

Discovery Testbed: an Observational Instrument for Broadband Research

Kate Keahey*, Nick Feamster†, Guilherme Martins†, Mark Powers†,
Marc Richardson†, Alexis Schrubbe†, Michael Sherman†

* Argonne National Laboratory
Lemont, United States of America
Email: keahey@mcs.anl.gov

† University of Chicago
Chicago, United States of America

Abstract—Investigating phenomena that require continuous collection of data from a large and widely distributed array of hard-to-reach sources – such as understanding the performance of end-user broadband – has traditionally been hard. The opportunities in this sphere are now changing with easy availability of single board computers (SBCs), that are reliable and cheap, and thus can in principle be deployed in places of interest at large scales to gain coverage yielding statistically significant results. The challenge that this idea raises is how to create, deploy, an operate this type of infrastructure, given that its makeup and deployment properties are very different from hardware deployed in datacenters.

This paper presents the design of FLOTO, an observational instrument that supports the deployment and operation of mainstream SBCs to collect data through large-scale deployments in the field. FLOTO allows users to deploy devices to collect data of interest; operate those devices securely in remote locations without physical access to device; supports multi-tenant sharing of devices between different data collecting applications and user groups, that makes it possible to adapt or re-purpose the observational function of the instrument; and provides data collection and sharing functionality allowing user communities to benefit from the collected data. We describe the design of FLOTO and present a case study of its deployment as an observational instrument to collect broadband data. We conclude by discussing the possible adaptations of this instrument to study different scientific questions.

Index Terms—infrastructure, instruments, broadband, scientific platforms

I. INTRODUCTION

Broadband plays a critical role in end-to-end network performance, application performance, and ultimately user experience; understanding its performance, reliability, and other characteristics helps us to not only understand overall performance of Internet applications but also develop new protocols and systems to improve it. A long-standing barrier to developing such understanding has been the lack of an instrument that could be flexibly deployed in areas of interest and customized to measure the quality of access networks such that it provides detailed and accurate answers to a broad range of research questions. Most existing testbeds, such as Chameleon [1], CloudLab [2], FABRIC [3], and the PAWR testbeds [4] are focused on supporting exploration, i.e., allowing experimenters to deploy a broad range of experiments

by providing deep reconfigurability and interactive access in the controlled conditions of a datacenter or other fixed deployment. In contrast, this type of research problem requires customizable observation, i.e., the ability to deploy a relatively narrow range of long-running tests, but at large scales and with robust support for a range of potentially deep remote adaptations in the conditions of a real-world deployment that may evolve as the nature and/or the subject of inquiry changes.

Until quite recently constructing such an instrument was impossible. In recent years however, low-power, small-form factor single board computers (SBCs) dropped in price while gaining in capability, ultimately making possible deployments “in the field” and at relatively large scales, that are no longer confined to a specific datacenter with single ownership and centralized operations. In particular, a targeted deployment of such devices, strategically deployed on home and enterprise networks, can collectively provide an observational instrument capable of generating unprecedented insight into how broadband behaves and evolves. RIPE Atlas [5] has provided the most successful example of such an instrument to date but it lacks the capability to customize its operations and tests. Building a customizable platform of this type challenges many of the assumptions we make about hardware uniformly managed in a datacenter, and means that it is not enough to build such a platform, but that we also have to create new scalable operations models to support it.

