10.1145/2504730.2504741

[29] F. Chen, R. K. Sitaraman, and M. Torres, "End-User Mapping: Next Generation Request Routing for Content Delivery," in *ACM Conference on Special Interest Group on Data Communication*, ser. SIGCOMM '15. New York, NY, USA: ACM, 2015, pp. 167–181. [Online]. Available: http://doi.acm.org/10.1145/2785956.2787500

[30] M. Calder, X. Fan, Z. Hu, E. Katz-Bassett, J. Heidemann, and R. Govindan, "Mapping the Expansion of Google's Serving Infrastructure," in *Proceedings of the 2013 Conference on Internet Measurement Conference*, ser. IMC '13. New York, NY, USA: ACM, 2013, pp. 313–326. [Online]. Available: http://doi.acm.org/10.1145/2504730.2504754

[31] D. Cicalese, J. Auge, D. Joumblatt, T. Friedman, and D. Rossi, "Characterizing IPv4 Anycast Adoption and Deployment," in *Proceedings of the 11th ACM International on Conference on Emerging Networking Experiments and Technologies*, ser. CoNEXT '15, 2015.

[32] X. Fan, E. Katz-Bassett, and J. Heidemann, "Assessing Affinity Between Users and CDN Sites," in *Traffic Monitoring and Analysis*. Springer International Publishing, 2015, vol. 9053, pp. 95–110. [Online]. Available: http://dx.doi.org/10.1007/978-3-319-17172-2_7

[33] M. Belshe, R. Peon, and M. Thomson, "Hypertext Transfer Protocol Version 2 (HTTP/2)," RFC 7540 (Proposed Standard), Internet Engineering Task Force, May 2015. [Online]. Available: http://www.ietf.org/rfc/rfc7540.txt

[34] R. Hamilton, J. Iyengar, I. Swett, and A. Wilk, "QUIC: A UDP-Based Secure and Reliable Transport for HTTP/2," Internet Engineering Task Force, Internet-Draft draft-tsvwg-quic-protocol-02, Jan. 2016, work in Progress. [Online]. Available: http://tools.ietf.org/html/draft-tsvwg-quic-protocol-02

[35] E. Rescorla, "The Transport Layer Security (TLS) Protocol Version 1.3," Internet Engineering Task Force, Internet-Draft draft-ietf-tls-tls13-11, Dec. 2015, work in Progress. [Online]. Available: http://tools.ietf.org/html/draft-ietf-tls-tls13-11

[36] k. claffy, Y. Hyun, K. Keys, M. Fomenkov, and D. Krioukov, "Internet Mapping: from Art to Science," in *IEEE DHS Cybersecurity Applications and Technologies Conference for Homeland Security (CATCH)*, Watham, MA, Mar 2009, pp. 205–211.

[37] F. Michaut and F. Lepage, "Application-oriented Network Metrology: Metrics and Active Measurement Tools," *Commun. Surveys Tuts.*, vol. 7, no. 2, pp. 2–24, Apr. 2005. [Online]. Available: http://dx.doi.org/10.1109/COMST.2005.1610543

[38] R. Prasad, C. Dovrolis, M. Murray, and K. Claffy, "Bandwidth Estimation: Metrics, Measurement Techniques, and Tools," *Netwrk. Mag. of Global Internetwkg.*, vol. 17, no. 6, pp. 27–35, Nov. 2003. [Online]. Available: http://dx.doi.org/10.1109/MNET.2003.1248658

[39] D. Thaler, R. Draves, A. Matsumoto, and T. Chown, "Default Address Selection for Internet Protocol Version 6 (IPv6)," RFC 6724, Sep. 2012. [Online]. Available: http://www.ietf.org/rfc/rfc6724.txt

[40] D. Wing and A. Yourtchenko, "Happy Eyeballs: Success with Dual-Stack Hosts," RFC 6555 (Proposed Standard), Internet Engineering Task Force, Apr. 2012. [Online]. Available: http://www.ietf.org/rfc/rfc6555.txt

[41] C. Huitema, "Teredo: Tunneling IPv6 over UDP through Network Address Translations (NATs)," RFC 4380 (Proposed Standard), Internet Engineering Task Force, Feb. 2006, updated by RFCs 5991, 6081. [Online]. Available: http://www.ietf.org/rfc/rfc4380.txt

[42] B. Carpenter and K. Moore, "Connection of IPv6 Domains via IPv4 Clouds," RFC 3056 (Proposed Standard), Internet Engineering Task Force, Feb. 2001. [Online]. Available: http://www.ietf.org/rfc/rfc3056.txt

[43] Mozilla Firefox 15 - Release Notes, "Fixed Bug 749209 - Happy Eyeballs implementation still not quite right," http://www.mozilla.org/en-US/firefox/15.0/releasenotes/buglist.html, [Online; accessed 25-January-2016].

[44] Opera 12.10 - Changelog, "IPv6 support enhanced: RFC-3484 and RFC-6555 ("Happy Eyeballs") implemented," http://www.opera.com/docs/changelogs/unified/1210, [Online; accessed 25-January-2016].

[45] Google Chrome - Revision 85934, "Add a fallback socket connect() for IPv6." http://src.chromium.org/viewvc/chrome?view=rev&revision=85934, [Online; accessed 25-January-2016].

[46] "Apple and IPv6 - Happy Eyeballs," https://www.ietf.org/mail-archive/web/v6ops/current/msg22455.html, [Online; accessed 25-January-2016].

[47] N. Sarrar, G. Maier, B. Ager, R. Sommer, and S. Uhlig, "Investigating IPv6 Traffic," in *Passive and Active Measurement*, ser. Lecture Notes in Computer Science, N. Taft and F. Ricciato, Eds. Springer Berlin Heidelberg, 2012, vol. 7192, pp. 11–20. [Online]. Available: http://dx.doi.org/10.1007/978-3-642-28537-0_2

[48] M. Butkiewicz, H. V. Madhyastha, and V. Sekar, "Characterizing Web Page Complexity and Its Impact," *IEEE/ACM Trans. Netw.*, vol. 22, no. 3, pp. 943–956, Jun. 2014. [Online]. Available: http://dx.doi.org/10.1109/TNET.2013.2269999

[49] S. J. Eravuchira, "Measuring Webpage Similarity from Dual-stacked Hosts," Masters Thesis, Jacobs University Bremen, Aug 2015.

[50] B. Chun, D. Culler, T. Roscoe, A. Bavier, L. Peterson, M. Wawrzoniak, and M. Bowman, "PlanetLab: An Overlay Testbed for Broad-coverage Services," *SIGCOMM Comput. Commun. Rev.*, vol. 33, no. 3, Jul. 2003. [Online]. Available: http://doi.acm.org/10.1145/956993.956995

[51] M. E. Fiuczynski, "PlanetLab: Overview, History, and Future Directions," *SIGOPS Oper. Syst. Rev.*, vol. 40, no. 1, pp. 6–10, Jan. 2006. [Online]. Available: http://doi.acm.org/10.1145/1113361.1113366

[52] C. Dovrolis, K. Gummadi, A. Kuzmanovic, and S. D. Meinrath, "Measurement Lab: Overview and an Invitation to the Research Community," *SIGCOMM Comput. Commun. Rev.*, vol. 40, no. 3, pp. 53–56, Jun. 2010. [Online]. Available: http://doi.acm.org/10.1145/1823844.1823853

[53] M. Murray and k. claffy, "Measuring the Immeasurable: Global Internet Measurement Infrastructure," in *Passive and Active Network Measurement Workshop (PAM)*. Amsterdam, Netherlands: RIPE NCC, Apr 2001, pp. 159–167.

[54] "Internet Measurement Infrastructure - CAIDA," http://www.caida.org/research/performance/measinfra, [Online; accessed 26-February-2016].

[55] S. Sundaresan, W. de Donato, N. Feamster, R. Teixeira, S. Crawford, and A. Pescapè, "Measuring Home Broadband Performance," *Commun. ACM*, vol. 55, no. 11, pp. 100–109, Nov. 2012. [Online]. Available: http://doi.acm.org/10.1145/2366316.2366337

[56] S. McKee, A. Lake, P. Laurens, H. Severini, T. Wlodek, S. Wolff, and J. Zurawski, "Monitoring the US ATLAS Network Infrastructure with perfSONAR-PS," *Journal of Physics: Conference Series*, vol. 396, no. 4, p. 042038, 2012. [Online]. Available: http://stacks.iop.org/1742-6596/396/i=4/a=042038

[57] M. Rimondini, C. Squarcella, and G. Battista, "Towards an Automated Investigation of the Impact of BGP Routing Changes on Network Delay Variations," in *Passive and Active Measurement*, ser. Lecture Notes in Computer Science, M. Faloutsos and A. Kuzmanovic, Eds. Springer International Publishing, 2014, vol. 8362, pp. 193–203. [Online]. Available: http://dx.doi.org/10.1007/978-3-319-04918-2_19

[58] M. A. Sánchez, J. S. Otto, Z. S. Bischof, D. R. Choffnes, F. E. Bustamante, B. Krishnamurthy, and W. Willinger, "Dasu: Pushing Experiments to the Internet's Edge," in *Proceedings of the 10th USENIX Conference on Networked Systems Design and Implementation*, ser. nsdi'13. Berkeley, CA, USA: USENIX Association, 2013, pp. 487–500. [Online]. Available: http://dl.acm.org/citation.cfm?id=2482626.2482672

[59] S. Shalunov, B. Teitelbaum, A. Karp, J. Boote, and M. Zekauskas, "A One-way Active Measurement Protocol (OWAMP)," RFC 4656 (Proposed Standard), Internet Engineering Task Force, Sep. 2006. [Online]. Available: http://www.ietf.org/rfc/rfc4656.txt

[60] A. Gupta, M. Calder, N. Feamster, M. Chetty, E. Calandro, and E. Katz-Bassett, "Peering at the Internet's Frontier: A First Look at ISP Interconnectivity in Africa," in *Passive and Active Measurement*, ser. Lecture Notes in Computer Science, M. Faloutsos and A. Kuzmanovic, Eds. Springer International Publishing, 2014, vol. 8362, pp. 204–213. [Online]. Available: http://dx.doi.org/10.1007/978-3-319-04918-2_20

[61] W. Matthews and L. Cottrell, "The PingER Project: Active Internet Performance Monitoring for the HENP Community," *Comm. Mag.*, vol. 38, no. 5, pp. 130–136, May 2000. [Online]. Available: http://dx.doi.org/10.1109/35.841837

[62] A. Faggiani, E. Gregori, L. Lenzini, S. Mainardi, and A. Vecchio, "On the feasibility of measuring the Internet through smartphone-based crowdsourcing," in *10th International Symposium on Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks (WiOpt), Paderborn, Germany, May 14-18, 2012*, 2012, pp. 318–323. [Online]. Available: http://ieeexplore.ieee.org/xpl/freeabs_all.jsp?arnumber=6260474

[63] X. Zhou and P. Van Mieghem, "Hopcount and e2e Delay: IPv6 Versus IPv4," in *Proceedings of the 6th International Conference on Passive and Active Network Measurement*, ser. PAM'05. Berlin, Heidelberg: Springer-Verlag, 2005, pp. 345–348. [Online]. Available: http://dx.doi.org/10.1007/978-3-540-31966-5_31

