Generico Payroll System Cloud Integration

Architecture and Components Report

This Document shall highlight the choices made to move Genericos Payroll System to the cloud. Each section shall provide in depth reasoning on why the decision was made, Also Pricing shall be discussed and any changes that are needed for the current implementation and a potential roadmap shall be laid out.

Name: Timothy Finn
FAN: FINN0066

Executive Summary

This document shall highlight the choices made to move Genericos current Payroll System to a cloud implementation. The discussion shall be focused on the architecture chosen including descriptions and justification of each component chosen. Finally, it shall be discussed any changes that will be made to the database or application and the remainder of the roadmap for the project.

This Documents intended audience is management or developers interacting the Payroll System with understanding of technical design concepts.

Name: Timothy Finn
FAN: Finn0066

Table of Contents

Executive Summary 1
Architecture and Components Report 3
Elastic Load Balancing 3
Elastic Beanstalk 3
Platform 3
Elastic Compute Cloud (Amazon EC2) 3
Relational Database Servicer(RDS) 4
Diagram of cloud implementation architecture. 4
Cloud Solution Cost 5
Elastic Computing Cost 5
Relation Database Service Cost 5
Elastic Bean Stalk Cost 5
Total Cost of Cloud Solution(Yearly) 6
Migration roadmap 7
First Stage of migration. 7
Database Schema Changes. 7
Changes Made to Legacy Application 8
Future Roadmap 9
Conclusion 10
Appendix A Group Report. 11
References 11

Name: Timothy Finn
FAN: FINN0066

Architecture and Components Report

Below shall discuss the chosen architecture and descriptions of why they were chosen to complete the tasks.

Elastic Load Balancing

Upon initial connection and successful completion of a security check of IP address, Amazon's elastic Load balancer will distribute incoming traffic across multiple Elastice compute cloud instances(EC2). Elastic Load Balancing was chosen to be integrated into the system as it provides High availability by distributing across multiple targets. [1]

Also, ELB allows Scalability to meet the user's demands. In future iterations of the project, ELB can be used with Amazons Virtual Private Cloud (Amazon VPC) to increase security or be monitored using Amazon Cloud Watch. [1].

Elastic Beanstalk

To simplfy, the process of moving the current system to the cloud Elastic Beanstalk was chosen. This simplifies many of the tasks in moving to the cloud as it reduces the management and organising of amazon's infrastructure and allows Generico in focusing on their payment system exclusively.

When creating an instance of Elastic Beanstalk Amazon automatically provisions an EC2 instance and deploys it with a software stack that Generico decides. Amazon also offers automatic scaling of the application to meet peaks in workload treffic, This could prove benificial in use cases of tax time or other instances where there is high demand for a small amount of time. [2]

Platform

The platform shall use a 64bit Amazon Linux 2017.03 version Operating system; This was chosen for maximum compatibility with the current Payroll System. Also, this operating system will come with Tomcat 8 and java 8 preinstalled to ensure that the application can run efficiently, and minimum amount of changes are required to the current System. [3]

Elastic Compute Cloud (Amazon EC2)

An EC2 instance is automatically generated by Amazon when using the elastic beanstalk service the current specifications of the device are configured with 8 gigabytes of ram and running on a t1.micro instance. This EC2 configuration was chosen as it provides the satisfactory performance requirements for the payroll system.

If needed performance could be scaled up by increasing the amount of ram or upgrading the EC2 instance to a higher performing CPU. [3]

Relational Database Servicer(RDS)

As performance is not mission critical a standard Hard Disk Drive was chosen to host the database needed to run the Payroll System. This was chosen to lower overall monthly costs but can easily be scaled up to meet a change in requirements needed for the Payroll System. Also, Backup options have been configured on the RDS to ensure data recoverability in the event of a roll back. [4]

M.4 was chosen as the database instance type this was chosen as it is Elastic Beanstalk optimized and suits general purpose instances. According to Amazon this instance is well suited for running small to mid-size databases and suited for Enterprise level applications. Size of the database has not been disclosed in price calculations as this will need to be discussed between management and Database Administrators on total amount to be placed on the cloud. [5]

Diagram of cloud implementation architecture.