This paper describes the initial experiences in constructing and deploying a *discovery testbed*, i.e., a large, distributed observational instrument – in this case, applied to broadband research. The testbed will ultimately consist of roughly a thousand edge devices deployed across multiple urban areas including Chicago, IL, Milwaukee, WI, and San Rafael, CA, as well as several university campuses, instrumented with broadband diagnostic tests to obtain observation data that will allow network scientists to gain insights into access network performance and development. In addition to technical questions leading to understanding and improving broadband performance, the instrument is intended to answer policy questions aimed at creating more equitable broadband deployment. The system supports exploration via the deployment of new devices in new locations; deployment of new applications

(in the context of our area of study new network tests) on selected subsets of devices; and the processing of datasets obtained by the instrument. The core of the system consists of a fleet management framework grafted onto an existing open core implementation, an application support framework, and data services. The case study application is an extended set of active measurements that we have developed, building on more than ten years of development in the BISmark project [6]–[8], providing an unprecedented capability for observing and understanding the dynamics and evolution of the Internet, and ultimately provide an invaluable tool for research and policymaking. We describe deployment across a variety of contexts and the data made available intended to create avenues of involvement in research that are currently difficult to bridge for lack of resources. On a different level, we also create a proof of concept and a blueprint for a discovery testbed at a large scale, pioneer new allocation and sharing methods, as well as a citizen science approach, that can then be adapted to different types of discovery which we discuss in the concluding section of the paper.

II. DESIGN OF FLOTO

FLOTO is a scientific observational instrument composed of multiple single board computers (SBCs), potentially equipped with or connected to a variety of sensors, that allows users to deploy applications on the SBCs that process and stream sensor data. To do so, this instrument supports three types of users: *device users*, who address new science questions by deploying the instrument in a different contexts or locations (e.g., by instrumenting a new campus or neighborhood); *application users* who address new science questions by developing new discovery and instrumentation applications (i.e., network diagnostic tests), deploy them on a defined subset of devices deployed in the fleet, and generate annotated datasets representing observations gathered; and *data users*, who address new science questions by analyzing datasets produced by the testbed and made publicly available.

To support device users we developed a framework that supports onboarding and offboarding devices easily; provides the ability to update and deploy new software versions to devices without direct physical access; monitor their status, and otherwise manage devices remotely within a platform built to withstand the security risks inherent in such deployments. Our implementation leverages the open core Balena fleet management product [9] that we extended to implement several device management features that support scaling in the number of deployments as well as users. They include support for multi-tenant usage (such as federated identity login and authorization); the ability to execute ad-hoc shell commands on device system containers; and creating and managing device collections. We also developed a device dashboard that allows device operators to manage devices easily by allowing them to view information about the device including status, type, configuration and general diagnostic output at a glance. The operator can also create and manage device profiles, or trigger device-specific actions such as rebooting or shutting down

the device, pinging it, changing its configuration, or restarting the services running on it. To make larger deployment scales easier to manage, the dashboard also allows operators to create collections/filters of devices with the same properties to drill down on device types (e.g., deployed within a certain area, with certain ownership, or with specific status value), update device variables, and generally manage device lifecycle. This functionality is available to device users via a command-line interface, or for ease of interaction a portal.

While this fleet management layer provides an effective and scalable tool for the management of devices, it does not yet support a multi-tenant, multi-application use pattern that allows different *application users* to run different types of diagnostic or data processing tools in different timeslots. To support this need, FLOTO provides a higher layer API that allows application users to run containerized tests on a selected set of devices by submitting new application packages configured to execute in Docker containers. Users can reserve timeslots in advance similar to datacenter resources [10]; the applications can then be scheduled for one time or periodic execution. Data produced by the application is collected first in temporary storage on the device, and then transferred out through a special-purpose data streaming application that can be scheduled at a timeslot different than data collection so as not to interfere with broadband measurements. The decision to make data transfer a standalone application gives the user control over when it can be scheduled and also opens the possibility of implementing their own, based on the provided reference implementation, to customize data streaming, e.g., where it crosses administrative boundaries or otherwise needs to meet specific destination needs. To support the development of new applications, FLOTO provides a repository of popular Docker images and build agents, including images with default system functions such as data evacuation. Users are able to compose multiple containers into a more complex workflow-based application, (e.g. two types of measurement + data evacuation) if desired, and this composition is then scheduled as above.

