The Living Laboratory: a Holistic Approach for Understanding the Performance of Future Mobile Services and Networks

Zary Segall*, Pietro Lungaro
Mobile Service Lab
Royal Institute of Technology (KTH)
Kista, Sweden
segall@kth.se, pietro@kth.se

Pavan Kumar*, Vikrant Nandakumar*
CSEE, University of Maryland
Baltimore County*, USA
pavan4@umbc.edu, vikrant1@umbc.edu

Abstract — **this paper presents the Mobile Services Living Laboratory - a practical approach for evaluating service provision in cellular networks. Our approach promotes an end-to-end view of the system by providing an effective means to store information from both terminal and network sides, together with integrated mechanisms for retrieving feedback on user-perceived service quality. The current implementation of the Living Laboratory is described and evaluated. Further, we show how the Living Laboratory is used to investigate both the effectiveness of context-aware opportunistic content delivery schemes in cellular networks, and the coexistence between M2M and user-generated traffic.**

Keywords: Mobile Networks; Context-awareness; Test-bed; Living Laboratory; Monitoring.

I. INTRODUCTION

In the last few years, there has been an unprecedented increase in the adoption of mobile data services. Key factors behind this success are both the development of powerful user devices such as smartphones and tablets, capable of delivering unprecedented levels of user experience, on the one hand, and on the other hand, flat rate pricing, which has made mobile Internet more affordable to wider consumer segments. While the general success of mobile data was somewhat predictable, this was not the case for its rate of adoption and the associated traffic growth in mobile networks. Mobile operators expected a much more gradual penetration, and significantly milder traffic growth [1]. Moreover, each new smartphone is loading the network with approximately 60 times more data than previous "feature" phones. The greatly increased demand smartphones put on the network coupled with brisk consumer adoption of these type devices has caught mobile operators off-guard, resulting in under-dimensioned networks, and, in most cases, severe congestion problems [1].

Together with smartphones, a novel way of delivering and consuming mobile content has rapidly emerged: the "app" paradigm. The success of apps can be quantified in terms of both number of downloads from the various "app stores" and in terms of user face-time: recent surveys clearly show that apps have rapidly become the main channel for wireless content consumption[2]. Similar to other publish/subscribe approachess, apps fetch information from their remote content servers via either push or pull mechanisms, and use the mobile network for completing the content delivery to the user terminals. In many cases, the transferred content is characterized by some degree of tolerance to delay, and many data transfers are performed as "background" operations. With increasing numbers of apps per phone, and with smartphones gaining considerable shares of the device market, it is becomes easy to understand the magnitude of the additional traffic volume injected in mobile networks.

While apps may be understood as the present manifestation of rapidly growing traffic trends, a large number of key players within the industry are unanimous in foreseeing, for the next few years, a dramatic growth of sensor-generated traffic in cellular systems. This M2M traffic is different from that of app traffic in terms of both the amount of data generated by individual sensors, and in that it is more likely to be stressing the uplink side of the communication, e.g. from the sensors and/or data collection units towards data fusion centers [7].

Understanding the potential impact of these novel traffic sources on the network dimensioning and architectures, together with the quality of experience (QoE) perceived by the end-users, is of paramount importance for the success and sustainability of future services in wireless mobile networks.

II. PAPER CONTRIBUTIONS

This paper presents the Living Laboratory, a novel research methodology to gain a holistic understanding of service provision, even before a service is actually deployed on the market. By merging information gathered from mobile operators' networks, mobile terminals and end users' qualitative assessments, the LL is designed to gain early insights on the performances of the different parts of the system concurring in the service provision.

To exemplify some of the potential applications of the LL concept, we further consider a scenario in which synthetic workloads are generated in a set of remotely controlled terminals. A series of investigations, specifically designed for understanding the performances of different context-aware schemes for content delivery, are presented. In particular, we study how exploiting signal quality information can benefit both terminal energy and QoE for opportunistic content delivery (e.g. content pre-fetching). Furthermore, by programming some of the controlled terminals to act as sensors, in terms of the amount of traffic injected and the periodicity of their reporting messages, we evaluate the impact of different penetration levels of M2M traffic on user-initiated traffic, e.g. YouTube video, served in the same cells.

