

MOBSIM

An Integrated Simulation Platform for the Evaluation of Mobile Radio Networks

Activity report

October/2011 – June/2012

Daniel Câmara - daniel.camara@inria.fr

Equipe Planète, INRIA Sophia-Antipolis

Contact: Walid Dabbous - Walid.Dabbous@sophia.inria.fr

Abstract: This document presents the work done over the ADT MobSim project during the nine first months of the project, from October 2011 to June 2012. My role over the MobSim project consisted basically of: (i) support to the other teams, (ii) creation of a tool for automatic build of the ns-3 platform and (iii) help ns-3 team with the development of the simulator. This report also points some of the next steps for the project and the work that still has to be done.

Table of Contents

I.	MOBSIM – Introduction and objectives	2
II.	The ns-3 simulator	2
III.	Project advancements	2
A.	Forecasted scheduling, as described on the proposal:	3
B.	Work done and scheduling of tasks:	4
C.	Plans for the end of this contract year:	6
D.	Plans for the second year:	6
E.	Personal improvements:.....	7
	References.....	8

I. MOBSIM – Introduction and objectives

The main objective of the ADT MOBSIM is to develop an integrated simulation platform for mobile wireless networks. There are three INRIA groups involved in the activities of this ADT: SOCRATE, from Lion, HIPERCOM from Paris, and PLANETE based at Sophia Antipolis. All three teams have developed in the past simulation modules for different simulators and that, in the end, were incompatible among them. The Mobsim intends to aggregate the knowledge and expertise of these three teams over a more strong and focused initiative. The idea is to create a framework for the ns-3 simulator (PLANETE) with modules of simulation for the physical layer (SOCRATE) e modules of access control and routing (HIPERCOM) for mobile networks.

Beyond the research that will be done during the development of the integrated platform, this ADT has also the ambition to improve and consolidate the position of INRIA as a supporter and contributor for the open source software movement, taking advantage of the visibility of the ns-3 project.

II. The ns-3 simulator

The ns-3 network simulator [1][2][3] is an open source, C++ based, GPL licensed and highly used network simulator. In the beginning of the years 2000 researchers start to lack confidence on stochastic simulations as a way to validate the performance of telecommunication networks protocols [4]. Two of the main reasons that lead to the development of ns-3 where: overcome the lack of realism of the available network simulators and provide reproducibility for networks emulation [3]. Ns-3 tries to close the gap between simulations and emulations/testbeds and has transparent support for real-time simulations. It allows the implementation of arbitrary models that can be interconnected with real world networks. Ns-3 extends the scope of the network simulations by allowing the use of the same protocol implementations in both simulation and testbed environments.

Ns-3, more information:

- license: GPLv2
- Code property : individual contributors
- web: <http://www.nsnam.org>
- wiki: <http://www.nsnam.org/wiki>
- blog: <http://nsnam.blogspot.com>
- code: <http://code.nsnam.org>
- bugzilla: <http://www.nsnam.org/bugzilla>

III. Project advancements

MOBISIM develops as planned, all the activities defined on the ADT proposal are being developed and are on time. The teams have cooperated sharing experience and helping each other to solve their problems. The Contiki adaptation to ns-3 was born form one of this exchanges, and is one of the best examples how wealthy was the exchange of ideas and mutual search for solutions.

The main activity proposed for the Planete team for the ADT is creation of a tool for automatic build of the ns-3 project. The advancement of the project is in accordance with the schedule put in the proposal, in fact we are even a little ahead of the chronogram. The developed build tool, called Bake, is divided it in two parts, a server side, where users can add modules, and a client part, that can be used by

users to make the build of the selected modules all in one. The whole process, from the upload of the module, by the developer, to the integrated build process is described in Figure 1.

It is important to highlight here that after starting the work on MOBSIM, we perceived that the ns-3 project needs were a little more complex and subtle we first thought. Mainly the server part, proposed on the ADT, had to be adapted to the needs of the project. The main problem, or worry, from the ns-3 development team is the addition of quality new modules. Modules that will be maintained by the project should have a minimum level of quality, and anyone who wants to add their modules for the project are free to do so, however, he/she should follow a well-defined process. The problem is that this process demands time and resources and not everyone is willing to go over it. In this way, potentially, a huge amount of code is being lost, only because that. We perceived that Bake, the tool we are proposing, could be used also to solve this problem.

