Open Source Platforms for Internet Monitoring and Measurement

Giuseppe Aceto, Alessio Botta, Walter de Donato, Pietro Marchetta, Antonio Pescapé, Giorgio Ventre
University of Napoli "Federico II" (Italy)
{*giuseppe.aceto,a.botta,walter.dedonato,pietro.marchetta,pescape,giorgio*}*@unina.it*

***Abstract*— Understanding the ever-changing scenario of computer networks and how they operate in the real world implies measuring and analyzing their characteristics. This in turn requires a set of advanced tools and methodologies to be shared among researches, along with the data derived from such activities. In this paper we first present some of the main issues and challenges in the field of Internet Monitoring and Measurement, then we present several open source platforms we have developed in the last 10 years for monitoring heterogeneous and large scale networks. Finally, we describe some of the data sets we made publicly available to the research community.**

I. Introduction

INTERNET measurements are fundamental for the networking research community. Through them, we acquire more knowledge on the incredibly complex and ever-changing system the Internet is. Through traffic analysis, topology discovery, study of routing dynamics, and measurement of links capacity, delay, etc. we can better understand how this system behaves and how it is evolving. The results of such activities lead to better control and optimization of current and to the design of future network architectures. The analysis of measurements has proved invaluable not only for traditional application fields like performance evaluation, but also for many others. In last years, for example, there has been a boost of network security research based on Internet measurements.

For research results to be significant and reliable, we need repeatable experiments, large shared data sets, comparable measures, and confirmation on more observation points and at different times of the properties discovered [1]; moreover, we need accurate, up-to-date, and sanitized data. This last point is not always possible. On the one side, there are issues related to the quality of the measurements: what to measure and what methodology to use. For example, some traffic analyses simply need flow-level data, or they are not significantly affected by packet sampling, while other kinds of studies might be. On the other side, two major obstacles to the large availability of data are related to the security/privacy and costs. Service providers do not want to disclose details on their networks (and on how such networks perform), and they need to carefully preserve their customers privacy. Furthermore, the costs to build and manage a proper measurement infrastructure and to make reliable and anonymized data available to the public are often very high.

However, the need for availability of measurement data to all researchers is indisputable [2]. Recently, there have been several discussions on the development of a large-scale community-oriented measurement infrastructure [3]. A lot of benefits would come from the adoption of a common framework, and a large-scale distributed infrastructure would yield up-to-date, statistically relevant, and comparable data. Also, costs would be much more affordable and clearly funded. However, the discussion of this matter generates more questions than answers [3]. In the mean time, several efforts to provide measurement data and tools are being made by different members of the research community . Some notable examples are represented by *CAIDA* [4], *NLANR* [5], and *MAWI-WIDE* [6] projects which cover a broad scope, or by more specific projects as *Crawdad* [7], which is devoted to build a community for wireless network measurements. Smaller projects from other research groups [8] also make useful data available.

It has been noted that the networking research community is relatively younger than the ones from other scientific disciplines, as biology and astronomy, which have partially solved the problem of coordinatively sharing experimental data (sometimes with privacy issues involved as well). This is a big challenge we still have to face. For example, which tools must be used to perform active network measurements? As regards passive measurements, some recent proposals argue that it would be much simpler to make ISPs and link administrators keep the data private, but allowing researchers to run tools on them to extract data innocuous with respect to privacy and security. But, which data (and how) must be collected [9] and archived? Moreover, measurement data often needs *sanitization*, which means removing outliers and anomalies caused by spurious data and errors in the various (hardware and software) processing phases. How to solve problems related to the large amount of data to manage? Several sampling techniques have been developed, but their impact on different kinds of analysis (e.g. for anomaly detection) is still under investigation. While, alternative techniques to reduce the amount of data resulting from measurements are being proposed [10], [11], [12].

While looking for general consensus on well-established procedures, the research community in parallel develops new measurement methodologies and techniques. New tools are proposed, and evaluated, and old ones are improved. Also, as the time goes by, we deepen our knowledge on the various aspects related to data collection and analysis (as coping with large data sets, data sanitization, etc.). Further, the Internet is a moving target, causing new measurement requirements to continually arise.

In this article, we briefly give an overview of our contribution to the community regarding some of these aspects. First, with respect to traffic analysis and characterization, our contribution is methodological, because it is related to specific approaches to what to measure and how. Moreover, in our work in this field, we have faced several of the cited issues, and sometimes we came up with original solutions. Second, we have developed, and we are still expanding, a set of platforms for network measurements (both active and passive) and for the processing of measurement data sets. Third, we have started a measurement and data collection framework to make the measurement, obtained by using our tools, publicly available. In the next section we will discuss all these points.

More precisely, in this work after discussing some of the main issues and challenges in the field of Internet Monitoring and Measurement, we present several open source platforms we have developed in the last 10 years for monitoring heterogeneous and large scale networks. Due to space constraints, we just briefly discuss the main features of each platform and we point the reader to a large number of our references in which we have deeply discussed and tested our platforms and their results over real networks. In summary, this paper represents a sort of survey of our work during the last decade in the field of Internet Monitoring and Measurement and of our contributions to the research community in terms of both platforms and data.

II. Tools and Data

A. Traffic Measurement and Analysis at Packet-level

Network traffic can be observed at different abstraction levels. It is possible to study aggregate traffic or, for example, to separate it into conversations, connections, flows, or packets. While still adopting a multi-level analysis approach, in our studies we focused our attention on packet-level. Packet-level traffic characterizations express traffic in terms of *inter-packet time* (IPT) and *packet size* (PS). There are several important advantages in such approach. First of all, it is very straightforward and concise. We do not need to make any assumptions regarding the kind of applications generating traffic, and the same methodology is easily extensible to study different application-level protocols and mixes of them. Moreover, observing traffic at packet-level allows to work at the deepest point of view. The results of the analysis and modeling can be applied in several contexts. Switching devices often operate on a packet-by-packet basis, and most network performance problems (e.g. delay, jitter, loss) happen at packet level. Packet-level models are also easily applicable to traffic emulation and simulation, which can be used to study network-related issues (measuring delay, jitter, packet loss etc.) or to test network equipment. Traffic at packet level remains observable after encryption made by, for example, end-to-end cryptographic protocols such as SSL or IPSec; this makes packet-level characterization and modeling robust approaches to traffic profiling for anomaly detection.