This work has been performed within the COSEM and SERMON projects, both supported with funding from VINNOVA, the Swedish Governmental Agency for Innovation Systems and Wireless@KTH

III. MONITORING TOOLS

A. State of the art

A very limited set of tools is currently available for researchers and system developers to gain holistic insights on the interplay between network resources, terminal capabilities and user perception during the provision of various services.

In general, the service evaluation in mobile networks can be performed either via simulation, emulation or direct experimentation. Simulation is typically preferred in those studies requiring accurate modeling of cellular network characteristics, since it is very difficult to gain access to accurate information concerning signal quality, network load and radio propagation conditions in the various cells of a real network. Until recently, the limited computational power of mobile terminals was a bottleneck for performance monitoring, but with the advent of smartphones and "programmable" operating systems and applications, new investigation tools are currently been designed.

Among these, the Forum Nokia "Virtual Developer Lab" [16] allows developers to connect to a set of virtual Nokia devices through the Internet and test the behavior of their applications before commercial release. Developers can reserve a time slot for a particular Nokia device and run automated tests for compatibility and troubleshooting. Similarly, "Device Anywhere" [17] offers the possibility of remote testing, but on a number of physical devices connected over different mobile networks. The devices are docked into a "Device Rack", accessible though the Internet. A third example of this trend is represented by the "Perfecto Mobile" system, in which devices are instead mounted on a cradle in a specific location and connected to the Perfecto system through data cables.

All the aforementioned examples represent a concrete step forward toward understanding and testing real system performance, however these might still represented a biased view on performance, since testing is mainly performed under artificial conditions inconsistent with everyday device usage (e.g. no mobility, specific cells and locations, etc.).

Another interesting, and very different, approach is the one proposed by Zokem [19]. By offering the end-users an advanced social application for geo-location and exchange of activity notification messages, Zokem has access to a set of invaluable information gathered directly from real user terminals. Some examples include face-time for the different applications, access patterns in time and space for individual users, web-browsing patterns including web-site URLs, and mobility patterns. All this is achieved in a transparent manner, since the application runs in the background on each terminal. While this approach is capable of extracting accurate information concerning user behavior, it still rather rudimentary for what is entailed in understanding network and terminal performances, together with an explicit feedback mechanism to gather the level of user service appreciation.

B. Living Laboratory Vision

In order to gather an end to end view of the service provision that includes network and terminal information and the user perceived QoE, the "Living Laboratory" (LL) concept, is proposed and evaluated in this paper.

The LL has the overall goal to provide a novel and complete characterization of the various system components occurring together in the provision of a given service. It consists of three different performance monitoring tools, designed for gathering relevant information on different aspects of the service provision:

- Terminal monitoring and load generator software
- Network monitoring software
- User QoE feedback mechanism

Currently implemented for the Android platform, the terminal monitoring application is installed on user terminals and runs transparently in the background. Various types of terminal information, including signal quality, battery levels and geo-location can be tracked and stored for investigatory purposes. At the same time, terminals can also be remotely controlled and synthetic workloads executed on them. These loads may be used to test various network performance and to reproduce interesting service combinations in various times and locations.

The LL framework is likely to create significant value for all the stakeholders involved in the service provision processes:

- Mobile operators can test how their current deployment is capable of coping with different types of new services before those services are available on the market. At the same time, even with current service offerings, the information gathered through the various sensors in terminals and network can be used for diagnostic purposes concerning coverage and capacity assessments in different parts of their network.
- Application developers can have access to user perception and potentially modify the design of their software to suit user preferences or network capabilities.
- End-users can express their feedback and have access to a more personalized service provision from both content developers and mobile operators.