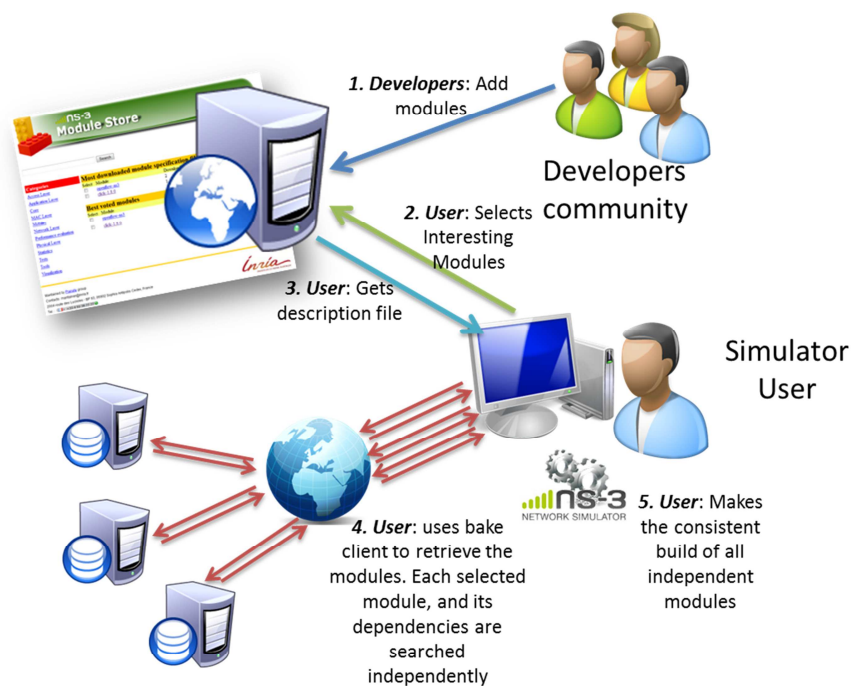


Figure 1. Generation of a customized build

A. Forecasted scheduling, as described on the proposal:

Main targets:

In 6 months: have a version that is stable and able to build ns-3

- Done on the schedule

In 12 months: have the tool adopted by the ns-3 consortium as the tool for build the code

- Bake should be adopted by the ns-3 project in August as the default tool to build the project, in 10 Months from the start of the project

Defined Milestones:

- Download modules described over an XML and recognize dependencies.
Forecasted time: T0 (1 month)

Real time: 1 Month

- Build downloaded modules with correct management of errors and optional dependencies.
Forecasted time: T2 (1 month)
Real time: 1 Month
- Server web to store the modules data in a databases and conversion of database data into a downloadable XML.
Forecasted time: T3 (1 month)
Real time: 1 Month
- Server web that is able to handle multi-users and rights of the users
Forecasted time: T4 (1 month)
Real time: Got for “free” with the development framework adopted
- Add search capabilities to the server web
Forecasted time: T4 (1 month)
Real time: 1 week
- Add capabilities to store more information about the status of the module on the server
Forecasted time: T5 (1 month)
Real time: Postponed in detriment of other functionalities
- Integration of the server and client applications to store results on the databases
Forecasted time: T5 (1 month)
Real time: 1 Month

B. Work done and scheduling of tasks:

- *Task:* Project follow-up
- *Duration:* Background activity over all the project (October - June)
 - Over this first year, as senior engineer, I had the role of project coordinator. I organized the project meetings, kept track of the work done and provided support for the MOBSIM ADT members. My main role on the project, as senior engineer, is to make it easier for the project members to use ns-3 and its functionalities. In fact even Bake, the tool that is being developed in the Planete group, comes also to reinforce this role. Bake will make it easier the integration of the code produced by the MOBSIM ADT.
 - During the project meeting the members of HIPERCOM team made us aware that it would be interesting for them to have sensor network nodes over ns-3, however, ns-3 lacks of support for this kind of network nodes. For this reason we decided to start a project to port Contiki to ns-3. Contiki is a popular operating system for sensor nodes.
 - I have also worked with SOCRATE team to define means to enable access to the data generated by their tools to the ns-3 community.
- *Task:* Formation
- *Duration:* 1 Month (October)
 - I never worked neither with Python nor ns-3, so I needed some time to learn about both and research other tools capable to perform the same kind of tasks we wanted to implement for the ns-3 project

- *Task:* Definition of the architecture
- *Duration:* 1 Month (November)
 - o Definition of the architecture and components of Bake

- *Task:* Client side Bake
- *Duration:* 2 months (December- January)
 - o Development of the client side Bake and adaptation of the existent code to support the requirements of the defined architecture

- *Task:* Project page
- *Duration:* Background work
 - o The creation of a page for the ADT where we can concentrate information and make all the material available for all the members. The page can be found at: http://planete.inria.fr/projects/mobsim_html/index.html

- *Task:* Preparation of the project meeting
- *Duration:* Background work over 0.5 months
 - o The organization of the first project meeting that was done at Sophia Antipolis (February 20, 2012)

- *Task:* Poster for the ns-3 workshop
- *Duration:* 0.5 months (March)
 - o The creation of a poster and a demo that was presented at the ns-3 workshop in Sirmione, Italy (March 23, 2012)

- *Task:* ns-3 Developers meeting
- *Duration:* Background work
 - o Presentation of Bake at the ns-3 developers meeting in Sirmione, Italy (March 24, 2012)

- *Task:* Server side Bake
- *Duration:* 3 months (April- June)
 - o Development of the server side Bake and adaptation of the existent client code to support the requirements of the server side

- *Task:* Google Summer of Code
- *Duration:* Background activity over 5 months (April- August)
 - o The ns-3 project was selected as one of the host institutions for the Google summer of code and I was chosen as one of the mentors for one of the 3 selected projects. This activity is related to the ADT in the sense that it reinforces the involvement of INRIA with the open source community

- *Task:* Adaptation of Contiki for ns-3
- *Duration:* Background activity over 3 months (May- July)
 - o I am supervising the activities of a student that is creating a middleware so that the Contiki applications and protocols will be able to be used over ns-3. Contiki is a well-known Operating System for sensors. This work will be afterwards used by the HIPERCOM team on their research.