As regards traffic characterization, we applied this approach in the study of traffic of different applications (HTTP, SMTP, IM, Worms, network games) and it showed some invariant properties with respect to time (*time invariance*) and to the observed network (*space invariance*), when sampling large and highly heterogeneous populations of clients and servers (to make our results partially independent from both network conditions and end-to-end congestion control) [13], [14], [15]. We are still expanding the categories of traffic to study, and we are experimenting on the usefulness of packet-level characterization for traffic fingerprinting and profiling, with possible applications in the context of classification and network security. As for traffic modeling, in [16] we developed a packet-level model based on Hidden Markov Models (HMM) which jointly models IPT and PS, taking into account mutual dependencies and time structure (by means of autocovariance). Preliminary results showed that the model has also interesting prediction capabilities.

B. Open Monitoring and Measurement Platforms

1) ***Plab****:* In the study of network traffic we felt the need for a measurement tool allowing to focus the analysis on packet-level statistics, while still being able to look at network traffic at different levels. Thus to capture and analyze traffic we developed Plab, a software platform written in C, based on the Libpcap library [17], running on FreeBSD, Linux, and MacOS-X. Capturing live traffic or analyzing trace files in *tcpdump* format, Plab is able to split traffic into different kinds of *sessions*. Depending on user-specified parameters, a session can be identified by:

- all packets sent and received by a host (*host mode*);
- all packets identified by source and destination IP addresses and ports with a default timeout of 60 seconds (*flow mode*);
- all packets exchanged by 2 hosts related to a specific service (e.g. TCP port 80), with a user definable timeout (*conversation mode*).

Given one of the above modes, sessions are assigned an ID, and for each session the IPT between packets flowing in the same direction and the PS os such packets are calculated. We call such data *packet-level* data series. Moreover, higher-level measures related to the sessions are stored, like the arrival time of each session, its duration, packet and bytes transmitted for each direction, etc. IPT and PS looking at the aggregate traffic as a whole are also calculated. In addition, many processing and filtering capabilities have been implemented, as the ability to decode optional TCP headers as the MSS, or to filter packets or entire sessions based on several criteria. Berkeley Packet Filter syntax (tcpdump-style) is supported at the end of the command line. We added also specific features which are useful for data sanitization. In our studies, we discovered spurious data probably due to hardware/software in some publicly available traffic traces. Among them, for example, we found full chunks of packet sequences duplicated inside the traces. Checks based on packets timestamps inconsistencies, allow to remove such duplicates. Data sets extracted by Plab are dumped into text files which can be directly imported under software environments for time series processing and statistical analysis (e.g. Matlab [18], R [19]). Plab is open-source software and

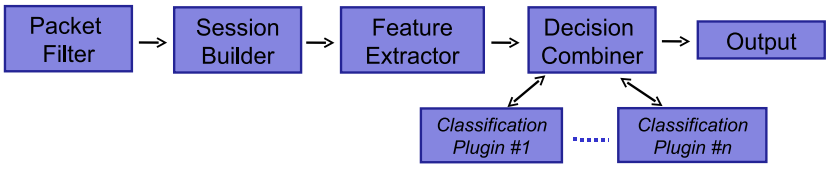

Fig. 1. TIE overall architecture (source [27]).

is available at our web site [20].

*2) **Statistical Library**:* Once measurement data has been collected, several approaches can be adopted in the analysis of network traffic and Internet measurements in general. Besides more general techniques of statistical analysis, which look at processes from a stochastic point of view by studying marginal distributions, covariances, scaling properties, etc., techniques from the fields of Signal and Image Processing and from Information Theory are successfully used to investigate Internet phenomena and to highlight interesting behaviors. In our studies, we usually work under the Matlab and Octave [21] environments. In this context, in addition to tools made available by the research community, we developed a number of Matlab functions useful for statistical analysis. Most of them have been written with the explicit purpose to ease the analysis of time series extracted from network traffic traces and active measurements. There are tools to extract, display (also by means of diagrams), and compare statistical properties, or for example, to build packet and byte rate time series sampled with different periods starting from IPT and PS. We also developed scripts which implement an entropy-based technique to heavily reduce large data sets while preserving main statistical properties. Tools for statistical fitting, based on Expectation Maximization, can be used as an aid to build analytical models starting from empirical data. They can then be applied, for example, into traffic emulation contexts (see next subsection). More details on the Statistical Library are available at its web site [22], from where it can be freely downloaded.

*3) **TIE**:* Traffic Classification is a hot topic in Internet research and it needs open platforms for experimenting with new algorithms and approaches [23], [24], [25], [26]. A large quantity of works have been published in the past few years on traffic classification. However, besides classifiers based on trasport-layer ports and on payload inspection, there are few implementations made available to the community that target alternative approaches. TIE (Traffic Identification Engine) is a platform for experimenting with and comparing traffic classification techniques. Its overall architecture is shown in Figure 1. As described in [27], the Packet Filter is able to both capture live traffic or read from a traffic trace, and it can filter packets depending on several criteria. Packets are then aggregated into separate sessions (flows, biflows, etc.) by the Session Builder, which keeps the status of each session updated. A set of feature extraction routines (e.g. updating statistics on inter-packet times) are performed by the Feature Extractor. The classification is performed by the Decision Combiner, which coordinates the activities of several classification plugins (each one executing a different classification technique). The Output generates final output files with modalities and data formats that depend on the operating mode. TIE allows the development of algorithms implementing different classification techniques as *classification plugins* that are plugged into a unified framework, allowing their comparison and combination [27]. The main features of TIE are the following: (i) it allows to currently run and combine multiple classification techniques; (ii) it has been designed to work also with live traffic (realtime mode); (iii) it is open source, it has a large community participation, geared towards data/tools sharing (developer's documentation and fully functional examples, classification plugins API, easy to add extraction of new classification features, easy to add new decision combination strategies, easy to provide ground-truth along with anonymized traces); (iv) it provides graph and web reports (see Figure 2 for one of the TIE graphical outputs); (v) it runs on Linux, FreeBSD, and MacOS X. To date, several classification techniques have already been implemented as TIE plugins [29], [30], [31], [32]. In literature there are other similar platforms, such as NeTraMark[28].