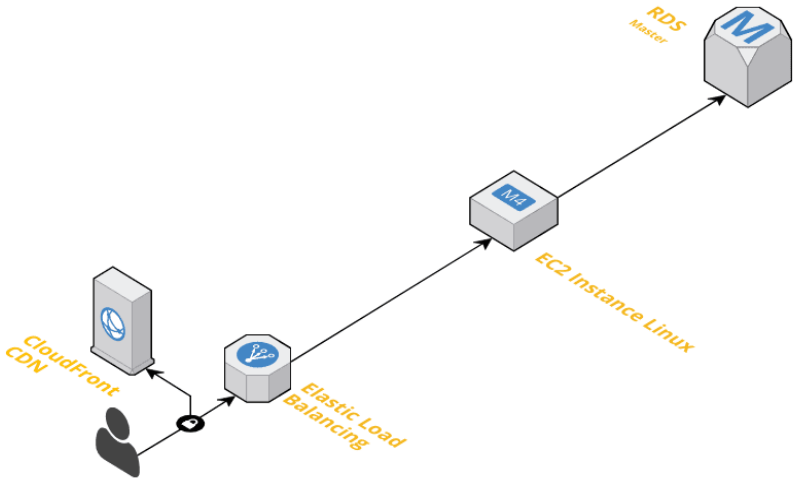

FIGURE 1 ABOVE SHOWS IMPLEMENTATION OF GENERICO PAYROLL SYSTEM ARCHITECTURE SITUATED ON AWS [6]

The above diagram illustrates the basic setup of Genericos Payment system components can be added or removed as necessary depending the on the needs of generico at the time.

Cloud Solution Cost

This Section shall outline the cost in employing the above selected architecture, As AWS is a cloud solution it offers a pay as you go approach for their services. Pay as you go does offer the freedom of only paying for what you need but consideration must be taken to ensure prices do not scale out of control. Amazon also offers to pay upfront costs which will offer a substantial discount to the below estimated costs. [1]

* All cots Calculated below are indictive only and are subject to change from AWS or usage requirements by Generico and as such should be considered a guideline.

Elastic Computing Cost

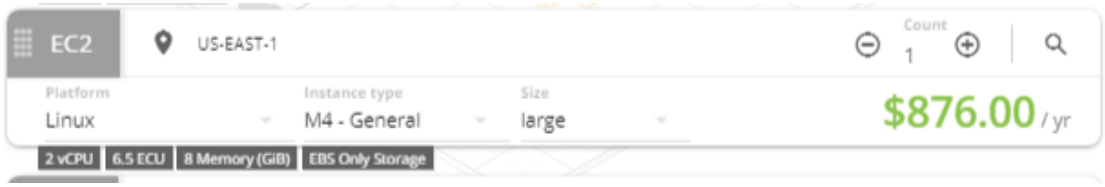

FIGURE 2 ESTIMATION OF ELASTIC COMPUTING COSTS [6]

Relation Database Service Cost

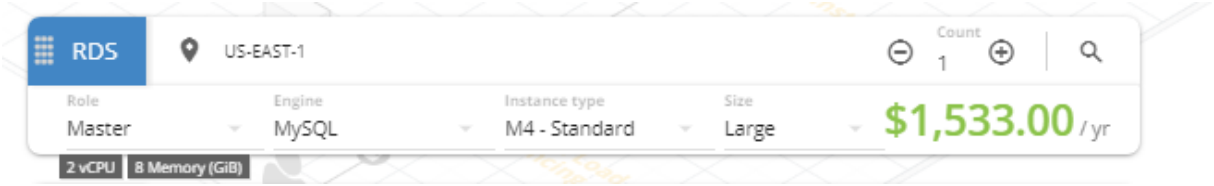

FIGURE 3 ESTIMATION OF DATABASE COST [6]

Elastic Bean Stalk Cost

FIGURE 4 ESTIMATION OF ELASTIC BEANSTALK COST [6]

