

HoBBIT: a Platform for Monitoring Broadband Performance from the User Network

Walter de Donato, Alessio Botta, and Antonio Pescapé

University of Napoli Federico II (Italy)
{walter.dedonato,a.botta,pescapè}@unina.it

Abstract. In the last years the problem of measuring and monitoring broadband performance has attracted a lot of attention. In this paper we present our recent work in this area, providing the following main contributions: firstly, we analyze the literature, illustrating the different existing approaches and the main requirements of platforms implementing these approaches; secondly, we describe the lessons learned designing and implementing a platform called HoBBIT; thirdly, we analyze and share the results obtained from the current deployment in Italy. The lessons learned and the design guidelines of the platform can be of help in light of the recent research and standardization efforts in this area. The results and the dataset we shared with the community provide information on the current status of broadband in Italy and allow other researchers to access fresh measurements from a real and large scale deployment.

1 Introduction

Recently, a great interest has been focused on the performance evaluation of broadband access networks, in order to shed light on (i) what is the real experience perceived by users on a large scale [1] and (ii) to provide empirical data for policymakers [2]. Most projects pursuing this objective perform active measurements to evaluate common QoS metrics. They operate such measurements by adopting different techniques and from different viewpoints, each of them affecting obtained results in a different way: most projects require users to cooperate by running a software on their computer or by deploying a special device in their home network, acting as vantage points (VPs) to perform the measurements¹.

In this paper we describe our recent research work, basically providing a twofold contribution. On the one hand, (i) we briefly analyze the literature by providing a taxonomy of approaches for evaluating broadband performance; (ii) we identify the most effective ones by evaluating their pros and cons; (iii) we outline the main characteristics and requirements of a platform suitable for applying such approaches. On the other hand, (iv) we describe the solutions we adopted in designing and developing a measurement platform called HoBBIT²

¹ In this work we explicitly target fixed broadband access networks and related issues. Mobile broadband is out of the scope of this paper.

² <http://hobbit.comics.unina.it>



Fig. 1. A taxonomy for broadband performance evaluation approaches based on the location of the VPs initiating the measurements.

(Host-Based Broadband Internet Telemetry); (v) we analyze the preliminary results obtained from the current deployment; (vi) we release open data sets to the community³.

To highlight and summarize the significance of the contribution proposed in this work, we underline that, to the best of our knowledge, it extends the results in literature (a) by providing design guidelines for implementing platforms for large scale measurements of broadband accesses, and (b) by collecting, analyzing, and sharing a rich set of data from real users in a specific geographical region for 3 years. The importance of the contribution should also be seen in the light of the work that various bodies are doing for standardizing the different aspects related to these platforms (architectures⁴, metrics⁵, methodologies⁶, use cases⁷, etc.) and it is witnessed by recent events like the 1st NMRG workshop on large scale network measurements⁸.

2 Platforms for broadband monitoring and measurement

By analyzing the literature and the main projects addressing the evaluation of access network performance, we defined a simple but effective taxonomy. As illustrated in Fig. 1, it classifies the approaches depending on where the VPs initiating the measurements are located: *WAN-based* approaches exploit VPs deployed in Internet at different locations, such as the network of the user’s *ISP* [3–5], *transit* networks [6, 7], or *stub* networks hosting real services [8]; *UN-based* approaches deploy VPs inside the user’s network, such as the home *gateway* [1, 9] or a software running on generic *hosts* (laptops, tablets, etc.) in the form of a standalone *application* [10–15] or a *browser* executing a web application [16–18].

The study of the literature allowed to understand that all the approaches have both advantages and drawbacks, but those working from user network result to be more effective⁹: they can better take into account the context in which measurements are performed. Among them, we identify gateway- and application-based as the most promising approaches: they allow to perform repeated exper-

³ <http://traffic.comics.unina.it/Traces>

⁴ LMAP, <https://datatracker.ietf.org/wg/lmap/charter/>

⁵ IPPM, <https://datatracker.ietf.org/wg/ippm/charter/>

⁶ BMWG, <https://datatracker.ietf.org/wg/bmwg/charter/>

⁷ HomeNet, <https://datatracker.ietf.org/wg/homenet/charter/>

⁸ <http://tinyurl.com/pqfkd7e>

⁹ We refer the reader to [19] for the details about such analysis.

iments on the same access link over a long time period, thus obtaining more statistically significant results. A trade-off exists between these two approaches: gateway-based ones are more accurate, but have some drawbacks (e.g., shipping devices is costly, gateways offer limited resources); client-based ones are less accurate (e.g., they cannot directly consider all the cross traffic potentially affecting measurement results), but enable to easily obtain capillary large-scale deployments. These aspects make the latter more effective for obtaining a fine-grained geographical view of broadband performance.