*4) **D-ITG**:* Approaches and systems for network workload generation are useful and effective only when they produce network workload that is realistic [33], [34], [35], [36], [37] and representative as much as possible of the real workload of the network scenario under study. Distributed Internet Traffic Generator (D-ITG) is a platform capable to produce IPv4 and IPv6 traffic by accurately replicating the workload of current Internet applications [38], [39], [40], [41]. Two main alternative approaches exist in literature for the generation of such workload: (i) *trace-based* generation (TCPReplay, TCPivo, TCPopera, etc.), in which flows exactly replicate the content and the timings of traffic traces previously collected in real scenarios; and (ii) *analytical model-based* generation (TG, MGEN, RUDE/CRUDE, D-ITG, etc.), in which flow and packet generation processes are based on statistical models. With D-ITG, the generation can either follow *simple stochastic models* for packet size (PS) and inter departure time (IDT), or *more complex analytical models* that mimic application-level protocol behavior, or instead it can follow *real traffic patterns captured in traffic traces*. In the first case, by specifying the distributions of IDT and PS random variables, it is possible to choose different renewal processes for packet generation: by using characterization and modeling results from literature and from our analysis (Sections II-B.1, II-B.2), D-ITG is able to replicate statistical properties of traffic of different well-known

applications (e.g Telnet, VoIP - G.711, G.723, G.729, Voice Activity Detection, Compressed RTP - DNS, network games). In the second case, D-ITG adopts the modeling approach presented in [16]. Specifically, customly pre-configured Hidden Markov Models (HMM) are used to reproduce the behavior of real traffic sources and then for the synthetic generation of their workload. The HMM reproduces different states in which a single source can be, and is used to associate to each state different statistical profiles of both PS and IDT, while state transitions happen with probabilities defined by a transition matrix. In the third case, D-ITG allows to replicate real traffic traces and to fully configure, at the same time, a number of parameters at layer 3 (e.g. the IP addresses), layer 4 (e.g. the transport layer ports), and at the application layer (e.g. the information in the SIP/SDP header). Beside the generation mode, at the transport layer, D-ITG currently support TCP (Transmission Control Protocol), UDP (User Datagram Protocol), SCTP (Stream Control Transmission Protocol), and DCCP (Datagram Congestion Control Protocol). Moreover, it supports ICMP (Internet Control Message Protocol) and allows to set the TOS (DS) and TTL IP header fields.

D-ITG overall architecture is shown in Figure 3. As described in [41], the main module is called sender/receiver. It can act as a traffic sender, receiver, or both (it is worth noticing that our platform can work with multiple sender/receiver instances, multiple senders can send to a single receiver, and a single sender can send to multiple receivers). The measurement information can be saved directly by the sender/receivers, or it can be sent - through the network - to a module called logger (useful to collect all the measures in a single point or in the case of hosts with limited storage capabilities e.g., sensors, smartphones, etc.). To cope with large scale experiments, the sender/receiver modules can be controlled directly by the user, or by another module called controller, which receives input from the user and interacts with the senders/receivers in order to orchestrate the measurements. This way, the user can completely control a large-scale distributed experiment from a single vantage point. A signaling channel between the main modules of the architecture (controller, sender/receiver, and logger) is used for the configuration, management and synchronization of the experiments. A module called analyzer is in charge of analyzing the results of the experiments both on-line and off-line extracting performance measures experienced by the probing traffic. Besides being a traffic generator, D-ITG can be used as an active measurement tool: one-way-delay (OWD), round-trip-time (RTT), packet loss rate, jitter, and throughput can be measured and analyzed using the various components of the D-ITG platform (sender, receiver, log server, and decoder). The support of different computing architectures (ARM, Intel XScale, x86@32bit and 64bit) and operating systems (Windows, OSX, FreeBSD, Linux, Montavista Linux, OpenWRT, snapgear, Linux Familiar) allows to use the distributed platform for experimentation in complex and heterogeneous environments. As for the experiment repeatability, D-ITG allows to use the same seed of the random number generator, to perform identical experiments on the same scenario or to study different scenarios under the same traffic generation conditions. D-ITG is able to reach high (receiver and sender) data rates. More precisely, with two Linux boxes connected with a Gb Ethernet the maximum bit rate achieved is higher than 950 Mbps and the packet rate is higher than 850 Kpps.

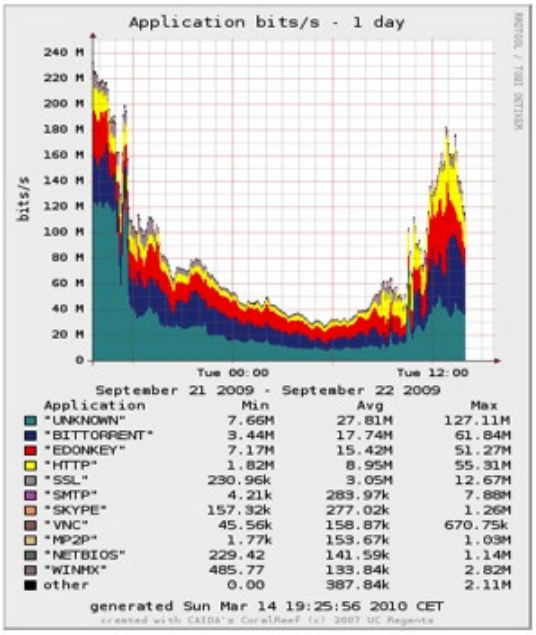

Fig. 2. TIE graphical output.

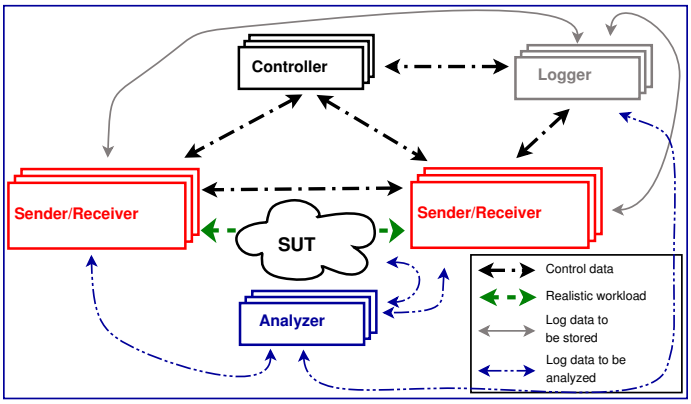

Fig. 3. D-ITG overall architecture (source [41]).

