# GPars - Groovy Parallel Systems

Jon Kerridge

Version 1.0, 2015-10-29

# Table of Contents

# Communication over Networks: Process Parallelism

**JCSP**, together with `JCSPNet` provide a transparent means of connecting processes, whether they be on the same processor, running concurrently, or if they are executing on separate processes in parallel.

Further, we can simulate processes executing in parallel on a single processor by running each process network in a different `Java Virtual Machine` (**JVM**). In this latter case, there will be no performance gain but the designer is able to assure themselves that the system is functioning as expected. Thus, we can test a multi-processor system in a single processor environment knowing that, at the level of inter-process communication, the system functions as expected but that it will perform in a different manner when placed on the multi-processor system. If the multi-processor system uses specific hardware then that functional test will have to either be simulated or wait until the system can be tested on the actual hardware.

## Process Definition Remains The Same

The definition of the processes that make up the process network, regardless of the processor upon which it will execute and whether or not it is connected to an internal or network channel, remains the same. Thus once a process has been tested, as described in Chapters 7 and 10, we can be assured that it will behave as demonstrated by that testing process.

The design of the `JCSPNet` package architecture means that the actual physical layer used to implement the underlying communication network is completely hidden from the application and in particular the processes and their connection over that network. The only requirement is that the script used to invoke a network of processes on a specific processor will utilise a specific `network-node-factory` compatible with the underlying communications architecture. Each process network running on a processor has to initialise itself as a node within the communications system.

`JCSPNet` utilises the underlying **Java** socket mechanism and thus inherently there are some limitations on the nature of networks that can be constructed. In this chapter, we explore these limitations. Initially, the means by which nodes identify themselves to the network and then define network communication channels is described.

---

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

---