

General Protocol For Master-Worker Interprocess Synchronization

Prakash Manandhar, Aug 5 2008

Here we are considering a “master” process (for example a GUI interface) that initiates, monitors and controls “worker” processes that take a long time to execute. Requirements:

- 1) master process needs to pass some parameters to the worker process
- 2) master process needs to spawn and terminate the worker process
- 3) worker process needs to relay status and progress of the “work” to the master process
- 4) “work” can be paused or canceled by the master process

I. CONNECTION

The master process starts the executable for the worker process and connects to the worker’s standard input for sending commands to the worker and connects to the worker’s standard output for receiving status from the worker. The worker can use the standard error for diagnostic purposes, the master can choose to leave the standard error output alone or connect to that too to display error messages. Parameter values can be passed as ‘command line arguments’ to the executable.

Commands between the Master and Worker are composed of ASCII characters including spaces and terminated by a new line.

II. COMMANDS FROM MASTER TO WORKER

When the process is first started a series of parameter values are passed as argument to the worker. Each parameter is passed as a pair of arguments, the first one is the *name* of the parameter which is immediately followed by the *value* of the argument.

When the connection has been made, the master sends the PING command to which the worker replies READY if it is ready to start a process or BUSY if it is working. The master can try several PING commands if it does not receive a READY or BUSY or PAUSE. The worker can also send back the READY command on it’s own when it finishes a job.

The master can send the CANCEL command to cancel an on going process. The worker responds by finishing any atomic task and then replies with a READY to indicate that the job has ended.

The master can send the PAUSE command to which the worker responds by finishing any atomic task and then replied back with a PAUSE. A paused process can be continued with the RESUME command. The worker can resume work and send status about progress.

The QUIT command is used by the Master to indicate the process should end. If the worker is BUSY, it should CANCEL and quit. Otherwise it can quit directly.

III. COMMANDS FROM WORKER TO MASTER

The READY, BUSY, PAUSE commands are as described above. An additional STATUS command is sent by the Worker as it finishes one atomic task and switches to the next. The STATUS command consists of three parameters two numbers and one text: STATUS *{total task quantity}* *{quantity completed}* *{status text}*, for example: STATUS 2 12 'Data for February processed.'\n

{end of protocol description}