

# Network Emulation for Application Development in Heterogeneous Networks

Thomas Reicher (1), Christian Schwingenschlögl (2) and Wolfgang Wein (1)

(1) Institute of Applied Software Engineering, Munich University of Technology, Boltzmannstrasse 1, D-85748 Garching, Germany, E-mail: [reicher@in.tum.de](mailto:reicher@in.tum.de)

(2) Institute of Communication Networks, Munich University of Technology, Arcisstrasse 21, D-80290 Munich, Germany, E-mail: [schwinge@lkn.ei.tum.de](mailto:schwinge@lkn.ei.tum.de)

## Abstract

This paper describes an emulation based approach to develop and test future applications and services in Ad-hoc and heterogeneous network environments. For the development of applications and services in Ad-hoc and heterogeneous networks it is important to consider the characteristics of each network type. This is contrary to the traditional goal of network stacks and middleware solutions which indeed want to hide network characteristics from the application developer. The setup of a development environment for such type of applications is complex and costly, and the development quite laborious. To ease the development of applications, services, and platforms we propose an emulation based approach. We present a prototype for such an emulation environment and illustrate the setup on a concrete example, the setup for the Virtual City Portal prototype. For the Virtual City Portal we employ digital broadcast networks, cell phone networks, and Ad-hoc networks.

## 1 Introduction

The development of applications and services in heterogeneous and dynamic, particularly in Ad-hoc networks faces developers with problems unknown from wired networks.

The traditional goal of network stacks and middleware solutions such as CORBA, Microsoft DCOM, Java RMI, or XML-RPC/SOAP was to hide network characteristics from the application developer. Developers should be able to develop distributed applications like local applications. While this assumption might be acceptable for today's well-built wired networks it is no longer true for wireless ones, particularly Ad-hoc networks, and heterogeneous networks with multi-access terminals. Multi-access is the capability to connect a terminal to several network attachment points of different technologies simultaneously for obtaining access to the same application service (Njes *et al.*, 2002).

Now, for the development of applications and services in Ad-hoc and heterogeneous networks it is important to consider the characteristics of each network type. There are several reasons: First, each network has its own profile for transmission bandwidth, packet drop rate, network reliability, network security, or cost.

Second, services for mobile users are not reliable. The connection between service provider and service user can be interrupted at any point and resumed later. In reliable networks this is handled by time-out thresholds and thrown software exceptions, if the threshold is exceeded. But in wireless networks disconnections are not exceptional at all (Grimm *et al.*, 2003).

This paper describes an emulation based approach for the development and testing of future applications and services in Ad-hoc and heterogeneous network environments. The emulation should be transparent for the developed application. Ideally, the developed and tested application should behave identically in the emulation environment and the real deployment environment. The configuration of the emulator can be coupled with a separate network simulator that provides dynamically

simulated network conditions for a simulated network, e.g. available nodes or bandwidth.

We explain our approach on a concrete example, the Virtual City Portal prototype that has been developed within the Softnet project (Bauch *et al.*, 2002).

## 2 The Virtual City Portal

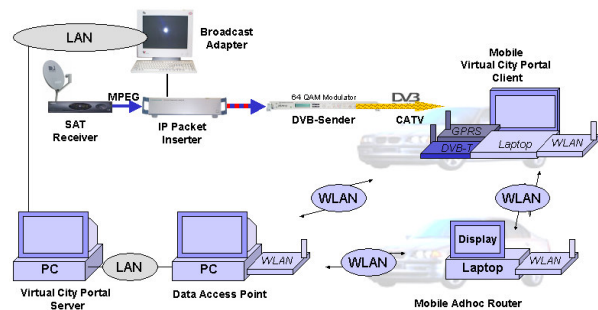


Figure 2.1 Setup of the Virtual City Portal prototype.

