File Transfer Protocol in Cloud Computing

[1]Ch.Madhu Babu, [2]O.Lakshmi Chandana
[1]Department of CSE,Mallareddy College of Engineeringa and Technology,Hyderabad.
[2]PG Student,Mallareddy Engineering College,Hyderabad
[1]madhugist@gmail.com,[2]nakshatra.bujji@gmail.com

***Abstract*—**The cloud computing is growing rapidly for it offers on-demand computing power and capacity. The power of cloud enables dynamic scalability of applications facing various business requirements. However, challenges arise when considering the large amount of existing applications. In this work we propose to move the traditional FTP service to the cloud. We implement FTP service on Windows Azure Platform along with the auto-scaling cloud feature. Based on this, we implement a benchmark to measure the performance of our CloudFTP. This case study illustrates the potential benefits and technical issues associated with the migration of the traditional applications to the clouds.

Keywords- *cloud computing; cloud migration; auto-scaling; dynamic scalability; Windows Azure; SaaS*

*****

I. Introduction

The Phrase “cloud computing”[1] describes it as a system platform or kind of software application. Cloud computing is an on-demand and cost saving computing with scalability, high-availability, and reduced management. Amazon’s Elastic Compute Cloud (EC2) is an example of IaaS (Infrastructure as a Service)[2] platform. It offers basic infrastructure component such as CPUs, memory, and storage. Google App Engine is an example of PaaS (Platform as a Service) platform. In Cloud Computing platform Cloud server is a physical server. Based on IaaS and PaaS platforms, a lot of time and money have been saved for start-up companies, such as foursquare and dropbox.

Cloud Computing is a Technology that uses the internet and central remote servers to maintain applications and data. One major challenge is how to migrate these traditional applications to the cloud. Current research focuses on the migration of specific applications such as high-performance applications [3], but little work has been proposed for the migration of general services.

In this paper, we present a case study moving the traditional FTP server to the cloud. We have implemented the Cloud FTP server on Windows Azure .Based on this, we have implemented a benchmark to measure the performance of our CloudFTP.

II. Cloud Services with Azure

Windows Azure is an environment run by Microsoft, Unlike IaaS provided by Amazon and PaaS offered by Google App Engine, Windows Azure uses the mixed PaaS and IaaS strategy, this paradigm makes developing in the cloud fully flexible. Developers could choose PaaS or IaaS depending on their own needs.

When developing cloud applications, three components are mainly used: a Compute service that runs cloud applications, a Storage service providing persistent storage and a Service Bus to exchange messages in a loosely coupled way.

The Windows Azure Platform consists of three main parts:

1. The Fabric: The Fabric basically a network of all the Microsoft machines that are dedicated to running the windows azure platform.
2. The storage: Storage in azure can be represented by tables, queues and binary large objects.
3. The API and SDK: This is the layer that developers work in.

Hosted services in Windows Azure are said to contain roles and there are two types of roles available: a worker role and a web role. Worker roles are frequently used for long-running parallel tasks that non-interactive. Unlike the high level parallel computing framework like Hadoop[4] and Dryad[5], worker roles are not constrained in the way they communicate with each other for each work role stands alone in a virtual server.

Unlike blobs and tables, which are used to store data, queues serve another purpose. A primary use is to allow web roles to communicate with worker roles, typically for notifications and to schedule work.

Figure 1 illustrates the typical design pattern of cloud applications on Windows Azure. The web roles are the front end to accept the user request, and then the task messages are written into the message queue and the worker roles reads from the message queue to take their jobs. Both web roles and worker roles take the storage service to store information.

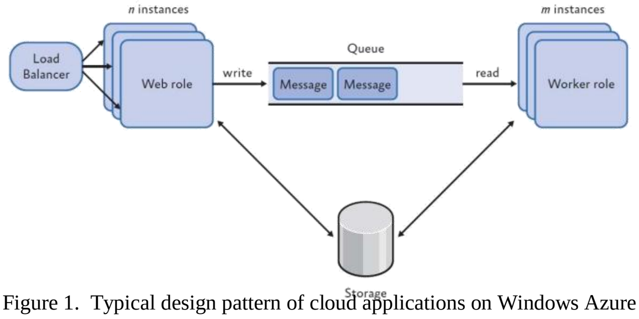

Figure 1. Typical design pattern of cloud applications on Windows Azure

III. Cloud FTP

A. FTP

File Transfer Protocol (FTP) is a standard network protocol used to transfer files from one host to another host over a TCP-based network, such as the Internet. The protocol is specified in RFC 959 [7].

