PlanetSim: An extensible framework for overlay network and services simulations*

Jordi Pujol Ahulló‡
jordi.pujol@urv.cat

Pedro García López‡
pedro.garcia@urv.cat

‡Universitat Rovira i Virgili
Av. Països Catalans, 26
43007 - Tarragona, Spain

1. INTRODUCTION

Research community on distributed systems, and in particular on peer-to-peer systems, needs tools for evaluating their own protocols and services, as well as against other protocols with the same precondictions. Since a (TCP/IP) experimental evaluation is not always feasible, simulation tools appeared. Nevertheless, the vast majority of them are ad-hoc customized simulators [3] and they are not for general overlay evaluation purposes, poorly documented or not extensible to other protocols and settings. Thus, in this paper we are mainly interested in extensible, scalable, high-level overlay and services simulation frameworks. In particular, we focus on structured (e.g. Chord, Pastry) and unstructured (e.g. Gnutella) peer-to-peer systems and services simulation onto our simulator PlanetSim.

We propose PlanetSim [2], an object-oriented, extensible, customizable, efficient framework for overlay network and services simulations implemented in Java programming language. In particular, we see these points as the main contributions of this simulator:

1. PlanetSim provides a clearly layered simulation framework, mainly network, node and application layers, employing the Common API (CAPI) [1] as the interface between node and application layers. Researchers can easily develop their own protocols and services, simulating them in a time-efficient way. To do so, PlanetSim does not consider packet-level details.

2. PlanetSim provides two ways of implementing new overlay protocols: an *algorithm-based* and a *behavior-based* approaches. We see the former as a traditional way to implement the overlay protocol itself all together by means of the node API. The latter approach enables the researcher to split every simple action a node must perform into different *behaviors*, defining when such behaviors are applicable.

3. As PlanetSim layered structure obeys to well-known design patterns in software engineering, we provide a framework with clear hotspots, so that modifications and extensions to PlanetSim at all levels are easy and well defined.

4. From the one hand, PlanetSim defines by default a naïve mechanism to gather results, avoiding complex mechanisms that may slow down simulations. From the other hand, we have defined an introspection scheme, so that researchers can gather as much statistical information as necessary.

The key idea behind our *behavior-oriented programming* model is to allow developers to encode the set of actions *any* node can perform in separate classes that can be added and removed at will, without recompiling the source code to specify the way in which nodes must behave in each simulation.

Additionally, to prove the framework extensibility, authors and other third parties have extended PlanetSim in different ways:

1. PlanetSim can extract statistics information accordingly by means of aspect-oriented programming (AOP), which penalizes neither in terms of computation time nor memory usage when disabled.

2. We have extended Planetsim to add latency awareness in order to evaluate more realistic context sensitive and more complex applications like content distribution networks.

3. Researchers can simulate various overlays within a single simulation, by means of the notion of *super-peer*, a special node that interconnects different overlays at the same time.

4. Simulations can be time-improved by leveraging a multiprocessor computer. To do so, PlanetSim has been extended to provide a multi-thread context where the simulation runs.

*This work has been partially funded by the European Union under the 6th Framework Program, POPEYE IST-2006-034241, and by the Spanish Ministry of Education and Science under P2PGRID project, TIN2007-68050-C03-03, and FPU National Program, AP-2006-04166.

SIMUTOOLS 2008, March 03-07, Marseille, France

DOI 10.4108/ICST.SIMUTOOLS2008.2935

2. REFERENCES

[1] F. Dabek, B. Zhao, P. Druschel, J. Kubiatowicz, and S. I. Towards a Common API for Structured Peer-to -Peer Overlays. In *Proc. IPTPS'03*, February 2003.

[2] P. García, C. Pairot, R. Mondéjar, J. Pujol, H. Tejedor, and R. Rallo. PlanetSim: A New Overlay Network Simulation Framework. In *Proc. SEM'04*, pages 123–136, September 2004.

[3] S. Naicken, B. Livingston, A. Basu, S. Rodhetbhai, I. Wakeman, and D. Chalmers. The state of peer-to-peer simulators and simulations. *ACM SIGCOMM Comp. Comm. Review*, 37(2), April 2007.