The Virtual City Portal aims at supporting mobile tourists that need guidance in a city. A foreign city can be very confusing, especially when visited by car. For example, it is quite difficult to find basic facilities such as a gas station or a good restaurant. What one would need in such a situation is an on-board application always at hand that contains current city specific services and information. With the current level of development of wireless network technologies and the integration of computer systems in vehicles, such an application can be realized. The Virtual City Portal is a new kind of an on-board application for

vehicles for city-specific services that empowers automobile travelers to access and use a city's key services and information without the need for time consuming seeking. For this the Virtual City Portal application relies on a heterogeneous wireless network environment, consisting of DVB-T, Wireless LAN (IEEE 802.11b) and GPRS, which enables it to adapt its offerings to the respective city. We describe the Virtual City Portal in more detail in Bogenberger *et al*, 2003.

### 3 The TUM NIST Net Emulator

The TUM NIST Net Emulator is used for emulating the behavior of distributed applications in heterogeneous, particularly wireless networks. The emulator should support the following requirements:

- **Operating system independence.** Since a lot of different hardware and software is used, the testing environment has to be completely independent of the operating system of a particular clients. Therefore it has to be based solely on the TCP/IP protocol set, which is used by all of the clients. Because of this, as few settings as possible should be manipulated on every client, as this would be very complex to realize for multiple operating systems.
- **Distributed testing.** The connection between any two network nodes can be completely different to any other connection. So it is necessary to control the connection from every node to any other one.
- **Emulated behavior.** In order to behave like a wireless LAN, the testing system has to be able to simulate the delay of TCP/IP packets, random drop of packets, bandwidth limitation and blocking connections temporarily. This should be controllable with a central graphical tool on one computer.
- **Scripting features.** The varying behavior of wireless LANs over, for example, time, position and environment should be emulated via user defined functions and scripts.
- **External interface.** In order to be coupled to network simulators it must be possible to take the output of simulations to set up and configure connections dynamically.

To meet these requirements we decided to employ a router-based approach where each data packet must pass the router. The router can shape the behavior of the network connection between any two nodes that are interconnected. We assume that no other network type we want to use is faster than the Fast or Gigabit Ethernet that connects the network nodes in our lab.

Technically, every client computer sustains its own small subnet, not overlapping with any other. A corresponding virtual network interface for each of those subnets is being set up on the control

computer, whose IP address is specified as the particular client's default gateway. Further the control computer can be configured as a DHCP server, taking on the client configuration, so that no settings have to be changed on the clients at all.

We chose to build upon the NIST Net emulator (NIST Net Homepage, 2002) as it provides the facilities necessary to control several features of installed network interfaces (delay, drop rate, duplication rate, bandwidth, and queue length). We developed a user interface for it, enabling comfortable graphical control as well as executing user defined scripts based on arbitrary time dependent functions (Wein, 2002).

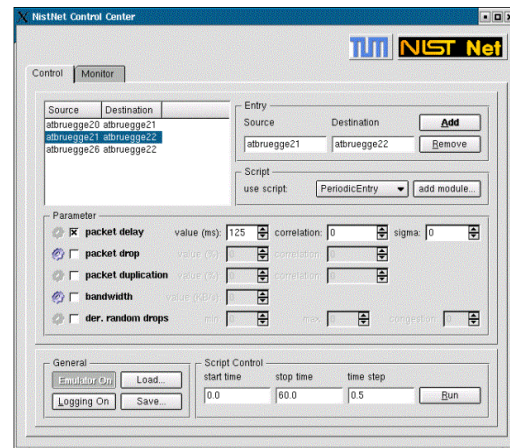


Figure 3.1 TUM NIST Net user interface (control tab).

