Tasks:

* Get CS guys to install GINI backend on machines and gather a list of hostnames (lab2-17.cs.mcgill.ca) ->Update $PATH file
* gbuilder GUI options: add checkbox for ‘run frontend only’ => force user to enter backend hostname before saving options
* gbuilder GUI: add provisions to handle ‘are you still there’ queries and reply to them via user input
* build a dispatcher (Python twisted script):
	+ IP interface
		- listen for and receive requests from gbuilder
		- replies to requests with hostname, IP or error (if all machines are turned off)
	+ Local software communication interface
		- communicated with scheduler
* build a scheduler
	+ User Interface:
		- Add/remove available workers
		- Loot at system log
		- Manually kill connections
	+ Polling engine: available machines:
		- upon dispatcher requests
		- every 2 minutes to maintain log
	+ Local software communication interface
		- Receive/reply to dispatcher
		- Send ‘are you still there’
		- Send propose better machines if excessive workloads

Present sequence:

1. gbuilder calls gloader and program starts

New proposed sequence:

1. gbuilder asks Dispatcher for a worker
2. dispatcher checks with scheduler for less busy worker
3. dispatcher returns IP of worker to gbuilder
4. gbuilder calls gloader via ssh tunnel and program starts
5. […] scheduler keeps a log and polls for open sessions – sends reminders every x time

Useful commands

$ ssh top > somefile.txt

$ ps

$ ps | grep someprocess > somefile.txt