Total Cost of Cloud Solution(Yearly)

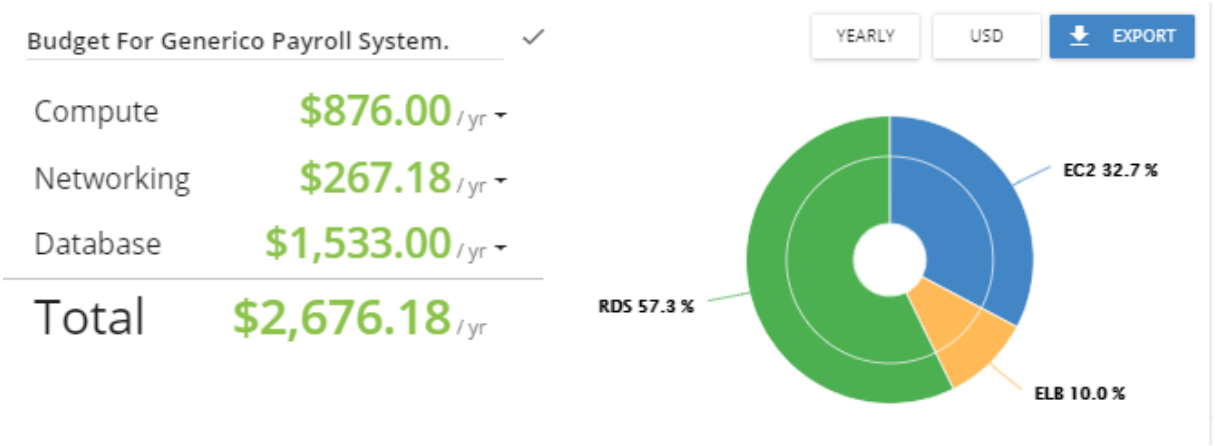

FIGURE 5 TOTAL COST OF IMPLEMENTING GENERICOS PAYROLL SYSTEM TO AMAZON WEB SERVICES [6]

Above highlights the total cost of implementation of the legacy system to a cloud based approach. Cloud database storage is the major driver of cost accounting for over half of the budget. Computing power is second coming from the EC2 instance and ELB with Networking coming in last.

As noted earlier all costs are subject to change and can be higher or lower depending on what architecture is chosen to be implemented. Also, if higher performance requirements are required, or more stringent backups are needed, then these shall also need to be considered. [1]

Migration roadmap

This section shall highlight the changes required in implementing Genericos payment solution to the cloud. Also discussing points about any changes that must be accommodated by the original system to function correctly in a cloud environment.

First Stage of migration.

The First Stage of migration to the cloud was to setup an account with AWS, after then it was to set up an Elastic Beanstalk implementation. With this step, AWS will automatically create an EC2 instance for use. When creating the instance, Amazons Linux instance is chosen, and Tomcat is selected when both options have been selected Amazon will add them to the newly created EC2 instance.

After elastic beanstalk has been configuring it is onto creating a Relational Database instance, with this instance, Generico can store all of its information onto a cloud platform. Importing data to a cloud platform will allow the Payroll System to access the database through AWS instead of relying on local servers this reduces on site costs and risks compared to Amazons powerful and dispersed network. This was achieved using MSQL workbench connecting to the RDS instance and run the database scripts.

Once both Services have been configured Security permissions need to be changed for the EB instance to communicate with the RDS instance, this can be setup in security group preferences. Once completed work then can focus on modifying the existing database and importing the data to the newly created RDS instance and also to modify the Payroll System application ensuring it receives its data from the newly created instance and doesn't rely on the old database location for its information.

Database Schema Changes.

Once the RDS database had been configured, it was necessary to make changes to the structure of the current database before importing. The current database had issues with normalisation so it needed to be normalised at least to third normal form once this had been completed it was decided instead of relying on two separate databases required to access employee information the data would be combined into one database.

Reducing two databases to one database was done to reduce complexity and the need for double the administrative work required to update and maintain two separate databases. Also needing only one database reduces the cost's associated with running

two database instances on the cloud. Finally, with only one database, it is far easier to maintain a database in proper normalised form instead of being required to make changes in separate databases.