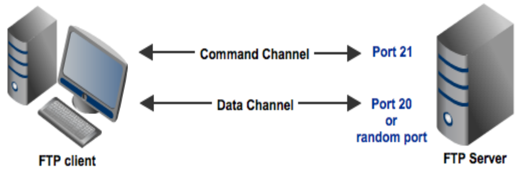

Figure 2. FTP Command and Data Channel

FTP is built on client server architecture and uses separate control and data connections between the client and the server. FTP may run in active or passive mode, which determines how the data connection is established:

Active mode: In active mode, the client creates a TCP control connection to the server and sends the server the client's IP address and an arbitrary client port number, and then waits until the server initiates the data connection over TCP to that client IP address and client port number.

Passive Mode: In passive mode, the client uses the control connection to send a PASV command to the server and then receives a server IP address and server port number from the server, which the client then uses to open a data connection from an arbitrary client port to the server IP address and server port number received. Passive mode may be used in situations where the client is behind a firewall and unable to accept incoming TCP connections.

B. System Architecture

In this section we present the system architecture of CloudFTP. CloudFTP follows the application model suggested for general Azure development as shown in Figure 3.

The users could upload and download files through ftp clients and the administrators could manage the cloud ftp through web portal. After FTP server boot up, whenever a connect request comes, there will be a slave thread spawned to handle all FTP request from the connection.

The worker role is responsible for transferring data. When an FTP client, such as FileZilla, requests a file, then the worker role sends the requested file data to the FTP client. And the number of worker role instances could be adjusted dynamically without affecting other instances. When a large number of clients attempt to connect, the number of worker roles would be increased. When the clients reduce, the number of worker roles would be decreased.

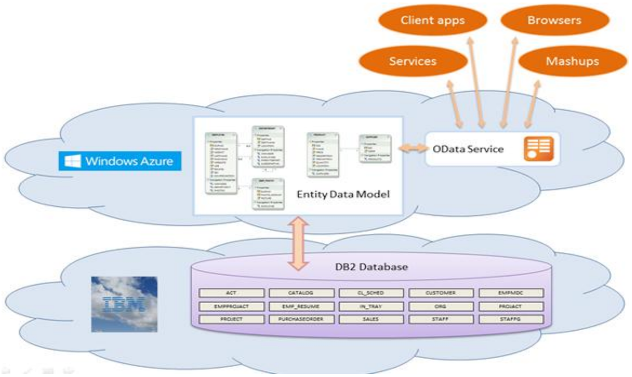

Figure.3 Window azure an d cloud computing FTP

The table storage service is in charge of storing users' information, configuration information and the performance information. The users' information includes user name, the password, and the home directory. The configuration information includes permissions of users and groups. All these three types of information are suitable for table storage service, for they are all structured.

Key features of this architecture include:

1. A globally distribution system that tracks both local and cloud based file versions and locations.
2. Dynamic file locking that are used to update the file status, namespace entries and payload changes.

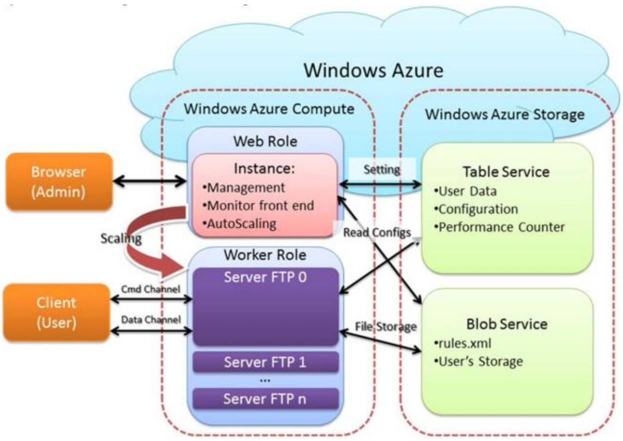

Figure 4. System architecture of Cloud FTP

C. Design of File System

In this section, we explain the design of file system in Windows Azure platform, which is the most important part of our cloud FTP. The Windows Azure Platform offers several storage services, i.e. blobs, tables, queues and SQL servers. Among these 4 storage services, we choose blob storage service as the backend of our file system, because blob storage service is able to store very large files and the key-value style storage provide us convenient ways to implement file system interfaces, such as find, create and delete files. The design of blob service based file system is shown in Figure 5.

