The RTSJ for Prototyping Real-Time Systems: A Case Study
Greg Bollella, Ph.D.

Abstract

Although developing to the application programming interface of an implementation of the Real-Time Specification for Java (RTSJ) might typically be considered more for production than prototyping the APIs provide a such a high-level and abstract interface, especially in the temporal domain, that they are also useful for rapid prototyping. The design of the RTSJ attempts to provide temporal semantics such that real-time developers need not spend significant effort thinking about time. This sounds, at first, backwards, but on reflection one can see that it is a profoundly new and productive view of the real-time developer's task. The APIs allow an almost declarative style of informing the underlying system about the periodicity, frequency, phasing, and asynchronicity needed by the application domain requirements. During construction of instances of the Scheduable class the application simply states the periodicity, period (frequency), phasing, for the instance and the underlying machinery takes care of all temporal issues.

My talk will first cover the RTSJ briefly, pointing out relevant and interesting sections and some of our innovations, such as the real-time garbage collector and then give a detailed discussion of the case study. We built a full feedback controller for a state-of-the-art industrial robot, the ABB IRB 340. This controller has a period of 1 millisecond and was fully realized in Java. The controller was constructed in about a month by 2 graduate students. A video of the demonstration of the IRB 340 at JavaOne 2007 can be found here, http://www.youtube.com/watch?v=xH1yUXd9krU.