D-ITG has often been used to evaluate the performance of wireless networked systems. More precisely, in [42], [43] it has been used to analyze the performance of heterogeneous wireless networks, considering handoffs too. In [44] it has been used to correctly assess the capacity of wireless links. In [45], [46] it has been used to evaluate the performance achieved by a metropolitan wireless network in the Berlin area. In [47] it has been used to asses the impact of middleboxes on the performance of wireless networks (3G and Satellite networks). Furthermore, data collected using D-ITG over real heterogeneous wireless networks has been used to derive the model presented in [48]. Finally, D-ITG has been used to evaluate the performance of transport protocols (e.g., SCTP) [49] and of

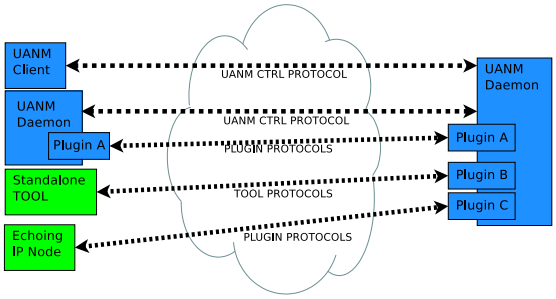

Fig. 4. UANM overall architecture (source [66], [67]).

networked embedded systems (e.g., network processors) [50].

An open-source multi-thread implementation of D-ITG is currently released at [51] for all the supported platforms. Other similar platforms are OSTINATO [52], RUDE/CRUDE [53], TG [54], MGEN [55], BRUTE [56], Iperf [57], TCPreplay [58], Swing [59], etc.

*5) **UANM**:* Among network path characteristics, the available bandwidth (also know as the unused capacity) is of notable interest for applications of traffic engineering, media streaming, network monitoring, etc., But its estimation poses significant practical issues, mainly related to dynamical nature of the available bandwidth and to the peculiarities of the active estimation techniques that have been proposed for it [60], [61], [62], [63], [64], [65]. In order to perform comparative analysis of known and future estimation techniques, as well as to provide a mean to automatically select and setup the best tool for varying measurement scenarios, we have designed and implemented UANM (Unified Platform for Network Measurement): a platform allowing the automated management of end-to-end available bandwidth estimation tools [66], [67]. UANM is capable of controlling the execution of the different techniques and benchmarking them in a fair environment; the techniques are implemented as one or more *plugins* (dynamically loadable modules) that expose to the experimenter a single coherent API (*plugin API*). The *plugin API* is offered to the developer and the experimenter, and abstracts the management of the measurement activity, comprising the setup of the control channel between the sender and the receiver, the setting of the measurement parameters, the actual measurement process, and the reporting of the outcome. The platform takes care of automating all those phases, also preventing mutual interference of concurrent measurements, and provides the user (a network manager, or third party software) with a simplified and goal-driven API (*user API*).

UANM overall architecture can be seen in Figure 4, which shows the components of the platform: (i) peer-to-peer network of measurement servers (*UANM Daemons*) that host and manage the *plugins*; (ii) clients that issue measurement requests and receive results from the measurement servers; and (iii) optionally third party applications that are engaged in measurements by the *plugins*. The *user API* for requesting the measurements is provided as a library to be included in external software, offering the knowledge of the estimated available bandwidth for applications of network monitoring, media streaming, server selection, etc.

Other platforms similar to UANM in some of its goals are NetQuest [68], FLAME [69]. The issue of calibration of available bandwidth estimation tools has been investigated in [65], introducing a new tool, YAZ, while in [62] a kernel-based implementation of an available bandwidth estimation tool, DIChirp, is proposed as a solution to improve the estimation accuracy. Scenario-specific techniques have also been proposed, such as WBest [63] for wireless networks, and Traceband [64] for low-overhead continuous monitoring.

More details about the platform are available at [70].

*6) **Hynetd**:* In most situations, active and passive measurements can not ignore the network topology. An accurate and complete knowledge of the network topology is crucial for fault diagnosis, performance evaluation, routing troubleshooting, modeling, simulation and analysis [71], [1]. Anyway, as largely demonstrated in literature [72], [73], [74], [75], [76], inferring the network topology is an extremely complex task and requires sophisticated solutions. Hynetd [77], [78] (Hybrid Network Topology Discovery) is a tool based on a hybrid methodology that effectively combines active and passive approaches to discover network topologies at router level, starting from the IP addressing space to be explored and a set of SNMP community names. The adopted approach aims at obtaining a high level of completeness and accuracy and at improving the efficiency of the discovery process (both in terms of discovery duration and traffic overhead). Hynetd achieves such goal by using (i)

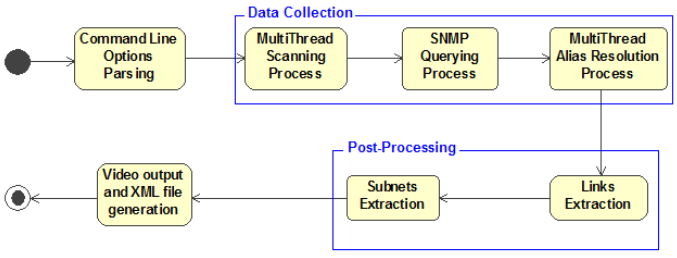

Fig. 5. Hynetd overall architecture (source [78]).