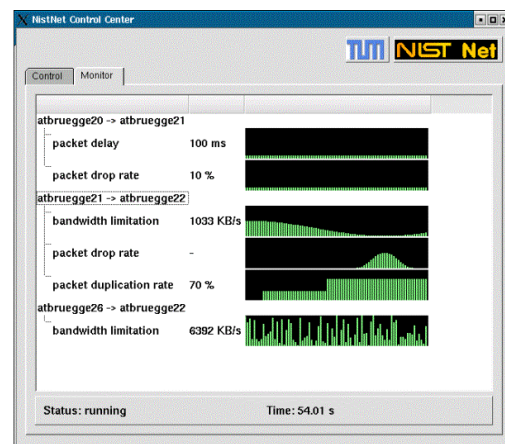


Figure 3.2 Graphical visualization of the emulated network characteristics (monitor tab).

### 4 The M\*A\*S\*H Ad-hoc Network Performance Simulator

Within the Softnet project the M\*A\*S\*H Ad-hoc Network Performance Simulator was developed (Castillo, 2002). This simulator will be used to get sufficiently realistic input data for the NIST Net emulator. The simulator particularly aims at

performance penalties of security extensions in Ad-hoc networks; however, it can as well be used for more generic simulations. The modular concept of the simulator allows simulating various routing protocols, mobility models, and transmission media. The various implemented mobility models allow e.g. simulating cars in city centers or on motorways as well as pedestrians or people in conferences rooms. Figure 4.1 shows the simulators structure for setting up various node classes that share similar settings. In Figure 4.2, a typical simulation-run is shown. On the left side, the configuration window for the different nodes can be seen; the right window shows the current network topology. The different colors of the nodes denote the membership in different node classes.

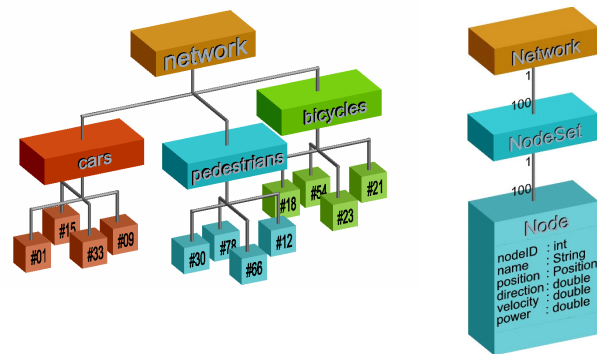


Figure 4.1 node structure of the MASH simulator

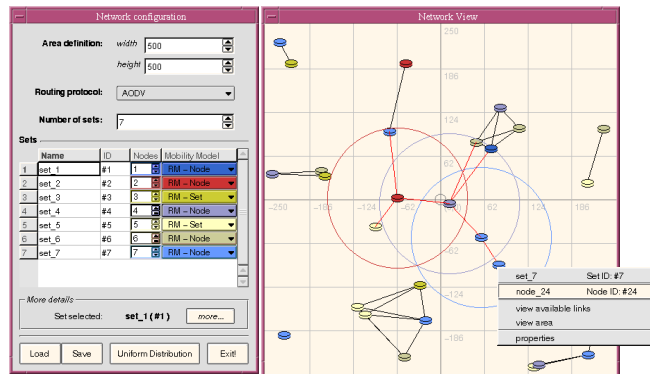


Figure 4.2 MASH configuration and network windows

Another feature of the MASH simulator is its highly simplified propagation model in order to save processing time and memory. As shown in Schwingenschloegl & Xi (2003), this approach does not result in unacceptably high errors, especially when simulating higher layer protocols.

### 5 Emulation Setup for the Virtual City Portal

To test the Virtual City Portal we map the physical architecture of figure 5.1 to an emulation setup with the TUM NIST Net emulator.

