CSC 8634 - Cloud Computing

Marc Birkett

17/12/2021

Introduction

This project is an exploratory data analysis (EDA) project into a multiple GPU node map rendering system. To bring structure to this project the CRISP-DM methodology will be followed as "it is soundly based on the practical, real-world experience of how people conduct data mining projects." (Chapman et al, 2000). To aid organisation and repeatability of the project various packages from the Tidyverse will be used, particularly ReadR, DPlyr, GGPlot2 and ProjectTemplate. The methodology splits a data mining project into 5 stages which will provide structure to this document, these are Business Understanding, Data Understanding, Data Preparation, Modelling and Evaluation.

Business Understanding

(Business objectives, assess situation, goals, project plan)

What is the need for the project? - Justify your choice of response (i.e. the nature of, and your plan for, your project). To give strength to your argument, you should reference to practice elsewhere (e.g. in academic literature, or industry practices.

Cloud computing has became more commonly used throughout all sectors in the UK since 2000. Multiple providers such as Amazon and Microsoft are now in a marketplace which seeks to offer externally hosted solutions on a Software as a Service (SaaS) basis, along with infrastructure and platforms to provide eslastic, scalable solutions to business need. Along with this there is an opportunity to bring rigour to the measurement and evaluation of cloud computing approaches. A Research and literature review into the subject was carried out via Google Scholar. The terms "statistical rigour, reproducible data analyses, performance evaluation in computer science" produced 7.94 million results. The addition of keywords including "cloud computing" and "supercomputer" brought this down to 44,600 records. A selection of highly referenced documents was reviewed.

Hoefler, Torsten and Belli (2015) state that the "measuring and reporting performance of parallel computers constitute the basis of scientific advancement of high performance computing ... and that the state of practice is lacking". Vitek and Kalibera, 2011, lamented "unrepeatable results, unreproduced results, lack of benchmarks, lack of experimental methodology", and, Papadopoulos et al, 2018, "although these important principles are simple and basic, the cloud community is yet to adopt them broadly to deliver sound measurement of cloud environments.". Given this lack of rigour, this paper will be approached as a exploratory data analysis project.

Problem Area

This paper conducts a performance evaluation of terapixel rendering in cloud super computing. The solution was rendering using an Infrastructure as a Service (IaaS) cloud environment and up to 1024 graphical process unit (GPU) nodes which was used to compute a realistic visualisation of Newcastle Upon Tyne and its environmental data as captured by the Newcastle Urban Observatory. The data was subsequently provided for analysis via comma separated value files. There will also subsequently be a dashboard created to allow investigation of the data set.

The completion of this paper will contribute to the some of knowledge regarding the measurement and assessment of metrics on cloud based supercomputers.

Current Solution

The project currently demonstrated that it "is feasible to produce a high quality terapixel visualization using a path tracing renderer in under a day using public IaaS cloud GPU nodes. Once generated the terapixel image supports interactive browsing of the city and its data at a range of sensing scales

from the whole city to a single desk in a room" (Forshaw, 2021). However, there has been no analysis of the metrics produced by the system regarding performance.

Objectives

Various examples of data we can investigate through an EDA process have been provided with the dataset, which are outlined here:

- Which event types dominate task runtimes?
- What is the interplay between GPU temperature and performance?
- What is the interplay between increased power draw and render time?
- Can we quantify the variation in computation requirements for particular tiles?
- Can we identify particular GPU cards (based on their serial numbers) whose performance differs to other cards? (i.e. perpetually slow cards).
- What can we learn about the efficiency of the task scheduling process?

I am particularly interested in the differing performance of various GPU cards in use. This work will be applicable to the investigation of other hardware within the use of the cloud and could be applied to my day to day work.

Success criteria and Project Plan

The data provided includes the complete cycle of the rendering of each graphic, along with the grid, in X and Y co-ordinates for each graphic and the granularity of the zoom. The success of this project will be identifying low performing GPU cards based on their render time. The data provided will be investigated to identify the performance of each card. The wider dataset will also then be investigated for mitigating factors such as load, through the potential complexity of the graphic being rendered and the point in the render process which takes the most time.

Data Understanding

(collect initial data, describe, explore, data quality) What, concisely, did you do? To begin to identify the execution time, the Application Checkpoint data was investigated, this contains timestamps for each step of the image render process. There are 5 event types each with a start and stop event recorded. These are Total Render - the complete render event, which is made up of 4 events which are, in order, Render, Saving Config, Tiling, Uploading. Initial investigation will cover the total render time, which will indicate the performance of the GPU, should we get that far we can look into the time of each stage of the render. In the absence of any continuous data in this table basic data quality checks were carried out, this identified the following:

- All of the START events have an associated STOP event, as there are the same number of both.
- All TOTAL RENDER events have a complete set of child events.
- Most host machines were used between 600 and 700 times during the course of the data, up to 670 times at the third quartile. However, there were 4 host machines that were used in 1240 and 1220 runs respectively as shown in Figure 1.
- There are some tasks which were ran twice, hence having 20 instances of jobId / taskId pairs in the dataset.

