University of California Santa Cruz

J. Baskin School of Engineering

Developing Resources with Microsoft Azure Cloud-Based Systems

Adrián Murillo

Testing, analyzing, reviewing, and exploring Microsoft Azure cloud computing platform, and comparing it to in-house computer systems and market competitors.

Intended for people interested in implementing or upgrading to Microsoft Azure or other cloud systems. Considerable knowledge of computers and networks needed, although no coding experience necessary.

IEEE Format

Instructor: Gerard Moulds
CSE 185E - Technical Writing for Computer Engineering

December 2019

Contents

1 ABSTRACT

Microsoft Azure is a cloud-based computing service where users are able to build, test, deploy, and manage applications and systems using Microsoft's cloud data servers. Azure provides customers with services categorized as PaaS (platform as a service), IaaS (infrastructure as a service), and Saas (software as a service), which encompass all the resources available by Microsoft. Additionally, it allows for the usage of a large variety of programming languages as well as access to external software, both Microsoft and third-party [4]. This system claims to be better than those of its primary market competitors, and a clear solution for upgrading localized IT services.

In this paper the capabilities and competitive advantage of using Microsoft Azure will be explored, and we will review the benefits from upgrading the IT systems of an in-house server to a cloud based one, such as Azure. For this we will first review the architecture, modeling, and services provided by Azure, and relate them to their pay-to-use price rates. Additionally, we will look into different sources and research projects that have reviewed the computing power, memory management, and development services in relationship to its cost and those of its competitors, and develop projects of our own to review the advantage of using such system and compare them to those of personal computers or a college UNIX timeshare.

2 INTRODUCTION

In the modern corporate world, companies are looking for more technological and secure ways of handling their IT services and their network and software operations. Hosting in-house IT services can have a multitude of disadvantages, such as low availability, fault tolerance, and stagnant scalability and agility, and price [**?**]. A popular service that is starting to gain momentum is that of cloud sourcing. Cloud sourcing is an arrangement were a company contracts a third-party cloud hosting provider to deliver and support IT services that could be provided internally. The term comes from ”outsourcing”, which refers to hiring an external service provided by a company to take care of tasks, but in this case it centers on the technology of cloud computing [6]. One of the most popular and effective cloud sourcing platforms is Microsoft Azure, created by Microsoft Corporation. Microsoft Azure is a cloud computing service that allows building, testing, deploying, and managing applications and services through Microsoft-managed data centers [4].

Microsoft Azure allows the development of multiple resources through its online portal based in portal.azure.com. The advantage of using Microsoft Azure over in-house servers is the advanced processing and computational power that it offers to the users, and the amount of usable memory that they can opt to have access to. Working with localized IT platforms can not only pose as a computational disadvantage in terms of processing power and availability, but also can have serious economic repercussions in a company.

Microsoft implements resources such as virtual machines, app services, firewall developments, and SQL databases, among others, to allow users to maximize the effectiveness of their software, together with the help to manage it and fix or upgrade it when needed. Additionally, it has the availability of linking multiple workloads and projects with one another across different Microsoft-supported applications, allowing the user to set specific bandwidths that serve for a large variety of specific tasks [3].

Compared to its competitors, Microsoft Azure also allows for better implementation of applications regardless of what system it is being adapted from. Not only does it show to have better management of storage and execution environments, but it adapts to the needs of the user as it implements the different services provided by Azure. Microsoft Azure poses to be the most cost-effective and reliable cloud based platform, and we will explore this claim throughout this research paper. We will be reviewing multiple research documents and books as well as developing our own projects for reviewing and conceptualizing the cost-effectiveness, reliability, and capacity of using Azure.

3 METHODS AND DEFINITIONS

3.1 Methodology

As explained before, in this research paper we will be analyzing the resources and capabilities of the Microsoft Azure platform, and explain different aspects of the computational architecture of the cloud systems and their resources. With this process we intend to explore the capacities of developing applications with Azure, and analyze the benefits of using this platform over in-house computer systems or other cloud platforms available in the market. For this we need to understand the computational architecture of the Azure platform, along with the different design characteristics that differentiate it with other computer systems. Additionally, we will need to evaluate the effectiveness of using

Azure resources and explore the capacities of using it. Finally, we will need to research the benefits of upgrading to Azure services in a macro level, to understand how a company would benefit from setting their IT services to a cloud-sourced one.