Changes Made to Legacy Application

Minimal Changes where needed to make the legacy application available in the cloud Main changes that were required were changes to the location sourced by the application to retrieve the data. As the database was now situated in the cloud, the application was needed to be required to search for data on the provided AWS link.

Moving the data to the cloud gives the application more uptime as the application now relies on the AWS network which is more widely available then Genericos own network. Changes were simple to implement and took the Developer team a minimal amount of time to complete. Highlights of the code changes are shown below.[3]

```
<sql:setDataSource var="database" driver="com.mysql.jdbc.Driver"
                   url="jdbc:mysql://rds-mysql-generico.clmxhcdlrmia.ap-
                   southeast-2.rds.amazonaws.com:3306/generico_database?zeroDateTimeBehavior=convertToNull"
                   user="admin"
                   password="password"
                   />
```

FIGURE 6 ADDING AWS RDS CONNECTION DETAILS TO THE LEGACY APPLICATION

```
<sql:query var="comp3732_employees" dataSource="${database}">
    SELECT * FROM comp3732_departments, comp3732_employees
    WHERE comp3732_employees.emp_id = comp3732_departments.emp_idfk
    AND comp3732_departments.deptName = ? <sql:param value="Development and Design"/>
</sql:query>
```

FIGURE 7 CHANGING DATA SOURCE TO USE NEW RDS DATABASE.

Future Roadmap

After successful completion of the initial roadmap, AWS provides a working URL that is available for use.

http://genericopayrollsystem.ap-southeast-2.elasticbeanstalk.com/

As the company's needs increase, the cloud architecture will need to be revaluated to meet demand. If the need arises and multiple datacenters are needed in different countries to meet the Businesses international growth Amazon services will need to be selected to be available in multiple zones. Below is a diagram showing what the company should expect in term of future upgradability.

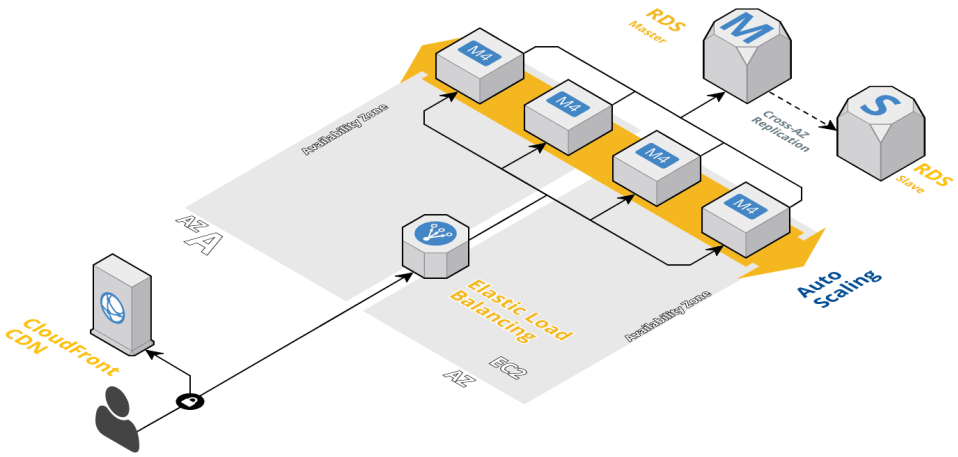

FIGURE 8 REVIEW OF FUTURE ARCHITECTURE [6]

As shown above, as demand increases so will the need for faster computing Elastic Load Balancing will ensure that requests are spread over multiple EC2 instances [3]. As the amount of data stored increases, it may be of benefit to increase the amount of storage available this will include the purchase of a secondary storage database which could be purposed