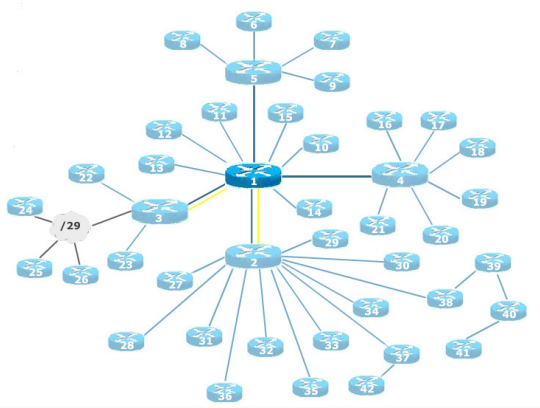

Fig. 6. Hynetd graphical output.

an algorithm named Backtrace, that efficiently implements the execution of many concurrent Traceroutes; (ii) an approach for IP alias resolution using Ping with Record Route IP option; (iii) some rules to reduce the number of IP pairs involved in the Ally algorithm [79]; (iv) an heuristic to recognize serial links in some specific conditions. Finally, as shown in Figure 5, by adopting a multi-step data collection phase involving highly multi-threaded operations, it is able to recover most of the time lost when waiting for the expiration of timeouts. Hynetd runs under the Linux operating system and has been released under the GPL terms [80]. Figure 6 reports one of its graphical outputs. Other similar platforms are [81], [82].

*7) **Hobbit** & **BISmark**:* Most people rely on Internet connectivity for everyday activities, thus making broadband access an essential resource. Although several tools to measure Internet performance exist [83], [84], [85], benchmarking the performance of broadband access networks is not as simple as running one-time "speed tests". *HoBBIT* and *BISmark* are two complementary platforms which address the measurement of access networks by adopting a similar architecture made of three components (see Figure 7): (i) a *management server*, whose role is to orchestrate all the experiments and to collect and organize all their results; (ii) multiple *measurement clients*, which run inside the home network and periodically execute experiments; (iii) several *measurement servers*, which are responsible for supporting the measurements conducted by the clients when they require a specific server-side component.

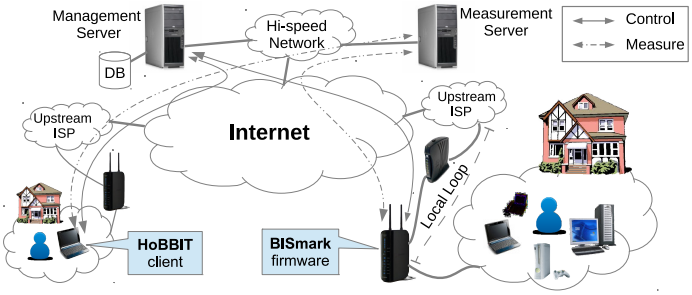

Fig. 7. BISmark and HoBBIT: common architecture overview.

HoBBIT (Host Based Broadband Internet Telemetry) is a platform capable of conducting experiments on broadband access networks by properly instructing measurement clients running on users' personal computers [86]. It provides a flexible framework to define experimental campaigns with a specific goal and a detailed schedule of the measurements requested. It further allows to reuse pre-existing measurement tools. All the performed measurements are collected and aggregated at different levels to produce both maps (see Figure 8 where the results of upstream throughput are graphically shown on the Italian geographical map) and plots, which are available to the users through the project website [87].
Other similar platforms are [88], [89], [90].

BISmark (Broadband Internet Service Benchmark) is a project led by Georgia Tech and the University of Napoli Federico II to develop an OpenWRT-based platform for running measurements of ISP performance, as well as traffic inside the home [91]. A BISmark router is capable of performing both active and passive measurements from a vantage point between the home network and the access ISP. This design offers an unobstructed view of both the ISP and the home networks. Moreover, the ability to run continuous measurements allows to account for many of the confounding factors that affect other measurement studies. The BISmark platform is publicly available and released under the GPL license [92].
The only other similar platform is [93].

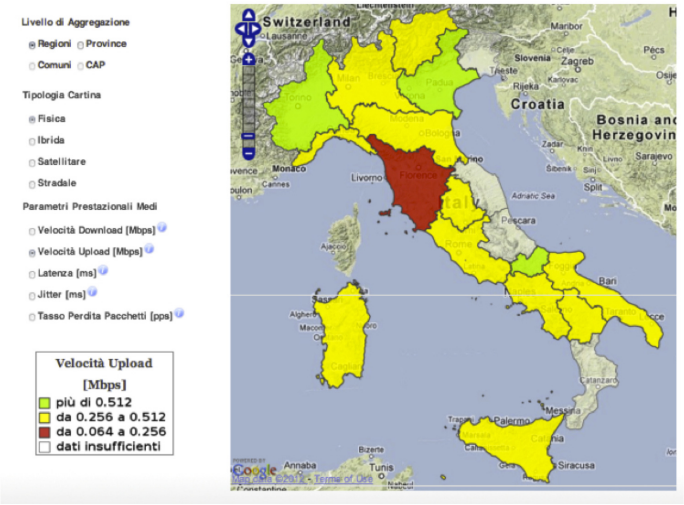

Fig. 8. HoBBIT geolocated-based graphical output.

C. Data

In this subsection we give a very brief overview of the measurement data we started to make available [94] to the community.

1) Traffic Traces: As regards traffic traces, we used Plab and Tcpdump to passively capture traffic from a link running at about 200 Mbps and connecting our University network (the *UNINA* network) to the rest of the Internet. The *UNINA* network has a main /16 address space plus few /24 networks. Using Plab, we were able to capture full layer-4 headers (e.g. TCP optional headers) without storing payload data for privacy issues. Moreover IP addresses were anonymized, preserving subnet membership, using the *tcpdpriv* tool from MAWI-WIDE. Besides trace files, we also make available time series of IPT and PS extracted from traffic. Some of this data, being related to aggregate traffic captured in absence of network anomalies, has also been used for anomaly detection studies.

2) Data Traces: Data traces related to measurements of delay, jitter, packet loss, and throughput, have been obtained by using D-ITG performing active measurements in various heterogeneous small-scale networks and in a large scale wireless network. With the term heterogeneous we mean different mixing of:

- access network technologies, (Ethernet, 802.11, UMTS, GPRS, ADSL, etc.) and intermediate links;

- transport and control protocols (TCP, UDP, SCTP, DCCP, ICMP);
- end-point hardware (palm-tops, smart phones, laptops, workstations);
- end-point operating systems.