We will be dividing this research in two segments. First, we will develop three projects using Microsoft supported programs, and deploying them through Azure cloud services. This projects will consist of: a web application using Visual Studio 2019, using HTML programming language; setting up and testing a UNIX virtual machine through Ubuntu Computer System, and an SQL database. To achieve this, we will need to first create an account on Azure.com and develop a starting portal. Then, we will review tutorials and user manuals provided by Microsoft or by members of the community to help us develop the applications. Also, we will be sourcing public code from different programmers for time and resource reasons. All of these projects will be developed with the main goal of reviewing the different services that Azure provides to its users, as well as test their effectiveness and reliability to link our projects to the platform.

The second part of this project will consist of the analytic review of other research projects that have tested and analyzed the different implications of developing software with Azure and its many supported systems. Additionally, I will study texts that help explain the macro and implementation advantages of Azure in a major scale level. This will give us a more clear understanding of the capacities and limitations of the Azure resources, in addition to a more clear understanding of the architecture, global name space, design, and data model, resource provisioning, load balancing, and replication systems implemented by Azure. We will be analyzing four primary texts, which consist of the following:

Chesire, Jim, *Microsoft Azure Fundamentals, Exam Ref AZ-900,* Pearson Education, Inc. 2019. Print

Insik Kim, et al. *Cloud Computing for Comparative Genomics with Windows Azure* Platform Evolutionary Bioinformatics. Libertas Academica. 2012. PDF.

Calder, Brad. et al. *Windows Azure Storage: A Highly Available Cloud Storage Service with Strong Consistency Microsoft. 2011. PDF.*

Huang, Cheng. et al. *Erasure Coding in Windows Azure Storage Microsoft Corporation. PDF.*

3.2 Technical Specifications

The design of Azure and its large variety of resources available for users are centered around the idea of cloud systems providing different services according to user needs. The base three types of cloud services provided by Azure are: Platform as a Service (PaaS), Infrastructure as a Sercice (IaaS), and Software as a Service (SaaS), which are concepts we will be using throughout this paper. This concepts explain the relationship between user access along with the software and processes provided. Consider the following image before trying to understand this concepts.

3.2.1 IaaS

Infrastructure as a Service is a term used in cloud computing as the categorization of any cloud service in which only the hardware is provided through cloud connections. This means that when a user requests a provider for a virtual machine, referred as VM, or a network framework they get access to all of it. The most common way of IaaS service is when a provider gives access to a company for the usage of their servers, where they define

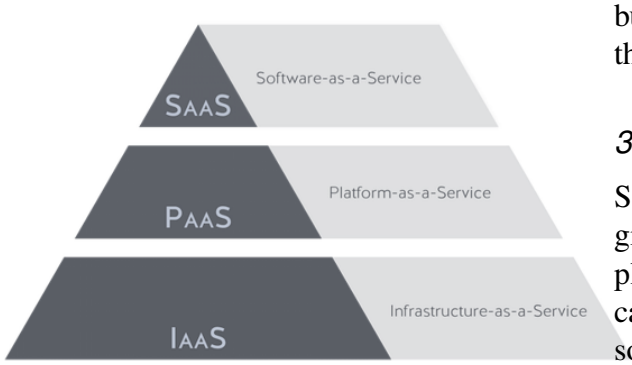

Fig. 1. Relationship between IaaS, PaaS, and SaaS, related to the amount of control the user has over their software

how much memory, RAM, and processing power they need. This usually is set to a pay-by-usage system.

As we can see on Figure 1, the service that allows for the most control over the system is IaaS. In our previous example, the user has complete access over the VM, which includes patching updates, setting memory spaces, and configuring the environment. While this allows for more freedom and control, it gives a lot of responsibility to the user. This is because the user needs to build or install the OS and applications if they want to use them.

3.2.2 PaaS

The second level of the control pyramid is given to Platform as a Service. In a PaaS environment, "a cloud provider still provides the infrastructure for you, but they also provide the operating system, software installed in the operating system to help you connect to databases and network systems, and many features that enable your to build and manage complex cloud applications" [**?**]. While this system has the best of both worlds, it still makes the user responsible for coding, managing, deploying, and building the applications, but it gives all of its software resources to help the user.

