Big Data and Analytics - Weatherlink
Personal Report
Matt Winchester
Milestone: Assignment 1

I wanted to find a dockerized solution for deploying a Hadoop cluster. There were lots of examples of people doing similar things online, but none of them seemed to work out of the box, so we had to work to get a solution. Tharun and I took charge of the second Pipeline we implemented. (Alper did the first, the Airbyte -> PosgreSQL approach)

To get this working, we modified a docker-compose example we found on github to use newer docker images for the hadoop cluster (including a single data node, a name node, a resource manager, and the spark job runner). The images for older versions did not seem to exist anymore, so we found newer images on docker hub and used those instead.

Tharun had not used docker before, so that was a fun thing to teach. We used docker compose to stand up an HDFS cluster on my laptop, and then figured out how to use Spark to run a Scala defined job to read csv data, manipulate it, and then store it on HDFS.

Each step was a little tricky to figure out, there were some networking issues we had to account for that did not seem to behave as intended. I defined a custom network for all of the container to share instead of just using localhost, but once it was working we had a way to keep the HDFS cluster running, and then as needed we were able to modify the Scala file and relaunch the spark job by re-running the Spark container. I liked this approach a lot, since it is platform agnostic. (Or at least it SHOULD be, we had some issues getting the dock images to deploy on Tharuns Windows PC, he had to install an ubuntu VM to get things working…) and allowed us to make changes to the spark job without having to redeploy the whole system.

Our spark job was very fast, the same dataset that Alper read in with Airbyte and stored to postgresql, we read in, manipulated by sorting, and then wrote back out to HDFS in a fraction of the time. His airbyte solution got the same csv file into postgresql in 5 minutes, our solution took 4 seconds.

I think we will not end up using HDFS for our project, but I really wanted to try it out since I think it is a fundamental Big Data tool and understanding it practically was a very good exercise. We will definitely need some ad-hoc batch processing capabilities for our project, so now we have practical experience with two potential solutions, both Spark and Airbyte.