Data is streamed from the device annotated with generic meta-data, such as timestamp and location of data acquisition, data ownership by application and project, and device information to a data repository that currently backends to Amazon S3. For better support of *data users*, we make annotated datasets generated by the project available via a data portal designed to make it easy to discover, process, and share relevant datasets. Data is owned by the project that created it and can be set to be publicly shareable by this project depending on the policies with which the data has been gathered and visualized with a Grafana front-end [11], facilitating easy deployment, analysis, and visualization of new tests results.

III. CASE STUDY: BROADBAND MEASUREMENT

End-user broadband infrastructure comprises a wide range of *access networks*, including fixed-line Internet access networks, satellite Internet networks, fixed-wireless access networks, and enterprise networks (such as campus networks).

Over the last decade, research into these types of networks has been comparatively limited due to the difficulties in collecting high-quality measurement data from (and about) them. However, understanding access network performance, reliability, and usage is critical not only to understanding overall performance of Internet applications but also to developing new protocols and systems to improve them. Access networks are also the portion of the infrastructure where widespread disparities exist in terms of deployment, availability, reliability, and many facets of performance that lead to the “digital divide” in our society.

Early studies using the BISmark package [7] made significant advances in understanding access network performance by developing an extensive suite of active performance measurement tests, including upstream and downstream throughput, packet loss rates, jitter, latency under load, DNS lookup times, as well as network availability and uptime. The insight of BISmark was to measure directly from the access network and collect data over time to reveal a holistic picture of end-user Internet performance from multiple vantage points.

Previous versions of BISmark were deployed directly in network routers; this deployment however was inherently limiting as routers are not always accessible or scalable – nor is it the case that all questions of interest can be answered from this vantage point. These limitations gave rise to a redesign of BISmark [6] to allow network tests to execute on an independent measuring box connected to routers or other access points, which gathers data from those points of deployment and makes them available for exploration. The “measuring box” must not only supply the processing capacity powerful enough to support the test suite and be capable of deployment scalable to thousands of devices, but also be robust enough to support seamless device activation and configuration, updates without direct physical access, and strong device and system security. Another critical requirement is the ability to customize and deploy new tests – potentially from multiple research groups – as our understanding of “what” and “how” to measure the Internet evolves. The development of FLOTO described in the previous section was guided largely by those considerations.

The initial deployment of FLOTO for network measurements took place at the University of Chicago campus in January 2023. It consisted of the deployment of 34 measurement boxes, implemented by Raspberry Pi 4s, and attached to switches in network closets across campus. For ease of deployment, the devices were fitted with Power over Ethernet (PoE) to allow for single-cable installation in network closets where power sources are constrained. Devices were flashed with a FLOTO image; once powered, they self-enrolled in the FLOTO fleet-management system, installed the modern BISmark package, and began reporting data to a centralized data store and analytics pipeline. The data were made available to the client (UChicago IT) via online dashboards and raw data exports.

Because devices are attached to ports on the campus access network, they experience the same “environment” in terms of congestion, outages, and network activity as actual clients of

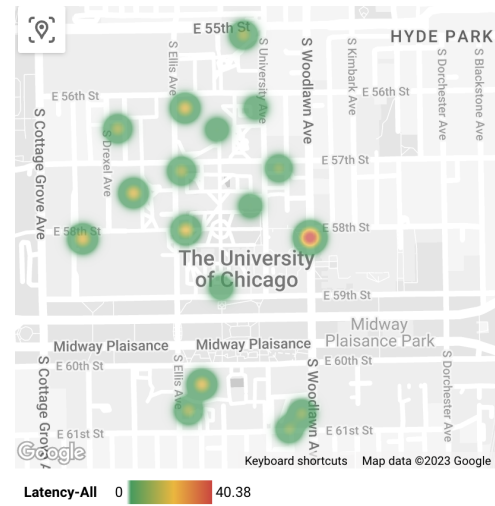


Fig. 1. Heatmap of latency in ms to google.com across all FLOTO devices at UChicago.

the campus network (regular users) and serve as a measurement proxy for user experience. The tests deployed include three popular speed test tools, various measurements of latency, jitter, and packet loss to well-known public destinations (google, news, etc.), and traceroutes to correlate issues with routing changes, for example if the access network provider no longer peers with a given upstream provider, removing a “shorter” network path (see Figure 1 for a visualization of latency over the University of Chicago campus).

