# Research Report of MIQS

## Introduction

Scientific applications often store datasets in self-describing data file formats, such as HDF5 and netCDF, however, existing solutions extract the metadata and store it in external database which offers a less efficient way to handle. In this project, I am going to achieve two tasks which are range query against numeric attribute values and compound metadata search with affix search against attribute name.

## Define the problem

In this project, the goal is to extend MIQS by implementing several specific functionalities, for me, I need to finish range query against numeric attribute values and compound metadata search with some specific conditions.

I was confused at the beginning because there is a very limited resource and documents relate to the existing project. Even the README on a Git repository can hardly lead to a successful compile of the development environment. It took me some time to figure out how to set up the dev environment and compile each section successfully on a mac OS. After the compilation of the code, I start to go through the source code in MIQS, and shortly I realize that It’s impossible to fully understand the whole picture within 20 hours (I will explain why I only have roughly 20 hours to work on this project).

After some research I try to find a basic workflow of the project which is, once I got metadata, no matter it loads from MDB or AOF, after loading the file, I need to parse it into a radix tree for future querying.

## Workflow Diagram

Text

Description automatically generated with medium confidence

## Tasks

* Range query against numeric attribute values
* Compound metadata search with affix search against attribute name and affix search/range query against attribute values

## Plan

|  |  |
| --- | --- |
| Date | Tasks |
| From 3-25 to 3-29 | Write project proposal and setup dev environment |
| From 3-30 to 4-10 | Finish range query against numeric attribute values |
| From 4-11 to 4-18 | Discuss with Jacob and finish Compound metadata search |

## Resources and Conclusion

Overall, it’s an interesting topic but it’s not doable within one month, because I have three classes this semester and this CS5352 also has three other projects required, plus I have my own thesis topic to work. So, the truth is I basically only have 20 hours to work on this research-oriented project. Before I dive into the project, there is no clear documents and diagrams to demonstrate the overall workflow of the system, the only document relates to the system I can find is the html version of the source code that generated automatically by the Doxy. As I went through the source code, I realize that there has not enough comments out there, also the design pattern of the code is hard to trace because it designed by procedure oriented rather than object-oriented, not even mention the S.O.L.I.D principal. I am not trying to find some excuse for me, but the truth is it’s hard to complete this task within the very limited period, and probably that’s why we have CS7000 to offer the opportunity to do such research-oriented study.