C. Plans for the end of this contract year:

Both the server and the client side Bake are ready to be used by the community. They are in a Beta testing phase, and the client Bake should be adopted by the ns-3 project, as standard building and installation tool already from the next release, to happen in August. From here to there I will be helping users that are whiling to add their codes over Bake and, as these are our first real “clients”, the greatest part of the next few months I will expend bug-fixing Bake client code, and adding new functionalities, as required by the users.

Among the functionalities that we already feel are missing in the client side Bake we can cite:

- Add the possibility for the installer to perform a pre and a post installation system calls. Sometimes, some application may need to perform SO operations, for example, copy files from one side to the other.
- Add conditional installation. For example, if the operating system does not have one specific characteristic, Bake should defer the package installation.

I will also work to polish a little more the server Bake that should be ready to be put in production in September.

D. Plans for the second contract year:

If the contract gets renewed we believe the tasks we plan to perform are the following:

- *Task:* Extend the integration of Contiki and ns-3
- *Duration:* October – May
- *Evaluation criteria:*
 - o A module for ns-3 that can create Contiki nodes over ns-3 networks. These nodes should have either a full Contiki stack or a mix Contiki-ns-3 stack.
- *Description:*
 - o The first task is to make the student’s developed code available to the community in the terms required by the ns-3 community
 - 1 Month
 - o Write the user’s manual and the development of meaningful examples for the users
 - 1 Month
 - o The project proposed to the student does not include the use of the ns-3 stack over the Contiki nodes. However, this is the kind of actions many researchers, including the ones involved on MOBSIM, would like to do.
 - Middleware for connecting the Contiki nodes over ns-3 MAC models
 - 1 Month
 - Middleware for connecting Contiki applications over ns-3 TCP stack
 - 1 Month
 - o The integration of simulated Contiki nodes and real network traffic
 - 2 Months
 - o Tests
 - 1 Months

- User's documentation
 - 1 Month
- *Task:* Integrate the MOBSIM framework
- *Duration:* Jun-August
- *Evaluation criteria:*
 - A fully integrated solution to evaluate mobile networks using ns-3 network simulator.
- *Description:*
 - During this term, as planned, I will perform the integration of the of the modules previously developed during the ADT within the ns-3 framework.
 - Integration of the modules developed by HIPERCOM team
 - 1 Month
 - Integration of the modules developed by SOCRATE team
 - 1 Month
 - Write MOBISIM User's manual
 - 1 Month
- *Task:* Academic publication
- *Duration:* August-September
- *Evaluation criteria:*
 - Two, or more, papers submitted to conferences or journals formalizing the work done within the context of the MOBSIM ADT.
- *Description:*
 - We should be able to formalize the experience with Bake into an academic paper to be submitted to a developer's journal or an open source conference
 - We should also be able to publish an academic paper to present the MOBSIM platform
- *Task:* Coordinate the project activities
- *Duration:* Background activity during all the project year, October 2012-September 2013
- *Evaluation criteria:*
 - Regular meeting minutes
 - Project page updated regularly
 - Delivery of successful project
- *Description:*
 - Prepare the meetings and coordinate the cooperation among the MOBSIM participants

E. Personal improvements:

- During the last months I have learned a lot and improved both my programming and management skills. Helping the other teams, advising students and trying to organize the project meetings really helped me to improve my management and people's skills. I believe I am much better on this kind of activities now than ever.

- I have never worked with Python before, and I am really glad I had the opportunity to work with Python on this project. Both the client and the server sides were built using Python.
- Apart from these points I have also never worked within any open source project before, nor did I have any experience with ns-3. Ns-3 is a fairly complex piece of software and being able to find your way over it is a challenge by itself. Being somehow forced to look into ns-3 code and helping to solve some bugs helped me a lot to improve my C++ skills.

References

- [1] CCNx, <http://www.ccnx.org/>, last visited April, 2012
- [2] Mathieu Lacage, Outils d'Expérimentation pour la Recherche en Réseaux, PhD Thesis, Université de Nice-Sophia Antipolis, November 2010
- [3] George F. Riley, Thomas R. Henderson, The ns-3 Network Simulator, Chapter 2, Modeling and Tools for Network Simulation Editors, Klaus Wehrle, James Gross, Springer-Verlag, ISBN: 978-3-642-12330-6, 2010
- [4] K. Pawlikowski, H. D. Joshua Jeong and J. S. Ruth Lee, On credibility of simulation studies of telecommunication networks , Communications Magazine, IEEE , vol.40, no.1, pp.132-139, Jan 2002