This heterogeneity is formalized through the concept of "service condition" introduced in [96], [97]. A clear understanding and reference for Quality of Service (QoS) parameters like delay, jitter, packet loss, and throughput is of paramount importance in several scenarios ad for Quality of Experience (QoE) evaluation [98], [99]. Also, thanks to a past joint research work [95] with the Network Group of the Deutsche Telekom Laboratories (Berlin), we collected measurements over a large-scale wireless network sited in Berlin: *MagNets* multi-hop backbone running on 802.11a/b/g.

III. Conclusion

In this paper we have discussed some of the main issues and challenges in the field of Internet Monitoring and Measurement and we have presented several open source platforms we have developed for monitoring heterogeneous and large scale networks. In our ongoing work, to follow the continuously changing nature of Internet, we are improving each of the presented platforms. For example, we are extending them for their use in scenarios like Cloud Computing, Software Defined Networks, Mobile and Ubiquitous applications.

IV. Acknowledgements

This work has been partially funded by LINCE project of the FARO programme jointly financed by the Compagnia di San Paolo and by the Polo delle Scienze e delle Tecnologie of the University of Napoli Federico II and by PLATINO, a MIUR project in the framework of the PON action.

References

[1] S. Floyd, V. Paxson, "Difficulties in simulating the Internet", *IEEE/ACM Transactions on Networking*, Vol. 9 , Issue 4, pp. 392 - 403, August 2001.
[2] N. Brownlee, KC Claffy, "Internet Measurement" *IEEE Internet Computing* Vol. 8, Issue 5, pp. 30 - 33, September 2004.
[3] KC claffy, M. Crovella, T. Friedman, C. Shannon, N. Spring, "Community-oriented network measurement infrastructure (CONMI) workshop report", *ACM SIGCOMM Computer Communication Review*, Vol. 36, Issue 2, pp. 41-48, April 2006.
[4] CAIDA. `http://www.caida.org`.
[5] NALNR. `http://www.nlanr.org`.
[6] Mawi. `http://tracer.csl.sony.co.jp/mawi`.
[7] Crawdad. `http://crawdad.cs.dartmouth.edu`.
[8] M. Faloutsos, "Public real data repositories and measurement tools", *ACM SIGCOMM Computer Communication Review archive* Vol. 36, Issue 2, pp. 37 - 40, April 2006.
[9] G. Varghese, C. Estan, "The measurement manifesto", *ACM SIGCOMM Computer Communication Review*, Vol. 34, Issue 1, pp. 9 - 14, January 2004.
[10] A. Pescapé," Entropy-Based Reduction of Traffic Data", IEEE Communications Letters, pp. 191-193, Vol.11, No.2 - February 2007.
[11] D. Tammaro, S. Valenti, D. Rossi, A. Pescapé, "Exploiting packet sampling measurements for traffic characterization and classification", International Journal of Network Management, 2012.
[12] G. Aceto, A. Botta, A. Pescapé, C. Westphal "An efficient storage technique for network monitoring data", IEEE International Workshop on Measurements & Networking (M&N 2011), Capri, Italy, October 2011.
[13] A. Dainotti, A. Pescapé, G. Ventre, "A Packet-level Characterization of Network Traffic", *11th IEEE International Workshop on Computer-Aided Modeling, Analysis and Design of Communication Links and Networks (CAMAD 2006).*
[14] Alberto Dainotti, Antonio Pescapé, Giorgio Ventre, "Worm Traffic Analysis and Characterization", 2007 IEEE International Conference on Communications (ICC 2007).
[15] A. Botta, A. Dainotti, A. Pescapé, G. Ventre, "Searching for Invariants in Network Games Traffic", Poster at Co-Next 2006 Student Workshop. 2-pages abstract published in Co-Next '06 Proceedings.
[16] A. Dainotti, A. Pescapé, P. Salvo Rossi, G. Iannello, F. Palmieri, G. Ventre, "An HMM Approach to Internet Traffic Modeling", *2006 IEEE GLOBECOM, Quality, Reliability, and Performance Modeling for Emerging Network Services Symposium.*
[17] tcpdump. `http://www.tcpdump.org`.
[18] Matlab.`http://www.mathworks.com`.
[19] R project. `http://www.r-project.org`.
[20] PLab. `http://www.grid.unina.it/software/Plab`.
[21] Octave. `http://www.octave.org`.
[22] Statistical Tools. `http://www.grid.unina.it/Traffic/Tools/statools.php`.
[23] A. Callado et al., "A Survey on Internet Traffic Identification", IEEE Commun. Surveys & Tutorials, vol. 11, no. 3, July 2009.
[24] T. T. T. Nguyen and G. Armitage, "A Survey of Techniques for Internet Traffic Classification Using Machine Learning," IEEE Commun. Surveys & Tutori- als, vol. 10, no. 4, 2008, pp. 56-76.
[25] A. Dainotti, A. Pescapé, KC Claffy, "Issues and Future Directions in Traffic Classification", IEEE Network, January 2012.
[26] M. Mellia, A. Pescapé, L. Salgarelli, "Traffic classification and its applications to modern networks", Computer Networks, Volume 53, Issue 6, 23 April 2009, Pages 759-760.
[27] A. Dainotti, W. De Donato, A. Pescapé, "TIE: a Community-Oriented Traffic Classification Platform", International Workshop on Traffic Monitoring and Analysis (TMA'09) @ IFIP Networking 2009 - May 2009, Aachen (Germany).
[28] Suchul Lee, Hyunchul Kim, Dhiman Barman, Sungryoul Lee, Chongkwon Kim, Ted Kwon, and Yanghee Choi "NeTraMark: a network traffic classification benchmark", SIGCOMM Comput. Commun. Rev. 41, 1 22-30.
[29] CoralReef. `http://www.caida.org/tools/measurement/coralreef/`.
[30] J. Levandoski, E. Sommer, and M. Strait, "Application Layer Packet Classifier for Linux." `http://l7-filter.sourceforge.net/`.
[31] G. Aceto, A. Dainotti, W. de Donato, A. Pescapé, "PortLoad: taking the best of two worlds in traffic classification", IEEE INFOCOM 2010 Work in Progress (WiP), San Diego, CA, USA, March 15-19, 2010.
[32] V. Carela-Español, P. Barlet-Ros, M. Solè-Simò, A. Dainotti, W. de Donato, A. Pescapè, "K-dimensional trees for continuous traffic classification", 2nd International Workshop on Traffic Monitoring and Analysis (TMA'10), Zurich, Switzerland, April 7, 2010.
[33] D. Ferrari, "On the Foundation of Artificial Workload Design," *Proc. ACM SIGMETRICS*, pp. 8-14, 1984.
[34] O. Shurrab, J. Pagna Disso, I. Awan, "A Realistic Approach to Background Traffic Generator", 27th Annual UK Performance Engineering Workshop 7-8 July 2011.
[35] M.C. Weigle, P. Adurthi, F. Hernandez-Campos, K. Jeffay, and F.D. Smith. "Tmix: A Tool for Generating Realistic TCP Application Workloads in ns-2", *ACM SIGCOMM Computer Communication Review (CCR)*, July 2006, Vol 36, No 3, pp. 67-76.
[36] P. Barford, M. Crovella, "Generating representative Web workloads for network and server performance evaluation", *SIGMETRICS Perform. Eval. Rev.* vol. 26 , Issue 1, pp. 151-160, June 1998.
[37] Y. Choi, J. A. Silvester, H. Kim, "Analyzing and Modeling Workload Characteristics in a Multiservice IP Network", IEEE Internet Computing, pp. 35-42, March/April, 2011.
[38] S. Avallone, A. Pescapé, G. Ventre, "Distributed Internet Traffic Generator (D-ITG): analysis and experimentation over heterogeneous networks", Poster at International Conference on Network Protocols, ICNP 2003 November 2003, Atlanta - Georgia (USA).
[39] D. Emma, A. Pescapé, G. Ventre, "Analysis and experimentation of an open distributed platform for synthetic traffic generation", 10th IEEE International Workshop on Future Trends of Distributed Computing Systems (FTDCS 2004), pp. 277-283, May 2004, Suzhou (China).
[40] A. Botta, A. Dainotti, A. Pescapé, "Do You Trust Your Software-based Traffic Generator?", IEEE Communications Magazine, vol.48, no.9, pp.158-165, Sept. 2010.

