

Masters of Science in Cloud Computing

Scalable Cloud Programming

Project

50%

Semester 3, 2022/23

1 Overview

In this project, you are required to develop a complex scalable cloud computing solution, which should be informed by best practice in the domain and documented in the form of a conference-style report. You will also be required to provide a complete archive of the code you developed and to prepare a video presentation demonstrating your working solution.

1.1 The data set

BGL is an open data set of logs collected from a BlueGene/L supercomputer at Lawrence Livermore National Labs. It is equipped with 131,072 processors and 32,768GB memory.

The log file can be downloaded from Zenodo¹. A sample line from the log file is shown below.

```
- 1121707460 2005.07.18 R23-M1-N0-C:J05-U01 2005-07-18-10.24.20.440509 R23-M1-N0-C:J05-U01 RAS  
KERNEL INFO generating core.7663
```

This can be parsed as show in table 1 below.

Table 1: Log file structure

Value	Interpretation
-	Alert message flag
1121707460	Timestamp
2005.07.18	Date
R23-M1-N0-C:J05-U01	Node
2005-07-18-10.24.20.440509	Date and Time
R23-M1-N0-C:J05-U01	Node (repeated)
RAS	Message Type
KERNEL	System Component
INFO	Level
generating core.7663	Message Content

Note that the first column may contain values other than the alert message flag.

¹<https://zenodo.org/record/3227177/files/BGL.tar.gz>

2 Tasks

For this project you are required to programmatically acquire, store, pre-process, and perform data computation tasks on the BGL data set using MPI, MapReduce or Spark frameworks and appropriate design patterns. The data computation tasks should provide answers to the questions listed below.

The questions you are required to answer are dependent the **last digit of your student number**, as follows:

Last digit of student number	Questions
0	4, 8, 9, 15 and 20
1	2, 5, 11, 13 and 20
2	1, 5, 9, 15 and 18
3	2, 6, 12, 16 and 17
4	4, 7, 9, 16 and 19
5	3, 8, 9, 16 and 18
6	1, 8, 11, 13 and 17
7	3, 6, 12, 14 and 19
8	2, 7, 10, 14 and 18
9	4, 6, 11, 15 and 18

Please ensure you choose the correct questions as no marks will be awarded for code and answers relating to incorrect selections.

Questions

1. How many fatal log entries in the month of December resulted from an "invalid or missing program image"?
2. How many fatal log entries in the month of September resulted from a "major internal error"?
3. How many fatal log entries that occurred on a Monday resulted from a "machine check interrupt"?
4. How many fatal log entries that occurred on a Friday resulted from a "kernel panic"?
5. For each month, what is the average number of seconds during which EDRAM errors were detected and corrected?
6. For each day of the week, what is the average number of seconds over which "re-synch state events" occurred?
7. For each week, what is the average number of seconds during which "ddr errors" were detected and corrected? Assume a week runs from Monday to Sunday.

8. For each hour of the day, what is the average number of seconds during which "torus receiver z+ input pipe errors" were detected and corrected?
9. What are the top 5 most frequently occurring dates in the log?
10. What are the top 5 most frequently occurring days of the week in the log?
11. What are the top 5 most frequently occurring nodes in the log?
12. What are the top 5 most frequently occurring hours in the log?
13. Which node generated the largest number APPSEV events?
14. Which node generated the smallest number of KERNRTSP events?
15. Which node generated the largest number of APPBUSY events?
16. Which node generated the smallest number of APPUNAV events?
17. On which date was the latest fatal kernel error resulting in an rts panic?
18. On which date was the earliest fatal kernel error where the message contains "Power Good signal deactivated"?
19. On which date was the latest fatal app error where the message contains "Error reading message prefix"?
20. On which date was the earliest fatal kernel error where the message contains the words "timed out"?

You may choose to extend beyond the questions listed above and to perform other computations that may provide useful insights into the data.

3 Deliverables

3.1 Project Report

You will present the results of your project in the form of report. Submission of the report is mandatory. Non-submission will result in you being marked as not present (NP) for the entire project.