Our initial plan was to deploy roughly 1,000 devices across the city of Chicago. However, rising interest in understanding network performance for technical as well as policy reasons, caused us to rethink our deployment strategy to accommodate additional sites in Milwaukee and San Rafael, in addition to smaller deployments. Furthermore, we refined our sampling strategy to focus on smaller geographic areas and higher density of devices to increase research opportunities.

In Chicago, We will deploy 200 devices to residents in the Logan Square neighborhood of north Chicago and in the South Shore neighborhood of south Chicago. In Milwaukee, we will deploy 300 devices in the Franklin Heights/Park West neighborhoods of north Milwaukee and Muskego Way and Historic Mitchell Street in south Milwaukee. For both deployments, we will control for variables that affect Internet performance, including Internet service provider, gateway equipment, and building type (single-family vs multi-dwelling), to identify sources of variation from collected data. In addition, traceroute and latency measurements from these deployments will contribute to research that examines whether spatially-clustered access networks demonstrate similar performance variability.

In San Rafael, California, we’re partnering with Merit Network and the University of Michigan to deploy 300 devices in underserved apartment buildings. This unique setting, with high-density complexes and aging coaxial wiring, offers opportunities to study performance variability within buildings

and how it correlates with network usage. For instance, we'll investigate whether multiple access networks on a single access point show uniform service degradation. Several smaller deployments (3-10 devices) will support research objectives from creating a 5G testbed to assessment of the quality of wireless Internet service supplied to an apartment building.

Beyond these research objectives, our goal is to share the data from these residential deployments with policymakers and researchers. This information will help inform decisions about federal funding for new broadband infrastructure and assessing service provider coverage claims. It will also support social science research into urban populations affected by underprovisioned broadband infrastructure and related digital equity challenges.

IV. DISCUSSION

An interesting question relating to any scientific instrument is how we can "turn the knobs" and adapt it to measure different phenomena, in different places, at different levels of resolution, and to answer different questions. In this section, we will discuss these types of customizations for FLOTO.

As in the case of a telescope that can be pointed at a different area of the sky, one way of adapting FLOTO to answer new questions is by *varying the deployment scope*. We currently do it in two ways: by either shipping devices, pre-flashed with a FLOTO image, to new deployment destinations, or by supporting a FLOTO image that users can put on already owned and potentially already deployed devices. The latter is particularly convenient for augmenting existing sensor deployments or when working with devices that are deeply integrated into existing infrastructure as is often the case with e.g., autonomous vehicles [12], [13]. Another simple way of extending the instrument is to *adapt its "sensing abilities" programmatically* by deploying new applications (e.g., other types of broadband tests in the case of the current application) either to fine-tune them to focus on phenomena of interest or keep pace with the development of our understanding of broadband. Those two adaptations speak directly to the FLOTO API design needs of device and application users described in Section II.

A more interesting question is whether FLOTO can be used to *measure phenomena other than broadband*. To achieve this kind of adaptation, on deployment, the FLOTO device needs to be combined with a new sensing "peripheral", i.e., a sensor or actuator such as e.g., a software-defined radio (SDR) providing the sensing capabilities in question. The basic requirements for combining such peripherals to FLOTO devices are the same as combining them to any SBC, i.e., that they attach via an interface that physically exists on the device (e.g. USB, GPIO, I2C, or SPI), and that there is driver support (either already in the Linux kernel or loadable via Linux driver module) that can make the peripheral accessible via a Linux interface (e.g., "/dev/something"). Attaching the peripheral to a user container can be achieved on container launch via the FLOTO infrastructure, though the system would have to be extended to add reasonable capabilities such as which

peripherals are available for specific devices (i.e., hardware discovery), authorization for peripheral use, and deployment and troubleshooting processes. This area will most require configurability and continuous integration as new peripherals become available and infrastructure capabilities need to be adapted to meet their needs.

