GPars - Groovy Parallel Systems

Jon Kerridge

Version 1.0, 2015-10-29

Table of Contents

External Events: Handling Data Multiple Sources

Traditionally, real-time systems that respond to external stimuli have utilised interrupts. Over the years, a great deal of effort has been expended in trying to make interrupt based systems more efficient and easier to program. However, the basic problem still remains that an interrupt causes the halting of the current program, saving its state and then starting an interrupt service routine.

The problems become more complex when an interrupt service routine is itself interrupted by a device with a higher priority. The approach has been to reduce the amount of time when interrupts are disabled. This in itself leads to further problems because it is very difficult then to foresee the precise nature of interactions between interrupts that can then take place. It is these indeterminate interactions that cause problems when systems are running, because it is impossible to test for all the possible interactions, especially in highly complex systems.

The framework built so far, uses parallelism and alternation to capture non-deterministic behaviour, provides a means of describing, implementing and analysing such event driven systems.

Rather than building a system that interrupts itself on receipt of an event notification; rather, build a system that expects such events to occur so that programmers can better reason about its behaviour. In effect, the external event is considered to be the same as a channel communication. Furthermore, the `client-server` design pattern gives us a handle by which the system can be analysed to ensure that unwanted interactions between events do not occur.

Want to read more of this chapter? Download this chapter's PDF here.