Coursework - EMATM0051 Large Scale Data Engineering [Data Science]

Version: 12.11.2021 v2.0

Changes:

12.11.2021 v2.0 – Initial version for 2021-22 unit

Summary

This coursework is divided into two parts:

Part 1: A written task (only) related to the knowledge gained in the AWS Academy Cloud Foundations course (weeks 1-7).

Part 2: A combined practical and written activity architecting a scaling application on the Cloud, where you will be required to use knowledge gained and a little further research to implement the scaling infrastructure, followed by a report that will focus on your experience in the practical activity together with knowledge gained in the entire LSDE course.

Weighting: This assessment is worth 100% of your total unit 20 credits.

Set: 13:00. Monday 15th Nov 2021.
Due: 13:00. Wednesday 12th Jan 2022.

Pre-requisites:

- You must have completed the AWS Academy Cloud Foundations course set in weeks 1-7
- You will require an AWS Academy Learner Lab account for the practical activity. You should receive an invite when this document is released. Please contact the LSDE Unit Director if you have no email or issues with the registration.
- A Secure Shell (SSH) client, such as MacOS Terminal or PuTTy on Windows, for server admin.

Submission:

Via the LSDE BlackBoard coursework assessment page, submit one zip file, named using your UOB username ('username.zip'), containing:

- a Report ('report.pdf') in PDF format containing:
  - Part 1
  - Part 2
- a Text File ('credentials.txt') containing your AWS Academy account credentials (username, password), to enable us to access and review your Learner Lab account as required.

In this document we provide a detailed explanation of the tasks, and the approach to marking.

Unit Director: Alan Forsyth

Task 1: (25%)

Write a maximum of 1000 words (minimum: 600) debating the statement:

"The Public Cloud is ideal for data processing"

Include your own descriptions of the following:

- At least 5 AWS features or services introduced in the Cloud Foundations course that make data processing in the public cloud advantageous.
- At least 3 scenarios where the public cloud is not optimal or should be avoided for data processing.

Task 2: Scaling the WordFreq Application (75%)

Overview

WordFreq is a complete, working application, built using the Go programming language.
*[**NOTE:** you are **NOT** expected to understand or permitted to modify the source code in any way]*

The basic functionality of the application is to count words in a text file. It returns the top ten most frequent words found in a text document.

The application uses a number of AWS services:

- S3: Text files are uploaded to an S3 bucket. The bucket has upload notifications enabled, such that when a file is uploaded, a message notification is automatically added to a wordfreq SQS queue
- SQS: There are two queues used for the application.
    - One is used for hold notification messages of newly uploaded text files from the S3 bucket. These messages are known as 'jobs', or tasks to be performed by the application, and specify the location of the text file on the S3 bucket.
    - A second queue is used to hold messages containing the 'top 10' results of the processed jobs.
- DynamoDB: A NoSQL database table is created to store the results of the processed jobs.
- EC2: The application runs on an Ubuntu Linux EC2 instance, which you will need to set up initially following the instructions given. This will include setting up and identifying the S3, SQS and DynamoDB resources to the application.

You will be required to initially set up and test the application, using instructions given with the zip download file. You will then need to implement auto-scaling for the application and improve its architecture based on principles learned in the CF course. Finally, you will write a report covering this process, along with some extra material.