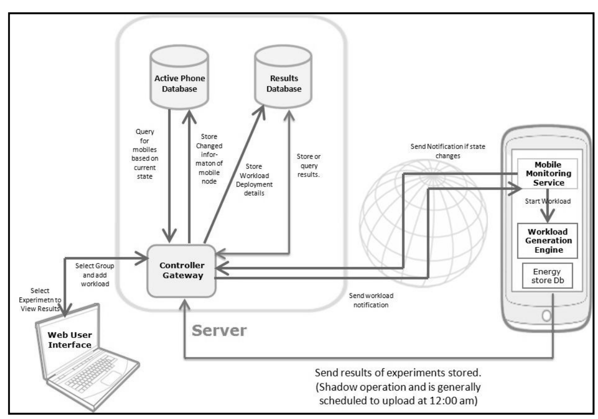

Figure 1. Description of the LL Testbed, individual components and architectures.

IV. CURRENT TESTBED IMPLEMENTATION

The current implementation of the LL includes the terminal monitoring software and the synthetic workload generator. Interfacing the current testbed implementation with the operators' monitoring infrastructure is part of the ongoing development within the COSEM project [?]. At the same time, the quality of experience feedback mechanism, designed to extract qualitative service experience evaluations from the end-user, is undergoing development within the SERMON project [?] and expected to be available during the end of 2011.

A. Architecture

The testbed architecture, shown in Figure 1, consists of three separate components which interoperate with each other. Controller Gateway Servlet (CGS), Active Phone Database (APD) and the Results Database (RD) constitute the server elements. The CGS acts as a gateway for all incoming and outgoing information between the server and the mobile nodes. CGS handles the requests sent to and from the terminals, such as sending network statistics (push), and getting workload information (pull), from the Results Database (RD). The APD is the snapshot of the current state of the mobile nodes. The RD contains the active workload deployment details along with the results obtained by running each one of them. Each mobile node has two components: the Mobile Monitoring Service (MMS) and the Workload Generation Engine (WGE). The MMS is responsible for keeping the APD updated with vital information about various parameters such as cell id, location, signal strength, battery capacity, battery voltage, battery drain percentage, network statistics, application usage information, etc., while the WGE is invoked by the user to start running workloads on the terminal. The entire system is exposed to the user through a web-based interface which has provisions to discover terminals, control them remotely, generate workloads, display results, query the Results Database (RD) and obtain processed statistics.

The test bed works seamlessly and independently of any service provider or device manufacturer. The system can manage mobile nodes spanning across different remote locations and allows different combinations of heterogeneous workloads. These run in the background without interrupting the user, while a workload scheduling feature additionally permits the users to deploy workloads at a future time. The system is secure with both device-level and user-level role-based authentication, and complexity scales efficiently on demand to accommodate a large number of mobile nodes.

B. Scalability

Two different implementations of the testbed have been considered, mainly taking into consideration the scalability of the system in respect of the number of concurrently supported terminals. In the "standalone" implementation, a client runs in each of the controlled terminals, while the server runs on a machine with limited capability, i.e. equipped with an AMD Athlon 64 TM processor and 1GB of memory.

In the "cloud" approach, the services of the Amazon Elastic Cloud (EC2) have been used instead. In the deployment of the server on EC2 we made use of one Standard Extra Large Instance and one High-memory Instance. The performances of the two implementations, expressed in terms of processing delay for different numbers of concurrent terminals are shown in Figure 2. The results show that the standalone implementation works fairly well for few concurrent terminals, and that delays in processing increase linearly with the number of terminals, eventually leading the server becoming overloaded. At the same time, in Figure 2 it is clearl that a significant processing speedup could be achieved by increasing the number of terminals. In particular, for handling 1000 concurrent terminals, the recorded speedup is approximately of a factor 15 times when compared to the standalone implementation.

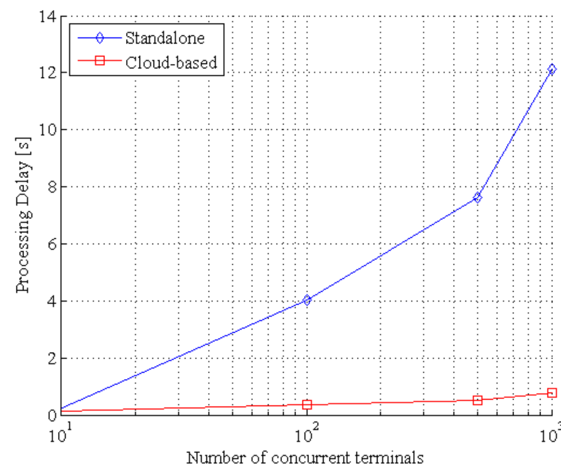

Figure 2. Processing delay performances, as function of the number of cuncorrently supported terminals, for both the Standalone and the Cloud-based implementations of the testbed.

