Product Backlog for WeatherPipe

Team 16
Stephen Harrell, Lala Vaishno De, Hanqi Du, Xiaoyang Lin

Problem Statement

Difficult-to-use data resources or expensive-to-access data have often confined weather researchers to limited data sets restricting their sample sets for testing their hypothesis and theories. The partnership of National Oceanic and Atmospheric Administration (NOAA) with Amazon that would make it openly accessible for anyone to access past radar data could potentially transform the way this problem has been approached thus far. We plan to write a general-purpose historical weather data analyzer that would employ Map Reduce (http://research.google.com/archive/mapreduce.html) to load datasets from Amazon S3 and directly run the required analyses. This application would thereby, be able to create novel historical analyses with dramatically less work than was possible in the past.

Definitions

1. **MapReduce:** A programming model used to process very large datasets across many computers and return a much smaller dataset.
    a. https://en.wikipedia.org/wiki/MapReduce
2. **AWS (Amazon Web Service):** A web based api that can provision many types of web services including S3 and EMR instances.
    a. https://aws.amazon.com/
3. **S3 (Simple Storage Service):** A web based object store used with AWS services.
    a. https://aws.amazon.com/s3/
4. **EMR (Elastic Map Reduce):** Dynamically provisioned Map Reduce clusters in AWS.
    a. https://aws.amazon.com/elasticmapreduce/
5. **NetCDF:** A file format used for array-oriented scientific data.
    a. http://www.unidata.ucar.edu/software/netcdf/

Background Information:

In the past accessing large amounts (5 years or more) of Radar data has been prohibitively expensive and only available to Principal Investigators (PI) with large grants and typically only able to analysis for the life of the grant. Most analyses done outside of these grant periods are relatively small in comparison using tools that are typically written to read one file at a time. [1] The project we are proposing will make analysis of this data cost effective by allowing anyone with an AWS account to pay for a run of their own analysis on the 30 years of data that is currently being hosted on s3. [2]

1. https://www.ncdc.noaa.gov/data-access/radar-data/radar-display-tools
2. https://aws.amazon.com/blogs/aws/announcing-the-noaa-big-data-project/

Requirements

- **Functional Requirements**

1. As a user, I want to be able to choose specific time periods from the data such as a range of dates or as a specific scheme of dates so that I can limit and choose the data set to the sample for my calculations.

2. As a user, I want to be able to do my analysis from the command line with a command line tool.

3. As a user, I want to be able to carry out simple operations (such as mean, median, mode, standard deviation, etc) on the chosen data set so that I can analyze historical data.

4. As a user, I want to have a simple config file to define inputs for each analysis, if time permits.

5. As a user, I want to be able to get results as a single value or in a table format so I can interpret the results, if time permits.

6. As a user, I want to be able to get results in the form of a histogram or a multivariate histogram or a 3D Space average diagram so that I can better understand the results of my analyses, if time permits.

7. As a user, I want to be able to save the results in a file from a certain run so that I can use it later.

8. As a user, I want to see a time remaining indicator so that I know how long it would take so I will know when it is done, if time permits.

9. As a user, I want to be able to be given the choice to restart, terminate the process at any time so that I can control the process, if time permits.

10. As a developer, I want an index of s3 datasets to be able to retrieve specific data related to dates and geospatial shapes so I can find the data that is relevant to my research, if time permits.

11. As a developer, I want to be able to retrieve radar data from s3 so that I can do analysis.

12. As a developer, I want to be able to authenticate an AWS account so that I can get access to compute resources.

13. As a developer, I want to be able to use EMR so that I can do analysis.

14. As a developer, I want to unit tests so as we make changes we can test them quickly, if time permits.

15. As a developer, I want to be able to handle failures by reporting so that I can figure out the failure.

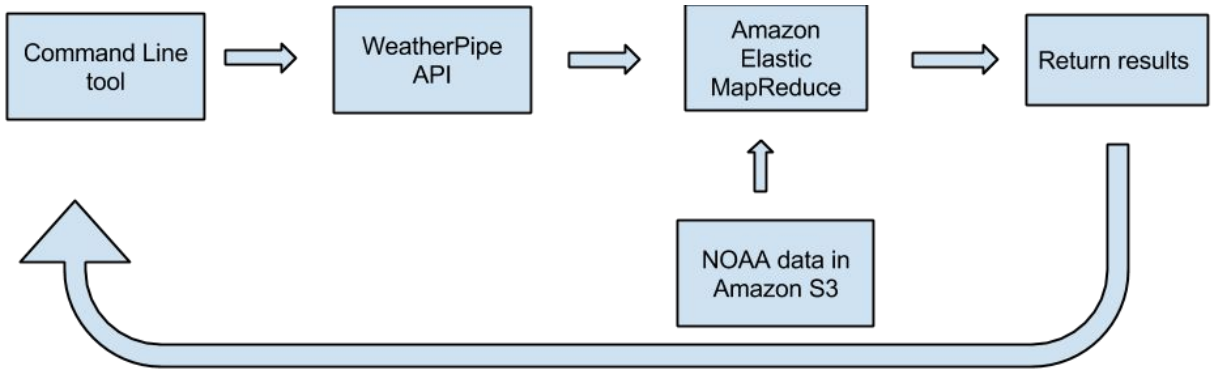
16. As a developer I want to be able to enumerate the different kinds of analysis that are possible so that I can generalize a solution.

17. As a developer I want to be able to enumerate the different kinds of data get output so that I can generalize a solution.

- **Non-Functional Requirements**

1. **Optimize by cost :** Optimize the cost of using amazon servers by finding the the most economical server-time combination that would minimize the cost of running the analysis in a suitable time frame. There are two types of research that dictate time frames. The first is operational research which is data that goes into a current forecast. The second is research that may go into a paper or journal article. The time frame for the first type of research is typically 4 hours and two weeks for the second one.

2. **Design:** By using a modular design such that radar data can be swapped with other data, this system can be used as a dock for weather processing using different sets of data from different sources. The command line tool will use the WeatherPipe API. This will allow us to develop other interfaces if we need to. The WeatherPipe API will take an Amazon security token, type of analysis, specific dates and location then it will submit the job to MapReduce. The analysis will pull the correct data from the NOAA repository of radar data in S3 and run the MapReduce. The results will be returned to the command line tool.



3. **Output Format:** The output of our initial analyses will be integers or arrays of integers. We will be enumerating different kinds of analysis beyond this with Dr Baldwin next week.

.

4. **Security:** We will be using Amazon AWS Security Tokens so we need to use them and delete them out of memory quickly.

5. **Demos/Tutorials:** We will write documentation to explain how to submit analyses as well as get results.

6. **Failure Modes:** For individual failures within a piece of the analysis we will retry at least 5 times before giving up. On the whole for the entire analysis we will give the researcher control over the failure rate and at which point the job should be abandoned wholy.

7. **Better analysis tools:** We will make the analyses pluggable so weather researchers can write their own analyses. Each Analysis requires it's own inputs and outputs. Although many of them overlap each analysis will require it's own config file, analysis code and output code.