[64] B. Augustin, X. Cuvellier, B. Orgogozo, F. Viger, T. Friedman, M. Latapy, C. Magnien, and R. Teixeira, "Avoiding Traceroute Anomalies with Paris Traceroute," in *Proceedings of the 6th ACM SIGCOMM Conference on Internet Measurement*, ser. IMC '06. New York, NY, USA: ACM, 2006, pp. 153–158. [Online]. Available: http://doi.acm.org/10.1145/1177080.1177100

[65] C. Pelsser, L. Cittadini, S. Vissicchio, and R. Bush, "From Paris to Tokyo: On the Suitability of Ping to Measure Latency," in *Proceedings of the 2013 Conference on Internet Measurement Conference*, ser. IMC '13. New York, NY, USA: ACM, 2013, pp. 427–432. [Online]. Available: http://doi.acm.org/10.1145/2504730.2504765

[66] B. Augustin, T. Friedman, and R. Teixeira, "Measuring Load-balanced Paths in the Internet," in *Proceedings of the 7th ACM SIGCOMM Conference on Internet Measurement*, ser. IMC '07. New York, NY, USA: ACM, 2007, pp. 149–160. [Online]. Available: http://doi.acm.org/10.1145/1298306.1298329

[67] M. Dischinger, A. Haeberlen, K. P. Gummadi, and S. Saroiu, "Characterizing residential broadband networks," in *Proceedings of the 7th ACM SIGCOMM conference on Internet measurement*, ser. IMC '07. New York, NY, USA: ACM, 2007, pp. 43–56. [Online]. Available: http://doi.acm.org/10.1145/1298306.1298313

[68] A. Schulman and N. Spring, "Pingin' in the Rain," in *Proceedings of the 2011 ACM SIGCOMM Conference on Internet Measurement Conference*, ser. IMC '11. New York, NY, USA: ACM, 2011, pp. 19–28. [Online]. Available: http://doi.acm.org/10.1145/2068816.2068819

[69] K. Lakshminarayanan and V. N. Padmanabhan, "Some Findings on the Network Performance of Broadband Hosts," in *Proceedings of the 3rd ACM SIGCOMM Conference on Internet Measurement*, ser. IMC '03. New York, NY, USA: ACM, 2003, pp. 45–50. [Online]. Available: http://doi.acm.org/10.1145/948205.948212

[70] M. Siekkinen, D. Collange, G. Urvoy-Keller, and E. W. Biersack, "Performance Limitations of ADSL Users: A Case Study," in *Proceedings of the 8th International Conference on Passive and Active Network Measurement*, ser. PAM'07. Berlin, Heidelberg: Springer-Verlag, 2007, pp. 145–154. [Online]. Available: http://dl.acm.org/citation.cfm?id=1762888.1762908

[71] G. Maier, A. Feldmann, V. Paxson, and M. Allman, "On Dominant Characteristics of Residential Broadband Internet Traffic," in *Proceedings of the 9th ACM SIGCOMM Conference on Internet Measurement Conference*, ser. IMC '09. New York, NY, USA: ACM, 2009, pp. 90–102. [Online]. Available: http://doi.acm.org/10.1145/1644893.1644904

[72] G. Maier, F. Schneider, and A. Feldmann, "NAT Usage in Residential Broadband Networks," in *Proceedings of the 12th International Conference on Passive and Active Measurement*, ser. PAM'11. Berlin, Heidelberg: Springer-Verlag, 2011, pp. 32–41. [Online]. Available: http://dl.acm.org/citation.cfm?id=1987510.1987514

[73] M. Dischinger, M. Marcon, S. Guha, K. P. Gummadi, R. Mahajan, and S. Saroiu, "Glasnost: Enabling End Users to Detect Traffic Differentiation," in *Proceedings of the 7th USENIX Conference on Networked Systems Design and Implementation*, ser. NSDI'10. Berkeley, CA, USA: USENIX Association, 2010, pp. 27–27. [Online]. Available: http://dl.acm.org/citation.cfm?id=1855711.1855738

[74] P. Kanuparthy and C. Dovrolis, "ShaperProbe: end-to-end detection of ISP traffic shaping using active methods," in *Proceedings of the 2011 ACM SIGCOMM conference on Internet measurement conference*, ser. IMC '11. New York, NY, USA: ACM, 2011, pp. 473–482. [Online]. Available: http://doi.acm.org/10.1145/2068816.2068860

[75] C. Kreibich, N. Weaver, B. Nechaev, and V. Paxson, "Netalyzr: illuminating the edge network," in *Proceedings of the 10th ACM SIGCOMM conference on Internet measurement*, ser. IMC '10. New York, NY, USA: ACM, 2010, pp. 246–259. [Online]. Available: http://doi.acm.org/10.1145/1879141.1879173

[76] M. Dhawan, J. Samuel, R. Teixeira, C. Kreibich, M. Allman, N. Weaver, and V. Paxson, "Fathom: a browser-based network measurement platform," in *Proceedings of the 2012 ACM conference on Internet*

measurement conference, ser. IMC '12. New York, NY, USA: ACM, 2012, pp. 73–86. [Online]. Available: http://doi.acm.org/10.1145/2398776.2398786

[77] L. DiCioccio, R. Teixeira, and C. Rosenberg, "Measuring Home Networks with Homenet Profiler," in *Proceedings of the 14th International Conference on Passive and Active Measurement*, ser. PAM'13. Berlin, Heidelberg: Springer-Verlag, 2013, pp. 176–186. [Online]. Available: http://dx.doi.org/10.1007/978-3-642-36516-4_18

[78] O. Goga and R. Teixeira, "Speed Measurements of Residential Internet Access," in *Proceedings of the 13th International Conference on Passive and Active Measurement*, ser. PAM'12. Berlin, Heidelberg: Springer-Verlag, 2012, pp. 168–178. [Online]. Available: http://dx.doi.org/10.1007/978-3-642-28537-0_17

[79] M. Jain and C. Dovrolis, "Pathload: A Measurement Tool for End-to-End Available Bandwidth," in *In Proceedings of Passive and Active Measurements (PAM) Workshop*, 2002, pp. 14–25.

[80] W. Li, R. K. Mok, R. K. Chang, and W. W. Fok, "Appraising the Delay Accuracy in Browser-based Network Measurement," in *Proceedings of the 2013 Conference on Internet Measurement Conference*, ser. IMC '13. New York, NY, USA: ACM, 2013, pp. 361–368. [Online]. Available: http://doi.acm.org/10.1145/2504730.2504760

[81] "SamKnows Sample Size Methodology," https://www.samknows.com/broadband/uploads/methodology/SamKnows_Sample_Size_Whitepaper_20140307.pdf, [Online; accessed 29-February-2016].

[82] "SamKnows Source Code," https://files.samknows.com/~gpl, [Online; accessed 29-February-2016].

[83] "SamKnows Dashboard," https://reporting.samknows.com, [Online; accessed 29-February-2016].

[84] "FCC Speed Test iOS App - SamKnows," https://itunes.apple.com/us/app/fcc-speed-test/id794322383?ls=1&mt=8, [Online; accessed 29-February-2016].

[85] "FCC Speed Test Android App - SamKnows," https://play.google.com/store/apps/details?id=com.samknows.fcc, [Online; accessed 29-February-2016].

[86] S. Bauer, D. Clark, and W. Lehr, "PowerBoost," in *Proceedings of the 2Nd ACM SIGCOMM Workshop on Home Networks*, ser. HomeNets '11. New York, NY, USA: ACM, 2011, pp. 7–12. [Online]. Available: http://doi.acm.org/10.1145/2018567.2018570

[87] Z. S. Bischof, J. S. Otto, M. A. Sánchez, J. P. Rula, D. R. Choffnes, and F. E. Bustamante, "Crowdsourcing ISP Characterization to the Network Edge," in *Proceedings of the First ACM SIGCOMM Workshop on Measurements Up the Stack*, ser. W-MUST '11. New York, NY, USA: ACM, 2011, pp. 61–66. [Online]. Available: http://doi.acm.org/10.1145/2018602.2018617

[88] Z. S. Bischof, J. S. Otto, and F. E. Bustamante, "Up, Down and Around the Stack: ISP Characterization from Network Intensive Applications," in *Proceedings of the 2012 ACM SIGCOMM Workshop on Measurements Up the Stack*, ser. W-MUST '12. New York, NY, USA: ACM, 2012, pp. 13–18. [Online]. Available: http://doi.acm.org/10.1145/2342541.2342546

[89] G. Bernardi, D. Fenacci, M. K. Marina, and D. P. Pezaros, "BSense: A Flexible and Open-source Broadband Mapping Framework," in *Conference on Networking - Volume Part I*, ser. IFIP'12. Berlin, Heidelberg: Springer-Verlag, 2012, pp. 344–357. [Online]. Available: http://dx.doi.org/10.1007/978-3-642-30045-5_26

[90] D. Genin and J. Splett, "Where in the Internet is congestion?" *CoRR*, vol. abs/1307.3696, 2013. [Online]. Available: http://arxiv.org/abs/1307.3696

[91] V. Bajpai and J. Schönwälder, "Measuring the Effects of Happy Eyeballs," Internet-Draft draft-bajpai-happy-01, Jul. 2013. [Online]. Available: http://tools.ietf.org/html/draft-bajpai-happy-01

[92] S. Grover, M. S. Park, S. Sundaresan, S. Burnett, H. Kim, B. Ravi, and N. Feamster, "Peeking Behind the NAT: An Empirical Study of Home Networks," in *Proceedings of the 2013 Conference on Internet Measurement Conference*, ser. IMC '13. New York, NY, USA: ACM, 2013, pp. 377–390. [Online]. Available: http://doi.acm.org/10.1145/2504730.2504736

[93] "BISmark - Network Dashboard," http://networkdashboard.org, [Online; accessed 29-February-2016].

[94] "BISmark - Source Code," https://github.com/projectbismark, [Online; accessed 29-February-2016].