While in [9] we presented a gateway-based platform, in this work we describe what we learned while designing and implementing an application-based platform for measuring broadband access networks. We believe that this contribution can be of great help for the research and standardization efforts ongoing in this field and can stimulate fresh and interesting discussions at the workshop.

Proposed Reference Architecture. As illustrated in Fig. 2, a UN-based platform should include three essential components: (i) *measurement clients* (henceforth *clients*) running inside the user’s network, which periodically ask for instructions about experiments to execute, and report back the obtained results; (ii) *measurement servers* (MRSs) acting as counterparts for active measurements, possibly deployed on networks with high-capacity links to the main backbones; (iii) a *management server*¹⁰ (MGS) responsible for: configuring and planning experiments to be performed by clients; collecting, organizing, and storing measurement results; monitoring the operating state of clients and MRSs; pushing software updates to clients.

Platform Requirements. Ideally, VPs should cover all the geographic locations of interest, ISPs, and service plans (henceforth *plans*). MRSs should be properly distributed to have at least one of them at the shortest network distance, while their number should be proportional to the number of clients in a given location.

We further identify the following requirements:

- *scalability*: it should be able to cope with a large number of clients;
- *accuracy*: it should provide accurate measurements while avoiding significant interferences among them;
- *portability*: clients should be as portable as possible, in order to potentially involve any user;
- *flexibility*: performed measurements should be dynamically tunable to behave according to the context;
- *autonomicity*: it should automatically obtain any required information (e.g., plan details, geographical area) without relying on user cooperation (which may lead to unverifiable mistakes);
- *manageability*: it should provide an easy way to manage experiments (e.g., definition, configuration, deployment);
- *traceability*: every measurement taken should include a timestamp and refer to a well-identified access network;
- *non-intrusiveness*: it should mitigate the impact of measurements on user traffic;
- *security*: it should not represent a potential vulnerability for the user network;

¹⁰ A single point of failure can be avoided by deploying multiple servers.

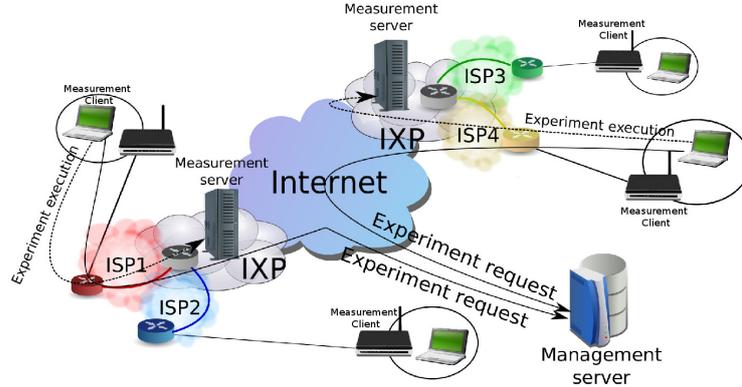


Fig. 2. Platform architecture overview.

- *privacy*: collected data should always be treated in order to preserve user privacy;
- *(geolocalized) visibility*: collected data should be properly presented to the user as an incentive to participate;
- *independence*: it should operate independently on ISPs funding or control to guarantee unbiased results;
- *inexpensiveness*: the deployment cost should be limited in order to allow the number of clients to scale quickly.

3 The HoBBIT Platform

According to the guidelines defined in Sec. 2, we designed and implemented HoBBIT, an application-based platform currently operated on the Italian territory¹¹. Volunteers can install a portable client application on their computer(s) for monitoring the performance of their access network(s).

