Luke Braby BxGrid Fall 2022 Research Writeup

This semester, I worked with Professors Flynn and Thain to create a system to allow researchers to more easily query and materialize data. After multiple iterations, I ended up with two programs (query.py to get data from BxGrid and store it in a csv file and materialize.py to store the queried files with their metadata in a directory tree).

To start the semester, I began by setting up my virtual environment and becoming familiar with mysql-python-connector. Not having ever worked with SQL, this was a bit of a slow process, but as the semester progressed I gained more experience thanks to Professor Weninger’s Database Systems Concepts course. Weeks one and two were dedicated to gaining experience querying the data and eventually chirping some files into a directory.

During weeks three and four, I worked out how to chirp files into a directory with a schema provided by the user. For example, if the specified schema was *subjectid/date*, files would be grouped by subject id and date and then placed into subdirectories according to those groupings. When chirping files, I made sure to try all possible replica locations that the file could be stored on. Lastly during these two weeks, I analyzed the performance of chirping files for varying counts.

For weeks five through seven, I began work on preliminary versions of “bxgrid in a box”. Initially, this looked more like querying a SQL database (ex “bgbox> materialize irises_still as date/weather where subjectid = ‘nd1S04473’”). However, after showing this version off in our weekly meeting, I decided to converted it into a command line tool. At this stage, it was only one command that would query data and materialize it all in one step. I created flags to allow the user to materialize into an already existing tree, to get metadata only, and to specify the head directory for the materialization. As these weeks progressed, general improvements (e.g. storing credentials/history and showing a progress bar with chirp feedback) were made to the tool.

During the last weeks of the project, I broke this initial single command line tool into two separate tools (one to query and one to materialize). The query tool stores data in a csv and allows for the user to make custom SQL queries should they want to. The materializer then reads in data from the csv and, like the previous version of this tool, chirps those files into a directory tree with a head directory and schema set by the user.

After creating this last version of the tool, I met with Ben Tovar to use Python’s chirp module instead of calling it using Python’s subprocess library, and I showed the finished project to Patrick Tinsley to get his input as a graduate student who would be using “BxGrid in a box”.

Now that the semester is wrapping up, there is still a bit more work to be done. When downloading a file, it would be helpful if that file was stored in a hidden folder and then accessed using symbolic links in the directory tree to allow for easier file reuse. Having a way to track the history of materializations (possibly in a way similar to git) would be another meaningful change, especially for when files are excluded after materialization.