The Cloud: How the Cloud Will Change the World of Finance

A Technical Report submitted to the Department of Computer Science

Presented to the Faculty of the School of Engineering and Applied Science

University of Virginia • Charlottesville, Virginia

In Partial Fulfillment of the Requirements for the Degree

Bachelor of Science, School of Engineering

Maya Nakhre

Fall, 2022

On my honor as a University Student, I have neither given nor received unauthorized aid on this assignment as defined by the Honor Guidelines for Thesis-Related Assignments

Briana Morrison, Department of Computer Science

The Cloud: How the Cloud Will Change the World of Finance

CS4991 Capstone Report, 2022

Maya Nakhre
Computer Science
The University of Virginia
School of Engineering and Applied Science
Charlottesville, Virginia USA
mtn4ghj@virginia.edu

Abstract

JP Morgan Chase and Co (JPMC), ranked one of the largest banks in the world, is transitioning to an almost entirely cloud-based system which will allow for a significant improvement in the scalability and accessibility of data. Since this type of shift would be a considerable digital transformation, the problem had to be broken up into smaller parts including using the help of services already available such as Amazon's AWS platform. As an intern, my approach was to first develop a way for clients to access the data in the cloud, specifically AWS, without needing ownership access. This project ultimately simplified over 100+ clients' workflows by enabling them to access data that was initially unavailable to them. The process of data mining and analysis will become much more streamlined. Future implementations for this project will include authentication and more security to validate a client's identity

1. Introduction

Imagine one of the largest banks in the world throwing out its entire infrastructure for the cloud. That is what JPMC is currently doing. With an investment of 12 billion dollars, JPMC is revamping its old data structure for something more mainstream. If the move to a cloud-based platform is successful, the estimated cost savings will be around 30 to 40 million a year.

That is not the only reason they are making this monumental change, however. Moving to the cloud will enable machine learning to work on facets of the banking system such as fraud management, risk, and marketing which will be extremely beneficial to the company as a whole. When moving the database to the cloud, the most important aspect is maintaining the clients' information which is what my project aimed to do. Enabling clients to query data in AWS accounts that they do not currently have access to will make it possible for them to better understand the business logic behind their work.

2. Relevant Work

Mason (2022) deals with an in-depth analysis of JPMC CEO Jamie Dimon's

decision, including the reasons moving to a cloud-based platform would be revolutionary for the banking system. The Bednarz (2022) article shows how much money JPMorgan spent in which areas. She mentions, for instance, the investment put into cloud engineers poached from companies such as Amazon, Google, and Microsoft. One of the main platforms I used in my project was Terraform, which is one of the new platforms that allow one to build infrastructure as code. Brikman (2016) explores why Terraform is being utilized more than platforms such as Puppet, SaltStack, or CloudFormation.

3. System Design

The system design for this project included many components, the most important being Terraform infrastructure and the SQL queries that were created using AWS client. The first tool I used was Terraform which created infrastructure as code and allowed AWS resources to be created. I then used a REST API to develop an interface specifically for client use. This interface allowed clients to query any data from the AWS cloud.

3.1 Review of System Architecture

The system architecture used in this project revolved heavily around the Terraform platform, as that provided the basis of the coding structure. Terraform uses AWS CLI, a method of making API calls to execute provisioning tasks. The tasks are created in .tf files and include different AWS instances. As seen in Figure 1, the instance created is EC2 so through this code block I am telling Terraform to provision this task to create an EC2 instance.

```
resource "aws_instance" "my_vm" {
 ami                    = "ami-065deacbcaac64cf2" //Ubuntu AMI
 instance_type          = "t2.micro"

 tags = {
   Name = "My EC2 instance",
 }
}
```

Figure 1: Provisioning an EC2 instance

The second form of architecture used to deploy the Terraform infrastructure was Jules Pipeline. Jules pipeline supports the continuous development of a system and the Jules pipeline deployment allowed for a working URL to exist where the queries could be written. The third form of system architecture used was Python and Django to establish queries that would go through the data and then display them to the client. By looking through the data catalog in the data lake clients will be able to get a better idea of what data exists and how to use it for their business needs.