Recommended Paper Structure

- **Abstract:** 150-250 words
- **Introduction:** The remainder of 1st page (+ up to 1 column). Provide a motivation for your work. Present and discuss the of the paper.
- **Related Work:** One page with 8 or more citations in total. This should not only summarise related work, but also critically evaluate (positive and negative aspects) of the cited works with respect to the
- **Methodology:** Describe how have you approached answering the questions. Additional (technical) details, such as design patterns employed should also be discussed here. Essentially, you should describe and justify how you applied scalable computing and design patterns to answer the questions.
- **Results:** Present the answers to the questions and discuss the performance/scalability of your solution.
- **Conclusions:** Present a summary of your findings, and discuss the implications/limitations of your work.
- **Bibliography**

Paper Formatting and Length

Submitted papers must adhere to the IEEE conference format and should be 8-10 double- column pages in length, including all figures and references.

Word and L^AT_EX templates are available from the IEEE website². Note that IEEE-style referencing, **not Harvard referencing**, should be used.

Tables and other text-based content **must not** be included as images. **Do not include code in your report.**

Papers over 10 pages in length will be subject to a 5% penalty, i.e. the maximum mark for the paper will be 70%.

²http://www.ieee.org/conferences_events/conferences/publishing/templates.html

Space-saving tips

- Never have a line less than half-full at the end of a paragraph. Almost any paragraph can be rewritten so that this is not the case!
- Graphs, flow diagrams and tables are easy to do sub optimally– draw them properly and decide if they really need to be as big as they are, or if they really should span both columns.
- Sub figures (e.g. 3 graphs as one figure prefixed a, b c that span both columns) are usually fairly space efficient.
- The L^AT_EX template is significantly cleverer than the Word one, and will do more work to save space.
- In L^AT_EX, paragraph spacing is heavily optimised. This also means that cutting out a line or two before a new section can cause paragraph spacing to be recalculated thus saving significant space.

3.2 Code Artefact

All code used on the project must be submitted as a single gz archive. The submitted code must be thoroughly commented.

The root directory of the archive should contain a plain text file name *readme.txt* that provides clear instructions to re-run your code to verify the results obtained.

Submission of the code artefact is mandatory. Non-submission will result in you being marked as not present (NP) for the entire project.

3.3 Video

Presentations will be conducted via video. Videos should be no longer than 10 minutes in length and should demonstrate each aspect of your code and provide a discussion/explanation of what the code is doing.

You may use any suitable tool to prepare the video, such as Snagit, OBS, Camtasia, ScreenFlow or Quicktime. Note the latter two are MacOS only.

Videos may be uploaded to YouTube. The visibility of the video must be set to **unlisted** and the video must not be included in any publicly-accessible playlists.

Alternatively, you may create the video as a Teams meeting recording. You must ensure that the recording is accessible to anyone at NCI who has the link.

Include a link to your video in your project report. Submission of the video is mandatory. Non-submission will result in you being marked as not present (NP) for the entire project.

4 Academic Integrity

Any written work created by others must be properly cited and should be paraphrased or summarised where possible, otherwise it should be included in quotes. Figures not created by you should include an acknowledgment detailing the name(s) of the creator(s). Small code snippets found on the internet should not be claimed as your own, but instead a comment should be included in the source code indicating where you obtained it. In general, your submitted code should be created by you.

The use of large language models such as ChatGPT is **strictly prohibited**.

Students are strongly advised to familiarise themselves with the Guide to Academic Integrity produced by the NCI Library ³.

Note: All submissions will be electronically screened for evidence of academic misconduct, e.g. plagiarism, collusion and misrepresentation. Any submission showing evidence of such misconduct will be referred to the college's academic misconduct committee for disciplinary action.

Your lecturer reserves the right to request a one-to-one viva presentation with any student should this be deemed necessary for any reason.

5 Marking

Your submission will be marked according to the rubric provided in the following two pages.