Since HoBBIT clients can move among different access networks (henceforth *connections*), we designed the platform by using a connection-centric approach. Each connection is associated to its geographical position (up to the zip code granularity) and to its network details (i.e., *ISP*, *plan*). Every user may own more *installations* of the client, each possibly contributing to the performance evaluation of several connections. As for measurements, they are referred to as *experiments*, which are organized in *campaigns* having a specific goal. Each experiment is periodically executed following a detailed *schedule*, which is defined by the specific campaign. Each experiment can produce more than one *measurement output* (i.e., a set of samples collected at a specific sampling rate) or a *failure log*. Finally, a set of MRSs is available for supporting the experiments with part of their resources (*slices*).

At runtime – every 5 minutes – each client asks the MGS for a list of experiments. If the list is not empty, it sequentially executes all the experiments and finally reports back all the results. All the messages exchanged between the

¹¹ The deployment can be easily extended to other geographical areas.

clients and the MGS are XML-encoded and transferred over the HTTPS protocol. In the following, by adopting a problem/solution scheme, we briefly describe the main lessons we learned while designing and implementing HoBBIT, with the aim to satisfy most of the requirements outlined in Sec.2. We refer the reader to [19] for a detailed discussion.

Tracking client-connection associations. HoBBIT clients can move among different connections or operate together behind the same connection. This is primarily true for mobile devices (e.g., laptops, tablets), which are today really common. It is then necessary to properly track their association with connections, supporting also plan upgrades and ISP changes.

Solution. The client computes a hash value starting from the MAC address of its current default gateway, which is used by the MGS to uniquely identify the connection. The hash is computed adding to the string representing the MAC address a progressive number, which is incremented every time a plan upgrade or an ISP change is detected or notified by the user. Hence, the history of a connection can be reconstructed. Since a gateway may have more than one network interface (e.g., wired and wireless), different hash values may be associated to the same connection. To cope also with such case, we implement an aliasing condition: we consider two connections in alias only if different clients report the same public IP address close enough in time¹². This solution enables different clients to be coordinated by the MGS for evaluating the performance of a connection. Hence, it helps to satisfy both accuracy and traceability requirements.

Detecting service plan and location. Every time a new connection is identified, it is necessary to detect its main properties: ISP, plan details, and geographic location. We found that asking such information directly to users is not sufficient: they tend to leave the information empty or, worst, to provide wrong answers. This is particularly true for plan details (e.g., advertised downstream and upstream rates, and access technology), since most users do not have enough technical knowledge to provide correct information.

Solution. In order to mitigate as much as possible the voluntary or accidental introduction of errors caused by users, we adopted a semi-automatic approach that tries to obtain most of the information automatically, while requesting the remaining information to the user in a guided way. We automatically detect: i) the ISP, using the public IP address of the client; ii) the plan, matching the results of an ad-hoc measurement campaign against a database of existing ISPs and plans; iii) the location up to the zip code granularity (only if GPS is available). Then, we present the inferred information to the user on a web page, where he can confirm, correct, or refine it. This procedure is performed every time an unknown connection is detected and is aimed at providing autonomy.

Providing a flexible measurement framework. In order to cope with heterogeneous and evolving network scenarios, the platform should be able to support any measurement tool and to dynamically adjust its parameters depending on the context. Moreover, measurements should be scheduled according to the

¹² Since some ISPs assign to users private IP addresses, we apply the aliasing condition only if we detect one private hop between the client and the Internet.

needs of the specific campaign, which may require to repeatedly execute an experiment at a specific time on a particular set of connections.

Solution. We designed the HoBBIT framework to give high flexibility in the definition, configuration and execution of the experiments and in their assignment to clients. On the one hand, each experiment is defined by the (P_i, P_o, S) tuple, where the P_i and P_o sets represent input and output parameters, and S is a wrapper script acting as an adaptation layer between the client and the underlying measurement tool. On the other hand, each campaign can be assigned to a subset of connections by means of configurable parameters (e.g., all the DSL connections in a specific area).

Providing accurate measurements. Obtaining accurate results is fundamental for any measurement platform and, when focusing on the access network, it is important to isolate most of the confounding factors coming from both the user network and the Internet. Moreover, accurate tools should be adopted and their execution should be controlled in order to avoid excessive interferences. Indeed, since many clients conduct measurements toward a few servers, it is important to properly coordinate the execution of the experiments to avoid interferences on the server side.

