Session Summary: End-to-end QoS in real-time systems

Rapporteur: Thomas Nolte*

1 Introduction

This first session of the 4th international workshop on real-time networks was dedicated to issues regarding large scale networks.

In this session, two papers were presented, covering (1) QoS for distributed soft real-time applications over an IP internetwork [1], and (2) experiences using a contract based reservation protocol on top of the Real-Time Ethernet Protocol (RT-EP) [2].

The format of the session was as follows: During the presentations the participants of the workshop asks questions freely, and after the two presentations a more general discussion is allowed.

Charing this first session was Thilo Sauter from the Research Unit for Integrated Sensor Systems at the Austrian Academy of Sciences in Austria.

2 IP Quality of Service Support for Soft Real-Time Applications

This first presentation was given by Binoy Ravindran from Bradley Department of Electrical and Computer Engineering at Virginia Tech in USA.

This work deals with large multi-hop IP based internetworks. The goal is to provide flexible QoS for soft real-time applications by the usage of a scalable QoS architecture, that implements scheduling at the end hosts, and stateless QoS in the core routers. A motivation is that earlier works providing real-time comes at the cost of being less flexible. By targeting soft real-time instead of hard, and by using stateless scheduling in the routers, the authors are targeting a broader application domain.

The soft timeliness requirements are expressed as time/utility functions (TUFs).

In the evaluations preformed, the authors show that their stateless QoS performs better than a best effort implementation, but worse compared with a statefull architecture. A drawback with a statefull architecture is that it requires the intermediate routers to be equipped with a UPA-style real-time packet-level scheduling algorithm. Using the proposed approach, this is not required with allows for greater flexibility, providing good QoS for soft real-time applications.

During the presentation, Binoy was asked to clarify some issues such as where the scheduling is performed and what about arbitration conflicts. The scheduling is performed at the end-nodes only. At the routers messages are scheduled according to FIFO.

There was a question regarding if the use of FIFO as a scheduling mechanism at the intermediate routers makes the proposed idea stateless or not. The idea of using FIFO scheduling is to have something simple running at the intermediate routers, simpler than having a complex scheduling mechanism. Complex scheduling at the intermediate routers is defined as statefull scheduling.

Other questions asked were whether or not it is possible to give end-to-end guarantees. It is not possible to give deterministic hard guarantees. Also, the target of the work is not "traditional" hard real-time. However, it is possible to give probabilistic guarantees.

3 Adding Contract-Based Reservation Services to a Hard Real-Time Ethernet Protocol

The second presentation was given by Juan López Campos from the Electronics and Computers Department at the University of Cantabria in Spain.

Juan presented the communications part of the FIRST (Flexible Integrated Real-Time Systems Technology) project. Here the Real-Time Ethernet Protocol (RT-EP) is used to provide real-time communications in the distributed flexible scheduling framework (DFSF) developed in the project. The communications are managed by the usage of contracts to set up timing and flexible scheduling requirements online. By the usage of contracts, servers are configured (sporadic servers) that manage the communications. The DFSF has been implemented in MaRTE OS and evaluated.

Juan got the question what the difference is between their token protocol and the Timed Token protocol, and why did

*Thomas Nolte is at MRTC, Department of Computer Science and Electronics, Mälardalen University, Västerås, Sweden. email: thomas.nolte@mdh.se

they implement a protocol of their own. The reason for their own implementation is that they moved from using CAN to using Ethernet. Then an implementation of an own token protocol was one suitable solution. Moreover, the protocol is designed for aperiodic traffic. What is desired is to reduce blockings to a minimum. These are reasons for why they implemented their own protocol.

Another question was if they had done experiments with regular Ethernet, without a protocol running on top, just to see how it performs. By using RT-EP, they are tricking the traffic on top of Ethernet. By running pure Ethernet the results might be interesting, maybe even better. Depending on the capacity used of an Ethernet network the probability of missing deadlines is really small. Also, if switches are used even better performance could be achieved. Juan motivates the choice of RT-EP by targeting hard real-time systems.

A question was asked to clarify what the authors meant by saying that RT-EP (their token-passing version of Ethernet) can be modelled together with the application. Juan answered that this modelling of application and RT-EP together means that the communication delays can be taken into account when analysing the application. The question clarified the difference between modelling and analysing. It is not that the application and RT-EP can be modelled together, but that they can be *analysed* together, and with respect to certain time parameters like worst-case response time, etc.

Finally, it was discussed whether the work presented support bandwidth management, and the discussions ended with that it probably can by negotiating contracts. Moreover, is this approach applicable to wireless networks? Yes, but that is part of future works.

4 Concluding Remarks

In the end of the session the discussions moved to more general topics on real-time research. Three of the covered topics were:

- The philosophical question on if it is possible to go away from deadlines was asked. Looking at the system, different entities in the system are having different time constraints. How to handle a distributed system in the most efficient way?

- The discussions moved on to implementation issues when using threads. What can be achieved by using jumping threads between the nodes of the system? What about distributable threads? What can be done with POSIX threads?

- The discussions moved on to QoS. Using IP version 6 one can do more compared to IP version 4. More QoS options are coming up. The question is what the reasonable tradeoffs are. There has been some work done with RT-channels. Here there is a move away from priorities.

Time flies and the discussions continued during the coffee break.

References

[1] K. Channakeshava, K. S. Phanse, L. A. DaSilva, B. Ravindran, S. F. Midkiff, and E. D. Jensen. IP Quality of Service Support for Soft Real-Time Applications. In *Proceedings of the* 4^{th} *International Workshop on Real-Time Networks (RTN'05) in conjunction with the* 17^{th} *Euromicro International Conference on Real-Time Systems (ECRTS'05)*, Palma de Mallorca, Spain, July 2005.

[2] M. González Harbour, J. M. Martínez, J. L. Campos, J. J. Gutiérrez, and J. L. Medina. Adding Contract-Based Reservation Services to a Hard Real-Time Ethernet Protocol. In *Proceedings of the* 4^{th} *International Workshop on Real-Time Networks (RTN'05) in conjunction with the* 17^{th} *Euromicro International Conference on Real-Time Systems (ECRTS'05)*, Palma de Mallorca, Spain, July 2005.