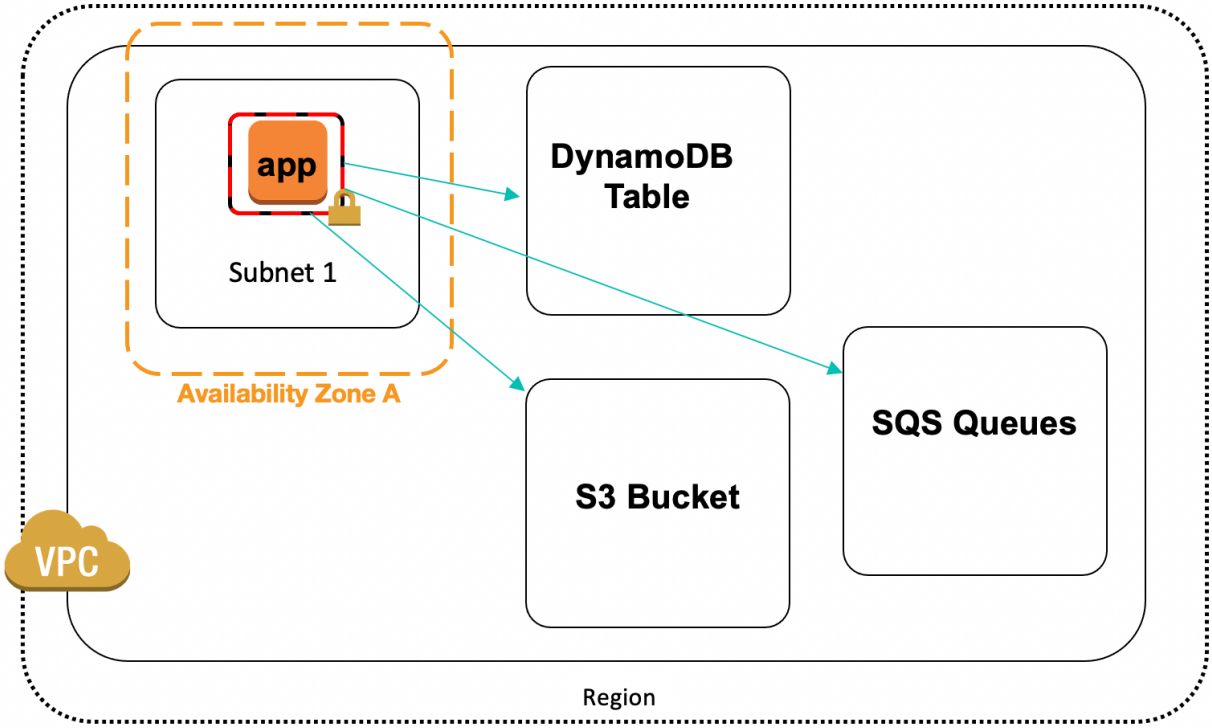

Figure 1 - WordFreq standard architecture

Task A – Install the Application

Ensure you have accepted access to your AWS Academy Learner Lab account and have at least $20 credit (you are provided with $300 to start with). If you are running short of credit, please inform your instructor.

Refer to the WordFreq installation instructions ('README.txt') in the coursework zip download on the BlackBoard site, to install and configure the application in your AWS Educate account. These instructions do not cover every step – you are assumed to be confident in certain tasks, such as in the use of IAM permissions, launching and connecting via SSH to an EC2 instance, etc.

You will set up the database, storage buckets, queues and worker EC2 instance.

Finally, ensure that you can upload a file using the 'run_upload.sh' script and can see the results logged from the running worker service, before moving on to the next task.

*[**NOTE:** The application code is in the Go language. You are **NOT** expected to understand or modify it. Any code changes will be ignored and may lose marks.]*

Task B – Design and Implement Auto-scaling

Review the architecture of the existing application. Each job process takes at least 10 seconds (artificially induced, but DO NOT modify the application source code!). To be able to process multiple uploaded files, we need to add scaling to the application.

This should initially function as follows:

- When a given maximum performance metric threshold is exceeded, an identical worker instance is launched and begins to also process messages on the queues.
- When a given minimum performance metric threshold is exceeded, the most recently launched worker instance is removed (terminated).
- There must always be at least one worker instance available to process messages when the application architecture is 'live'.

Using the knowledge gained from the Cloud Foundations course, architect and implement auto-scaling functionality for the WordFreq application. Note that this will not be exactly the same as Lab 6 in Module 10, which is for a web application. You will not need a load balancer, and you will need to identify a different CloudWatch performance metric to use for the 'scale out' and 'scale in' rules. The 'Average CPU Utilization' metric used in Lab 6 is not necessarily the best choice for this application.