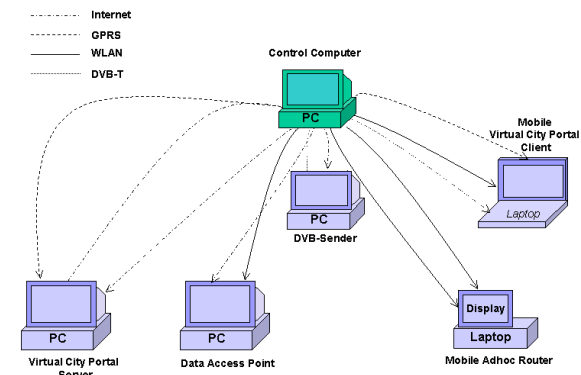


Figure 5.1 Setup of the Virtual City Portal prototype. Each loop represents the logical connection between nodes.

Figure 5.1 shows the logical setup of the emulation environment. For each connection between two nodes there is line from one node via a network hub to the other node. We illustrate each type (Internet, DVB-T, GPRS, WLAN) with a different dashes. Physically, a node needs only be connected once with the hub. To emulate a multi-access terminal a node must provide a virtual interface for each access. For each virtual interface a own subnet must be installed to make sure that every data packet is sent to the router and not directly to a physically connected node.

### 6 Discussion

To test applications in heterogeneous networks, a relatively static setup might be sufficient to define the types of each pair of connection before the testing starts. For more dynamic environments with dynamically changing network topologies we need a more sophisticated approach. One solution would be to enhance the scripting features of our TUM NIST Net emulator. What is needed is to feed the emulator with data about new connections between nodes and their configuration. This is what several existing simulators already provide (University of Southern California, 2003, About GloMoSim – Homepage, 2001). The goal is to couple a network simulator with the emulator so that we can also provide an emulation environment for networks with dynamic setup. Currently we investigate coupling TUM NIST Net and M\*A\*S\*H for this purpose.

### 7 Acknowledgements

This work was partly supported by the Bayerische Forschungsstiftung in the SOFTNET project, part of the compound project FORSOFT 2.

## 8 References

Bauch, M., Reicher, T., Schwingenschloegl, C., Sties, P. and Kosch, T. (2002), *SOFTNET - Software Engineering for Soft Networking*, in SCI 2002. The 6th World Multi-Conference on Systemics, Cybernetics and Informatics, Orlando, Florida, USA, Jul. 2002.

Bogenberger, R., Kellerer, W., Kosch, T., Reicher, T., Schwingenschloegl, C., Sties, P. and Wagner, M. (2003), *Virtual City Portal – A Multi-Network Personal Information System for Automobile Users*, in Workshop on Multiradio Multimedia Communications, Communication Technology for Vehicles, Feb. 2003.

Castillo, J.J.N. (2002), *Development of a Simulation Tool for Ad Hoc Security Performance Evaluation*, Diploma Thesis, Technische Universität München, 2002.

About *GloMoSim* – *Homepage*.  
<http://pcl.cs.ucla.edu/projects/glomosim/>, 2001.

Grimm, R., Davis, J., Lemar, E., Macbeth, A., Swanson, S., Anderson, T., Bershad, B., Borriello, G., Gribble, S. and Wetherall, D. (2003), *Programming for Pervasive Computing Environments*, To be published, (2003).

NATIONAL INSTITUTE OF STANDARDS AND TECHNOLOGY, USA, *NIST Net Home Page*. <http://snad.ncsl.nist.gov/itg/nistnet>, 2002.

University of Southern California (2003), *The Network Simulator ns-2 – Homepage*. <http://www.isi.edu/nsnam/ns/>, 2003.

Njes, R.T., Moessner, K., Lohmar, T. and Wolf, M. (2002), *OverDRIVE - Spectrum Efficient Multicast Services to Vehicles*, in IST Mobile Summit, Thessaloniki, 16-19. June, 2002

Schwingenschloegl, C. and Xi, J. (2003), *Effects of PHY Layer Abstraction in Ad-Hoc Network Simulation*, Accepted for SCI 2003

Wein, W. (2002), *Network Testing Environment for Distributed Applications*. Systementwicklungsprojekt, Technische Universität München, 2002.