[95] S. Avallone, S. Guadagno, D. Emma, A. Pescape, and G. Ventre, "D-ITG Distributed Internet Traffic Generator," in *Proceedings of the The Quantitative Evaluation of Systems, First International Conference*, ser. QEST '04. Washington, DC, USA: IEEE Computer Society, 2004, pp. 316–317. [Online]. Available: http://dx.doi.org/10.1109/QEST.2004.14

[96] S. Sundaresan, N. Feamster, R. Teixeira, A. Tang, W. K. Edwards, R. E. Grinter, M. Chetty, and W. de Donato, "Helping Users Shop for ISPs with Internet Nutrition Labels," in *Proceedings of the 2Nd ACM SIGCOMM Workshop on Home Networks*, ser. HomeNets '11. New York, NY, USA: ACM, 2011, pp. 13–18. [Online]. Available: http://doi.acm.org/10.1145/2018567.2018571

[97] H. Kim, S. Sundaresan, M. Chetty, N. Feamster, and W. K. Edwards, "Communicating with Caps: Managing Usage Caps in Home Networks," in *Proceedings of the ACM SIGCOMM 2011 Conference*, ser. SIGCOMM '11. New York, NY, USA: ACM, 2011, pp. 470–471. [Online]. Available: http://doi.acm.org/10.1145/2018436.2018526

[98] S. Roy and N. Feamster, "Characterizing correlated latency anomalies in broadband access networks," in *Proceedings of the ACM SIGCOMM 2013 conference on SIGCOMM*, ser. SIGCOMM '13. New York, NY, USA: ACM, 2013, pp. 525–526. [Online]. Available: http://doi.acm.org/10.1145/2486001.2491734

[99] S. Sundaresan, N. Magharei, N. Feamster, R. Teixeira, and S. Crawford, "Web Performance Bottlenecks in Broadband Access Networks," in *Proceedings of the ACM SIGMETRICS/International Conference on Measurement and Modeling of Computer Systems*, ser. SIGMETRICS '13. New York, NY, USA: ACM, 2013, pp. 383–384. [Online]. Available: http://doi.acm.org/10.1145/2465529.2465745

[100] S. Sundaresan, N. Magharei, N. Feamster, and R. Teixeira, "Accelerating Last-mile Web Performance with Popularity-based Prefetching," in *Proceedings of the ACM SIGCOMM 2012 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communication*, ser. SIGCOMM '12. New York, NY, USA: ACM, 2012, pp. 303–304. [Online]. Available: http://doi.acm.org/10.1145/2342356.2342421

[101] M. Chetty, S. Sundaresan, S. Muckaden, N. Feamster, and E. Calandro, "Measuring Broadband Performance in South Africa," in *Proceedings of the 4th Annual Symposium on Computing for Development*, ser. ACM DEV-4 '13. New York, NY, USA: ACM, 2013, pp. 1:1–1:10. [Online]. Available: http://doi.acm.org/10.1145/2537052.2537053

[102] S. Sundaresan, N. Feamster, and R. Teixeira, "Measuring the Performance of User Traffic in Home Wireless Networks," in *Passive and Active Measurement*. Springer International Publishing, 2015, vol. 8995, pp. 305–317. [Online]. Available: http://dx.doi.org/10.1007/978-3-319-15509-8_23

[103] "Dasu - Source Code," http://www.aqualab.cs.northwestern.edu/running-code, [Online; accessed 29-February-2016].

[104] M. A. Sánchez, J. S. Otto, Z. S. Bischof, and F. E. Bustamante, "Dasu - ISP Characterization from the Edge: A BitTorrent Implementation," in *Proceedings of the ACM SIGCOMM 2011 Conference*, ser. SIGCOMM '11. New York, NY, USA: ACM, 2011, pp. 454–455. [Online]. Available: http://doi.acm.org/10.1145/2018436.2018517

[105] M. A. Sánchez, J. S. Otto, Z. S. Bischof, D. R. Choffnes, F. E. Bustamante, B. Krishnamurthy, and W. Willinger, "A Measurement Experimentation Platform at the Internet's Edge," *IEEE/ACM Trans. Netw.*, vol. 23, no. 6, pp. 1944–1958, 2015. [Online]. Available: http://dx.doi.org/10.1109/TNET.2014.2354348

[106] M. Sánchez, J. Otto, Z. Bischof, and F. Bustamante, "Trying Broadband Characterization at Home," in *Passive and Active Measurement*, ser. Lecture Notes in Computer Science, M. Roughan and R. Chang, Eds. Springer Berlin Heidelberg, 2013, vol. 7799, pp. 198–207. [Online]. Available: http://dx.doi.org/10.1007/978-3-642-36516-4_20

[107] Z. S. Bischof, F. E. Bustamante, and R. Stanojevic, "Need, Want, Can Afford: Broadband Markets and the Behavior of Users," in *Proceedings of the 2014 Conference on Internet Measurement Conference*, ser. IMC '14. New York, NY, USA: ACM, 2014, pp. 73–86. [Online]. Available: http://doi.acm.org/10.1145/2663716.2663753

[108] "Nettitutka," http://www.netradar.org/fi, [Online; accessed 29-February-2016].

[109] S. Sonntag, J. Manner, and L. Schulte, "Netradar - Measuring the wireless world," in *11th International Symposium and Workshops on Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks, WiOpt 2013, Tsukuba Science City, Japan, May 13-17, 2013*, 2013, pp. 29–34. [Online]. Available: http://ieeexplore.ieee.org/xpl/freeabs_all.jsp?arnumber=6576402

[110] S. Sonntag, L. Schulte, and J. Manner, "Mobile network measurements - It's not all about signal strength," in *Wireless Communications and Networking Conference (WCNC), 2013 IEEE*, April 2013, pp. 4624–4629. [Online]. Available: http://dx.doi.org/10.1109/WCNC.2013.6555324

[111] L. Wang and J. Manner, "Energy-efficient Mobile Web in a Bundle," *Comput. Netw.*, vol. 57, no. 17, pp. 3581–3600, Dec. 2013. [Online]. Available: http://dx.doi.org/10.1016/j.comnet.2013.08.006

[112] A. Faggiani, E. Gregori, L. Lenzini, V. Luconi, and A. Vecchio, "Smartphone-based crowdsourcing for network monitoring: Opportunities, challenges, and a case study," *Communications Magazine, IEEE*, vol. 52, no. 1, pp. 106–113, January 2014. [Online]. Available: http://dx.doi.org/10.1109/MCOM.2014.6710071

[113] ——, "Lessons Learned from the Design, Implementation, and Management of a Smartphone-based Crowdsourcing System," in *Proceedings of First International Workshop on Sensing and Big Data Mining*, ser. SENSEMINE'13. New York, NY, USA: ACM, 2013, pp. 2:1–2:6. [Online]. Available: http://doi.acm.org/10.1145/2536714.2536717

[114] E. Gregori, L. Lenzini, V. Luconi, and A. Vecchio, "Sensing the Internet through crowdsourcing," in *Pervasive Computing and Communications Workshops (PERCOM Workshops), 2013 IEEE International Conference on*, March 2013, pp. 248–254. [Online]. Available: http://dx.doi.org/10.1109/PerComW.2013.6529490

[115] F. Disperati, D. Grassini, E. Gregori, A. Improta, L. Lenzini, D. Pellegrino, and N. Redini, "SmartProbe: A Bottleneck Capacity Estimation Tool for Smartphones," in *IEEE International Conference on Green Computing and Communications and IEEE Internet of Things and IEEE Cyber, Physical and Social Computing*, ser. GREENCOM-ITHINGS-CPSCOM '13. Washington, DC, USA: IEEE Computer Society, 2013, pp. 1980–1985. [Online]. Available: http://dx.doi.org/10.1109/GreenCom-iThings-CPSCom.2013.371

[116] M. Botts, G. Percivall, C. Reed, and J. Davidson, "GeoSensor Networks," S. Nittel, A. Labrinidis, and A. Stefanidis, Eds. Berlin, Heidelberg: Springer-Verlag, 2008, ch. OGC® Sensor Web Enablement: Overview and High Level Architecture, pp. 175–190. [Online]. Available: http://dx.doi.org/10.1007/978-3-540-79996-2_10

[117] L.-J. Chen, T. Sun, L. Lao, G. Yang, M. Y. Sanadidi, and M. Gerla, "Estimating Link Capacity in High Speed Networks," in *Proceedings of the 5th International IFIP-TC6 Conference on Networking Technologies, Services, and Protocols; Performance of Computer and Communication Networks; Mobile and Wireless Communications Systems*, ser. NETWORKING'06. Berlin, Heidelberg: Springer-Verlag, 2006, pp. 98–109. [Online]. Available: http://dx.doi.org/10.1007/11753810_9

[118] "RIPE TTM User Survey Results," https://labs.ripe.net/Members/dfk/ripe-ttm-user-survey-results, [Online; accessed 29-February-2016].

[119] T. McGregor, S. Alcock, and D. Karrenberg, "The RIPE NCC Internet Measurement Data Repository," in *Proceedings of the 11th International Conference on Passive and Active Measurement*, ser. PAM'10. Berlin, Heidelberg: Springer-Verlag, 2010, pp. 111–120. [Online]. Available: http://dl.acm.org/citation.cfm?id=1889324.1889336

[120] "RIPE Atlas - Future Plans," https://atlas.ripe.net/about/future-plans, [Online; accessed 29-February-2016].

[121] "RIPE Atlas - Source Code," https://atlas.ripe.net/get-involved/source-code, [Online; accessed 29-February-2016].

[122] "RIPE Atlas - Anchors," https://atlas.ripe.net/about/anchors, [Online; accessed 29-February-2016].

[123] "RIPE Atlas - Frequently Asked Questions (FAQ)," https://atlas.ripe.net/about/faq, [Online; accessed 29-February-2016].

[124] "RIPE Atlas - Roadmap," http://roadmap.ripe.net/ripe-atlas, [Online; accessed 29-February-2016].

[125] "RIPE Atlas - Internet Maps," https://atlas.ripe.net/results/maps, [Online; accessed 29-February-2016].

[126] "RIPE Atlas - Graphs," https://atlas.ripe.net/results/graphs, [Online; accessed 29-February-2016].

[127] "RIPE Atlas - REST API," https://atlas.ripe.net/docs/rest, [Online; accessed 06-November-2015].

[128] "RIPE Atlas - Analysis," https://atlas.ripe.net/results/analyses, [Online; accessed 29-February-2016].

[129] M. Candela, M. Bartolomeo, G. Battista, and C. Squarcella, "Dynamic Traceroute Visualization at Multiple Abstraction Levels," in *Graph Drawing*, ser. Lecture Notes in Computer Science, S. Wismath and A. Wolff, Eds. Springer International Publishing, 2013, vol. 8242, pp. 496–507. [Online]. Available: http://dx.doi.org/10.1007/978-3-319-03841-4_43

[130] A. Lutu, M. Bagnulo, C. Pelsser, and O. Maennel, "Understanding the Reachability of IPv6 Limited Visibility Prefixes," in *Passive and Active Measurement*, ser. Lecture Notes in Computer Science, M. Faloutsos and A. Kuzmanovic, Eds. Springer International Publishing, 2014, vol. 8362, pp. 163–172. [Online]. Available: http://dx.doi.org/10.1007/978-3-319-04918-2_16

[131] A. Lutu, M. Bagnulo, and O. Maennel, "The BGP Visibility Scanner," in *Computer Communications Workshops (INFOCOM WKSHPS), 2013 IEEE Conference on*, April 2013, pp. 115–120. [Online]. Available: http://dx.doi.org/10.1109/INFCOMW.2013.6562877

[132] N. Brownlee, "On Searching for Patterns in Traceroute Responses," in *Passive and Active Measurement*, ser. Lecture Notes in Computer Science, M. Faloutsos and A. Kuzmanovic, Eds. Springer International Publishing, 2014, vol. 8362, pp. 67–76. [Online]. Available: http://dx.doi.org/10.1007/978-3-319-04918-2_7

[133] A. Faggiani, E. Gregori, A. Improta, L. Lenzini, V. Luconi, and L. Sani, "A study on traceroute potentiality in revealing the Internet AS-level topology," in *Networking Conference, 2014 IFIP*, June 2014, pp. 1–9. [Online]. Available: http://dx.doi.org/10.1109/IFIPNetworking.2014.6857118

[134] E. Gregori, A. Improta, L. Lenzini, L. Rossi, and L. Sani, "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*, ser. IMC '12. New York, NY, USA: ACM, 2012, pp. 253–264. [Online]. Available: http://doi.acm.org/10.1145/2398776.2398803

[135] C. Anderson, P. Winter, and Roya, "Global Network Interference Detection Over the RIPE Atlas Network," in *4th USENIX Workshop on Free and Open Communications on the Internet (FOCI 14)*. San Diego, CA: USENIX Association, Aug. 2014. [Online]. Available: https://www.usenix.org/conference/foci14/workshop-program/presentation/anderson

[136] M. D. Bartolomeo, V. D. Donato, M. Pizzonia, C. Squarcella, and M. Rimondini, "Mining Network Events using Traceroute Empathy," *CoRR*, vol. abs/1412.4074, 2014. [Online]. Available: http://arxiv.org/abs/1412.4074

[137] C. J. Bovy, H. T. Mertodimedjo, G. Hooghiemstra, H. Uijterwaal, and P. V. Mieghem, "Analysis of End-to-end Delay Measurements in Internet," in *Proc. 3rd Passive and Active Measurement Conference (PAM 2002)*, 2002.