3.2.3 SaaS

Software as a Service is when the provider gives access to the user to software that the platform has complete access to. In most cases, the platform provides the best possible software that they can develop, and availability from pretty much any device. In this type of services, the users are only able to access the I/O nodes that the programs produces, in other words, the provider manages the applications. This is particularly helpful for hiring services for people who have no computer experience to use, as it is simple, reliable, and effective, as the provider will make sure to give the best experience and tools to the user. An example of this service is GMail or OutLook, where users link to a cloud system filled with billions of emails, where users can only access their own accounts.

4 Results

4.1 Testing Azure

By creating an account in the Azure portal we gained access to all of the resources provided by Microsoft, as well as the possibility to link our resources to other Microsoft supported programs, such as GitHub, Visual Studio, Microsoft Power Shell, and the CPU Command Line. We were given $200 credit for creating a new account in a pay-to-use charging system, which we opted to only use the free versions of the resources and only use the upgraded versions for reference or if stricly necessary. Testing the Azure resources gives us insight to how the different services of SaaS, PaaS, and IaaS are provided by Azure, and how they work.

4.1.1 Web Application

Right of the bat Azure provides users a large variety of options to develop Web App Services. Users are eligible to use .NET, JAVA, NODE.js, PHP, and Python on Windows, or .NET Core, Node.js, PHP or Ruby on Linux systems. Additionally, users can deploy their websites from almost anywhere, as the ability to link source code from the Power Shell, the local PC command line or from a GitHub repository. There are also multiple development IDEs for which we chose Visual Studio 2019 for this project, since is the IDE that is developed by Microsoft and is recommended for deploying in Azure. This IDE is not only a great option for developing source material for website design, but also it achieves the capacities and freedom of most available IDEs while having direct access to the Azure Cloud Services. We decided to develop our website using the HTML program building format.

For designing our website we utilizing a ASP.NET framework with a MVC design architecture (Model, View, Controller). We created a simple website using JavaScript programming language. In this project we defined a really simplistic design of a website, just enough to allow us to deploy it. We developed it using the CSS stylesheet language. For reference, the URL of the website is:

`https://mvcmovie20191130045126.azurewebsites.net/`

This is a trivial implementation of a website. Visual Studio 2019 is a SaaS service so when we deploy it through Azure, the system helps us fill in the blanks with their default HTML template. The Azure platform lets us decide the deployment location, for which we chose North Europe. Additionally, we chose the F1 plan, which allows us 1GB of memory, and 60 minutes of dynamic memory usage per day, while sharing a .NET based infrastructure.

Azure web services allows us to scale out or scale up, since we have the possibility of developing linking websites through their database framework or to upscale our resource usage to a higher usage plan. If up-scaling is needed Azure provides a maximum of 7.5 GB of memory, A-series computer equivalent, and a total of 400 ACU. Additionally, we can opt to have our own private URL port if needed, completely independent from the Azure default building software. The next figure shows a screenshot of the title page in our website:

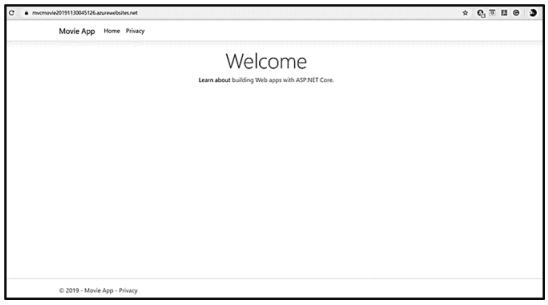

Fig. 2. Screenshot of Home Page of developed website

By using Visual Studio 2019 we can automatically link our source to our publishing allocation in the cloud platform, which allows for better managing and developing of web applications.

For this section we actually managed to create a secondary website from a public access repository, sourced from user **bcaruthers** in GitHub. The following is the SSH clone URL:

`https://github.com/bcaruthers/hotel-website.git`

This website is a simple template for a hotel website, in which sections and controllers are included to traverse through the application. This time we decided to use the

Microsoft command line from our Lenovo Y-520, which runs a Windows OS. We used the v2 command-line tool of Azure CLI, which allows access to our website services through our processor. Even thought this might seem as a IaaS, it is still a PaaS, since we have access to all installed modules on the Azure platform. At the same time, we are able to manipulate the configuration of the website through the AZ CLI access.