[41] A. Dainotti, A. Botta, A. Pescapé, "A tool for the generation of realistic network workload for emerging networking scenarios", Computer Networks (Elsevier), Volume 56, Issue 15, 15 October 2012, Pages 3531-3547.

[42] M. Bernaschi, F. Cacace, A. Pescapé, S. Za, "Analysis and Experimentation over Heterogeneous Wireless Networks", First IEEE International Conference on Testbeds and Research Infrastructures for the DEvelopment of NeTworks and COMmunities (TRIDENTCOM'05), February, 2005, Trento (Italy).

[43] G. Iannello, A. Pescapé, G. Ventre, L. Vollero, "Experimental analysis of heterogeneous wireless networks", WWIC 2004, Wired/Wireless Internet Communications 2004 - LNCS vol. n. 2957, ISBN: 3-540-20954-9, pp. 153 - 164 - February 2004, Frankfurt (Germany).

[44] L. Angrisani, A. Botta, A. Pescapé, M. Vadursi, "Measuring Wireless Links Capacity", IEEE 1st International Symposium on Wireless Pervasive Computing, 2006 - 16-18 Jan. 2006, pp. 1 - 5.

[45] R. Karrer, I. Matyasovszki, A. Botta, A. Pescapé, "Experimental evaluation and characterization of the magnets wireless backbone", ACM WINTECH 2006, pp. 26-33.

[46] R. Karrer, I. Matyasovszki, A. Botta, A. Pescapé, "MagNets: experiences from deploying a joint research-operational next-generation wireless access network testbed", In Proc. of 3rd International Conference on Testbeds and Research Infrastructures (TridentCom), Orlando, FL, May 2007.

[47] A. Botta, A. Pescapé, "Monitoring and measuring wireless network performance in the presence of middleboxes", The 8th International Conference on Wireless On-demand Network Systems and Services (WONS), Bardonecchia (TO), Italy, January 2010.

[48] G. Iannello, F. Palmieri, A. Pescapé, P. Salvo Rossi, "End-to-End Packet-Channel Bayesian Model applied to Heterogeneous Wireless Networks", IEEE Globecom 2005 General Conference - ISBN 0-7803-9415-1 - December 2005, St. Louis (MO, USA).

[49] A. Dainotti, S. Loreto, A. Pescapé, G. Ventre, "SCTP performance evaluation over Heterogeneous Networks", Concurrency and Computation: Practice and Experience (Wiley) , Vol 19, Issue 8 (June 2007), pp. 1207 - 1218.

[50] A. Botta, W. de Donato, A. Pescapé, G. Ventre, "Networked Embedded Systems: a Quantitative Performance Comparison", IEEE Globecom 2008, New Orleans (LA), USA, 30 November - 4 December, 2008.

[51] D-ITG. http://www.grid.unina.it/software/ITG.

[52] Ostinato. http://code.google.com/p/ostinato/.

[53] Rude-Crude. http://rude.sourceforge.net/.

[54] Tg. http://www.postel.org/tg/tg.html.

[55] Mgen. http://cs.itd.nrl.navy.mil/work/mgen/.

[56] Brute. http://code.google.com/p/brute/.

[57] Iperf. http://sourceforge.net/projects/iperf/.

[58] Tcpreplay. http://tcpreplay.synfin.net/.

[59] Swing. http://cseweb.ucsd.edu/~kvishwanath/Swing/.

[60] M. Jain and C. Dovrolis, "Pathload: A measurement tool for end-to-end available bandwidth", Passive and Active Measurements (PAM) Workshop, 2002.

[61] V. Ribeiro, R. Riedi, R. Baraniuk, J. Navratil, and L. Cot, "pathchirp: Efficient available bandwidth estimation for network paths," in Passive and Active Measurement Workshop, 2003.

[62] Y. Ozturk and M. Kulkarni, "DIChirp: direct injection bandwidth estimation," Int. J. Netw. Manag., vol. 18, no. 5, 2008, Pages 377-394.

[63] M. Li, M. Claypool, and R. Kinicki, "WBest: A bandwidth estimation tool for IEEE 802.11 wireless networks," in Local Computer Networks, 2008. LCN 2008. 33rd IEEE Conference on, Oct. 2008, Pages 374-381.

