Aniruddha Pispati

Extracting Leftovers from Large Datasets on Blue Waters

Introduction:

Blue Waters contains stored in a collection of relatively big files, around 50GB each. The records are variable in length and are not sorted. Each record contains multiple data points. The system monitoring data in the datasets of Blue Waters contains such important data points that need to be extracted; however, we need to find an efficient method for the extraction, since doing it in a straightforward way will take too much computing time. Different methods had to looked up and thought about to find the quickest way possible. After indexing the data files and storing them in a database, we need to perform various statistical operations to get valuable information and results from the data.

Abstract:

This study will effectively save a lot of computing time. The project is divided into three components- the file reader, the indexer and the web interface for interaction. From there, we find the fastest way possible to extract the data and make the appropriate modifications. The quickest method is noted down and is to be analyzed for future purposes. Statistical information is also obtained from this data to show concrete results

Methodology:

Our proposed solution was to index the data files by preprocessing the data (although it could be done on-the-fly as well) and store the indexes in a database to accelerate all subsequent queries. Two separate scripts were written- one for reading the entire file and one for indexing the data and then storing these indexes. The code for the scripts is written in C (in MPI for the parallel processing to send messages) and are still be modified to enhance results.

An interface was also designed for user interaction. The database used was PostgreSQL, while Django was used for the backend and Mezzanine (a Django CMS) was used for the frontend. The user would need to enter the node required using a comma-separated list. Statistical information can be obtained by selecting the choices from a dropdown list containing operations like dropdown list with min, max, sum and mean.

Literature Review:

Similar research has been done before in this field. In one such experiment, a new kind of database was created for the efficient querying on data files. “Before being able to submit queries, the data must first be loaded, which transforms it from the raw format to the database page format” (Ioannis Alagiannis, Renata Borovica-Gajic, Miguel Branco, Stratos Idreos, Anastasia Ailamaki 113).

Previous research has shown that there are several proposed efficient and scalable spatial indexing techniques for big data stored in distributed storage systems. A data structure called geohash can be used to develop a lightweight spatial index for big data stored in a database called an HBase. Sorting approaches have also been used by sorting spatial objects in order to be able to index them.

Previously, various techniques have been looked into for data mapping, such as the Pyramid technique. Multidimensional data points can be mapped to one-dimensional space to exploit single dimensional indexing structures such as the B+-tree. Research into Generalized structure for data Mapping and query Processing (GiMP) shows that it could support extensible mapping methods and query processing.

Results:

Once the project is completed, we shall find that essentially our methodology makes for the extraction to be taking place in the shortest amount of time. Our proposed solution makes the extraction very efficient. With the indexes of the data now stored in a database, the data could be made useful to other groups. The results that we get from the statistics involved will give us important data points and useful information.

Discussion:

The main significance of the results would be the output- the statistical results obtained at the end. Information can be extracted through several operations to find data like mean, median, standard deviation and maximum. The work done on the project should provide a fast and a simple way for support staff to access OVIS data which will help the Blue Waters group at solving a lot of issues with user codes, system health and reporting (a suite of High Performance Computing (HPC) monitoring, analysis, and feedback tools).

The research done will also help with research in other areas. This is done with investigations into operations of Blue Waters, which in turn will help us to produce a list of requirements for new supercomputers.

Conclusion:

We find that by using MPI to code the reading and indexing of the file, we save a huge amount of time vis-à-vis directly extracting the necessary data in a straight-forward way. The statistical data obtained at the end should be analyzed further. More research should be done in similar fields where this method could be applied when examining the operations of Blue Waters.

References:

- Generalized multidimensional data mapping and query processing

  By Rui Zhang, Panos Kalnis, Beng Chin Ooi, Kian-Lee Tan

Aniruddha Pispati

- Efficient spatial query processing for big data

  By Kisung Lee, Raghu K. Ganti, Mudhakar Srivatsa, Ling Liu

- A sorting approach to indexing spatial data

  By Hanan Samet

- NoDB: Efficient Query Execution on Raw Data Files

  By Ioannis Alagiannis, Renata Borovica-Gajic, Miguel Branco, Stratos Idreos, Anastasia Ailamaki