Instructions

Difficulties

The project ran successfully on all but one of the computers we ran it on (it worked on ssh-linux.ece.ubc.ca, as well has our linux, mac and windows machines). The computer that was unsuccessful was a Windows 7 PC. Errors that occurred on the unsuccessful machine were garbled responses from the SmartMeter and time out errors in the terminal between the smart meter and server. Also, the port that the sockets bind to is static, and so you must make sure the processes are dead before rerunning the project or else you will encounter binding exceptions.

If you are unable to run the project on a windows machine, try ssh-linux.ece.ubc.ca. If that fails, notify us and we can demonstrate it working on one of our computers.

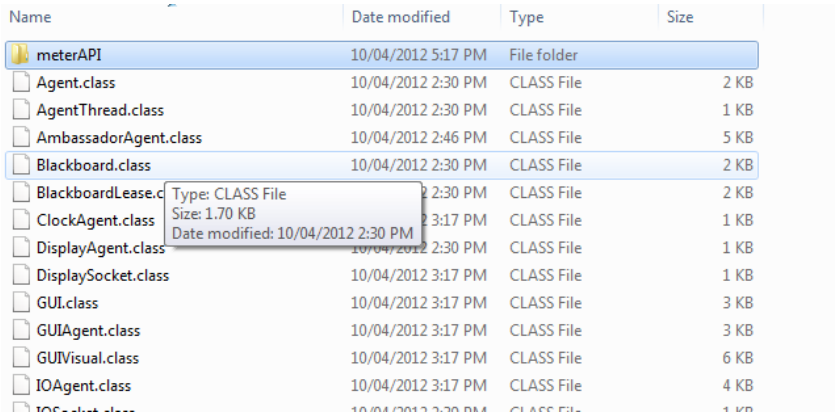
Compiling

When compiling, make sure that meterAPI/* is a folder in build path, and it includes the API class files.

A precompiled JAR file is available here: http://dl.dropbox.com/u/49412219/final-jar-submission.jar .

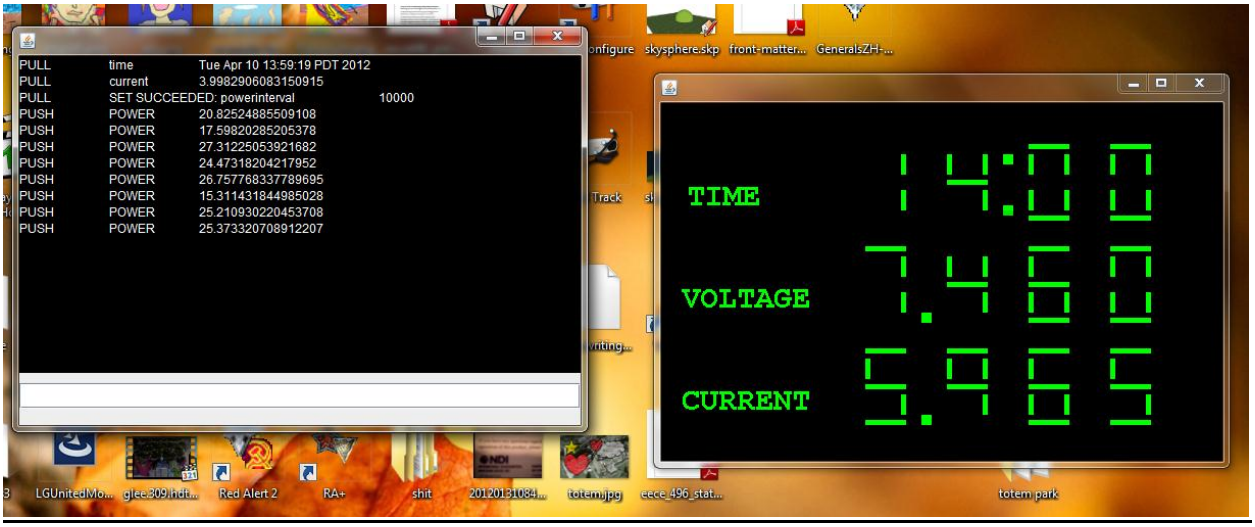
Running

For compiled class files, make sure the folder meterAPI/* is in the runpath, and it includes the API class files like shown in the below screenshot.



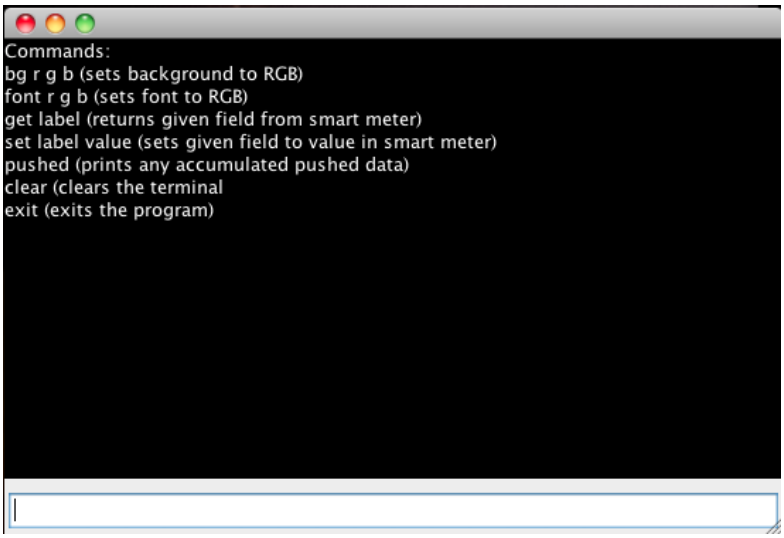
For the JAR file, make sure the folder meterAPI/* is in the same directory it is ran from.

Using



The window on the left is the Server Terminal, the window on the Right is the emulated 8-segment display on the SmartMeter.

In the Terminal, the PUSH entries are the values sent from the smart meter automatically, which are the hourly average powers. The PULL entries are the entries requested from the server using the get command. The terminal commands are shown in the below figure.



As explained in the report, the “set” and “get” commands in the Server terminal allow the server to become a remote agent. The sets and gets act on the blackboard. Due to time constraints, the sets and gets can only perform a subset of the possible manipulations on the blackboard.

You can get the following: “time”, “power”, “voltage”, “current”, and the quality of power attribute “variancepower”. So to get time, you would type “get time”.

Sets can act on “powerinterval”, which is the interval between power pushes, “watchdoginterval”, timeout interval for a heartbeat that triggers a system restart. The latter value can be modified to force a system reset.

Gets will return strings, doubles, and sometimes tuples.

SmartMeter – Server Protocol

Hourly average power is pushed from the smart meter regularly. Pulls are done through the terminal with get.