[64] C. D. Guerrero and M. A. Labrador, "Traceband: A fast, low overhead and accurate tool for available bandwidth estimation and monitoring," Computer Networks, vol. 54, no. 6, Apr. 2010, Pages 977-990.

[65] J. Sommers, P. Barford, and W. Willinger, "Laboratory-based calibration of available bandwidth estimation tools," Microprocessors and Microsystems, vol. 31, no. 4, pp. 222-235, Jun. 2007.

[66] A. Botta, A. Pescapé, G. Aceto, M. D'Arienzo, "UANM: a platform for experimenting with available bandwidth estimation tools", 15th IEEE Symposium on Computer and Communications, June 2010 Riccione (ITALY).

[67] G. Aceto, A. Botta, A. Pescapé, M. D'Arienzo, "Unified Architecture for Network Measurement: the case of available bandwidth", Elsevier Journal of Network and Computer Applications, Volume 35, Issue 5, September 2012, Pages 1402-1414.

[68] H. H. Song, L. Qiu, and Y. Zhang, "NetQuest: a flexible framework for large-scale network measurement," IEEE/ACM Trans. Netw., vol. 17, no. 1, 2009, Pages 106-119.

[69] A. Ziviani, T. B. Cardozo, and A. T. Gomes, "Rapid prototyping of active measurement tools," Computer Networks, vol. 56, no. 2, Feb. 2012, Pages 870-883.

[70] Uanm. http://www.grid.unina.it/Traffic/uanm.php

[71] R. A. Steenbergen, "A practical guide to (correctly) troubleshooting with traceroute," In NANOG 45, 2009.

[72] B. Donnet, T. Friedman, "Internet topology discovery: A survey," Communications Surveys & Tutorials, IEEE, 9(4):56-69, 2007.

[73] KC claffy, Y. Hyun, K. Keys, M. Fomenkov, and D. Krioukov, "Internet mapping: from art to science," CATCH 2009.

[74] P. Marchetta, P. Mérindol, B. Donnet, A. Pescapé, J-J Pansiot, "Topology Discovery at the Router Level: A New Hybrid Tool Targeting ISP Networks," JSAC, Vol 29 Issue 9, pp 1776 - 1787, Oct. 2011.

[75] M. Luckie, A. Dhamdhere, KC claffy, D. Murrell, "Measured impact of crooked traceroute," SIGCOMM Computer Communication Reviews 41, 1 14-21.

[76] M.H. Gunes, K. Sarac, "Resolving Anonymous Routers in Internet Topology Measurement Studies," pp.1076-1084, INFOCOM 2008.

[77] D. Emma, A. Pescapé and G. Ventre, "Discovering topologies at router level", *5th IEEE International Workshop on IP Operations & Management (IPOM 2005)*, LNCS 3751, pp. 118-129 October 2005.

[78] A. Botta, W. de Donato, A. Pescapé, G. Ventre, "Discovering Topologies at Router Level: Part II", Globecom 2007, Washington, D.C., 26-30 November, 2007.

[79] N. Spring, R. Mahajan, D. Wetherall, "Measuring ISP topologies with Rocketfuel", SIGCOMM'02. ACM, New York, NY, USA, 133-145, 2002.

[80] Hynetd. http://www.grid.unina.it/software/TD.

[81] B. Huffaker, D. Plummer, D. Moore, K. Claffy, *"Topology Discovery by active probing"*, CAIDA, University of California, 1998.

[82] Y. Breitbart, M. Garofalakis, B. Jai, C. Martin, R. Rastogi, A. Silbershatz, *"Topology Discovery in Heterogeneus IP Networks: The NetInventory System"*, IEEE/ACM Transactions on Networking, Vol. 12, no. 3, 2004.

[83] Netalyzr. http://netalyzr.icsi.berkeley.edu/.

[84] Matt Mathis et al. Network Path and Application Diagnosis. http://www.psc.edu/networking/projects/pathdiag/.

[85] Richard Carlson. Network Diagnostic Tool. http://e2epi.internet2.edu/ndt/.

[86] W. de Donato *"Large Scale Benchmarking of Broadband Access Networks: Issues, Methodologies, and Solutions"*, University of Napoli Federico II, 2011.

[87] HoBBIT. http://hobbit.comics.unina.it/.

[88] Neubot. http://neubot.org/.

[89] Grenouille. http://www.grenouille.com/.

[90] BSense. http://broadbandforall.net/.

[91] S. Sundaresan, W. de Donato, N. Feamster, R. Teixeira, S. Crawford, and A. Pescapè. Broadband internet performance: A view from the gateway. In *Proc. ACM SIGCOMM*, Toronto, Ontario, Canada, August 2011.

[92] BISmark Project. http://projectbismark.net/.

[93] SamKnows. http://www.samknows.com/.

[94] Traffic Traces. http://www.grid.unina.it/Traffic/Traces/.

[95] Magnets project. http://www.deutsche-telekom-laboratories.de/~karrer/magnets.html.

[96] A. Botta, D. Emma, A. Pescapé, G. Ventre, Systematic Performance Modeling and Characterization of Heterogeneous IP Networks, Journal of Computer and System Sciences (Elsevier) Volume 72, Issue 7 , November 2006, Pages 1134-1143.

[97] A. Botta, A. Pescapé, G. Ventre, 'Quality of Service Statistics over Heterogeneous Networks: Analysis and Applications', Special Issue of Elesevier EJOR on 'Performance Evaluation of QoS-aware Heterogeneous Systems', Volume 191, Issue 3, 16 December 2008, Pages 1075-1088.

[98] C. N. Pitas, D. E. Charilas, A. D. Panagopoulos, P. Chatzimisios and P. Constantinou, "ANFIS-based Quality Prediction Models for AMR-Telephony in Public 2G/3G Mobile Networks", accepted in IEEE Global Communications (GlobeCom 2012) conference, Anaheim, California, Dec. 2012.

[99] G. Baltoglou, E. Karapistoli and P. Chatzimisios, "IPTV QoS and QoE Measurements in Wired and Wireless Networks", accepted in IEEE Global Communications (GlobeCom 2012) conference, Anaheim, California, Dec. 2012.