Solution. In order to isolate most of the confounding factors, we adopt the following strategy: (i) we try to deploy dedicated MRSs in networks having high-capacity links to the main backbones (in order to push the bottleneck toward the access link) and in different locations (in order to be able to select the MRS closest to the client both in terms of end-to-end latency and hops); (ii) we avoid conducting measurements if user-generated traffic and CPU usage exceed thresholds defined for the connection (depending on upstream and downstream rates, access network capacity, etc.). (iii) since we cannot take into account all the interferences, we try to repeat as many measurements as possible in the same conditions (e.g., same daytime and day of week), in order to count on big numbers for isolating outliers; (iv) for each measurement, we take note of the involvement of a wireless link (e.g., Wi-Fi) between the host and the access link.

As for the accuracy of the tools, we purposely designed the measurement framework to support any underlying pre-existing tool. Hence, standard and well-tested tools are and can be adopted depending on the context.

To cope with server-side interferences, we designed a centralized scheduling algorithm executed by the MGS while assigning experiment targets to clients¹³. We assign MRSs a network capacity made of reservable fixed-size slices. For each slice we keep a timestamp representing the time when a slice will be free again. When a client asks for an experiment target MRS, the MGS reserves – according to the experiment duration and required bandwidth – the set of slices having the lower timestamp values on a single MRS. If not enough slices are immediately available, the reply includes a delay time, i.e. the time to wait before starting the experiment.

¹³ Because of space constraints, we refer the reader to [19] for a detailed description of the algorithm

Making the platform scalable. A platform conducting periodic measurements and potentially involving a huge number of clients has to properly address scalability from different viewpoints: i) network load introduced by both control and measurement traffic; ii) available computational and networking resources on MRSs; iii) storage and processing of the amount of data collected on the MGS(s).

Solution. While the scheduling algorithm described above allows to address the first two viewpoints by properly dimensioning the number of MRSs and tuning the periodicity of the experiments, further work is necessary to address the third one. Indeed, the amount of data collected over time tends to increase very quickly, and processing it in realtime becomes unfeasible without special arrangements. Since the objective is to maintain the finest granularity on the measurements performed (i.e. the samples produced by every single experiment) while offering different levels of aggregation on different axes (i.e. temporal and geographical), we structured the HoBBIT database in order to store and aggregate data in an efficient manner (e.g., materializing views, partitioning tables). To cope with the increasing amount of samples produced by every single experiment, we are also studying and comparing different reduction approaches.

Making the platform easily manageable. To properly control and configure a complex platform made of several distributed nodes, it is fundamental to have a (logically) single management point, which should allow to easily reconfigure or update any part of the platform, in order to support remote troubleshooting and debugging without user intervention.

Solution. HoBBIT has been purposely designed to control any aspect of the platform from the MGS(s). It provides a web-based management interface which gives full control on the configuration of campaigns, experiments and related aspects, while giving visibility on monitored connections, installations running on them, and collected measurement results. Moreover, we can set any client in debug mode (to send detailed debug logs back to the MGS) and automatically update them in automatic or on-demand (i.e., only for specific clients) modes¹⁴. This solution provides manageability.

Involving as many users as possible. The success of platforms like HoBBIT is determined by the ability to involve as many users as possible, possibly using different ISPs and plans, and with a large geographical coverage. Main problems here are the diversity of user equipments (PCs, OS, etc.) and the necessity to find effective incentives [20, 21].

Solution. In order to make the client portable to the most widespread operating systems, we made the following choices: i) we developed the client using the Qt libraries, which provide high portability and flexibility; ii) we selected the `bash` and `gawk` interpreters for implementing wrapper scripts, because they are natively available on Linux and Mac OSX, while they have Cygwin-based versions for Windows¹⁵; iii) we select measurement tools according to their portability.

¹⁴ The HoBBIT client is not installed system wide and does not require administration privileges to perform updates.

¹⁵ Both the Cygwin library and the interpreters are embedded into the client.

Table 1. Experiments part of the BPE campaign.

ID	Measured metrics	Adopted transport protocol	Required bandwidth (kbps)	Measure duration (sec)	Sampling rate (msec)
#1	Round-trip Latency, Jitter & Packet loss	UDP	0	30	1000
#2	Upstream throughput	UDP	$f(AR_{up})$	15	
#4		TCP			
#3	Downstream throughput	UDP	$f(AR_{dw})$	15	
#5		TCP			

As for providing incentives for the users, we give them visibility and guided interpretation on the performance measured, enabling the comparison with nearby connections.