³<https://libguides.ncirl.ie/academicintegrity>

GRADING RUBRIC - Scalable Cloud Programming Project - Semester 3 2022/23

CRITERION	Upper H1	H1	H2.1	H2.2	PASS	FAIL
Abstract 5%	An excellent abstract that provides a concise summary of the objectives, approach and key findings of the project.	A very good abstract that provides a reasonably concise summary of the objectives, approach and key findings of the project.	A good abstract that offers a satisfactory summary of the objectives, approach and key findings of the project.	An adequate abstract that provides a reasonable outline of the objectives, approach and key findings of the project.	An adequate abstract that provides a somewhat unclear outline of the objectives, approach and key findings of the project.	A poor abstract that fails to provide a summary of the objectives, approach and key findings of the project.
Introduction 5%	A very comprehensive but succinct introduction that very clearly outlines the objectives of the project. All objectives are met.	A comprehensive but succinct introduction that clearly outlines the objectives of the project. All objectives are met.	A reasonably comprehensive introduction that clearly outlines the objectives of the project. All objectives are met.	An adequate introduction that provides a reasonable outline of the objectives of the project. All objectives are met.	The objectives are reasonably well specified and are partially met.	It is not possible to discern the project objectives, and/or if the objectives were met.
Related Work 10%	The discussion of related work is excellent, and presents a very in-depth critical review of highly relevant work on the same or similar data sets.	The discussion of related work is very good, and presents a reasonably in-depth critical review of highly relevant work on the same or similar data sets.	The discussion of related work is good, and presents a critical review of reasonably relevant work on the same or similar data sets.	The discussion of related work is good but the work cited is somewhat lacking in relevance and/or is not subject to critical review.	The discussion of related work is adequate, but the work cited is largely lacking in relevance and/or is not subject to critical review.	The discussion of related work lacks depth, and/or the choice of work cited seems somewhat arbitrary.
Methodology 30%	The rationale for the approach taken to completing each task is exceptionally well documented. Suitable design patterns are identified and used for all tasks.	The rationale for the approach taken to completing each task is very well documented. Suitable design patterns are identified and used for all tasks. Additional questions over and above those listed in section 2 are well presented and discussed.	The rationale for the approach taken to completing each task is well documented. Suitable design patterns are identified and used for all tasks.	The rationale for the approach taken to completing each task is reasonably well documented. Suitable design patterns are identified and used for most tasks.	The rationale for the approach taken to completing each task is adequately documented. Suitable design are not identified or used for the most part.	The rationale for the approach taken to completing each task is poorly documented. Suitable design patterns are not identified or used.

GRADING RUBRIC - Scalable Cloud Programming Project - Semester 3 2022/23

CRITERION	Upper H1	H1	H2.1	H2.2	PASS	FAIL
Results 15%	The results of the analysis, including answers to additional questions over and above those listed in section 2, are exceptionally well presented and discussed.	The results of the analysis, including answers to additional questions over and above those listed in section 2, are very well presented and discussed.	The results of the analysis are very well presented and discussed.	The results of the analysis are reasonably well presented and discussed.	The results of the analysis are adequately presented and discussed.	The results of the analysis are poorly presented and discussed..
Writing 10%	Exceptionally well-written, with no language errors. All figures are well conceived and readable. The IEEE template is adhered to. Report does not exceed the length limits. References are appropriately and correctly used.	Very well-written, with no significant language errors. All figures are well conceived and readable. The IEEE template is adhered to. Report does not exceed the length limits. References are appropriately and correctly used.	The report has a few language and/or style errors. Figures are well presented. IEEE template and length limit are adhered to. References are complete, and correctly used.	The report is readable with some language and/or style errors. Some figures and tables may be hard to read or presented in a sub-optimal manner. IEEE template is largely adhered to. References are mostly complete and correctly used.	While not unreadable, there are a large number of language and/or style errors. Figures and tables may be hard to read, without appropriate numbering and captions. The IEEE template is largely adhered to. References are mostly complete and correctly used.	The report is has many typographical errors, and/or poor use of English. The IEEE template may have been broken. Figures and tables may be hard to read and lacking appropriate numbering and captions. References (if any) are incomplete.
Code Artefact 15%	The code artefact is complete. The code is very well structured and exceptionally well commented.	The code artefact is complete. The code is well structured and very well commented.	The code artefact is complete. The code is reasonably well structured and well commented.	The code artefact is complete. The code is reasonably well structured and adequately commented.	The code artefact is complete. The code is poorly structure and inadequately commented.	The code artefact is incomplete. The code is poorly structured or lacks comments.
Video Evidence 10%	The video shows each element of the project working and is accompanied by a clear and thorough explanation.	The video shows each element of the project working and is accompanied by a clear and reasonably thorough explanation.	Most elements of the project are demonstrated in the video. There is an accompanying explanation but this may lack clarity or detail in parts.	Most elements of the project are demonstrated in the video. There is an accompanying explanation but this may lack clarity and detail in parts.	Most elements of the project are demonstrated in the video. There is no accompanying explanation or it totally lacks clarity and detail.	The video fails to demonstrate almost all elements of the project. There is little or no accompanying explanation.