[138] A. Ziviani, S. Fdida, J. Rezende, and O. Duarte, "Towards a Measurement-Based Geographic Location Service," in *Passive and Active Network Measurement*, ser. Lecture Notes in Computer Science, C. Barakat and I. Pratt, Eds. Springer Berlin Heidelberg, 2004, vol. 3015, pp. 43–52. [Online]. Available: http://dx.doi.org/10.1007/978-3-540-24668-8_5

[139] X. Zhou and P. Mieghem, "Reordering of IP Packets in Internet," in *Passive and Active Network Measurement*, ser. Lecture Notes in Computer Science, C. Barakat and I. Pratt, Eds. Springer Berlin Heidelberg, 2004, vol. 3015, pp. 237–246. [Online]. Available: http://dx.doi.org/10.1007/978-3-540-24668-8_24

[140] A. Hanemann, J. W. Boote, E. L. Boyd, J. Durand, L. Kudarimoti, R. Lapacz, D. M. Swany, S. Trocha, and J. Zurawski, "PerfSONAR: A Service Oriented Architecture for Multi-domain Network Monitoring," in *Conference on Service-Oriented Computing*, ser. ICSOC'05. Berlin, Heidelberg: Springer-Verlag, 2005, pp. 241–254. [Online]. Available: http://dx.doi.org/10.1007/11596141_19

[141] "PerfSONAR - Hardware Selection," http://psps.perfsonar.net/toolkit/hardware.html, [Online; accessed 29-February-2016].

[142] "PerfSONAR - Downloads," https://forge.geant.net/forge/display/perfsonar/Downloads, [Online; accessed 29-February-2016].

[143] "Bandwidth Test Controller (BWCTL)," http://software.internet2.edu/bwctl, [Online; accessed 29-February-2016].

[144] "PingER (Ping End-to-end Reporting)," http://www-iepm.slac.stanford.edu/pinger, [Online; accessed 29-February-2016].

[145] "One-Way Ping (OWAMP)," http://software.internet2.edu/owamp, [Online; accessed 31-December-2015].

[146] "DFN - Customer Network Management," http://www.cnm.dfn.de, [Online; accessed 29-February-2016].

[147] "UNINETT - NEMO (Netmonitor)," http://drift.uninett.no/kart/nemo, [Online; accessed 29-February-2016].

[148] "Visual PerfSONAR," http://www.perfsonar.net/visualperfSONAR.html, [Online; accessed 29-February-2016].

[149] "PerfSONARUI," http://docs.perfsonar.net/manage_extra_tools.html#perfsonarui, [Online; accessed 29-February-2016].

[150] "eduGAIN," http://edugain.org, [Online; accessed 29-February-2016].

[151] "ESnet perfSONAR Dashboard," http://ps-dashboard.es.net, [Online; accessed 29-February-2016].

[152] A. Hanemann, V. Jeliazkov, O. Kvittem, L. Marta, J. Metzger, and I. Velimirovic, "Complementary Visualization of perfSONAR Network Performance Measurements," in *Internet Surveillance and Protection*, Aug 2006, pp. 6–6. [Online]. Available: http://doi.ieeecomputersociety.org/10.1109/ICISP.2006.5

[153] J. Zurawski, M. Swany, and D. Gunter, "A scalable framework for representation and exchange of network measurements," in *Testbeds and Research Infrastructures for the Development of Networks and Communities*, 2006, pp. 9 pp.–417. [Online]. Available: http://dx.doi.org/10.1109/TRIDNT.2006.1649176

[154] J. Zurawski, J. Boote, E. Boyd, M. Glowiak, A. Hanemann, M. Swany, and S. Trocha, "Hierarchically Federated Registration and Lookup within the perfSONAR Framework," in *Integrated Network Management, 2007. IM '07. 10th IFIP/IEEE International Symposium on*, May 2007, pp. 705–708. [Online]. Available: http://dx.doi.org/10.1109/INM.2007.374832

[155] B. Tierney, J. Metzger, J. Boote, E. Boyd, A. Brown, R. Carlson, M. Zekauskas, J. Zurawski, M. Swany, and M. Grigoriev, "perfSONAR: Instantiating a Global Network Measurement Framework," Oct. 2009.

[156] P. Calyam, L. Kumarasamy, C.-G. Lee, and F. Ozguner, "Ontology-Based Semantic Priority Scheduling for Multi-domain Active Measurements," *Journal of Network and Systems Management*, pp. 1–35, 2013. [Online]. Available: http://dx.doi.org/10.1007/s10922-013-9297-x

[157] P. Calyam, L. Kumarasamy, and F. Ozguner, "Semantic scheduling of active measurements for meeting network monitoring objectives," in *Network and Service Management (CNSM)*, Oct 2010, pp. 435–438. [Online]. Available: http://dx.doi.org/10.1109/CNSM.2010.5691256

[158] P. Calyam, S. Kulkarni, A. Berryman, K. Zhu, M. Sridharan, R. Ramnath, and G. Springer, "OnTimeSecure: Secure middleware for federated Network Performance Monitoring," in *Network and Service Management (CNSM)*, Oct 2013, pp. 100–104. [Online]. Available: http://dx.doi.org/10.1109/CNSM.2013.6727815

[159] I. Monga, C. Guok, W. Johnston, and B. Tierney, "Hybrid networks: lessons learned and future challenges based on ESnet4 experience," *Communications Magazine, IEEE*, vol. 49, no. 5, pp. 114–121, May 2011. [Online]. Available: http://dx.doi.org/10.1109/MCOM.2011.5762807

[160] A. Oslebo, "Share and visualize your data using the perfSONAR NC framework," in *Network Operations and Management Symposium (NOMS), 2012 IEEE*, April 2012, pp. 838–852. [Online]. Available: http://dx.doi.org/10.1109/NOMS.2012.6211998

[161] J. Andreeva, C. B. Iglesias, S. Campana, A. D. Girolamo, I. Dzhunov, X. E. Curull, S. Gayazov, E. Magradze, M. M. Nowotka, L. Rinaldi, P. Saiz, J. Schovancova, G. A. Stewart, and M. Wright, "Automating ATLAS Computing Operations using the Site Status Board," *CoRR*, vol. abs/1301.0101, 2013. [Online]. Available: http://dx.doi.org/10.1088/1742-6596/396/3/032072

[162] J. Zurawski, S. Balasubramanian, A. Brown, E. Kissel, A. Lake, M. Swany, B. Tierney, and M. Zekauskas, "perfSONAR: On-board Diagnostics for Big Data," in *IEEE International Conference on Big Data*, Oct 2013.

[163] R. Dourado, L. Sampaio, and J. Suruagy Monteiro, "On the composition of performance metrics in multi-domain networks," *Communications Magazine, IEEE*, vol. 51, no. 11, pp. 72–77, November 2013. [Online]. Available: http://dx.doi.org/10.1109/MCOM.2013.6658655

[164] A. Morton and E. Stephan, "Spatial Composition of Metrics," RFC 6049 (Proposed Standard), Internet Engineering Task Force, Jan. 2011, updated by RFC 6248. [Online]. Available: http://www.ietf.org/rfc/rfc6049.txt

[165] P. Kanuparthy, D. Lee, W. Matthews, C. Dovrolis, and S. Zarifzadeh, "Pythia: detection, localization, and diagnosis of performance problems," *Communications Magazine, IEEE*, vol. 51, no. 11, pp. 55–62, November 2013. [Online]. Available: http://dx.doi.org/10.1109/MCOM.2013.6658653

[166] P. Kanuparthy and C. Dovrolis, "Pythia: Diagnosing Performance Problems in Wide Area Providers," in *USENIX Annual Technical Conference*. Berkeley, CA, USA: USENIX Association, 2014, pp. 371–382. [Online]. Available: http://dl.acm.org/citation.cfm?id=2643634.2643672

[167] H. Yu, F. Liu, S. Naegele-Jackson, T. Coulouarn, T. Kulkarni, J. Kleist, W. Hommel, and L. Dittmann, "GEANT perfSONAR MDM-based circuit monitoring in a multidomain environment," *Communications Magazine, IEEE*, vol. 52, no. 5, pp. 174–181, May 2014. [Online]. Available: http://dx.doi.org/10.1109/MCOM.2014.6815909

[168] P. Calyam, M. Dhanapalan, M. Sridharan, A. Krishnamurthy, and R. Ramnath, "Topology-Aware Correlated Network Anomaly Event Detection and Diagnosis," *J. Netw. Syst. Manage.*, vol. 22, no. 2, pp. 208–234, Apr. 2014. [Online]. Available: http://dx.doi.org/10.1007/s10922-013-9286-0

[169] "Trevor Burbridge - Large Scale Broadband Measurement at RIPE66," https://ripe66.ripe.net/archives/video/1259, [Online; accessed 29-February-2016].

[170] M. Bagnulo, T. Burbridge, S. Crawford, P. Eardley, J. Schoenwaelder, and B. Trammell, "Building a standard measurement platform," *Communications Magazine, IEEE*, vol. 52, no. 5, pp. 165–173, May 2014. [Online]. Available: http://dx.doi.org/10.1109/MCOM.2014.6815908

[171] M. Bagnulo, T. Burbridge, S. Crawford, P. Eardley, and J. Schönwälder, "A framework for large-scale measurements," in *2013 Future Network & Mobile Summit, Lisboa, Portugal, July 3-5, 2013*, 2013, pp. 1–10. [Online]. Available: http://ieeexplore.ieee.org/xpl/freeabs_all.jsp?arnumber=6633550

[172] M. Bagnulo, P. Eardley, T. Burbridge, B. Trammell, and R. Winter, "Standardizing large-scale measurement platforms," *SIGCOMM Comput. Commun. Rev.*, vol. 43, no. 2, pp. 58–63, Apr. 2013. [Online]. Available: http://doi.acm.org/10.1145/2479957.2479967

[173] "IETF 85 Proceedings - Combined Plenary," http://www.ietf.org/proceedings/85/combined-plenary.html, [Online; accessed 29-February-2016].

[174] "IETF 86 Proceedings - Large-Scale Measurement of Broadband Performance (lmap) (WG)," http://www.ietf.org/proceedings/86/lmap.html, [Online; accessed 29-February-2016].

[175] "IETF LMAP Charter -01," http://www.ietf.org/charter/charter-ietf-lmap-01.txt, [Online; accessed 29-February-2016].