V. EXPERIMENTAL RESULTS

Two different studies, focusing on investigating the performances of context-aware schemes, are here presented to illustrate some of the potential applications of the LL framework. In order to mitigate the effects of apps and M2M communication on traffic, we proposed and investigated in [3]-[6] a set of context-aware solutions for opportunistic content delivery. The basic idea is to optimize content delivery, in terms of investments in terminal and network resources, by exploiting the times in which there are excess resources available at the BSs. Since a large portion of the transferred content has some degree of tolerance to delays, operators can opportunistically wait for users to roam closer to BSs and/or to less loaded cells, and then perform a more efficient content delivery. The importance of opportunistic content delivery in future mobile networks has also been addressed in a series of previous publications, mainly targeting pre-fetching solutions in cellular networks, e.g. [9]-[15]. In some cases ([9],[11]), local storage of information at the terminals has been proposed to reduce the consumption of wireless resources for frequently accessed data items. In other investigations, pre-fetching solutions have been suggested for reducing the effects of channel quality fluctuations ([10]) and improving the performance of (streaming) protocols in wireless environments ([12],[13]). In some other cases (e.g. [14]-[15]), exploiting mobility information for performing content pre-fetching in heterogeneous networks has been investigated, where pre-fetching was mainly performed within WLAN coverage.

A. *Context-aware content delivery*

In Figure 3 we show the energy costs associated with receiving video files of different qualities. The costs in terms of battery drainage can be significant, if signal strength is not considered when deliverying the content to the user terminals. By exploiting this context information, and waiting for user terminals to roam closer to the BSs, operators can reduce the overall terminal energy costs by 75%, for low quality videos, and 33% for high quality .

TABLE I. SUMMARY OF TEST VIDEO CHARACTERISTICS

Version	Resolution [pixels]	Format	Size [MB]
LQ 3GP	176x144	3GP	0.85
MQ 3GP	176x144	3GP	1.07
HQ 3GP	320x240	3GP	3.14
HQ MP	480x360	MP4	6.48

While in the final paper an extensive set of experimental results will be discussed and analyzed, in this section we focus on a specific study case concerning a video delivery service. Four different versions of the same video, with actual content duration equal to 103 seconds, are considered and the corresponding parameters are summarized in Table 1. The experiments presented here are all performed within the same cell with Android Nexus Ones adopted as users' terminals. In Figure 3 the average energy costs (in battery percentage) required for pre-fetching the different versions of the video are shown. In this example, the context information is the signal quality measured at the terminal, and two cases are considered: high quality (83%), experienced by users close to the BS, and low quality (12%), with users in the outer parts of the cell. The results show that including signal quality reports in the pre-fetching schedulers can save between 33% and 75% of the battery energy, depending on the specific video quality.

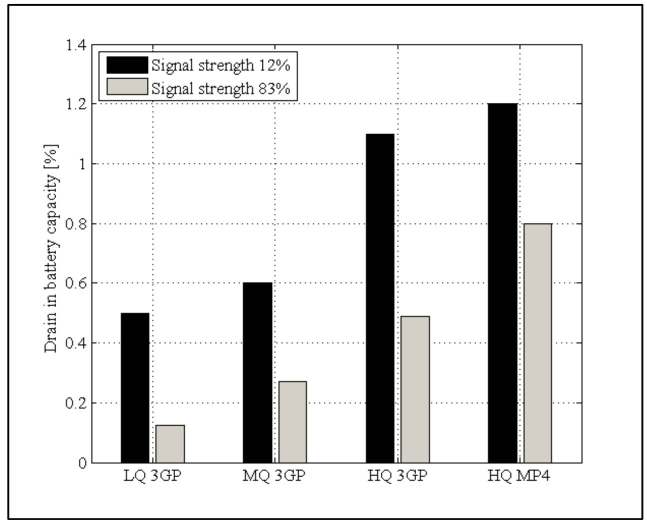

Figure 3. The Average percentage of battery drain associated with pre-fetching the 4 different video types when the terminals are in good and bad signal quality locations.