Preserving user experience. Frequent measurements are important to compensate the effect of confounding factors and to observe time-dependent trends. On the other side, it is fundamental to avoid a noticeable impact on user experience, to keep them involved in the long term.

Solution. The HoBBIT client has been purposely designed to provide non-intrusiveness. On the one hand, the user interface normally consists in a system tray icon displaying the current status (e.g., measure in progress, idle, etc.), while limited user interactions are required only when a new connection is detected. Moreover, interacting with the icon, the user can always know when the next set of experiments will be performed and decide to temporarily disable their execution for a predefined period (e.g., from half hour up to four hours). On the other hand, experiments are performed only if computational and networking resources are not over two different thresholds, defined according to the host capabilities and to the plan respectively. If after a number of retries such conditions are not satisfied, the experiment is aborted.

4 Experimental analysis

Since December 2010 HoBBIT is running a measurement campaign named Basic Performance Evaluation (BPE). Its objective is to measure a basic set of performance metrics on a regular basis and for a long period. We present preliminary results from about 190K experiments performed on 310 selected connections monitored by the current deployment, which counts about 400 clients.

4.1 BPE campaign details

Tab. 1 reports details about BPE. Round-trip latency, jitter and packet loss are evaluated generating 10 isochronous UDP packets per second for 30 seconds, while upstream and downstream throughput are measured imposing for 15 seconds a constant bitrate – 5% higher than the advertised rate (AR) – with both

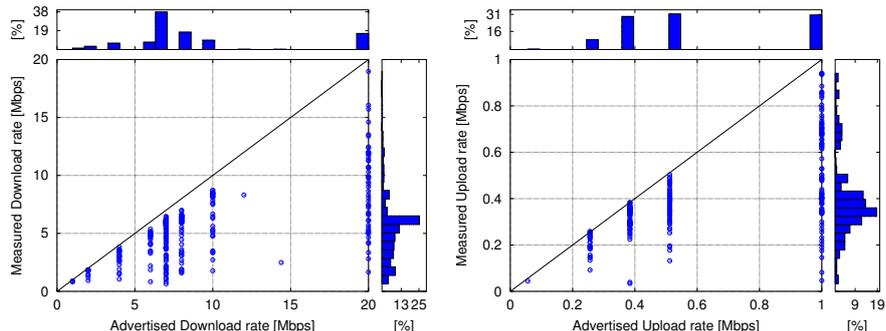


Fig. 3. Average discrepancy between advertised and actual performance.

UDP and TCP protocols. Such measurements are performed every half hour toward three MRSs located at the University of Napoli by embedding the D-ITG tool [22].

4.2 Preliminary results from the real deployment

Analyzing the data collected, we tried to answer to the following important questions about broadband in Italy.

To what extent ISPs offer the advertised performance? Most ISPs advertise only the downstream and upstream capacity of their connections. We compared them with the average throughput results obtained for each monitored connection. Fig. 3 reports the comparison of measured and advertised rates in downstream (left) and upstream (right), where x and y axes respectively report advertised and measured speeds, each point identifies a monitored connection, and the histograms highlight their percentage with respect to both speeds. At first glance, it is evident how no connection meets or outperforms the advertised maximum rates. The discrepancy between actual and advertised performance is highly variable, ranging from 10% to 90% for downstream and from 5% to 100% for upstream. It is also noticeable that users owning high-end plans observe a higher performance penalty than those owning low-end plans.

Clustering the observed ADSL performance by plans, some other characteristics are more evident. Fig. 4 – divided in four groups according to the advertised upstream capacity for ease of reading – reports the average rates measured in both directions, where each point (red 'o' marker) represents the measured rates of a connection and a line connects each of them to the corresponding advertised rates (black '*' marker). In this kind of plot nearly horizontal (vertical) lines highlight a very asymmetric discrepancy with respect to the downstream (upstream) direction. On the one hand, Fig.4(a) highlights how 20 Mbps plans are in general very far from the promised maximum rates, which is a consequence of a technological constraint: the distance between the modem and the DSLAM is often too long to avoid excessive signal attenuation. Furthermore, 25% of monitored high-end plans (more expensive) achieve rates advertised by low-end plans