[176] M. Boucadair and C. Jacquenet, "Large scale Measurement of Access network Performance (LMAP): Requirements and Issues from a Network Provider Perspective," Internet Engineering Task Force, Internet-Draft draft-boucadair-lmap-considerations-00, Feb. 2013, work in Progress. [Online]. Available: http://tools.ietf.org/html/draft-boucadair-lmap-considerations-00

[177] K. Nagami, S. Kamei, K. Koita, T. Jitsuzumi, and I. Mizukoshi, "Use Case from a measurement provider perspective for LMAP," Internet Engineering Task Force, Internet-Draft draft-nagami-lmap-use-case-measurement-provider-00, Jul. 2013, work in Progress. [Online]. Available: http://tools.ietf.org/html/draft-nagami-lmap-use-case-measurement-provider-00

[178] R. Huang, "Use Case for Large Scale Measurements Used in Data Collection of Network Management Systems," Internet Engineering Task Force, Internet-Draft draft-huang-lmap-data-collection-use-case-00, Jun. 2013, work in Progress. [Online]. Available: http://tools.ietf.org/html/draft-huang-lmap-data-collection-use-case-00

[179] P. Eardley, A. Morton, M. Bagnulo, T. Burbridge, P. Aitken, and A. Akhter, "A Framework for Large-Scale Measurement of Broadband Performance (LMAP)," RFC 7594 (Informational), Internet Engineering Task Force, Sep. 2015. [Online]. Available: http://www.ietf.org/rfc/rfc7594.txt

[180] A. Pras and J. Schoenwaelder, "On the Difference between Information Models and Data Models," RFC 3444 (Informational), Internet Engineering Task Force, Jan. 2003. [Online]. Available: http://www.ietf.org/rfc/rfc3444.txt

[181] T. Burbridge, P. Eardley, M. Bagnulo, and J. Schönwälder, "Information Model for Large-Scale Measurement Platforms (LMAP)," Internet Engineering Task Force, Internet-Draft draft-ietf-lmap-information-model-07, Nov. 2015, work in Progress. [Online]. Available: http://tools.ietf.org/html/draft-ietf-lmap-information-model-07

[182] R. Enns, M. Bjorklund, J. Schoenwaelder, and A. Bierman, "Network Configuration Protocol (NETCONF)," RFC 6241 (Proposed Standard), Internet Engineering Task Force, Jun. 2011. [Online]. Available: http://www.ietf.org/rfc/rfc6241.txt

[183] J. Schoenwaelder, "Considerations on using NETCONF with LMAP Measurement Agents," Internet Engineering Task Force, Internet-Draft draft-schoenw-lmap-netconf-00, Feb. 2013, work in Progress. [Online]. Available: http://tools.ietf.org/html/draft-schoenw-lmap-netconf-00

[184] V. Bajpai and R. Krejci, "Managing SamKnows probes using NETCONF," in *Network Operations and Management Symposium (NOMS), 2014 IEEE*, May 2014, pp. 1–2. [Online]. Available: http://dx.doi.org/10.1109/NOMS.2014.6838279

[185] M. Bjorklund, "YANG - A Data Modeling Language for the Network Configuration Protocol (NETCONF)," RFC 6020 (Proposed Standard), Internet Engineering Task Force, Oct. 2010. [Online]. Available: http://www.ietf.org/rfc/rfc6020.txt

[186] J. Schönwälder and V. Bajpai, "A YANG Data Model for LMAP Measurement Agents," Internet Engineering Task Force, Internet-Draft draft-ietf-lmap-yang-02, Nov. 2015, work in Progress. [Online]. Available: http://tools.ietf.org/html/draft-ietf-lmap-yang-02

[187] ——, "Using RESTCONF with LMAP Measurement Agents," Internet Engineering Task Force, Internet-Draft draft-ietf-lmap-restconf-01, Nov. 2015, work in Progress. [Online]. Available: http://tools.ietf.org/html/draft-ietf-lmap-restconf-01

[188] A. Bierman, M. Bjorklund, and K. Watsen, "RESTCONF Protocol," Internet Engineering Task Force, Internet-Draft draft-ietf-netconf-restconf-09, Dec. 2015, work in Progress. [Online]. Available: http://tools.ietf.org/html/draft-ietf-netconf-restconf-09

[189] A. Oslebo, "A YANG based Data Model for the LMAP Controller," Internet Engineering Task Force, Internet-Draft draft-oslebo-lmap-control-yang-01, Oct. 2014, work in Progress. [Online]. Available: http://tools.ietf.org/html/draft-oslebo-lmap-control-yang-01

[190] B. Claise, B. Trammell, and P. Aitken, "Specification of the IP Flow Information Export (IPFIX) Protocol for the Exchange of Flow Information," RFC 7011 (INTERNET STANDARD), Internet Engineering Task Force, Sep. 2013. [Online]. Available: http://www.ietf.org/rfc/rfc7011.txt

[191] M. Bagnulo and B. Trammell, "An LMAP application for IPFIX," Internet Engineering Task Force, Internet-Draft draft-bagnulo-lmap-ipfix-01, Feb. 2013, work in Progress. [Online]. Available: http://tools.ietf.org/html/draft-bagnulo-lmap-ipfix-01

[192] R. Alimi, R. Penno, Y. Yang, S. Kiesel, S. Previdi, W. Roome, S. Shalunov, and R. Woundy, "Application-Layer Traffic Optimization (ALTO) Protocol," RFC 7285 (Proposed Standard), Internet Engineering Task Force, Sep. 2014. [Online]. Available: http://www.ietf.org/rfc/rfc7285.txt

[193] J. Seedorf, D. Goergen, R. State, V. Gurbani, and E. Marocco, "ALTO for Querying LMAP Results," Internet Engineering Task Force, Internet-Draft draft-seedorf-lmap-alto-02, Oct. 2013, work in Progress. [Online]. Available: http://tools.ietf.org/html/draft-seedorf-lmap-alto-02

[194] D. Goergen, R. State, and V. Gurbani, "Aggregating large-scale measurements for Application Layer Traffic Optimization (ALTO) Protocol," Internet Engineering Task Force, Internet-Draft draft-goergen-lmap-fcc-00, Jul. 2013, work in Progress. [Online]. Available: http://tools.ietf.org/html/draft-goergen-lmap-fcc-00

[195] M. Bagnulo, T. Burbridge, S. Crawford, J. Schönwälder, and V. Bajpai, "Large MeAsurement Platform Protocol," Internet Engineering Task Force, Internet-Draft draft-bagnulo-lmap-http-03, Sep. 2014, work in Progress. [Online]. Available: http://tools.ietf.org/html/draft-bagnulo-lmap-http-03

[196] V. Liu, D. Lingli, D. Liu, S. Liu, and C. Li, "REST Style Large MeAsurement Platform Protocol," Internet Engineering Task Force, Internet-Draft draft-liu-lmap-rest-03, May 2015, work in Progress. [Online]. Available: http://tools.ietf.org/html/draft-liu-lmap-rest-03

[197] V. Paxson, G. Almes, J. Mahdavi, and M. Mathis, "Framework for IP Performance Metrics," RFC 2330 (Informational), Internet Engineering Task Force, May 1998. [Online]. Available: http://www.ietf.org/rfc/rfc2330.txt

[198] "CAIDA - Internet Tools Taxonomy," http://www.caida.org/tools/taxonomy, [Online; accessed 29-February-2016].

[199] J. Mahdavi and V. Paxson, "IPPM Metrics for Measuring Connectivity," RFC 2678 (Proposed Standard), Internet Engineering Task Force, Sep. 1999. [Online]. Available: http://www.ietf.org/rfc/rfc2678.txt

[200] A. Morton, L. Ciavattone, G. Ramachandran, S. Shalunov, and J. Perser, "Packet Reordering Metrics," RFC 4737 (Proposed Standard), Internet Engineering Task Force, Nov. 2006, updated by RFC 6248. [Online]. Available: http://www.ietf.org/rfc/rfc4737.txt

[201] J. Bellardo and S. Savage, "Measuring packet reordering," in *ACM SIGCOMM Workshop on Internet measurement*, ser. IMW '02. New York, NY, USA: ACM, 2002, pp. 97–105. [Online]. Available: http://doi.acm.org/10.1145/637201.637216

[202] G. Almes, S. Kalidindi, and M. Zekauskas, "A One-way Delay Metric for IPPM," RFC 2679 (Proposed Standard), Internet Engineering Task Force, Sep. 1999. [Online]. Available: http://www.ietf.org/rfc/rfc2679.txt

[203] C. Demichelis and P. Chimento, "IP Packet Delay Variation Metric for IP Performance Metrics (IPPM)," RFC 3393 (Proposed Standard), Internet Engineering Task Force, Nov. 2002. [Online]. Available: http://www.ietf.org/rfc/rfc3393.txt

[204] G. Almes, S. Kalidindi, and M. Zekauskas, "A One-way Packet Loss Metric for IPPM," RFC 2680 (Proposed Standard), Internet Engineering Task Force, Sep. 1999. [Online]. Available: http://www.ietf.org/rfc/rfc2680.txt

[205] R. Koodli and R. Ravikanth, "One-way Loss Pattern Sample Metrics," RFC 3357 (Informational), Internet Engineering Task Force, Aug. 2002. [Online]. Available: http://www.ietf.org/rfc/rfc3357.txt

[206] H. Uijterwaal, "A One-Way Packet Duplication Metric," RFC 5560 (Proposed Standard), Internet Engineering Task Force, May 2009, updated by RFC 6248. [Online]. Available: http://www.ietf.org/rfc/rfc5560.txt

[207] S. Savage, "Sting: a TCP-based network measurement tool," in *USENIX Symposium on Internet Technologies and Systems*, ser. USITS'99. Berkeley, CA, USA: USENIX Association, 1999, pp. 7–7. [Online]. Available: http://dl.acm.org/citation.cfm?id=1251480.1251487

[208] G. Almes, S. Kalidindi, and M. Zekauskas, "A Round-trip Delay Metric for IPPM," RFC 2681 (Proposed Standard), Internet Engineering Task Force, Sep. 1999. [Online]. Available: http://www.ietf.org/rfc/rfc2681.txt

[209] A. Morton, "Round-Trip Packet Loss Metrics," RFC 6673 (Proposed Standard), Internet Engineering Task Force, Aug. 2012. [Online]. Available: http://www.ietf.org/rfc/rfc6673.txt

[210] P. Chimento and J. Ishac, "Defining Network Capacity," RFC 5136 (Informational), Internet Engineering Task Force, Feb. 2008. [Online]. Available: http://www.ietf.org/rfc/rfc5136.txt

[211] M. Mathis and M. Allman, "A Framework for Defining Empirical Bulk Transfer Capacity Metrics," RFC 3148 (Informational), Internet Engineering Task Force, Jul. 2001. [Online]. Available: http://www.ietf.org/rfc/rfc3148.txt

[212] B. Constantine, G. Forget, R. Geib, and R. Schrage, "Framework for TCP Throughput Testing," RFC 6349 (Informational), Internet Engineering Task Force, Aug. 2011. [Online]. Available: http://www.ietf.org/rfc/rfc6349.txt

[213] M. Mathis and A. Morton, "Model Based Metrics for Bulk Transport Capacity," Internet Engineering Task Force, Internet-Draft draft-ietf-ippm-model-based-metrics-07, Oct. 2015, work in Progress. [Online]. Available: http://tools.ietf.org/html/draft-ietf-ippm-model-based-metrics-07

[214] K. Hedayat, R. Krzanowski, A. Morton, K. Yum, and J. Babiarz, "A Two-Way Active Measurement Protocol (TWAMP)," RFC 5357 (Proposed Standard), Internet Engineering Task Force, Oct. 2008, updated by RFCs 5618, 5938, 6038. [Online]. Available: http://www.ietf.org/rfc/rfc5357.txt

[215] S. Niccolini, S. Tartarelli, J. Quittek, T. Dietz, and M. Swany, "Information Model and XML Data Model for Traceroute Measurements," RFC 5388 (Proposed Standard), Internet Engineering Task Force, Dec. 2008. [Online]. Available: http://www.ietf.org/rfc/rfc5388.txt

[216] J. Quittek and K. White, "Definitions of Managed Objects for Remote Ping, Traceroute, and Lookup Operations," RFC 4560 (Proposed Standard), Internet Engineering Task Force, Jun. 2006. [Online]. Available: http://www.ietf.org/rfc/rfc4560.txt

[217] A. Morton, "Rate Measurement Test Protocol Problem Statement and Requirements," RFC 7497 (Informational), Internet Engineering Task Force, Apr. 2015. [Online]. Available: http://www.ietf.org/rfc/rfc7497.txt

[218] "IETF IPPM Charter -05," http://www.ietf.org/charter/charter-ietf-ippm-05.txt, [Online; accessed 29-February-2016].