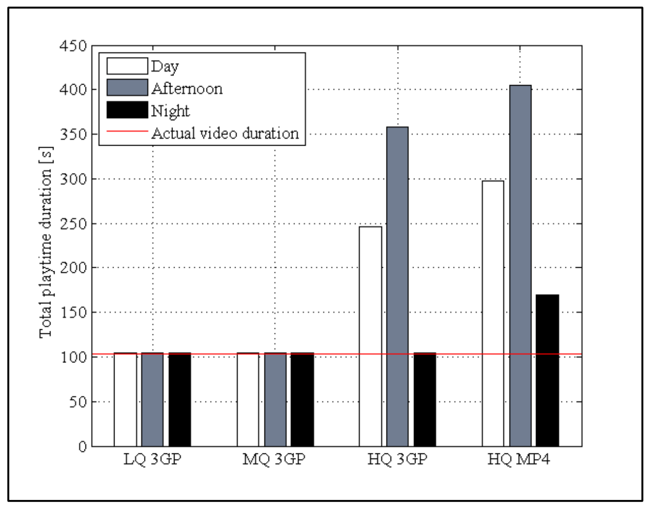

Figure 4. Duration of the playtime associated with deliverying four different video qualities in diffeent hours of the day. The actual video duration, without interruptions, is 103 s.

On the other hand, signal quality alone does not describe the actual traffic conditions of a cell. In Figure 4, the playtime durations, experienced by the users when consuming the various video versions during different times of the day, are shown. In particular, the results referring to the ``day'' time are averages of the performances obtained between 10am and 11am, ``afternoon'' refers to the interval 4pm-5pm and finally ``night'' time is between 11pm and 12pm. While the playtime duration for the lower quality videos is not affected by the traffic variations in the cell, the higher quality videos are instead characterized by frequent interruptions, except during the night hours. Streaming the HQ MP4 video during the ``afternoon'' forces the user to spend roughly 75% of the time waiting with an unfilled buffer, at a considerable energy cost.

Instead, being able to predict the wanted video and delivering it in advance to the user terminal during night time, or while roaming through a low loaded cell, would have brought a significant improvement in both user perceived service quality and extended battery life. The app paradigm simplifies information prediction and therefore is a natural candidate for an opportunistic content delivery.

B. *Sensor Traffic*

In Figure 5, we instead present a case study in which two terminals are co-located within the same cell. While the workload in the first terminal represents human generated traffic (the download of a small file), the second one injects in the network a synthetic workload corresponding to different levels of sensor traffic activity in the same cell. The penetration levels considered in the experiments are relative to the network dimensioning assumed in [7], while the sensor mix and traffic associated with each sensor type are based on the scenarios presented in [8]. The results show that with increasing sensor traffic, the performances experienced by the users progressively deteriorate, reaching almost 100% increased download duration when as many sensors as mobile users are assumed to be colocated within coverage of the same BS (100% penetration).

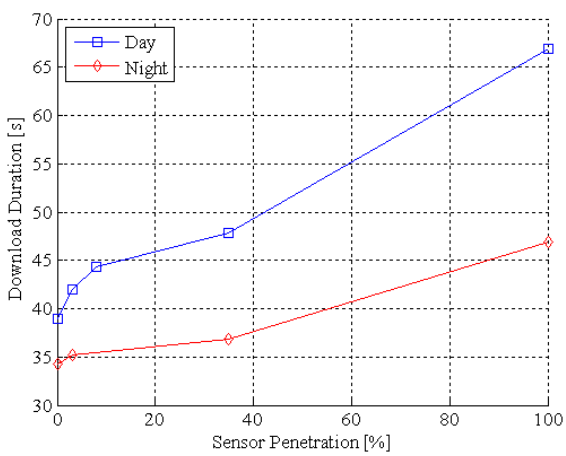

Figure 5. The total duration of a video download is shown for different penetration levels of sensor traffic that is simultaneously served in the same cell. The penetration levels and sensor traffic composition are extracted from [7] and [8].