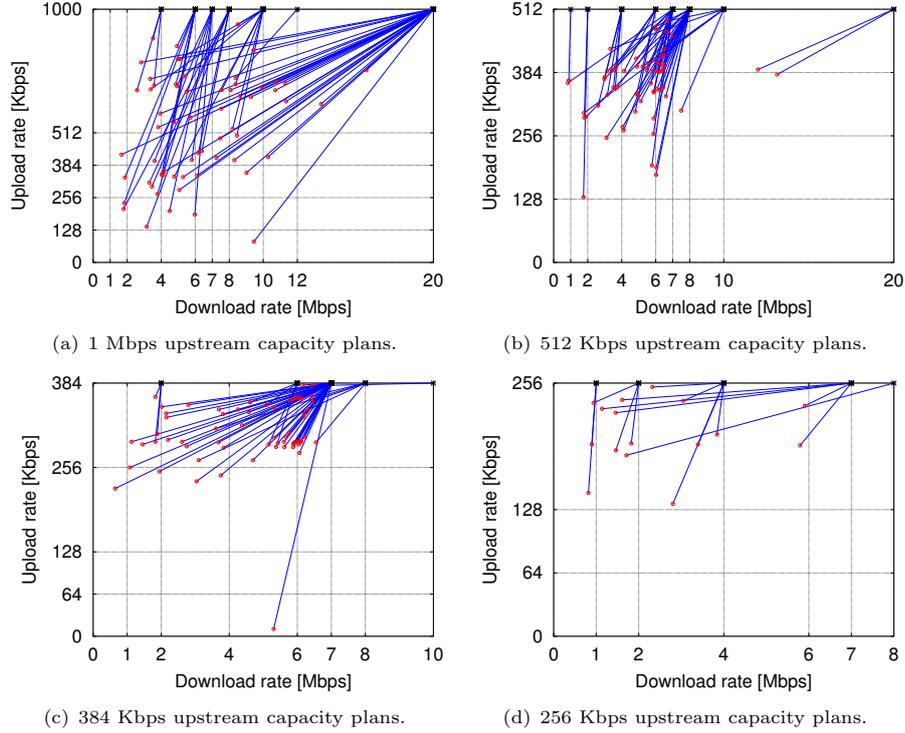


Fig. 4. Average ADSL throughput clustered by plan.

(cheaper), causing a waste of money for the user. On the other hand, Fig.4(c) and 4(d) show how low advertised upstream rates are more likely to be achieved. From the above results, the answer is that actual performance unlikely reaches the maximum advertised rates and the difference between actual and measured rates is not negligible in many cases.

What unadvertised QoS metrics do different ISPs provide? By looking at unadvertised QoS parameters, different ISPs may offer significantly diverse performance. Fig. 5(b), 5(c), and 5(d) report the empirical cumulative distributions of such metrics for four major Italian ISPs, for which HoBBIT monitored the amount of connections reported in Fig. 5(a). According to Fig. 5(b) all the ISPs provide latencies below 100 ms in 80% of cases, but only the Wind operator provides under 40 ms latencies in more than 50% of cases. On the other hand, according to Fig. 5(c) and Fig. 5(d), Wind provides high jitter values in more than 40% of cases and some packet losses in 25% of cases. A possible explanation of such results is that Wind configures ADSL plans in fast mode, which provides low latencies at the cost of higher loss probability.

To what extent performance depends on geographic location? We analyzed the maps reported on the HoBBIT website. Fig. 6 shows average performance aggregated per region (nation wide) and per municipality (in the area around Napoli). While in average the downstream throughput (Fig. 6(a)) is be-