[219] J. Fabini and A. Morton, "Advanced Stream and Sampling Framework for IP Performance Metrics (IPPM)," RFC 7312 (Informational), Internet Engineering Task Force, Aug. 2014. [Online]. Available: http://www.ietf.org/rfc/rfc7312.txt

[220] A. Morton and B. Claise, "Packet Delay Variation Applicability Statement," RFC 5481 (Informational), Internet Engineering Task Force, Mar. 2009. [Online]. Available: http://www.ietf.org/rfc/rfc5481.txt

[221] K. Pentikousis, E. Zhang, and Y. Cui, "IKEv2-Derived Shared Secret Key for the One-Way Active Measurement Protocol (OWAMP) and Two-Way Active Measurement Protocol (TWAMP)," RFC 7717 (Proposed Standard), Internet Engineering Task Force, Dec. 2015. [Online]. Available: http://www.ietf.org/rfc/rfc7717.txt

[222] M. Bagnulo, B. Claise, P. Eardley, A. Morton, and A. Akhter, "Registry for Performance Metrics," Internet Engineering Task Force, Internet-Draft draft-ietf-ippm-metric-registry-05, Oct. 2015, work in Progress. [Online]. Available: http://tools.ietf.org/html/draft-ietf-ippm-metric-registry-05

[223] M. Bagnulo, T. Burbridge, S. Crawford, P. Eardley, and A. Morton, "A Reference Path and Measurement Points for Large-Scale Measurement of Broadband Performance," RFC 7398 (Informational), Internet Engineering Task Force, Feb. 2015. [Online]. Available: http://www.ietf.org/rfc/rfc7398.txt

[224] H. Schulzrinne, S. Casner, R. Frederick, and V. Jacobson, "RTP: A Transport Protocol for Real-Time Applications," RFC 3550 (INTERNET STANDARD), Internet Engineering Task Force, Jul. 2003, updated by RFCs 5506, 5761, 6051, 6222. [Online]. Available: http://www.ietf.org/rfc/rfc3550.txt

[225] T. Friedman, R. Caceres, and A. Clark, "RTP Control Protocol Extended Reports (RTCP XR)," RFC 3611 (Proposed Standard), Internet Engineering Task Force, Nov. 2003. [Online]. Available: http://www.ietf.org/rfc/rfc3611.txt

[226] A. Clark and Q. Wu, "Measurement Identity and Information Reporting Using a Source Description (SDES) Item and an RTCP Extended Report (XR) Block," RFC 6776 (Proposed Standard), Internet Engineering Task Force, Oct. 2012. [Online]. Available: http://www.ietf.org/rfc/rfc6776.txt

[227] A. Clark, K. Gross, and Q. Wu, "RTP Control Protocol (RTCP) Extended Report (XR) Block for Delay Metric Reporting," RFC 6843 (Proposed Standard), Internet Engineering Task Force, Jan. 2013. [Online]. Available: http://www.ietf.org/rfc/rfc6843.txt

[228] A. Clark and Q. Wu, "RTP Control Protocol (RTCP) Extended Report (XR) Block for Packet Delay Variation Metric Reporting," RFC 6798 (Proposed Standard), Internet Engineering Task Force, Nov. 2012. [Online]. Available: http://www.ietf.org/rfc/rfc6798.txt

[229] A. Clark, S. Zhang, J. Zhao, and Q. Wu, "RTP Control Protocol (RTCP) Extended Report (XR) Block for Burst/Gap Loss Metric Reporting," RFC 6958 (Proposed Standard), Internet Engineering Task Force, May 2013. [Online]. Available: http://www.ietf.org/rfc/rfc6958.txt

[230] "TR-069 CPE WAN Management Protocol - Issue: 1 Amendment 5," http://broadband-forum.org/technical/download/TR-069_Amendment-5.pdf, [Online; accessed 29-February-2016].

[231] D. Fellows and D. Jones, "DOCSIS cable modem technology," *Communications Magazine, IEEE*, vol. 39, no. 3, pp. 202–209, Mar 2001. [Online]. Available: http://dx.doi.org/10.1109/35.910608

[232] "TR-143 Enabling Network Throughput Performance Tests and Statistical Monitoring - Issue: 1," http://www.broadband-forum.org/technical/download/TR-143.pdf, [Online; accessed 29-February-2016].

[233] "Broadband Forum - Technical Work in Progress," http://www.broadband-forum.org/technical/technicalwip.php, [Online; accessed 29-February-2016].

[234] "Workshop - Large-scale Measurements: From Standards to Implementations," http://workshop.leone-project.eu, [Online; accessed 29-February-2016].

[235] "The IEEE 802.16 Working Group on Broadband Wireless Access Standards," http://www.ieee802.org/16, [Online; accessed 29-February-2016].

[236] "IEEE 802.16's Project 802.16.3: Mobile Broadband Network Performance Measurements," http://www.ieee802.org/16/mbnpm, [Online; accessed 29-February-2016].

[237] R. Huang, Q. Wu, H. Asaeda, and G. Zorn, "RTP Control Protocol (RTCP) Extended Report (XR) Block for MPEG-2 Transport Stream (TS) Program Specific Information (PSI) Independent Decodability Statistics Metrics Reporting," RFC 6990 (Proposed Standard), Internet Engineering Task Force, Aug. 2013. [Online]. Available: http://www.ietf.org/rfc/rfc6990.txt

[238] A. Morton, "Active and Passive Metrics and Methods (and everything in-between, or Hybrid)," Internet Engineering Task Force, Internet-Draft draft-ietf-ippm-active-passive-06, Jan. 2016, work in Progress. [Online]. Available: http://tools.ietf.org/html/draft-ietf-ippm-active-passive-06

[239] "JCA-CIT Output Documents (2013-2016)," http://www.itu.int/en/ITU-T/jca/cit/Pages/output-201304.aspx, [Online; accessed 29-February-2016].

[240] A. Faggiani, E. Gregori, L. Lenzini, V. Luconi, and A. Vecchio, "Network Sensing Through Smartphone-based Crowdsourcing," in *Proceedings of the 11th ACM Conference on Embedded Networked Sensor Systems*, ser. SenSys '13. New York, NY, USA: ACM, 2013, pp. 31:1–31:2. [Online]. Available: http://doi.acm.org/10.1145/2517351.2517397

[241] B. Briscoe, A. Brunstrom, A. Petlund, D. Hayes, D. Ros, I.-J. Tsang, S. Gjessing, G. Fairhurst, C. Griwodz, and M. Welzl, "Reducing Internet Latency: A Survey of Techniques and their Merits," *Communications*

Surveys Tutorials, IEEE, vol. PP, no. 99, pp. 1–1, 2014. [Online]. Available: http://dx.doi.org/10.1109/COMST.2014.2375213

[242] B. Trammell, P. Casas, D. Rossi, A. Bär, Z. Houidi, I. Leontiadis, T. Szemethy, and M. Mellia, "mPlane: an intelligent measurement plane for the Internet," *Communications Magazine, IEEE*, vol. 52, no. 5, pp. 148–156, May 2014. [Online]. Available: http://dx.doi.org/10.1109/MCOM.2014.6815906

[243] I. Bermudez, S. Traverso, M. Munafo, and M. Mellia, "A Distributed Architecture for the Monitoring of Clouds and CDNs: Applications to Amazon AWS," *Network and Service Management, IEEE Transactions on*, vol. 11, no. 4, pp. 516–529, Dec 2014. [Online]. Available: http://dx.doi.org/10.1109/TNSM.2014.2362357

[244] S. Niccolini, F. Huici, B. Trammell, G. Bianchi, and F. Ricciato, "Building a decentralized, cooperative, and privacy-preserving monitoring system for trustworthiness: the approach of the EU FP7 DEMONS project [Very Large Projects]," *Communications Magazine, IEEE*, vol. 49, no. 11, pp. 16–18, November 2011. [Online]. Available: http://dx.doi.org/10.1109/MCOM.2011.6069700

[245] G. Bianchi, E. Boschi, D. Kaklamani, E. Koutsoloukas, G. V. Lioudakis, F. Oppedisano, M. Petraschek, F. Ricciato, and C. Schmoll, "Towards Privacy-Preserving Network Monitoring: Issues and Challenges," in *Personal, Indoor and Mobile Radio Communications*, Sept 2007, pp. 1–5. [Online]. Available: http://dx.doi.org/10.1109/PIMRC.2007.4394186

[246] S. Knight, H. Nguyen, N. Falkner, R. Bowden, and M. Roughan, "The Internet Topology Zoo," *Selected Areas in Communications, IEEE Journal on*, vol. 29, no. 9, pp. 1765–1775, October 2011. [Online]. Available: http://dx.doi.org/10.1109/JSAC.2011.111002

[247] P. Borgnat, G. Dewaele, K. Fukuda, P. Abry, and K. Cho, "Seven Years and One Day: Sketching the Evolution of Internet Traffic," in *INFOCOM 2009, IEEE*, April 2009, pp. 711–719. [Online]. Available: http://dx.doi.org/10.1109/INFCOM.2009.5061979

[248] E. Dart, L. Rotman, B. Tierney, M. Hester, and J. Zurawski, "The Science DMZ: A Network Design Pattern for Data-intensive Science," in *Proceedings of the International Conference on High Performance Computing, Networking, Storage and Analysis*, ser. SC '13. New York, NY, USA: ACM, 2013, pp. 85:1–85:10. [Online]. Available: http://doi.acm.org/10.1145/2503210.2503245

[249] H. Yan, R. Oliveira, K. Burnett, D. Matthews, L. Zhang, and D. Massey, "BGPmon: A Real-Time, Scalable, Extensible Monitoring System," in *Conference For Homeland Security, 2009. CATCH '09. Cybersecurity Applications Technology*, March 2009, pp. 212–223. [Online]. Available: http://dx.doi.org/10.1109/CATCH.2009.28

[250] A. Lodhi, N. Larson, A. Dhamdhere, C. Dovrolis, and k. claffy, "Using peeringDB to Understand the Peering Ecosystem," *SIGCOMM Comput. Commun. Rev.*, vol. 44, no. 2, pp. 20–27, Apr. 2014. [Online]. Available: http://doi.acm.org/10.1145/2602204.2602208

[251] T. McGregor, H.-W. Braun, and J. Brown, "The NLANR Network Analysis Infrastructure," *Communications Magazine, IEEE*, vol. 38, no. 5, pp. 122–128, May 2000. [Online]. Available: http://dx.doi.org/10.1109/35.841836

[252] V. Bajpai and J. Schönwälder, "Understanding the Impact of Network Infrastructure Changes Using Large-Scale Measurement Platforms," in *Emerging Management Mechanisms for the Future Internet*, Jun. 2013. [Online]. Available: http://dx.doi.org/10.1007/978-3-642-38998-6_5

[253] V. Bajpai and J. Schönwälder, "Measuring TCP connection establishment times of dual-stacked web services," in *Proceedings of the 9th International Conference on Network and Service Management, CNSM 2013, Zurich, Switzerland, October 14-18, 2013*, 2013, pp. 130–133. [Online]. Available: http://dx.doi.org/10.1109/CNSM.2013.6727822

[254] I. Csabai, A. Fekete, P. Hága, B. Hullár, G. Kurucz, S. Laki, P. Mátray, J. Stéger, G. Vattay, F. Espina, S. Garcia-Jimenez, M. Izal, E. Magaña, D. Morató, J. Aracil, F. Gómez, I. Gonzalez, S. López-Buedo, V. Moreno, and J. Ramos, "ETOMIC Advanced Network Monitoring System for Future Internet Experimentation," in *Testbeds and Research Infrastructures*. Springer Berlin Heidelberg, 2011. [Online]. Available: http://dx.doi.org/10.1007/978-3-642-17851-1_20

[255] Internet Society, "World IPv6 Day 2011," http://worldipv6day.org, [Online; accessed 25-January-2016].