VI. CONCLUSIONS

The concept of Living Laboratory has been presented and discussed in this paper. In order to exemplify the potential impact of this approach in providing novel insights on service provision we have investigated two scenarios concerning context-aware content delivery and M2M communication networks. Based on those results we project that the LL framework is likely to create significant value for all the stakeholders involved in the service provision processes: mobile operators, application developers and ultimately end consumers.

REFERENCES

[1] Roben Farzad, "AT&T's iPhone Mess", article on Businessweek, 15th of February2010, available online at: http://www.businessweek.com/magazine/content/10_07/b4166034389519.htm

[2] Zokem report on smartphone usage, 14th of February 2011. Available online at: http://www.zokem.com/2011/02/gsma-partners-with-zokem-at-the-biggest-mobile-event-of-the-year-to-report-the-latest-in-mobile-usage/

[3] P. Lungaro, Z. Segall and J. Zander, "ContextShift: a Model for Efficient Delivery of Content in Mobile Networks", in Proceedings of the Wireless Communications & Networking Conference (WCNC10), Sydney, April 2010.

[4] P. Lungaro, Z. Segall and J. Zander, "Predictive and Context-based Multimedia Content Delivery in Future Cellular Networks", in Proceedings of the Vehicular Technology Conference (VTC10), Taipei, May 2010.

[5] P. Lungaro, Z. Segall and J. Zander, "Context-aware Resource Schedulers for Opportunistic Content Delivery in Cellular Networks", in Proceedings of the 3rd international conference on Communication Theory, Reliability, and Quality of Service (CTRQ10), Athens, June 2010.

[6] P. Lungaro, Z. Segall and J. Zander, "ActiveCast - a network and user aware mobile content delivery system", in Proceedings of the 2nd international conference on Ubiquitus and Future Networks (ICUFN), June 2010, Jeju Island, Korea

[7] J. Krnic and S. Krco "Impact of WSN Applications' Generated Traffic", in Proceedings of PIMRC 2008.

[8] J Marron, D. Minder and the Embedded WiSeNts Consortium, "Embedded WiSeNts IST-004400 – Application Scenarios", 2006-02-21.

[9] D. Barbara, T. Imielinski, ``Sleepers and Workaholics: Caching Strategies in Mobile Environments", in Proceedings of SIGMOD94, 1994.

[10] S. Gitzenis, N. Bambos, ``Power-Controlled Data Prefetching/Caching in Wireless Packet Networks", in Proceedings of INFOCOM02, 2002.

[11] Y. Lin et al., ``Effects of Cache Mechanism on Wireless Data Access", IEEE Transactions on Wireless Communications, Vol.2, November 2003.

[12] F. Fitzek, M. Reisslein, ``A Prefetching Protocol for Continuous Media Streaming in Wireless Environments", IEEE Journal on Selected Areas in Communications, Vol.19, November 2001.

[13] J. Hu, G. Feng, K. Yeung, ``Hierarchical Cache Design for Enhancing TCP Over Hetereogeneous Network with Wired and Wireless Links", IEEE Transactions on Wireless Communications, Vol.2, March 2003.

[14] S. Drew and B. Liang, ``Mobility-aware web prefetching over heterogeneous wireless networks", in Proceedings of PIMRC04, Sept. 2004.

[15] B. Liang, S. Drew and D. Wang, ``Performance of multiuser network-aware prefetching in heterogeneous wireless systems", in Wireless Networks, Springer, Vol. 15, January 2009.

[16] "Forum Nokia Virtual Developer Lab", website available at the URL: https://www.deviceanywhere.com/nokia_vdl/Nokia_virtual_developer_lab.html.

[17] "Device Anywhere", website available at the following URL: http://www.deviceanywhere.com/

[18] "Perfecto Mobile", website available at the URL: http://www.perfectomobile.com/portal/cms/Services/nokia.html

[19] "Zokem", available at the following URL: http://www.zokem.c