Task C - Perform Load Testing

Once you have set up your auto-scaling infrastructure, test that it works. The simplest method is to create around 50 large text files by finding a few publicly available big text files (do not use private or sensitive data!), then make multiple copies, renaming them each time (<filename1>_A.txt, <filename1>_B.txt, <filename2>_A.txt, etc.).
Suitable sources: Any of the corpora archives on the Canterbury Corpus page (e.g. The Large Corpus):
https://corpus.canterbury.ac.nz/descriptions/
Other options are mentioned on this StackOverflow page:
https://stackoverflow.com/questions/44492576/looking-for-large-text-files-for-testing-compression-in-all-sizes

Ensure:

- Files are in text (.txt) format only, no binary files should be used.
- Filenames have no spaces or non-alphanumeric characters (hypens, underscores are ok).
- Files are no larger than about 10MB, otherwise they will take a while to upload, but no smaller than 1MB.

You can 'purge' all files from your S3 bucket, then click 'Upload' and drag all 50 selected files onto the S3 upload page.
*[**NOTE:** An optional more efficient method is to install the AWS CLI on your PC and use aws s3 cp commands – see the note about using the AWS CLI with Learner Labs at the end of this document]*

- Watch the behaviour of your application to check the scale out (add instances) and scale in (remove instances) behaviour works.
- Take screenshots of your uploaded files, the SQS queue page showing message status, the Auto Scaling Group page showing instance status and the EC2 instance page showing launched / terminated instances during this process.

- Try to optimise the scaling operation, for example so that instances are launched quickly when required and terminated soon (but not immediately) when not required. Note down settings you used and the fastest file processing time you achieved.
- Try using a few different EC2 instance types – with more CPU power, memory, etc. Note down any changes in processing time.

  ***[NOTE: The Learner Lab accounts officially only allow a maximum of 9 instances running in one region, including auto-scaling instances**. Learner Lab accounts are limited in which EC2 Types and AWS services they can use. This is explained in the Lab Readme file on the Lab page; section 'Service usage and other restrictions'.]*

Task D - Optimise the WordFreq Architecture

As a consulting data engineer, you are asked for advice from two companies who wish to use an application like WordFreq for large scale processing of text files.

a) Based on **only** AWS services and features learned from the Cloud Foundations course, describe how you could re-design the WordFreq application's current cloud architecture (i.e. not changing the application functionality itself) to suit **ONE** of the following two company scenarios:

   - **Company A**: Requires a highly resilient, highly performant, very secure application for extremely reliable and immediate data processing of critical text documents. No long-term data backups required. **[OR]**
   - **Company B**: Wants an extremely cost-effective and efficient (but still scalable and resilient) application for occasional use. Processing does not need to be immediate. Basic security and long-term data backups required.

   Your description for each company should ideally include diagrams and include the AWS services required together with a high-level explanation of features & configuration for each, related to the following categories:

   - Data processing performance of the application as appropriate.
   - Resilience and availability of the application against component failure.
   - Security of the application for data protection and to prevent unauthorised access.
   - Ensuring the application is as cost-effective as required.

b) Attempt to apply as much of your design as possible to your Task C WordFreq application architecture in the Learner Lab. If you do not have permissions or access to a particular service or feature, mention this after your design description in part a) above.

***[NOTE:** Ensure that your WordFreq application's auto-scaling is still functional when finished!]*

Task E - Create the Final Report

Write a report of no more than 3500 words or 20 A4 pages (there is NO minimum), including:

- A brief summary of how the application works (without any reference to the code functionality)
- Your design process to architect the scaling behaviours (task B)
- An overview of the testing and your results, including screenshots (task C)
- Your architectural description for Company A **OR** Company B (task D)
- Details of any issues you had and whether you resolved them.

Add one final section of the report: *Further Improvements*

- Based on services and frameworks covered in the full LSDE course, identify two alternative data processing applications that would be far more performant and robust for this processing task.
  [Do not implement these ideas, just describe their advantages over WordFreq in a few paragraphs].

The report should be included within the zip file as a PDF. It does not need to follow any academic format, but you should use grammar and spelling checkers on it and make good use of paragraphs and sub-headings. Double-spacing is not required. Use diagrams where they make sense and include captions & references from the text.