[256] "World IPv6 Launch - Measurements," http://www.worldipv6launch.org/measurements, [Online; accessed 11-January-2016].

[257] "Comcast Reaches Key Milestone in Launch of IPv6 Broadband Network," http://corporate.comcast.com/comcast-voices/comcast-reaches-key-milestone-in-launch-of-ipv6-broadband-network, [Online; accessed 11-January-2016].

[258] L. Colitti, S. H. Gunderson, E. Kline, and T. Refice, "Evaluating IPv6 Adoption in the Internet," ser. PAM, 2010. [Online]. Available: http://dl.acm.org/citation.cfm?id=1889324.1889339

[259] S. Zander, L. L. Andrew, G. Armitage, G. Huston, and G. Michaelson, "Mitigating Sampling Error when Measuring Internet Client IPv6 Capabilities," in *Proceedings of the 2012 ACM Conference on Internet Measurement Conference*, ser. IMC '12. New York, NY, USA: ACM, 2012, pp. 87–100. [Online]. Available: http://doi.acm.org/10.1145/2398776.2398787

[260] ——, "Investigating the IPv6 Teredo Tunnelling Capability and Performance of Internet Clients," *SIGCOMM Computer Communication Review*, vol. 42, no. 5, pp. 13–20, Sep. 2012. [Online]. Available: http://doi.acm.org/10.1145/2378956.2378959

[261] M. Karir, G. Huston, G. Michaelson, and M. Bailey, "Understanding IPv6 Populations in the Wild," in *Passive and Active Measurement*, ser. Lecture Notes in Computer Science, M. Roughan and R. Chang, Eds. Springer Berlin Heidelberg, 2013, vol. 7799, pp. 256–259. [Online]. Available: http://dx.doi.org/10.1007/978-3-642-36516-4_27

[262] "Hurricane Electric - Global IPv6 Deployment Progress Report," http://bgp.he.net/ipv6-progress-report.cgi, [Online; accessed 11-January-2016].

[263] K. Cho, M. Luckie, and B. Huffaker, "Identifying IPv6 Network Problems in the Dual-stack World," ser. NetT, 2004. [Online]. Available: http://doi.acm.org/10.1145/1016687.1016697

[264] H. Alzoubi, M. Rabinovich, and O. Spatscheck, "Performance Implications of Unilateral Enabling of IPv6," ser. PAM, 2013, vol. 7799. [Online]. Available: http://dx.doi.org/10.1007/978-3-642-36516-4_12

[265] A. Keranen and J. Arkko, "Some Measurements on World IPv6 Day from an End-User Perspective," RFC 6948 (Informational), Internet Engineering Task Force, Jul. 2013. [Online]. Available: http://www.ietf.org/rfc/rfc6948.txt

[266] "OpenWrt," http://openwrt.org, [Online; accessed 29-February-2016].

[267] "ALEXA Top 1M websites," http://s3.amazonaws.com/alexa-static/top-1m.csv.zip, [Online; accessed 29-February-2016].

[268] A. Durand, P. Fasano, I. Guardini, and D. Lento, "IPv6 Tunnel Broker," RFC 3053 (Informational), Internet Engineering Task Force, Jan. 2001. [Online]. Available: http://www.ietf.org/rfc/rfc3053.txt

[269] "RIPE NCC - Routing Information Service (RIS)," http://www.ripe.net/ris, [Online; accessed 29-February-2016].

[270] J. Livingood, "Considerations for Transitioning Content to IPv6," RFC 6589, Apr. 2012. [Online]. Available: http://www.ietf.org/rfc/rfc6589.txt

[271] "Google no longer returning AAAA records?" https://www.mail-archive.com/ipv6-ops@lists.cluenet.de/msg01775.html, [Online; accessed 11-January-2016].

[272] "MaxMind - IP Geolocation and Online Fraud Prevention," http://www.maxmind.com, [Online; accessed 29-February-2016].

[273] "Experiences of host behavior in broken IPv6 networks," http://www.ietf.org/proceedings/80/slides/v6ops-12.pdf, [Online; accessed 25-January-2016].

[274] S. Zander, L. L. H. Andrew, G. J. Armitage, G. Huston, and G. Michaelson, "Investigating the IPv6 teredo tunnelling capability and performance of internet clients," *Computer Communication Review*, vol. 42, no. 5, pp. 13–20, 2012. [Online]. Available: http://doi.acm.org/10.1145/2378956.2378959

[275] Christopher Palmer, "Teredo Microsoft Present and Future," http://www.ietf.org/proceedings/88/slides/slides-88-v6ops-0, [Online; accessed 10-February-2016].

[276] O. Troan and B. Carpenter, "Deprecating the Anycast Prefix for 6to4 Relay Routers," RFC 7526 (Best Current Practice), Internet Engineering Task Force, May 2015. [Online]. Available: http://www.ietf.org/rfc/rfc7526.txt

[277] Geoff Huston, "Measuring IPv6 Performance," https://www.nanog.org/sites/default/files/Huston_Is_Ipv6.pdf, [Online; accessed 10-February-2016].

[278] Apple Mailing Lists, "Lion and IPv6," http://lists.apple.com/archives/ipv6-dev/2011/Jul/msg00009.html, [Online; accessed 25-January-2016].

[279] Emile Aben, "Hampering Eyeballs - Observations on Two Happy Eyeballs Implementations," https://labs.ripe.net/Members/emileaben/hampered-eyeballs, [Online; accessed 10-February-2016].

[280] Geoff Huston, "Dual Stack Esotropia," http://labs.apnic.net/blabs/?p=47, [Online; accessed 10-February-2016].

[281] ——, "Bemused Eyeballs: Tailoring Dual Stack Applications for a CGN Environment," http://www.potaroo.net/ispcol/2012-05/notquite.html, [Online; accessed 10-February-2016].

[282] F. Baker, "Testing Eyeball Happiness," RFC 6556 (Informational), Internet Engineering Task Force, Apr. 2012. [Online]. Available: http://www.ietf.org/rfc/rfc6556.txt

[283] S. Zander, L. L. H. Andrew, G. J. Armitage, G. Huston, and G. Michaelson, "Mitigating sampling error when measuring internet client IPv6 capabilities," ser. IMC '12, 2012, pp. 87–100. [Online]. Available: http://doi.acm.org/10.1145/2398776.2398787

[284] "Facebook News Feeds Load 20-40% Faster Over IPv6," http://www.internetsociety.org/deploy360/blog/2015/04/facebook-news-feeds-load-20-40-faster-over-ipv6, [Online; accessed 11-January-2016].

[285] Dan Drown, "SB6183 dropping IPv6 traffic," http://blog.dan.drown.org/sb6183-dropping-ipv6-traffic, [Online; accessed 03-February-2016].

[286] "Amsterdam Internet Exchange - IPv6 Traffic," https://goo.gl/pr9HlW, [Online; accessed 05-May-2016].

[287] "NANOG - IPv6 traffic percentages?" http://goo.gl/w1mJ8y, [Online; accessed 05-May-2016].

[288] P. Gill, M. F. Arlitt, Z. Li, and A. Mahanti, "Youtube traffic characterization: a view from the edge," in *Proceedings of the 7th ACM SIGCOMM Internet Measurement Conference, IMC 2007, San Diego, California, USA, October 24-26, 2007*, 2007, pp. 15–28. [Online]. Available: http://doi.acm.org/10.1145/1298306.1298310

[289] M. Cha, H. Kwak, P. Rodriguez, Y. Ahn, and S. B. Moon, "I tube, you tube, everybody tubes: analyzing the world's largest user generated content video system," in *Proceedings of the 7th ACM SIGCOMM Internet Measurement Conference, IMC 2007, San Diego, California, USA, October 24-26, 2007*, 2007, pp. 1–14. [Online]. Available: http://doi.acm.org/10.1145/1298306.1298309

[290] V. K. Adhikari, S. Jain, and Z.-L. Zhang, "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*, ser. IMC '10. New York, NY, USA: ACM, 2010, pp. 431–443. [Online]. Available: http://doi.acm.org/10.1145/1879141.1879197

[291] A. Finamore, M. Mellia, M. M. Munafò, R. Torres, and S. G. Rao, "YouTube everywhere: impact of device and infrastructure

synergies on user experience," in *Proceedings of the 11th ACM SIGCOMM Internet Measurement Conference, IMC '11, Berlin, Germany, November 2-, 2011*, 2011, pp. 345–360. [Online]. Available: http://doi.acm.org/10.1145/2068816.2068849

[292] P. Juluri, V. Tamarapalli, and D. Medhi, "Measurement of quality of experience of video-on-demand services: A survey," ser. IEEE Communications Surveys and Tutorials, 2016. [Online]. Available: http://dx.doi.org/10.1109/COMST.2015.2401424

[293] V. Adhikari, S. Jain, Y. Chen, and Z.-L. Zhang, "Vivisecting YouTube: An active measurement study," in *INFOCOM, 2012 Proceedings IEEE*, March 2012, pp. 2521–2525. [Online]. Available: http://dx.doi.org/10.1109/INFCOM.2012.6195644

[294] P. Juluri, L. Plissonneau, Y. Zeng, and D. Medhi, "Viewing YouTube from a metropolitan area: What do users accessing from residential ISPs experience?" in *2013 IFIP/IEEE International Symposium on Integrated Network Management (IM 2013), Ghent, Belgium, May 27-31, 2013*, 2013, pp. 589–595. [Online]. Available: http://ieeexplore.ieee.org/xpl/freeabs_all.jsp?arnumber=6573037

[295] P. Juluri, L. Plissonneau, and D. Medhi, "Pytomo: A Tool for Analyzing Playback Quality of YouTube Videos," in *Proceedings of the 23rd International Teletraffic Congress*, ser. ITC '11. International Teletraffic Congress, 2011, pp. 304–305. [Online]. Available: http://dl.acm.org/citation.cfm?id=2043468.2043517

[296] H. Nam, K.-H. Kim, and H. Schulzrinne, "QoE Matters More Than QoS: Why People Stop Watching Cat Videos," ser. INFOCOM '16, 2016.

