Term Project: CS480 Principles of Data Management
High-throughput Data Storage

1 Introduction
The objective of the Term Project is to provide a solution to the challenge of high-throughput data storage. Data volumes have been increasing exponentially. The rates at which data are produced and consumed by monitoring devices and software have also increased substantially. Data is typically written to stable storage to allow post-processing and visualizations. Efficient access to the stored data elements is a key requirement for data analytics or visualizations.

As part of this term project, you should design and build a storage system that can handle 100s of thousands of small files efficiently. The size of any given file in the system will not exceed 50KB. Since the final target of dataset will be voluminous, searching dataset through the file name will be extremely hard. The system should provide simple time series query interface to support advanced data search. You are not allowed to use cloud storage systems such as HDFS and Cassandra, or databases either relational or object-based. The storage framework must be designed by you from scratch.

Note: I will update this document to clarify any questions that you may have.

2 Elements of the term paper

2.1 Store small files
Your system should be able to store 100s of thousands small files efficiently. The directory structure that you use is completely your design choice. Your system must be compatible with the Unix/Linux file system to simplify the testing.

2.2 Features
Your system should provide a few command line interfaces. This includes:

- **store_file**: This command saves file(s) specified with its local path(s)
- **store_dir**: This command saves files under the specified directory with its local path. If the directory has a nested directory structure, this command should work for all the files recursively.
- **retrieve [start_time] [end_time]**: This command returns all of the data containing timestamps that fall between the start_time and end_time.
- **count [start_time] [end_time]**: This command returns the number of data items containing time stamps falling between start_time and end_time

Each system must also introduce one special feature that can distinguish your system from others.

2.3 Software Design
You must also provide your own software design. Description of the design must be included in the final report. Design features reported in this document may include:

- Data staging
- Metadata management
2.4 Testing your system
To test your system, you should create a test dataset. Each file in this dataset should include a timestamp. Individual files can contain one or many data items. You can define your own data format that allows you to demarcate data items. You must document your data format including the reasons for your choice.

2.5 Evaluating your system
The final goal of this project is to achieve a high-throughput data storage. In order to measure the throughput rate, you will need to know the size of data (input and output) and the turnaround time for your query. This measurement can be done using a test harness.

2.6 Analysis questions
To analyze your system you must include following information in the final report.

- Software design of the system: Include information about the class structure and any design patterns that you have used.
- Complexity analysis for each of the commands.
- Data flow analysis. This describes the flow of data in your system. In some cases, you may end up with multiple figures.
- Design decisions that enable the system to achieve the stated goal: Include details about algorithms, data structures, and optimizations that you have in place that allow your system to meet the objectives.
- How it is different from existing approaches: Describe the innovations in your approach. Identify what is unique in your solution.

2.7 Software Development
You may use any programming language of your choice. This information must be included in the phase 2 report. You are encouraged to use a version control system (such as cvs, subversion, or git) to collaborate with your teammate.

3 Submissions

3.1 Phase 1. Reading and Summarizing Motivating Articles (due on 1/28 by noon)
You should read motivating articles from the following book.

*The Fourth Paradigm: Data-Intensive Scientific Discovery*

Read 3 articles listed below and provide a 400-word summary for each of those 3 articles.
Article 1: 
Jim Gray on eScience: A Transformed Scientific Method 
Edited by Tony Hey, Stewart Tansley, and Kristin Tolle 
pp. xvii

Article 2: 
Pick one article (except for Introduction) from Chapter 1. Earth and Environment

Article 3: 
Pick one article (except for Introduction) from Chapter 2. Health and Wellbeing.

3.2 Phase 2. Kick-off Planning (due on 2/11 by noon) (revision due on 2/18 by noon if needed) 
You should pick a teammate and submit a planning document. This should include,
• Mission (specific goal of your task) 
  Your strategy to achieve this.
• Functions targeted by your software 
• Special features of your software 
• Plan for testing 
• Manipulating test data 
• Source of test data 
• Infrastructure for testing 
• Evaluation method 
• Plan for analysis 
• Project timeline (bi-weekly plan) 
• The two members of the team and their respective roles 
• Development plan (source/version control)

In this phase, each team will have an interview with the instructor and TA.

3.3 Phase 3. Mid-point Progress Report (Due on 3/11 by noon) 
You and your teammate should submit a mid-point progress report. It should include the following:
• A description of the progress that you have made so far.
• Are you on your schedule based on the project timeline specified in Phase 1? 
• Modified project timeline with justification for the changes. If your timelines have not changed, 
you do not need to include this. 
• Design of the pilot software (your experiences so far and any preliminary results) 
• Survey sources for your analysis.

3.4 Phase 4. Final Report (Due on 4/16 by noon) 
• Analysis report (about 4000 words excluding citations) 
• Pilot software with source code. This must also include instructions to compile the software, 
  execute sample programs, and run the benchmark programs. 
• Your team will also provide a short demo to the instructor and TA.

3.5 Phase 5. Presentation (16th week in the class) 
Presentation (~15 minutes per team) 
Feedback

4 Grading 
After each of the submissions, partial scores will be posted on RamCT within 5 days.
The following table contains details about the grading criteria.

<table>
<thead>
<tr>
<th>Phase</th>
<th>Grading criteria</th>
</tr>
</thead>
<tbody>
<tr>
<td>Phase 1</td>
<td>Does each summary meet the word count requirement? (10)</td>
</tr>
<tr>
<td>(20/200)</td>
<td>Do the summaries include core idea in these articles? (10)</td>
</tr>
<tr>
<td>Phase 2</td>
<td>Does the report include information listed in the submission section? (10)</td>
</tr>
<tr>
<td>(20/200)</td>
<td>Did the team have an interview with instructor and TA? (5)</td>
</tr>
<tr>
<td></td>
<td>Are suggestions incorporated after the interview? (5)</td>
</tr>
<tr>
<td>Phase 3</td>
<td>Does the report include the information listed in the submission section? (20)</td>
</tr>
<tr>
<td>Phase 4</td>
<td><strong>Written Report</strong></td>
</tr>
<tr>
<td></td>
<td>Does the report include a good introduction and motivation for this project? (10)</td>
</tr>
<tr>
<td></td>
<td>Does the report include the information listed in the submission section? (20)</td>
</tr>
<tr>
<td></td>
<td>Does the report include good analysis and test results? (10)</td>
</tr>
<tr>
<td><strong>Software</strong></td>
<td>Does the software perform the specified features? (20)</td>
</tr>
<tr>
<td></td>
<td>Does the software’s special feature contribute to the product significantly (10)?</td>
</tr>
<tr>
<td></td>
<td>Does the software’s performance support high-throughput data access? (10)</td>
</tr>
<tr>
<td></td>
<td>Was the software demonstration well organized? (5)</td>
</tr>
<tr>
<td></td>
<td>Could the students answer the questions during the demonstration? (5)</td>
</tr>
<tr>
<td>Phase 5</td>
<td>Were the slides well organized? Good flow and description (10)</td>
</tr>
<tr>
<td></td>
<td>Did the slides include good results from this project? Was the analysis and discussion of these results good? (10)</td>
</tr>
<tr>
<td></td>
<td>Did the students interact with the audience and answer questions posed during the session? (10)</td>
</tr>
</tbody>
</table>

### 5 Late Policy
Each submission of the phases can be submitted up to a maximum of 24 hours past the deadline with a 10% deduction for the late submission.