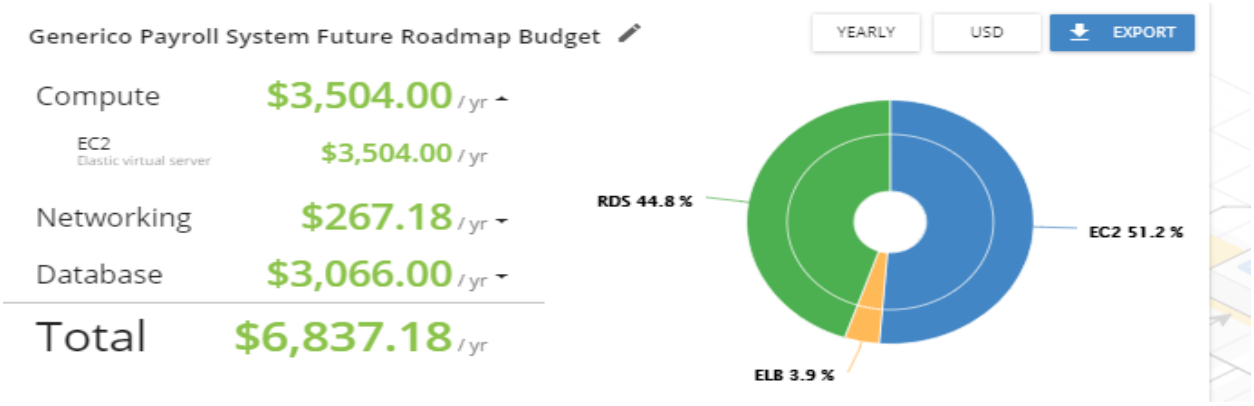

FIGURE 9 TOTAL COST OF LONG TERM GENERICO PAYROLL SYSTEM (ESTIMATION YEARLY) [6]

As Shown in the previous image as the computing needs of Genericos Payroll System so does the cost, As Described you can find an estimation of the total cost of the above implementation of a Future Generico Payroll System. These prices are indicative only and could fluctuate with the need of more computing power or increased storage space.

Also, some thoughts which can affect cloud viability and security should also be followed:

- Implement Login access and update security policies
- Evaluate and monitor performance indicators
- Re-evaluate design choices
- Re-evaluate costs monthly.

Performance should be consistently monitored to ensure that satisfactory levels are being maintained. Also, cost should be monitored carefully to ensure that cost remains in line with Genericos expectations. An Evolutionary Process for integrating(EPIC) approach should be taken to ensure that the cloud system is meeting user demands. [7]

Conclusion

Genericos payroll system was simple to implement on the cloud, using elastic beanstalk a lot of the overhead and management tasks were streamlined creating efficient ways of implementing legacy systems on the cloud. Care should be taken due to any further regulatory requirements and performance evaluations should be taken at regular intervals to ensure adequate performance. Also, Budget should be monitored to ensure no overspending is occuring or to ensure changes to AWS terms and conditions. [7]

Appendix A Group Report.

This report was created by Timothy Finn based on the experience I have with completing the Practical component and with references from Amazon. Timothy Finn has completed 100 % of the work submitted for grade calculation and was approved for an extension to the assignment by Romana Challins.

References

[1] Amazon. (2017, 10 30). *Elastic Load Balancing*. Retrieved from Amazon Corporate WebSite: https://aws.amazon.com/elasticloadbalancing/

[2] Amazon. (2017, 10 30). *Elastic Beanstalk*. Retrieved from Amazon public Corpartion website: http://docs.aws.amazon.com/elasticbeanstalk/latest/dg/Welcome.html

[3] Amazon. (2017, 10 30). *EC2 instance*. Retrieved from Amazon Corparte public website: https://aws.amazon.com/ec2/

[4] Amazon. (2017, 10 30). *Relational Database*. Retrieved from Amazon Coporate Website: https://aws.amazon.com/rds/

[5] Amazon. (2017, 10 30). *Instance Types*. Retrieved from Amazon Coporation: https://aws.amazon.com/ec2/instance-types/

[6] Cloud Craft. (2017, 10 30). *Cloud Craft Coporation*. Retrieved from AWS Cost Simulation: https://cloudcraft.co/app

[7] Challans, R. (2017). EPIC- Lecture. *Cloud Computing Lecture.* Adelaide.