[297] I. Livadariu, A. Elmokashfi, and A. Dhamdhere, "Characterizing IPv6 control and data plane stability," ser. INFOCOM '16, 2016.

[298] "YouTube Data API," https://goo.gl/EBCbR8, [Online; accessed 27-Apr-2016].

[299] "Leone - From global measurements to local management (Final Report)," http://goo.gl/x9WreS, [Online; accessed 27-Apr-2016].

[300] "Google - Peering and Content Delivery," https://isp.google.com/about/ggc.html, [Online; accessed 31-December-2015].

[301] S. Deering and R. Hinden, "Internet Protocol, Version 6 (IPv6) Specification," RFC 2460 (Draft Standard), Internet Engineering Task Force, Dec. 1998, updated by RFCs 5095, 5722, 5871, 6437, 6564, 6935, 6946, 7045, 7112. [Online]. Available: http://www.ietf.org/rfc/rfc2460.txt

[302] Lee Howard, "The State of IPv6 Only at RIPE 67," https://ripe67.ripe.net/archives/video/8, [Online; accessed 11-January-2016].

[303] T. A. Johnson and P. Seeling, "Desktop and mobile web page comparison: characteristics, trends, and implications," *IEEE Communications Magazine*, vol. 52, no. 9, pp. 144–151, 2014. [Online]. Available: http://dx.doi.org/10.1109/MCOM.2014.6894465

[304] F. Mahi, "RIPE Atlas Midsummer Update 2014," *RIPE Labs*, Jul. 2014. [Online]. Available: https://labs.ripe.net/Members/fatemah_mafi/ripe-atlas-midsummer-update-2014

[305] P. Vixie and D. Dagon, "Use of Bit 0x20 in DNS Labels to Improve Transaction Identity," Internet Engineering Task Force, Internet-Draft draft-vixie-dnsext-dns0x20-00, Mar. 2008, work in Progress. [Online]. Available: http://tools.ietf.org/html/draft-vixie-dnsext-dns0x20-00

[306] V. Bajpai, S. J. Eravuchira, and J. Schönwälder, "Lessons Learned From Using the RIPE Atlas Platform for Measurement Research," *SIGCOMM Comput. Commun. Rev.*, vol. 45, no. 3, pp. 35–42, Jul. 2015. [Online]. Available: http://doi.acm.org/10.1145/2805789.2805796

[307] J. P. Rula, Z. S. Bischof, and F. E. Bustamante, "Second Chance: Understanding Diversity in Broadband Access Network Performance," in *Proceedings of the 2015 ACM SIGCOMM Workshop on Crowdsourcing and Crowdsharing of Big (Internet) Data*, ser. C2B(1)D '15. New York, NY, USA: ACM, 2015, pp. 9–14. [Online]. Available: http://doi.acm.org/10.1145/2787394.2787400

[308] S. Bortzmeyer, "How Many RIPE Atlas Probes Believe They Have IPv6 (But Are Wrong)?" *RIPE Labs*, Jun. 2013. [Online]. Available: https://labs.ripe.net/Members/stephane_bortzmeyer/how-many-atlas-probes-believe-they-have-ipv6-but-are-wrong

[309] X. A. Dimitropoulos, D. V. Krioukov, G. F. Riley, and K. C. Claffy, "Revealing the Autonomous System Taxonomy: The Machine Learning Approach," *CoRR*, vol. abs/cs/0604015, 2006. [Online]. Available: http://arxiv.org/abs/cs/0604015

[310] RIPE NCC Staff, "RIPE Atlas Measurement Creation API," https://atlas.ripe.net/docs/measurement-creation-api, [Online; accessed 06-November-2015].

[311] Y. Rekhter, B. Moskowitz, D. Karrenberg, G. J. de Groot, and E. Lear, "Address Allocation for Private Internets," RFC 1918 (Best Current Practice), Internet Engineering Task Force, Feb. 1996, updated by RFC 6761. [Online]. Available: http://www.ietf.org/rfc/rfc1918.txt

[312] R. Steenbergen, "NANOG 59 Tutorial: Troubleshooting with Traceroute," https://nanog.org/meetings/abstract?id=2218, 2013, [Online; accessed 06-November-2015].

[313] Z. S. Bischof, J. S. Otto, and F. E. Bustamante, "Up, Down and Around the Stack: ISP Characterization from Network Intensive Applications," in *Proceedings of the 2012 ACM SIGCOMM Workshop on Measurements Up the Stack*, ser. W-MUST '12. New York, NY, USA: ACM, 2012, pp. 13–18. [Online]. Available: http://doi.acm.org/10.1145/2342541.2342546

[314] L. DiCioccio, R. Teixeira, M. May, and C. Kreibich, "Probe and Pray: Using UPnP for Home Network Measurements," in *Proceedings of the 13th International Conference on Passive and Active Measurement*, ser. PAM'12. Berlin, Heidelberg: Springer-Verlag, 2012, pp. 96–105. [Online]. Available: http://dx.doi.org/10.1007/978-3-642-28537-0_10

[315] C. Kreibich, N. Weaver, B. Nechaev, and V. Paxson, "Netalyzr: Illuminating the Edge Network," in *Proceedings of the 10th ACM SIGCOMM Conference on Internet Measurement*, ser. IMC '10. New York, NY, USA: ACM, 2010, pp. 246–259. [Online]. Available: http://doi.acm.org/10.1145/1879141.1879173

[316] D. G. Altman and J. M. Bland, "Statistics notes: Diagnostic tests 1: sensitivity and specificity," *BMJ*, vol. 308, no. 6943, p. 1552, 1994. [Online]. Available: http://www.bmj.com/content/308/6943/1552

[317] Y.-C. Chiu, B. Schlinker, A. B. Radhakrishnan, E. Katz-Bassett, and R. Govindan, "Are We One Hop Away from a Better Internet?" in *Proceedings of the 2015 ACM Conference on Internet Measurement Conference*, ser. IMC '15. New York, NY, USA: ACM, 2015, pp. 523–529. [Online]. Available: http://doi.acm.org/10.1145/2815675.2815719

[318] T. Holterbach, C. Pelsser, R. Bush, and L. Vanbever, "Quantifying Interference Between Measurements on the RIPE Atlas Platform," in *Proceedings of the 2015 ACM Conference on Internet Measurement Conference*, ser. IMC '15. New York, NY, USA: ACM, 2015, pp. 437–443. [Online]. Available: http://doi.acm.org/10.1145/2815675.2815710

[319] P. Marchetta, A. Botta, E. Katz-Bassett, and A. Pescapé, "Dissecting Round Trip Time on the Slow Path with a Single Packet," in *Passive and Active Measurement*. Springer International Publishing, 2014, vol. 8362, pp. 88–97. [Online]. Available: http://dx.doi.org/10.1007/978-3-319-04918-2_9

[320] I. Canadi, P. Barford, and J. Sommers, "Revisiting Broadband Performance," in *Proceedings of the 2012 ACM Conference on Internet Measurement Conference*, ser. IMC '12. New York, NY, USA: ACM, 2012, pp. 273–286. [Online]. Available: http://doi.acm.org/10.1145/2398776.2398805

[321] S. Roy and N. Feamster, "Characterizing Correlated Latency Anomalies in Broadband Access Networks," in *Proceedings of the ACM SIGCOMM 2013 Conference on SIGCOMM*, ser. SIGCOMM '13. New York, NY, USA: ACM, 2013, pp. 525–526. [Online]. Available: http://doi.acm.org/10.1145/2486001.2491734

[322] R. Bonica, D. Gan, D. Tappan, and C. Pignataro, "ICMP Extensions for Multiprotocol Label Switching," RFC 4950 (Proposed Standard), Internet Engineering Task Force, Aug. 2007. [Online]. Available: http://www.ietf.org/rfc/rfc4950.txt

[323] S. Jiang, D. Guo, and B. Carpenter, "An Incremental Carrier-Grade NAT (CGN) for IPv6 Transition," RFC 6264 (Informational), Internet Engineering Task Force, Jun. 2011. [Online]. Available: http://www.ietf.org/rfc/rfc6264.txt

[324] J. Weil, V. Kuarsingh, C. Donley, C. Liljenstolpe, and M. Azinger, "IANA-Reserved IPv4 Prefix for Shared Address Space," RFC 6598 (Best Current Practice), Internet Engineering Task Force, Apr. 2012. [Online]. Available: http://www.ietf.org/rfc/rfc6598.txt

[325] RIPE NCC Staff, "RIPE Atlas Probe API: v1," https://atlas.ripe.net/api/v1/probe, [Online; accessed 06-November-2015].

[326] ——, "RIPE Stat API," https://stat.ripe.net, [Online; accessed 06-November-2015].

[327] B. Donnet, M. Luckie, P. Mérindol, and J.-J. Pansiot, "Revealing MPLS Tunnels Obscured from Traceroute," *SIGCOMM Comput. Commun. Rev.*, vol. 42, no. 2, pp. 87–93, Mar. 2012. [Online]. Available: http://doi.acm.org/10.1145/2185376.2185388

[328] R. Kisteleki, "RIPE Atlas Update: HTTP Measurements, CLI Tools, DomainMON and More," *RIPE Labs*, Nov. 2015. [Online]. Available: https://labs.ripe.net/Members/kistel/ripe-atlas-update-http-measurements-cli-tools-domainmon-and-more

[329] "RIPE Atlas - User Defined Measurements," https://atlas.ripe.net/docs/udm, [Online; accessed 06-November-2015].

[330] "RIPE Atlas - Global Network Coverage," https://atlas.ripe.net/results/maps/network-coverage, [Online; accessed 06-November-2015].

[331] "PeeringDB," https://www.peeringdb.com, [Online; accessed 31-December-2015].

[332] V. Raisanen, G. Grotefeld, and A. Morton, "Network performance measurement with periodic streams," RFC 3432 (Proposed Standard), Internet Engineering Task Force, Nov. 2002. [Online]. Available: http://www.ietf.org/rfc/rfc3432.txt

[333] N. Wells, "BusyBox: A Swiss Army Knife for Linux," *Linux J.*, vol. 2000, no. 78es, Oct. 2000. [Online]. Available: http://dl.acm.org/citation.cfm?id=364412.364422

[334] "Vaibhav Bajpai - Lessons Learned From Using the RIPE Atlas Platform for Measurement Research," https://ripe68.ripe.net/archives/video/240, [Online; accessed 29-February-2016].

[335] "RIPE Atlas - Register a Probe," https://atlas.ripe.net/register, [Online; accessed 29-February-2016].

[336] E. Chan, X. Luo, W. Fok, W. Li, and R. Chang, "Non-cooperative Diagnosis of Submarine Cable Faults," in *Passive and Active Measurement*, ser. Lecture Notes in Computer Science, N. Spring and G. Riley, Eds. Springer Berlin Heidelberg, 2011, vol. 6579, pp. 224–234. [Online]. Available: http://dx.doi.org/10.1007/978-3-642-19260-9_23

[337] V. I. Levenshtein, "Binary codes capable of correcting deletions, insertions, and reversals," Tech. Rep. 8, 1966.

COLOPHON

This thesis was typeset with LaTeX 2ε using Hermann Zapf's *Palatino* and *Euler* type faces. The listings are typeset in *Bera Mono*, originally developed by Bitstream, Inc. The typographic style was inspired by *The Elements of Typographic Style*.