Lab2 – Part 1 Report

Walter lozier

i. Introduction - For this lab I used Hive as a means of storing and querying large data sets.

ii. Objectives - The objectives for this lab was to create a number of tables and import a data set into them using Hive. Then, again using Hive and the built-in functions, come up with 10 queries of the data set.

iii. Approaches and Methods - For this lab, the approach I used was to rely on my previous SQL experience and simply modify the queries I am used to into the Hive syntax, which is actually incredibly similar.

iv. Workflow - The workflow for this lab was very simple 1. Download data sets from [https://www.kaggle.com/claudiodavi/superhero-set/data#heroes\_information.csv](https://www.kaggle.com/claudiodavi/superhero-set/data" \l "heroes_information.csv) 2. Create tables in hive based on the data set (same number of columns and types) 3. Import data into hive tables 4. Run queries on the tables

v. Data sets - [https://www.kaggle.com/claudiodavi/superhero-set/data#heroes\_information.csv](https://www.kaggle.com/claudiodavi/superhero-set/data" \l "heroes_information.csv)

vi. Parameters - The parameters for this lab were to use one of the assigned data sets. Use Hive built-in functions for your queries. And run at least 10 queries to answer interesting questions about the data.

vii. Evaluation and Discussion - It is hard for me to truly evaluate the performance of Hive since it is necessary for me to run Hive through a virtual machine, which has severe performance impacts. For the medium sized data sets I worked on in this lab, the performance was not very good compared to SQL queries. Perhaps with much larger data sets the performance differences would become more apparent.

viii. Conclusions - Hive is a powerful tool for analyzing very large data sets while keeping the familiar syntax and operation of SQL. Sub-queries were more difficult for me in Hive than I was used to in SQL, but joins are still very simple.