We found in both implementations an easy access to the Azure network, as the accessible workloads are set to rooted origins in either our IDE or our processor. Additionally, Azure makes it very easy to adapt source from any repository, by managing multiple programming languages and code architectures. Furthermore, Azure provided repair and bug fix services for both building and managing the web application, which responds to unhealthy servers, malware, harmful access points, and many other causes of problems with a website.

4.1.2 Unix Virtual Machine

The Azure platform also allows the development of a VM (virtual machine) sourced in any of their multiple server locations in the world. For this project we decided to implement a standard Ubuntu Linux Server 18.04 LTS. For creating an Ubuntu VM, there was no free option, so we chose the smallest size which was a D2s_v3 server for general purpose, with vCPU 2, a RAM memory of 0.5GB, and a max IOPS of 3200, and a temporary storage limit of 1 GB. This CPU was comprised of 2 data disks. All of this resulted in an estimated cost of $3.87.

Once I set up the VM, Azure allowed to deploy and manage it anywhere in the world, similarly to the web services stated before. The system that was provided was a complete blank slate of a Linux system, which holds true for all IaaS services. The VM also had its own accessible timeshare, which we were able to access directly from our Power Shell or our command prompt execution, without needing to log-in to the portal every time we want to access it. The usage of the VM was pretty straight forward; I uploaded several previous coding projects (File compression and decompression program using LWZ, buffered reader using bloom filters and hash tables, and 4 different sorting programs), installed a few compilers and programs, developed and ran bash scripts, and changed several system variables to my personal preference. In addition, it was beneficial to use a Ubuntu VM since Ubuntu has access to most programs, shells, compilers, among others. From the Azure portal we could link our VM to any resource group or resource project in any account, which proved to be advantageous. Finally, I managed and accessed the website portrayed in subsection 4.1.1 with ease, even while accessing the VM through my own command line.

The scalability of the VM is outstanding, as Azure provides up to 3400 GB of RAM, 17880 GB of temporary storage, 64 data disks, and a vCPU of 128. Additionally, it provides specialized VM for either memory or computing power, allowing users to adapt their VMs to what they need the most. Of course the prices of such computers are very high, going up to $20,000 a month, but in a company level, the amount of advantages of using such system would make it worth the price. All things considered, the Ubuntu VM used in this project didn't run into any problems. Obviously the processing power of our VM was underwhelming, but that is because we chose the smallest VM package possible, which ended up costing around $0.4 after all usage was finished. As expected, the VM ran everything without any errors or strange outcomes.

The next step towards analysing the capacities and advantages of using Azure was by developing an SQL database, which is another one of the resource available for Azure customers. It would be a waste of time to experiment the different execution and storage capabilities of an SQL database hosted in a cloud system, since we already know how SQL databases work, and we are conscious of the scalability of an SQL database by spending more money on more storage space as well as computing power. For this section of the research I mostly focused on how Azure allows for easy implementation of an SQL database, in addition to the different advantages of deploying using their cloud based systems.

4.1.3 SQL Database

Azure provided a 100MB database with access to 10 DTUs in a SaaS service. The estimated cost per month was set to $4.99. I set the deployment server to North Europe. Azure also allows users to allocate memory in their database using SQL elastic pools, which are ”a simple and cost effective solution for managing the performance of multiple databases within a fixed budget. An elastic pool provides compute (eDTUs) and storage resources that are shared between all the databases it contains. Databases within a pool only use the resources they need, when they need them, within configurable limits”[**?**]. This has the advantage of making the price of usage in accordance to how much the user is needing, which we opted not to use this time.

To save time I transferred the contents of a previous SQL database to the one created by Azure, and ran scripts previously designed for managing said database. The database used was a previous SQL project that hold a database for the information on countries, their capitals, leaders, and land sizes. I used Microsoft SQL Server Management Studio 2019, which allows easier access to the database as well as linkage to other programs. Again we used a Microsoft developed IDE since it has additional features to connect directly to the Azure platform. As expected the system worked flawlessly. The interesting findings were that users can access their SQL databases through the command line, Power Shell, or even an allocated cloud VM, same as the one developed in subsection 4.1.2. Additionally, we could link said database to other services such as websites, applications, container instances, resource groups, etc. Finally, Azure helps with garbage collection and memory management that allows for more efficient usage, which is something that users have to develop by themselves when using an in-house server.