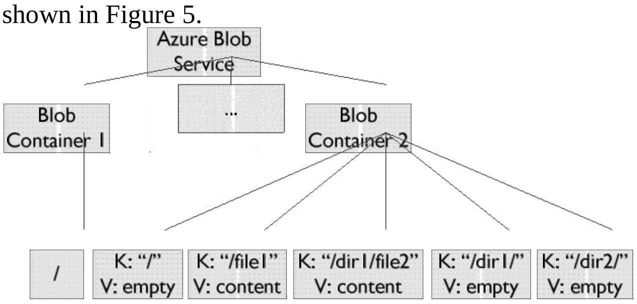

Figure 5. File System Design

In our implementation, a file system corresponds to a blob container in Windows Azure Blob Storage Service. For a file in such a file system, we use a blob to store it and the key of the blob is the absolute path of that file and the value of the blob is file content, besides, we also use a blob to store a directory. The absolute path of a directory is stored as a blob key, which ends with a "/", and the blob that stores directory has an empty value.

D. Design of FTP Server

In this section we present the design of FTP server in Windows Azure platform. We use a worker role with an infinite main loop to listen at the request port and once a client sends a connect request, the server creates a new thread to handle that request. The codes running in the new thread implement all commands in standard FTP. Many storage services can implement our VFS interface. In our project, we use the Azure Storage Service to implement it. Figure 6 depicts the design of FTP server.

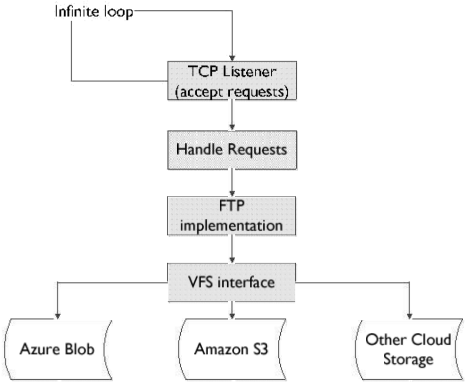

Figure 6. FTP server design

E. Passive Mode

Passive FTP is more secure form of data transfer in which the flow of data is set up and initiated by the FTP client rather than FTP server. Developing on Windows Azure Storage service is just like playing with a black box, you are not as free as developing on local operating systems.

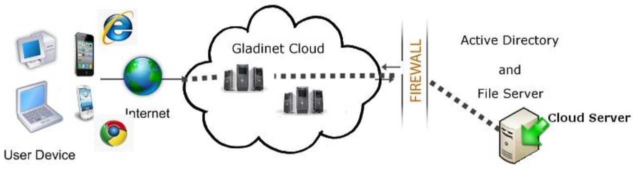

Figure 7. Configuration of FTP in Cloud

V. Related Work

In summary, though cloud migration is not a totally new idea, there is no case study to present potential advantages and technical issues on cloud migration, especially on Windows Azure platform.

VI. CONCLUSIONS

This paper presents a case study to migrate traditional applications to the cloud. We implement CloudFTP on Windows Azure along with the auto-scaling feature. We also implement a benchmark to evaluate the performance of the cloud ftp server. From the design and the evaluation results, we summarized potential benefits and risks to migrate traditional applications to the cloud. The summary could help cloud developers migrate traditional applications quickly and safely, especially on Windows Azure platform.

References

[1] M. Armbrust, A. Fox, R. Griffith *et al.*, “A view of cloud computing,” *Communications of the ACM*, vol. 53, no. 4, pp. 50-58, 2010.

[2] S. Bhardwaj, L. Jain, and S. Jain, “Cloud computing: A study of infrastructure as a service (IAAS),” *International Journal of engineering and information Technology*, vol. 2, no. 1, pp. 60-63, 2010.

[3] W. Lu, J. Jackson, and R. Barga, "Azureblast: a case study of developing science applications on the cloud." pp. 413-420.

[4] T. White, *Hadoop: The definitive guide*: Yahoo Press, 2010.

[5] M. Isard, M. Budiu, Y. Yu *et al.*, “Dryad: distributed data-parallel programs from sequential building blocks,” *ACM SIGOPS Operating Systems Review*, vol. 41, no. 3, pp. 59-72, 2007.

[6] M. Stonebraker, “SQL databases v. NoSQL databases,” *Communications of the ACM*, vol. 53, no. 4, pp. 10-11, 2010.

[7] J. Postel, and J. Reynolds, “Rfc 959: File transfer protocol (ftp),”
InterNet Network Working Group, 1985.

[8] B. Calder, J. Wang, A. Ogus *et al.*, "Windows Azure Storage: a highly available cloud storage service with strong consistency." pp. 143-157.

[9] N. Leavitt, “Is cloud computing really ready for prime time?,” *Computer*, vol. 42, no. 1, pp. 15-20, 2009.