[IMPORTANT: All text not originally created by you must be cited, leading to a final numbered reference section (based on e.g. the British Standard Numeric System) to avoid accusations of plagiarism.]

AWS Academy Learner Lab

You are given an AWS Academy Learner Lab account for this coursework. Each account has $300 assigned to it, which is updated every 24 hours and displayed on the Academy Lab page.

To access the lab from AWS Academy, select *Courses > AWS Academy Learner Lab > Modules > Learner Lab - Foundation Services*. On this page click 'Start Lab' to start a new lab session, then the 'AWS' link to open the AWS Console once the button beside the link is green.

Please note:

- Ensure you shut down (stop or terminate) EC2 instances when you are not using them. These will use the most credit in your account in this exercise. Note that the Learner Lab will stop running instances when a session ends, then restart them when a new session begins.
- AWS Learner Lab accounts have only a limited subset of AWS services / features available to them, see the Readme file on the Lab page (Service usage and other restrictions) or this AWS Academy Page.
- If you have installed the AWS CLI on your PC and wish to access your Learner Lab account, you will need the credentials (access key ID & secret access key) shown by pressing the AWS Details button on the Lab page. Note that these only remain valid for the current session.
- If you have any issues with AWS Academy or the Learner Lab, please book an Office Hours session or use the LSDE Discussion Forums to seek help FIRST, use the LSDE Teams site chat or email the instructors if there is no other option.

Support

The normal options for support are available for you during term time, up until Week 13 (January 12th):

- Book Office Hours for tech questions/support in LSDE Teams site form.
    - If available dates are not shown, view page in browser: https://outlook.office365.com/owa/calendar/ematlsdeofficehours@bristol.ac.uk/bookings/
    - Office Hours: 15 min sessions with a member of staff or a TA.
- LSDE BlackBoard Discussions Forum

Marking

Below are the marking bands with maximum possible mark range achievable given approximate scope of work.

+80% Outstanding report and implementation. Extensive exploration, analysis and implementation demonstrating deep understanding and reading outside of the CF course and lectures.

70 - 80% Excellent report. Well architected, fully functional auto-scaling, great optimisation techniques, very good understanding of cloud principles gained in the CF course.

60 - 70% Report of correct length, fully functional auto-scaling, good optimisation techniques, good understanding of cloud principles gained in the CF course.

50 - 60% Report of correct length, basic but functional auto-scaling, some good ideas about optimisation techniques, correct understanding of main cloud principles in the CF course.

<50% (Fail) Report is not at an appropriate standard, auto-scaling not implemented. Objectives of the assignment have not been demonstrated.

Academic Offences

Academic offences (including submission of work that is not your own, falsification of data/evidence or the use of materials without appropriate referencing) are all taken very seriously by the University. Suspected offences will be dealt with in accordance with the University's policies and procedures. If an academic offence is suspected in your work, you will be asked to attend an interview with senior members of the school, where you will be given the opportunity to defend your work. The plagiarism panel are able to apply a range of penalties, depending the severity of the offence. These include: requirement to resubmit work, capping of grades and the award of no mark for an element of assessment.

Extensions and Extenuating Circumstances

If the completion of your assignment has been significantly disrupted by serious health conditions (including mental health impairment), personal problems, or other similar issues, you may be able to apply for an extension for assessment submission or consideration of extenuating circumstances (in accordance with the normal university policy and processes).

- Extensions allow limited additional time to be granted before submission. They must be requested before the normal assessment submission date. See the following page: https://www.bristol.ac.uk/students/support/academic-advice/assessment-support/request-a-coursework-extension/
  Note that all assessment extension requests in AY 2021-22 require evidence.

- Extenuating Circumstances (EC) recognises a significant disruption and can facilitate extensions, additional support and care services, waiving of late submission penalties, extension of studies, etc. Students should contact the LSDE Unit Director and their tutor and apply for consideration of EC as soon as possible when the problem occurs. Please review the following university page: https://www.bristol.ac.uk/students/support/academic-advice/assessment-support/extenuating-circumstances/