Lastly, the instrument can be adapted to *support users with different interests, levels of skill, and access* by providing different ways of ingress into the system and supporting resource sharing. For example, users who don't have the need or the means to deploy their own devices or are not interested in developing new applications can still benefit from publicly shared datasets generated by the instrument; FLOTO is working towards making such annotated datasets available via a portal. Device deployments can also be shared and potentially better amortized by advertising timeslots that are not used by the primary deployment team via a "matchmaking" service that advertises availability on one end, and the requirements of deployment-ready applications on the other.

ACKNOWLEDGEMENT

Results presented in this paper were obtained using the FLOTO project (award 2213821) supported by the National Science Foundation. This material is based upon work supported by the U.S. Department of Energy, Office of Science, under contract number DE-AC02-06CH11357.

REFERENCES

- [1] K. Keahey, J. Anderson, Z. Zhen, P. Riteau, P. Ruth, D. Stanzione, H. Gunawi, C. Hammock, and M. Joe, "Lessons Learned from the Chameleon Testbed," in *2020 USENIX Annual Technical Conference (USENIX ATC 20)*, 2020, pp. 219–233.
- [2] D. Duplyakin, R. Ricci, A. Maricq, G. Wong, J. Duerig, E. Eide, L. Stoller, M. Hibler, D. Johnson, K. Webb *et al.*, "The design and operation of {CloudLab}," in *2019 USENIX annual technical conference (USENIX ATC 19)*, 2019, pp. 1–14.
- [3] I. Baldin, A. Nikolich, J. Griffithen, I. I. S. Monga, K.-C. Wang, T. Lehman, and P. Ruth, "FABRIC: A National-scale Programmable Experimental Network Infrastructure," *IEEE Internet Computing*, vol. 23, no. 6, pp. 38–47, 2019.
- [4] "PAWR," <https://advancedwireless.org/>.
- [5] "RIPE Atlas," <https://atlas.ripe.net/>.
- [6] "BISmark v. 2.0 ("Netrics")," <https://github.com/internet-equity/netrics>.
- [7] S. Sundaresan, S. Burnett, N. Feamster, and W. De Donato, "{BISmark}: A testbed for deploying measurements and applications in broadband access networks," in *2014 USENIX Annual Technical Conference (USENIX ATC 14)*, 2014, pp. 383–394.
- [8] N. Feamster and N. Marwell, "Benchmarks or equity? a new approach to measuring internet performance," in *Conference on Communications, Information, and Internet Policy (TPRC)*, Washington, DC, sep 2022, pp. 1–10. [Online]. Available: <https://tprcweb.com/>
- [9] "Balena," <https://www.balena.io/>.
- [10] K. Keahey, P. Riteau, J. Anderson, and Z. Zhen, "Managing Allocatable Resources," in *2019 IEEE 12th International Conference on Cloud Computing (CLOUD)*. IEEE, 2019, pp. 41–49.
- [11] "Grafana," <https://grafana.com/>.
- [12] W. Fowler, K. Keahey, and A. Esquifiel Morel, "Road to reliability: Optimizing self-driving consistency with real-time speed data," in *In Submission to the International Conference for High Performance Computing, Networking, Storage and Analysis (SC'23 Poster)*. IEEE Press., 2023.
- [13] J. Tsen, J. Anderson, L. Bobadilla, and K. Keahey, "One fish, two fish: Choosing optimal edge topologies for real-time autonomous fish surveys," in *In Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis (SC'21 Poster)*. IEEE Press., 2021.