4.1.4 Further Projects

In this project we developed only 3 resources using Azure services. In another project I could link the SQL database to be accesible from omy website, but that would take a lot of additional work. The possibilities of developing much more projects is available in Azure, as it can also develop resources such as: OS systems, Virtual Networks, Load Balancers, FireWalls, Citrix Virtual Apps, Route Tables, and many more. Aside from this project I will continue to familiarize myself with the Azure platform, and develop more complex and varied projects on my own. Additionally, the possibility of getting an Azure Certification could be beneficial for further understanding of the systems.

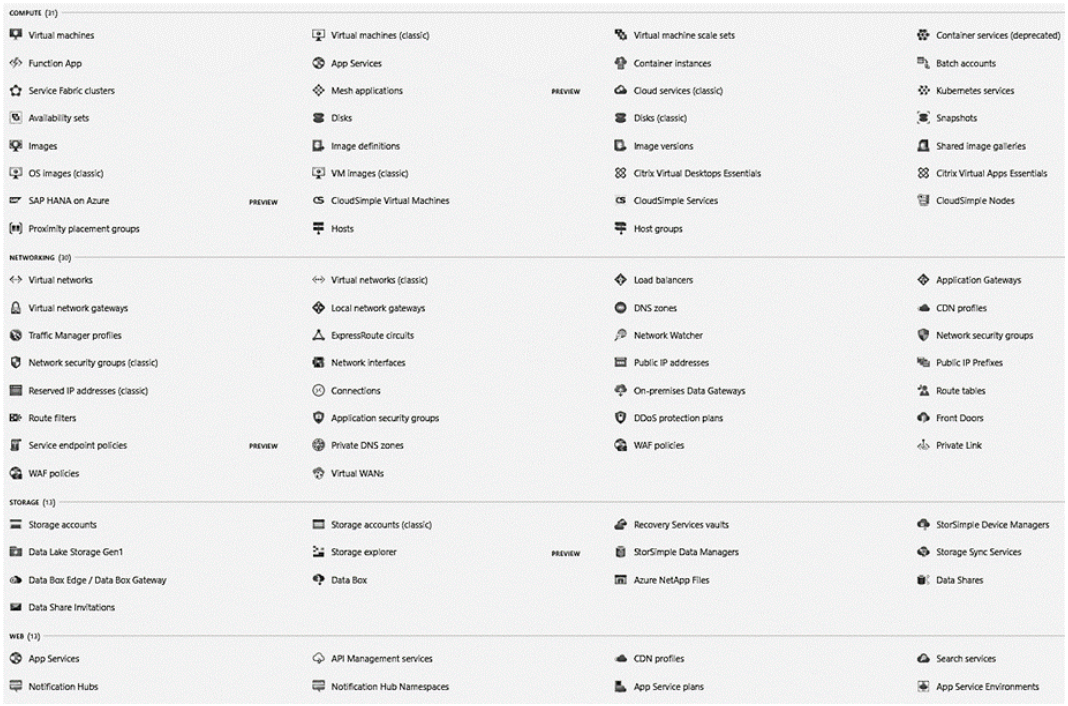

Fig. 3. Examples of Azures resources available for customers (Picture taken from the Azure Online Portal)

4.2 Literature Review

In continuation to the anaylis of Azure, I will study and review different texts where the architecture, model, design, and implementations of Azure is analyzed. This will allow us to understand even further as to why is Azure much more powerful, accessible, and price-efficient than its competitors or in-house computer systems. I will be exploring the findings of several research papers and books focusing Azure and its design in the following subsections.

4.2.1 Azure Architecture and Design

As the technology corporations try to deliver to the user the best service among their competitors, Microsoft has been improving their Azure web services since its release in 2008. They suggest that their cloud storage platform and development services have a computational design advantage against its competitors. In the research paper *Windows Azure Storage: A Highly Available Cloud Storage Service with Strong Consistency* by Microsoft researchers, it is described the architecture behind WAS (Windows Azure Storage), their global namespace, and data model, as well as its resource provisioning, load balancing, and replication systems [3]. This is done with the purpose of explaining how the key design features behind the process of building WAS was guided by internal and external customer feedback. This includes but is not limited to: strong consistency, global and scalable namespace/storage, disaster recovery, multi-tenancy, and reduced cost of storage.