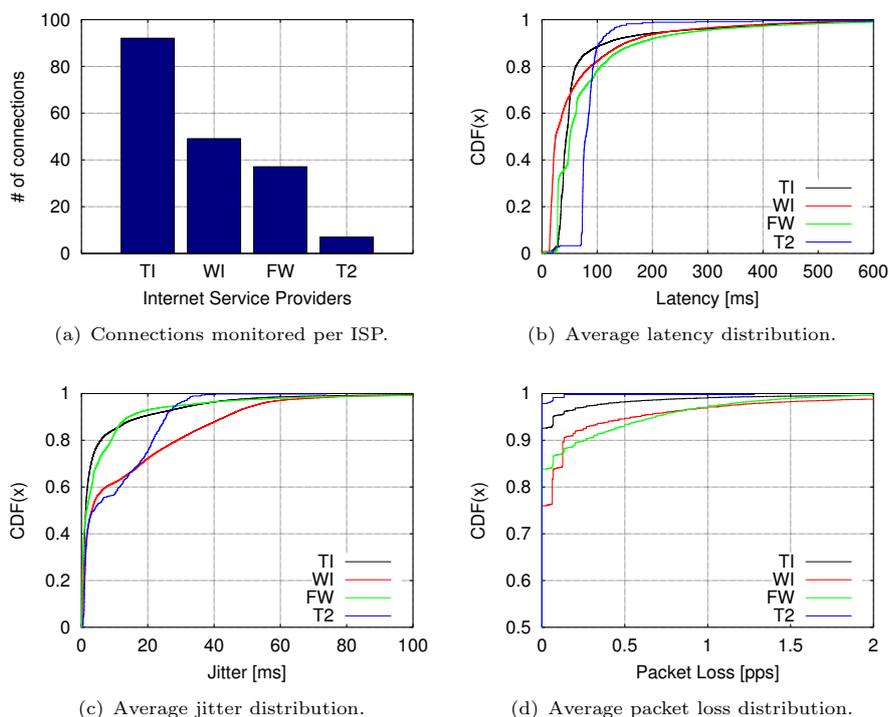


Fig. 5. Average unadvertised QoS metrics from 4 major Italian ISPs: Telecom Italia (TI), Wind (WI), Fastweb (FW), Tele2 (T2).

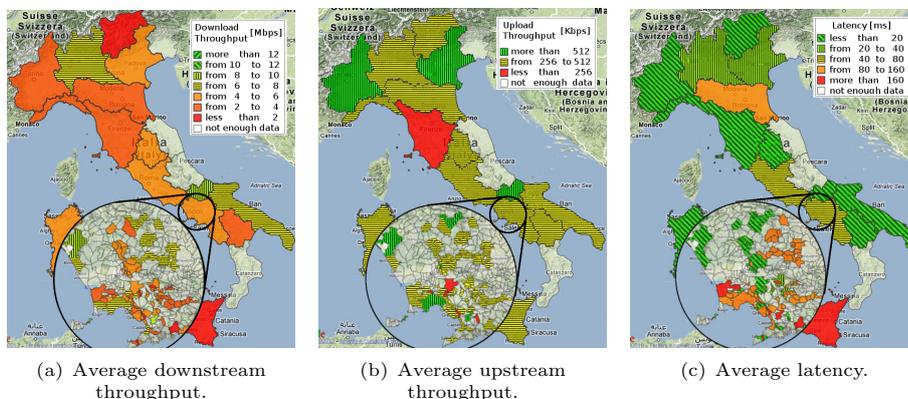


Fig. 6. Average performance ranges measured: Italian regions and municipalities around Napoli.

tween 4 and 6 Mbps in most regions, a few of them observe values lower than 2 Mbps and the same happens at municipality level. The average upstream throughput (Fig. 6(b)) is higher than 512 Kbps only in three regions and a few

municipalities, while most of other areas provide values between 256 and 512 Kbps. Finally, only a few areas report average latencies (Fig. 6(c)) above 40 ms, both at region and municipality level.

5 Conclusion

Shedding light on broadband performance on a large scale is still a challenging task. In this work, starting from the definition of a taxonomy for the existing approaches, we presented the lessons we learned while designing and implementing HoBBIT, which tries to satisfy most of the requirements we identified as necessary to measure and monitor broadband performance from the user network, using an application-based approach. We discussed the main issues encountered and the practical solutions we adopted for them. Thanks to the deployment of HoBBIT on the Italian territory, we also collected and shared fresh results on the performance of broadband in Italy. By analyzing these measurements, we tried to answer three basic questions about the performance perceived by users. We detected that (i) in average most plans promise maximum rates which are far from actual average rates, (ii) 25% of HoBBIT users may obtain the same performance – and save money – by choosing a cheaper plan, and (iii) unadvertised QoS metrics can make the difference among similar plans. Finally, we underline that to obtain more reliable insights about performance on a geographic basis, a more capillary deployment is necessary. Accordingly, in our ongoing work we are investigating incentive mechanisms to reach a wider user participation. We further plan to perform an experimental comparison among HoBBIT and similar platforms, and to design and deploy in it new measurement techniques for detecting network neutrality violations and Internet censorship [23].