3.2 Process of Creating the Queries

Once the .tf files were created and all the AWS resources were provisioned it was time to actually make all the resources in my AWS account. Since Terraform uses AWS CLI, I used that call to connect my Terraform resources to the AWS account. I created AWS resources such as dynamodb, IAM modules, and EC2 instances. The dynamodb instance was the main structure created that would hold all the data in the data lake. Thereafter I started writing queries that would scan through the table and deliver what the client wanted. I built

the queries using AWS client, a remote access VPN method used by a remote workforce to access resources within my AWS account.

AWS client connected the dynamodb tables to my code so that I was able to write SQL statements that scanned through the tables. As seen in figure 2, I used the scan method which goes through the dynamodb tables and then chooses the data element that matches what the client put in as a parameter.

```
aws dynamodb scan \
    --table-name Movies \
    --projection-expression "title" \
    --filter-expression 'contains(info.genres,:gen)' \
    --expression-attribute-values '{":gen":{"S":"Sci-Fi"}}' \
    --page-size 100  \
    --debug
```

Figure 2: The dynamodb scan structure

Results

The intended outcome of this project was to improve the knowledge that clients had on the data lake. Every day data gets moved into the data lake and into AWS accounts such as the dynamodb tables and clients do not know what exists. By creating this query method, clients will be able to get a better idea of what data exists just by inputting a few parameters of their choosing. This project will help clients find exactly the data elements they are looking for which, in turn, will help them develop and grow their businesses.

Conclusion

The Cloud has proven to be one of the most important assets for businesses to grow and expand efficiently in today's world and JPMorgan is no different. Being able to access, edit, and control clients' data from anywhere in the world is a feature and a benefit that every powerful bank should be able to take advantage of. The innovative solutions that come from Cloud creation are exactly what JPMorgan needs to usher in this new era of banking with millions of users.

Future Work

One of the next steps with this project is to include comprehensive security measures. These measures include encryption details and two-factor authentication to make sure that the only people who can access data are those permitted to access it. These next steps will make the data lake of JPMorgan significantly more secure and reliable.

UVA Evaluation

The one class that most helped me with understanding this project was Cloud Computing, a 4000-level course offered at UVA. Although the curriculum did not include everything that I worked on in the project, it still provided me with a foundation for the basics of the Cloud. I was able to go into the project with a basic knowledge of AWS resources such as EC2 instances and S3 buckets. Some additional training that would have been helpful would be more information about SQL and MySQL and how those programs work since they seem to be utilized extensively in the workforce.

Acknowledgments

One of the main individuals who helped me through this project was Yufei Zhou, one of the principal engineers for the Data lake team at JPMorgan. He was instrumental in helping me understand what I had to do in the project and how to start.

Relevant Work

[1] Sumeet Ninawe. 2022. Terraform Tutorial – Getting Started With Terraform on AWS. (Jul 2022). Retrieved November 8,2022 from https://spacelift.io/blog/terraform-tutorial.

[2] AWS.Working with scans in DynamoDB. Retrieved November 1, 2022 from https://docs.aws.amazon.com/amazondynamodb/latest/developerguide/Scan.html.

[3] Ann Bednarz. 2022. Terraform Tutorial – Getting Started With Terraform on AWS. (Feb 2022). Retrieved November 1, 2022 from https://www.networkworld.com/article/3649774/jpmorgan-chase-spent-2-billion-on-brand-new-data-centers-last-year.html.

[4] Emily Mason. 2022. Terraform Tutorial – Getting Started With Terraform on AWS. (Sep 2022). Retrieved November 8,2022 from https://www.forbes.com/sites/emilymason/2022/09/14/jp-morgan-rushes-to-modernize-as-cloud-native-competitors-move-in-on-its-payments-business/?sh=7ebff9fc767e.