Sprint 2 Retrospective
TEAM 16
WeatherPipe

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

1. What Went Well

In this sprint we were able to get most user stories finished and added functionality to our prototype. We were able to get a signifigant portion of unittests finished, have an auto build system and implemented a way to send objects between the mapper, reducer and command line tool.

a. list completed user stories here

1. We completed our user story 1(a), which is from last sprint, which allows users to do the analysis from the command line with a command line tool.

2. We completed our user story 1(b), which is from last sprint, which allows users to be able to get statistics and estimated cost from the tool.

3. We completed our user story 1(c), which is from last sprint, which allows us to handle failures by reporting so that we can figure out the failure.

4. We completed our user story 2 which allows users to do analysis on more than one days worth of data.

5. We completed our user story 3 which allows users to have an integrated build system so that the MapReduce jar is built with the command line tool.

6. We completed our user story 4 which allows users to control many parameters of the running of the analysis.

7. We completed our user story 5 which allows users to do analysess that have arrays for outputs.

8. We completed the first part of the user story 6 which allows us to use the unit test to test all the functions that is possible for.

9. We completed our user story 7 which allows users to modify two simple classes to change my analysis without having to modify the bulk of the project.

10. We completed our user story 8 which allows users to make faster searches on NEXRAD data files, therefore, allowing a faster and larger analysis to be carried out.

<u>Additional features that were also added in this sprint:</u>

1. We added better status updating and cleaned up warnings/non-important error messages.
2. We added the ability of map and reduce to work on custom objects (output configurations need to be changed in order to do that).
3. We added technical documentation allowing users to build and run the program on their own Unix-based machines or at a lab at Lawson Building, Purdue University.

2. What Did Not Go Well

As some of the user stories required us to perform jobs that we had never done before, we often under-estimated the time that we would need to complete the task. As a result, some tasks did not get completed:

a. We didn't complete the second part of the user story 6, which is to create regression test for entire command line tool.
    i. We ran out of time in making a tool for regression testing as greater effort was made in keeping the codebase functional and working at all times.

b. We didn't complete the second part of the user story 4, which is to check to see if that the data is available for the specified radar station.
    i. We ran out of time as well as had additional problems as mentioned in part c below.

c. We realized that some of the team members were not building the program and testing for their changes before pushing their code.
    i. This caused a a massive effort in integrating broken or half-working code in order to keep the codebase functional at all times.
    ii. It also made it more difficult to keep on par with our individual tasks as it needed time to fix the codebase.

3. How we can improve

In Sprint 3, we hope to continue improving our communication skills and focus as a team to get as much work done as we plan to. In addition, we must also keep the code base working and functional at all times.

a. We will be creating a help index with all possible flags so that everyone understands and can individually run the map reduce jobs with different configurations. This will help in standardizing flag rules and solve any confusions about flags.

b. We have created build-and-run instructional documents that would be used in the next sprint by all team members to ensure that individual increments are tested and working before they are pushed onto the codebase. This would ensure that no huge effort is needed to integrate a team member's code.

c. We will convert the test suite to be run from outside the program, instead of from within the program as that would permit a larger variety of testing to be carried out. In addition, we hope that this would force a more test-driven approach for the team to develop code that passes the test cases.

d. We will be finishing up the user stories from the last sprint and adding new stories that are yet to be completed from the product backlog. Additionally, as we gain more knowledge about this system, we would like to keep the sprint documents updated when we make changes to our sprint goals.