Brendan Koning

2/12/2015

Project 2 Design Document

The Process Manager program has an array of serverProcess structs in order to keep track of all of the servers that have been created. The struct contains information about each server including the minimum, maximum, and current duplicates, the name of the server, and the PID of the server. This information is used to check that a user command does break any restrictions, such as not exceeding the maximum number of duplicates. This information is also used when a user wants to terminate a server by name by linking the name and the PID. The Process Manager itself creates new servers and records the information about these servers and tells the servers what to do. When a user has a request, the Process Manager ensures that the request does not break any restrictions and then gives the request to the desired server using a POSIX signal. Each of the main servers is responsible for creating and aborting individual processes as well as monitoring if any of its duplicates abort unexpectedly. If a process does abort unexpectedly the main server creates a new copy. The SIGTERM signal has been repurposed for the main server threads to allow it to abort its child processes before terminating itself when it is told by the Process Manager to abort.

In order to print information about what is occurring with each of the servers and the Process Manager I developed a method that takes only a message and prints out the message with the process that sent the message. This makes it simple for the programmer to print out messages without having to keep track of what process they are currently working with.

Printing the status of the Process Management system is done using signals. The Process Management system sends a signal to each of the servers individually, which then prints out the desired information about itself.