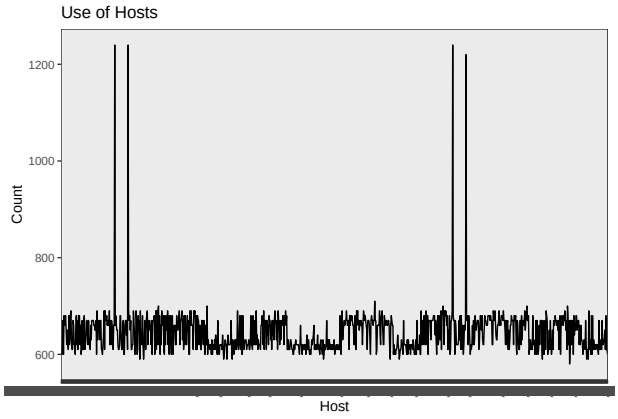


Figure 1: Azure Hosts use during Render

Data Preparation

(select, clean, construct, integrate, format) What, concisely, did you do?

Modelling

(technique, test design, build, assess) R-Shiny What, concisely, did you do?

Evaluation

(evaluate, review, next steps) How successful has it been? Provide evidence, using appropriate evaluation methodologies, and comment on the strengths/weaknesses of your evidence in answering this question. What are the future implications for work in this area? If applicable, which areas of extension work are now possible due to the foundational work you have performed in this project?

References

Alessandro V. Papadopoulos, Senior Member, IEEE, Laurens Versluis, Member, IEEE, Andre Bauer, Nikolas Herbst, Member, IEEE, Joakim von Kistowski, Member, IEEE, Ahmed Ali-Eldin, Cristina L. Abad, Member, IEEE, Jose Nelson Amaral, Senior Member, IEEE, Petr Tuma, Member, IEEE, and Alexandru Iosup, Member, IEEE, 2018, Methodological Principles for Reproducible Performance Evaluation in Cloud Computing, https://drive.google.com/file/d/151guslA9SYV-8BJNMXa1udvMrF4jn2ae/view Accessed: 21/12/2021

Forshaw, Matt, 2021, Performance evaluation of Terapixel rendering in Cloud (Super)computing [https://github.com/NewcastleDataScience/StudentProjects202122/blob/master/TeraScope/Summary.md#background] (https://github.com/NewcastleDataScience/StudentProjects202122/blob/master/TeraScope/Summary.md#background) Accessed: 21/12/2021

Hoefler, Torsten, and Roberto Belli, 2015, "Scientific benchmarking of parallel computing systems: twelve ways to tell the masses when reporting performance results." In Proceedings of the international conference for high performance computing, networking, storage and analysis, p. 73. ACM, 2015.

Jan Vitek, Tomas Kalibera, 2011, Repeatability, Reproducibility and Rigor in Systems Research https://www.cs.kent.ac.uk/pubs/2011/3174/content.pdf Accessed: 21/12/2021

Pete Chapman (NCR), Julian Clinton (SPSS), Randy Kerber (NCR), Thomas Khabaza (SPSS), Thomas Reinartz (DaimlerChrysler), Colin Shearer (SPSS) and Rüdiger Wirth (DaimlerChrysler), 2000, CRISP-DM Step-by-step data mining guide

R Markdown

This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. For more details on using R Markdown see http://rmarkdown.rstudio.com.

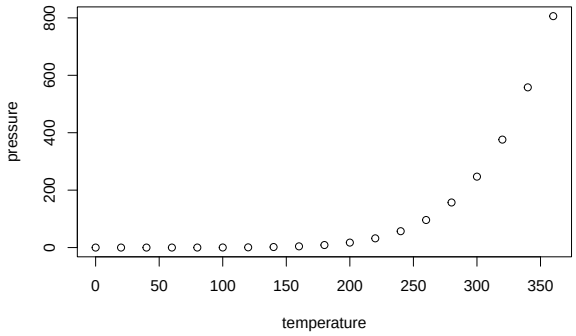
When you click the **Knit** button a document will be generated that includes both content as well as the output of any embedded R code chunks within the document. You can embed an R code chunk like this:

```
summary(cars)
```

```
##      speed           dist       
##  Min.   : 4.0   Min.   :  2.00  
##  1st Qu.:12.0   1st Qu.: 26.00  
##  Median :15.0   Median : 36.00  
##  Mean   :15.4   Mean   : 42.98  
##  3rd Qu.:19.0   3rd Qu.: 56.00  
##  Max.   :25.0   Max.   :120.00
```

Including Plots

You can also embed plots, for example:



Note that the `echo = FALSE` parameter was added to the code chunk to prevent printing of the R code that generated the plot.