

Simulation Concepts to Model Real-Time Properties of Symmetric Multiprocessor Systems

Second Year Report

Lars Albertsson
Computer and Network Architectures Laboratory
Swedish Institute of Computer Science
lalle@sics.se

June 5, 2001

Introduction

Within the research project "Simulation Concepts to Model Real-Time Properties of Symmetric Multiprocessor Systems" we explore the use of complete system simulation for analysing high performance systems with quality of service requirements. By using Simics, a system-level instruction set simulator, operating systems and time sensitive applications may be analysed without intrusion. Our goal is to develop tools and methods for performance analysis and debugging of complex, multiprocessor based real-time systems, using simulation as a building block.

The project plan mentions three main objectives:

- Development of tools for analysing quality of service sensitive applications within simulation models of multiprocessor systems.
- Development of environments for feeding modelled user input to simulation of interactive applications.
- Evaluation of proposed modifications to make conventional SMP capable operating systems support QoS guarantees.

Results

The activity under the first two years have concentrated on the first objective, building tools based on simulation. After obtaining experience with the research field, we think that the most interesting results will be produced by investigating how new types of tools can be built. The research will therefore continue in this direction, by exploring other types of tools for time-sensitive programs.

These specific accomplishments have been made during the first two years:

- Design and prototype of a simulation based “temporal debugger”, capable of analysis of operating system execution time flow. As a case study, we have used the temporal debugger to analyse short time critical sequences in Linux. These results were presented at MASCOTS [5] and the Real-Time Linux Workshop [4].
- Detailed design of a temporal debugger for analysis of user applications. The key component of the debugger is a technique for mapping low-level data, provided by the simulator, to application level data useful to the debugger. The debugger was presented briefly at RTAS [2]. A more detailed presentation has been submitted for consideration [3].
- Demonstration of the temporal debugger by analysing missed deadlines of an MPEG video decoder. It is also demonstrated how the debugger environment can be used to measure performance statistics in the simulated system, and how these statistics are useful for profiling real-time applications and explain missed deadlines [3].
- A support meta-tool (tool to build tools) for probing of a simulated system. This meta-tool has been essential in the construction of the temporal debugger, and facilitates building other tools for probing programs in the simulated system. It is also useful for building other types of support software, such as models for application input.
- A study in related research regarding practical methods for constructing, validating, and predicting real-time system behaviour [1].

Industrial participation

The project is based on the use of Simics, developed by Virtutech, which is a spin-off company from SICS. As the project represents a novel use of Simics, it exposes services the simulator may need to support. It is therefore a source of input to discussions on future directions of Simics. Peter Magnusson, assistant supervisor, is the creator of Simics as well as the manager of Virtutech. He is familiar with the issues involved in complete system simulation, and plays a key role in the education.

Publications

- [1] Lars Albertsson. An overview of practical research approaches to real-time system engineering. Technical report, Swedish Institute of Computer Science, 2001.
- [2] Lars Albertsson. Simulation-based debugging of soft real-time applications. In *Proceedings of the Real-Time Application Symposium*. IEEE Computer Society, IEEE Computer Society Press, May 2001.
- [3] Lars Albertsson. Temporal debugging and profiling of multimedia applications. Submitted for Multimedia Computing and Networking, 2002.
- [4] Lars Albertsson and Peter S. Magnusson. Simulation-based temporal debugging of Linux. In *Proceedings of the Second Real-Time Linux Workshop*, December 2000.

- [5] Lars Albertsson and Peter S. Magnusson. Using complete system simulation for temporal debugging of general purpose operating systems and workloads. In *Proceedings of MASCOTS 2000*. IEEE Computer Society, IEEE Computer Society Press, August 2000.



Tuesday, June 05, 2001

ARTES
Box 325
SE-751 05 Uppsala
Sweden

Attn: Roland Grönroos

To Whom It May Concern:

Virtutech hereby wishes to declare it's continued endorsement and support of project no P1-9805 "Simulation Concepts to Model Real-Time and Dependability Properties of Symmetric Multiprocessor Systems" within the PAMP programme.

The use of complete system simulation to analyze high performance systems with quality of service requirements is of general industrial interest, and is of particular interest to Virtutech.

The use of simulation to support the design and implementation of high performance systems is well established, and Virtutech is having great success marketing Simics™ for such applications. The use within any real-time domain raises difficult issues where input from research projects is highly valuable.

The potential is very large indeed. Today quality of service requirements on high performance systems are slowly but steadily increasing. There simply do not exist adequate tools and methods to handle the requirements that this blend causes to arise. And this at a time when such systems are becoming the default compute/communication platform across industry.

We're very impressed by the work done to date by the principal investigator, Lars Albertsson at the Swedish Institute of Computer Science (SICS). Virtutech is happy to continue supporting this project with access to Simics engineers and technology.

Sincerely,

Peter S. Magnusson
Chairman and CEO
Virtutech AB