

CSC714 – Real-Time Computer Systems

Term Project Proposal

Anita Nagarajan
Guru Ranganathan
Vasanth Asokan

Introduction:

The main objective of this project is to test and evaluate the QOS guarantees for an application on an XScale Ipaq running Win CE over an 802.11b network. The experiment as such and the metrics obtained would give an insight into the efficacy of Win CE's RTOS capabilities. Additionally, we hope to implement some kind of middleware on top of Win CE, attempting to improve the QOS capabilities.

Background:

WinCE is Microsoft's version of its desktop OS for small devices like handhelds and embedded devices [1]. There are a number of features which Win CE exports for real-time applications [2].

Features include,

- 1) Interruptible ISR handling.
- 2) Thread quantum specification
- 3) Improved timer granularity
- 4) Priority inheritance for resources
- 5) 256 priority levels.
- 6) Priority per thread, rather than per process.

There are some other aspects of Win CE, whose impact on real-time guarantees are unknown. Some of these are,

- 1) Paged file system with potential page-faults during time critical tasks.
- 2) Any file system access is protected by a mutex.
- 3) Demand paging.

Therefore, evaluating QOS guarantees on Win CE needs to take into consideration the above features. Additionally a real-time application that needs to go over a wireless network (802.11b) has to take into account the various potential pitfalls that the wireless medium and protocol could impart on it.

Methodology:

We propose to go about the project according to the following plan of work.

1) Develop Test Applications:

We plan to implement a real-time oriented audio application on top of Win CE and 802.11b. This could be anything from a simple streaming or interactive audio application between the Ipaq and another fixed/wireless node. The application running on the Ipaq might be server oriented or client oriented. The application will use Win CE' s RTOS features. We plan to also implement other applications which simulate aperiodic/sporadic load on the system. These could be like, IO intensive applications like simple file transfer over wireless or other CPU oriented tasks. The effects of varying the background load will be analyzed.

2) Evaluate Metrics:

Some of the metrics that are important for a wireless real-time application are delay, jitter, packet loss, bit-rate, relative resource utilization and others. The measurement of these metrics will enable us to estimate the capabilities of the given real-time system and operating environment.

3) Enhance QOS capabilities:

We plan to investigate running some kind of middleware on top Win CE which can improve the QOS guarantees for the application based on the the results we get for the above metrics. We aim to build the middleware such that the earlier varying background load' s effects are minimized. Some interesting ideas are, proportional share scheduling [4], feedback control scheduling [5] etc.

References:

- [1] Marko Mattilla, Helsinki University of Technology, “*Windows CE 3.0 as Real-time Operating System in Automation Applications*”
- [2] Paul Yao, “*Windows CE 3.0: Enhanced Real-Time Features Provide Sophisticated Thread Handling*”, MSDN Magazine.
<http://msdn.microsoft.com/msdnmag/issues/1100/RealCE/default.aspx>
- [3] Rishi, “*QOS Parameters in Wireless Networks*”, University of Cincinnati
http://www.ececs.uc.edu/~rtoshniw/qos_wireless.pdf
- [4] Jeffay, K. Donelson Smith, F. Moorthy, A. Anderson, J., “*Proportional share scheduling of operating system services for real-time applications*”, Page(s): 480-491, RTSS' 98
- [5] D.Steere, A.Goel, J.Gruenberg, D.Mcnamee, C.Pu, J.Walpole, “*A feedback-driven proportion allocator for Real-Rate scheduling*”, Proceedings of the Third Symposium on Operating System Design and Implementation (OSDI' 99),New Orleans, February 1999.

More Information:

The Project Page can be reached at <http://freebird.thispointer.com/TheProject.html> or alternatively at <http://www4.ncsu.edu/~vasokan/TheProject.html>.