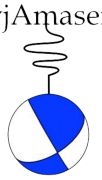
PyjAmaseis



Project Log Book – Semester One

Week 1

- Project Briefs were sent out via Email by Mano

Week 2

- First meeting for BTech 451 with Mano where project structure explained and details given out

Week 3

- Second meeting for BTech 451 with Mano - Python Interface for Seismology project was assigned to me

Week 4

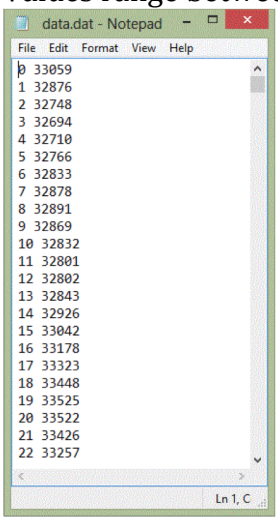
- I had first meeting with Dr Kasper van Wijk and Dr Patrice Delmas
- Project motivation and direction explained
- Immediate milestones discussed, these were setting up the environment with TC1 and Python, and receiving data from Arduino
- Project requirements briefly outlined
    - Develop a Python Application that can work as a Seismology Data Analysis, Recording and Sharing suite.
    - Core features to implement are:
        - Data Collection from TC1
        - Plotting Live
        - Saving into mseed
        - Sharing by uploading to server

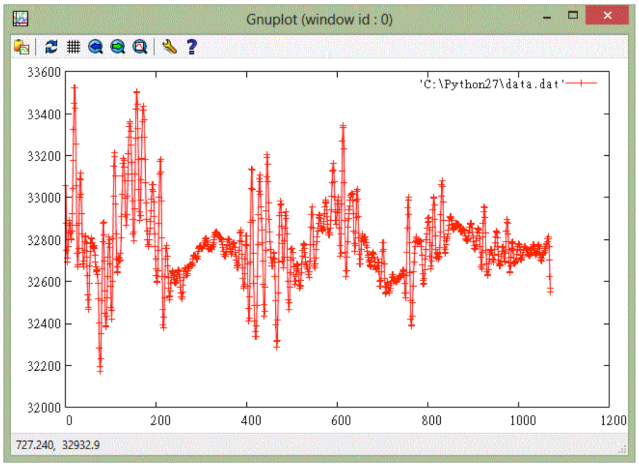
Week 5

- Had Meeting with Dr Kasper van Wijk
- Was given a TC1 seismometer to work with
- Setup python environment and downloaded PySerial module required to start reading data coming from Arduino (connected to TC1) via a serial (USB) port
- Created a simple python application (PyjAmaseis v1) that read and prints values from TC1

Week 6

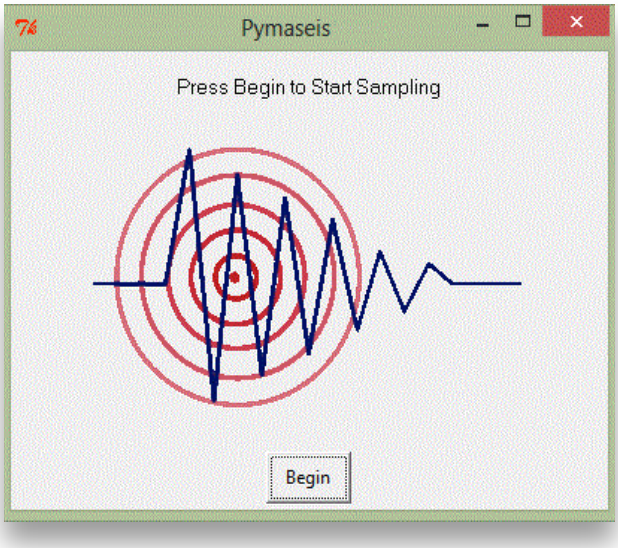
- PyjAmaseis reads and saves writes data into a .dat formated file then creates a static plot of the data after the end of x amount of minutes
- GNUplot used to plot data.dat file at the end of data collection
- Values range between 31000 to 34000





Week 7

- Had Meeting with Dr Kasper van Wijk
- PyjAmaseis v2 - Simple GUI added with an image and button on the menu
- Start sampling button on menu windows opens channel and starts reading from serial port - Ctrl+c stops the application - then sampling completed window shown



Week 8

- Had Meeting with Dr Kasper van Wijk
- Prepared and practiced presentation
- BTech Introductory Seminar

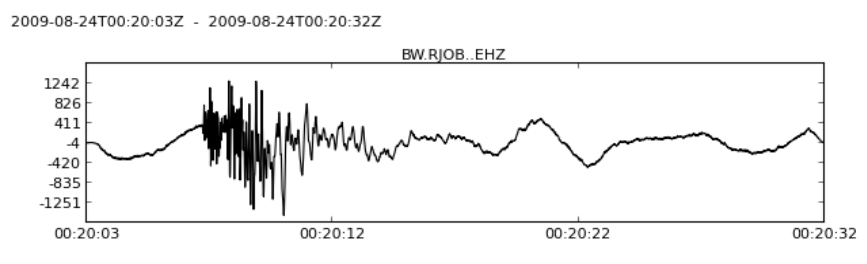
Week 9

- Meeting with Jami Johnson to go over the process of saving trace information (seismic data from TC1) into mseed (mini standard for exchanging earthquake data) format
- I received a python source file with the procedure on saving a data from a trace into mseed. Scan.py
- I learnt that a trace is an object similar to HTTP requests with headers and a payload. The trace contains headers an array of data (samples collected from TC1)
- With the help of the ObsPy frame work I can create mseed files by saving trace objects that I create into mseed file formats

ObsPy
A Python Framework for Seismology

Week 10

- Functionality successfully implemented - Saving mseed files every x amount of time
- After saving each mseed a static plot is created using the saved trace via the command trace.plot()



Week 11

- Was unable to commit time to project due to being occupied with work from other courses and projects

Week 12

- Was unable to commit time to project due to being occupied with work from other courses and projects