The different mechanisms for efficiently using WAS are provided by three different storage abstractions: Blobs, Tables, and Queues, which translate to the essential services for cloud based solutions. This mechanisms allow for all sorts of storage and work-

flow control for a large variety of applications.

Moreover, they affirm that since WAS runs in a disparate set of workloads with different usage profiles on the same set of hardware, storage cost is reduced, giving the customer the best cost efficiency. This is because the amount of resources to be provisioned is significantly less than the sum of the peak resources required to run all of the workloads on the dedicated hardware.

In short, the way WAS is designed allows for more complex application design and management, as it allows to deploy IT services with cost efficiency in mind. Not only that but it delivers a reliable service that consists on pay-to-use servers that allow for more calculated usage. This allows companies and developers the best option in the market in terms of the possibilities of implementation of their workflow mechanisms and cost efficiency.

Microsoft Azure provides users with the possibility of developing major or minor scale projects using a large variety of virtual systems as well as servers. The processing power and its effectiveness in relationship to cost are important advantages that Azure has over its competitors. One problem that many cloud-based platforms have is the management of memory, where data has to be stored in hardware based servers that cost a lot of money and take a lot of power and maintenance to keep running. This often has the effect of an increase of price on memory spaces in the timeshare for the company, which ultimately ends up costing more to the user. Microsoft Azure is steadily getting close to an Exabyte of cloud storage within the Azure services, which can become costly and troublesome for any company, even for big companies such as Microsoft (One Exabyte is equal to 10^{18} bytes, or 1024 Terabytes). Luckily, Microsoft Azure implements memory management and compression algorithms in order to keep the cost of building and maintaining servers to a minimum, which allows for the price of storage to remain as low as possible for customers.

According to a Microsoft Corporation research titled *Erasure Coding in Windows Azure Storage*, the cloud storage platform of Azure manages its memory storage by implementing erasure coding to compress and simplify the data as it gets stored, and then utilize reconstruction algorithms to deliver the data to the user as soon as they need it. In this article, Huang, Cheng and others portray the way in which Azure implements this erasure codes besides LRCs (Local Reconstruction Codes) and explain how it manipulates the different banks of memory into reconstructed binary strings where raw deconstructed data is kept for later use.[3]

According to researchers at Microsoft Corporation [3], memory within Azure is appended into distributed file systems called stream layers. Data is added to working active extends which then are written into 3 full files by the underlying stream layer. This is done to keep the data durable and accessible at real-time. Then, as soon as an active extend is filled to a default space size (typically 1GB) it gets sealed by the erasure codes and it is no longer available for direct usage. Then, the erasure algorithm lazily deconstructs the user data stored in memory into fragmented nodes accessible only through reconstruction, by using LRC. Finally, the 3 copies created by the used stream layer are deleted, releasing the memory for stream usage. When the user needs to access that data again, reconstruction algorithms separately rebind the data in a working tree where each node is attempting a different approach. Once at least one node reconstructs the data it is handed to the user and the rest is dumped. Three new copies

are assigned to the current working execution directory of the user.

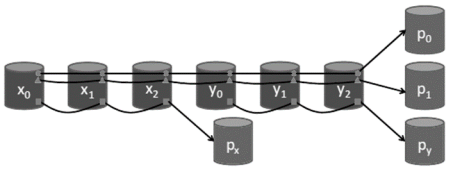

Fig. 4. A (6, 2, 2) LRC Example. (k = 6 data fragments, l = 2 local parities and r = 2 global parities[3])

The main purposes of LRC in memory management for the Azure platform are the following:

1) Reduce the minimal number of fragments that need to be read from to reconstruct a data fragment. This provides the following benefits:
    i) reduces the network overhead and number of I/Os to perform a reconstruction;
    ii) reduces the time it takes to perform the reconstruction since fewer fragments need to be read. The time to perform the reconstruction is often dominated by the slowest fragments (the stragglers) to be read from.
2) Provide significant reduction in storage overhead to 1.33x while maintaining higher durability than a system that keeps 3 replicas for the data.

[3]

The erasure codes maximize speed when accessing sealed memory fragments. The main disadvantage of using LRCs and Erasure codes is that sometimes the access to fragmented sealed data may have higher latency when dealing with more complex memory management, specifically with a hot storage node or when there is a lost or offline data fragment. This way of compressing and managing memory allows Azure to reduce the size of its servers, ultimately cutting down costs which keep the prices for the consumer as low as possible. According to the researchers, Azure provides similar storage access as its competitors for a much better price, or much higher storage access for the same price [1]. All of this shows that Azure not only is more cost efficient than its competitor in terms of processing power and resources, but also in memory space.