Acknowledgment

This work has been partially funded by MIUR in the PLATINO (PON01_01007) and SMART HEALTH (PON04a2.C) projects.

References

1. SamKnows, <http://www.samknows.com/broadband/>.
2. K. Flamm, A. Friedlander, J. Horrigan, and W. Lehr, *Measuring broadband: Improving communications policymaking through better data collection*. Pew Internet & American Life Project, 2007.
3. K. Cho, K. Fukuda, H. Esaki, and A. Kato, “The impact and implications of the growth in residential user-to-user traffic,” in *ACM SIGCOM*, 2006.
4. M. Siekkinen *et al.*, “Performance limitations of adsl users: A case study,” in *the Passive and Active Measurement Conference (PAM)*, 2007.
5. G. Maier *et al.*, “On dominant characteristics of residential broadband internet traffic,” in *ACM Internet Measurement Conference*, 2009.
6. M. Dischinger, A. Haeberlen, K. P. Gummadi, and S. Saroiu, “Characterizing residential broadband networks,” in *Proc. ACM SIGCOMM Internet Measurement Conference*, San Diego, CA, USA, Oct. 2007.

7. D. Croce, T. En-Najjary, G. Urvoy-Keller, and E. Biersack, "Capacity estimation of adsl links," in *CoNEXT*, 2008.
8. "Youtube," http://www.youtube.com/my_speed.
9. S. Sundaresan, W. de Donato, N. Feamster, R. Teixeira, S. Crawford, and A. Pescapè, "Measuring home broadband performance," *Communications of the ACM*, vol. 55, no. 11, pp. 100–109, 2012.
10. Grenouille, "Grenouille," <http://www.grenouille.com/>.
11. D. Han *et al.*, "Mark-and-sweep: Getting the inside scoop on neighborhood networks," in *Proc. Internet Measurement Conference*, Vouliagmeni, Greece, Oct. 2008.
12. G. Bernardi and M. K. Marina, "Bsense: a system for enabling automated broadband census: short paper," in *Proc. of the 4th ACM Workshop on Networked Systems for Developing Regions (NSDR '10)*, 2010.
13. "Neubot," <http://www.neubot.org>.
14. "Ne.me.sys." <https://www.misurainternet.it/nemesys.php>.
15. D. Joumblatt *et al.*, "Characterizing end-host application performance across multiple networking environments," in *INFOCOM, 2012 Proceedings IEEE*. IEEE, 2012, pp. 2536–2540.
16. "Speedtest.net," <http://www.speedtest.net>.
17. C. Kreibich, N. Weaver, B. Nechaev, and V. Paxson, "Netalyzr: Illuminating the edge network," in *Proc. Internet Measurement Conference*, Melbourne, Australia, Nov. 2010.
18. "Glasnost: Bringing Transparency to the Internet," <http://broadband.mpi-sws.mpg.de/transparency>.
19. W. de Donato, "Large scale benchmarking of broadband access networks: Issues, methodologies, and solutions," *PhD dissertation, University of Napoli Federico II*, 2011.
20. J.-S. Lee and B. Hoh, "Dynamic pricing incentive for participatory sensing," *Pervasive and Mobile Computing*, vol. 6, no. 6, 2010.
21. D. R. Choffnes, F. E. Bustamante, and Z. Ge, "Crowdsourcing service-level network event monitoring," in *ACM SIGCOMM Computer Communication Review*, vol. 40, no. 4. ACM, 2010, pp. 387–398.
22. A. Botta, A. Dainotti, and A. Pescapè, "A tool for the generation of realistic network workload for emerging networking scenarios," *Computer Networks*, vol. 56, no. 15, pp. 3531–3547, 2012.
23. G. Aceto *et al.*, "User-side approach for censorship detection: home-router and client-based platforms," *Connaught Summer Institute on Monitoring Internet Openness and Rights, University of Toronto*, 2013.