4.2.2 Comparative Analysis

We can see how WAS can be used to develop major scale applications that require high processing power as well as large amounts of dynamic memory in the project developed by Libertas Academica, described in their paper *Cloud Computing for Comparative Genomics with Windows Azure Platform* [1]. In this project, researchers reviewed the Microsoft Azure platform in comparison to the Amazon Web Services, which is their primary market competitor, by deploying an application using their corresponding run-time environments. This was made with the goal of analysing the details needed to run the application, and observing how the data management, storage services and the execution environments handle the programs.

The case study consisted of the running of a computationally intensive comparative genomics application called RoundUp. The process was divided into two stages: running the application using a single Azure instance for prototyping and testing; and a test run with the scaled application with multiple instances with Azure High Performance Computing Scheduler. This stages were split into two executions were one was done by using WAS and the other was by using Amazon Web Services, their primary market competitor.

The researchers found that the cross-platform migration from a Unix System to a

Windows-based Azure platform was successful and obstacle free, as Azure has the tools to run the same programs using a Unix Virtual Computer System [1].

They achieved to analyze how the platform behaves while running a computationally intensive task of orthology inference using the RoundUp algorithm, and relate its computing capabilities to its cost-effectiveness. They recognize as superior the ratio of computing capabilities-to-cost of the Azure Platform against its competitor, and even believe that the usage of servers and multi-layered processes could have been administrated in a better way to reduce costs even further, possibly opting to implement their systems through an IaaS database. The only problem that they faced while running the application using Azure was that the runtime environment doesn't allow for control of the individual computing nodes to the user, which limits the control that they have throughout the execution. This again was produced by the usage of an PaaS system, and upgrading to a IaaS would have fixed this problem.

4.2.3 Macro Advantages of Azure

We considered the differences of implementing Azure on a minor scale using simple projects in addition to reviewing other projects that reviewed the effectiveness, computing power, and cost efficiency of Azure when used in a more advanced research setting, which requires one-time-use high processing power. I will now explore the advantages of upgrading to Azure in a company level.

When a company needs a lot of processing power or large amounts of memory storage, the possibility of building in-house servers are always an option. They may use these computers to handle their online servers, manage and store their databases, automate different aspects of operations, or allow for communication among employees. The obvious disadvantage of using in-house systems rely on their lack of scalability, agility, and development. This means that when a company wants to upgrade their computer systems, they have to invest large amounts of money in the building, managing, and maintenance of these systems, besides of the immense power consumption[5]. Additionally, the company would have to hire a large team of IT workers to manage and maintain the servers, and constantly fix issues. The building of servers in a company most of the times takes a long time for a full return of investment. This makes it so companies can't upgrade their systems often, making them obsolete as the years go by. By using cloud services, this obvious problems are taken care of, as systems like Azure constantly update their infrastructures and give their users access to only what they need to use at the time. This is called scalability, which is the ability to upgrade at any time [5]. Moreover, when using in-house systems the possibly of malfunctioning servers increases, as well as the vulnerability to power outages, viruses, and data corruption and fragmentation. Azure services will handle these problems for the company as their fault tolerance and disaster recovery services give the servers high availability.

Another challenge is the development of effective software. Lets say that a company wants to adapt a Google sign-in feature on their website, it would take a lot of experienced software engineers to create such plug-in, which would cost the company a lot of money [5]. In this scenario, Azure provides this and many other types of linkages and software tools that developing efficient software is much easier and cost efficient. Moreover, Azure provides users with the latest software

technology, making obsolete any self developed software that a company might use. This is true especially for non-tech companies, where their focus has to be elsewhere rather than in their IT department.

Besides, Azure proves to be superior when it comes to giving customers exactly how much they need at a certain point. Azure allows users to set scaling automatically whenever the servers used require higher processing power or larger memory storage. This concept of automatically scaling is called elasticity [5]. Likewise, their IaaS, PaaS, and SaaS focused services allow users to have access to as much control and resources is needed, making it more cost-efficient to pay only for services that are being used

Finally, as we explored in section 4.1 ”Testing Azure”, the services provided by Azure services go beyond just providing cloud servers for software development and usage. They give access to efficient and reliable software that allows for much easier implementation and development, making the process of developing IT services more time-efficient. Azure provides so many different resource options and plans that there is a service for pretty much any project.

All in all, Azure Services provide users with a lot of benefits in comparison to developing in-house servers, and the ones covered in this section scratch only the surface of the countless cons of upgrading IT services to cloud servers.

5 Discussion

The services provided by Microsoft Azure proved to be reliable and cost-efficient when used in a minor scale, aside from being scalable and elastic to what I needed. Moreover, the services provided in the platform allowed for any sorts of project development and implementation from any system or format. The resources developed in this researched project proved that the Azure servers allow for specific and adapted formatting for what we needed. The Azure web services were easily accessible, and provided a lot of tools for managing and upgrading the website. Also, the web template and default settings gave a solid starting point to the development of the website, saving valuable time that was spent developing a second website. The server linked to our website was also available for access from multiple sources. The SQL database proved to be reliable and efficient, and suing Microsoft SQL Database Development tool allowed for more options when it came to managing our database. The VM developed in this project also proved to be reliable, as I managed and configured the CPU as much as I wanted.

The main advantage that I found when developing resources with Azure is the many possibilities of linking resources with one another between members of the same resource group. The possibility of linking my SQL database to my website through the created VM was a possibility, but for time reasons I decided not to develop. All of these proved also the efficiency and dependability of using Azure services, as I ran into no errors during my implementation.

In another note, it was seen how Azure’s architecture and logic design allows for more efficient usage, specially when it comes to memory. This allows Azure to reduce costs for their customers. Also, the comparative study proved that Azure has higher computing power and memory access in relation to cost that its primary competitor, Amazon Web Services.

The superiority of Azure over in-house systems and other cloud services in terms of computing power, reliability, scalability,

elasticity, agility, and fault control make it the best option for developing major scale IT services. Lastly, the capacity to adapt to the user's needs through their large variety of services and their focus on either PaaS, IaaS, and SaaS services proved to be almost necessary when it comes to macro software and network development, especially when it comes to companies.

6 Conclusion

The advantage of using Microsoft Azure over in-house servers is the advanced processing and computational power that it offers to the users, and the amount of usable memory that they have access to. Additionally, it has the availability of linking multiple workloads and projects with one another across different Microsoft-supported applications, allowing the user to set specific bandwidths that serve for a large variety of specific tasks. Compared to its competitors, Microsoft Azure also allows for better implementation of applications regardless of what system it is being adapted from. Not only does it show to have better management of storage and execution environments, but it adapts to the needs of the user as it implements the different services provided by Azure.

References

[1] Kim, Insik, et. al. *Cloud Computing for Comparative Genomics with Windows Azure Platform* Evolutionary Bioinformatics. Libertas Academica. 2012. PDF.

[2] Calder, Brad, et.al. *Windows Azure Storage: A Highly Available Cloud Storage Service with Strong Consistency* Microsoft. 2011. PDF.

[3] Huang, Cheng. et. al. *Erasure Coding in Windows Azure Storage* Microsoft Corporation. PDF.

[4] Wikipedia, The Free Encyclopedia, *Microsoft Azure*, https://en.wikipedia.org/wiki/Microsoft_Azure.

[5] Chesire, Jim. *Microsoft Azure Fundamentals: Exam Ref AZ-900*, Microsoft Corporation. Pearson Education, Inc. 2019. Print.

[6] Rouse, Margaret. et. al. *Cloud Sourcing*, WhatIs.com. URL. https://searchcloudcomputing.techtarget.com/definition/cloud-sourcing

[7] Doerrfeld, Bill. *cloud-stack-pyramid-saas-paas-iaas-as-a-service-02*, Nodric Apis, https://nordicapis.com/living-in-the-cloud-stack-understanding-saas-paas-and-iaas-apis/cloud-stack-pyramid-saas-paas-iaas-as-a-service-02/. URL.

.

Acknowledgements

Great thanks to Gerard Moulds from UC Santa Cruz for the assigning of this project, as well as Colin Mitchel who peer reviewed this research paper.

The development of this research paper was made possible by the multiple tutorials and training modules available online for Azure Cloud Systems by Microsoft Corporation. Additionally, for the writing of this research paper we utilized LaTex text editor by OverLeaf.com.