Name: Raj Patel

Project Advisor: Dr. Erick Maxwell

Group Name: MedCap

Storage and Processing Frameworks for Streamed Data

Introduction

Streamed data encompasses information gathered from one or more sources which will be stored and continuously analyzed for future use. The availability of distributed measurement devices has created a need for batch analytics via cloud storage and computing. Several frameworks have become available among cloud providers to store, process, and categorize data. This paper reviews the commercial applications and their underlying technologies available in modern cloud infrastructures.

Applications of Data Storage and Processing

Cloud Focus

Modern data storage and processing frameworks are cloud-based due to the demand required to process large streams or batches of data. Cloud providers utilize clusters of physical servers and storage media in order to offer high throughput and large-scale computing. Cloud providers offer resources which can be used to store and process input data and provide categorized results via analytics [1].

Framework Applications

Two large-scale cloud providers are Amazon Web Services (AWS) and Microsoft Azure. These companies offer cloud computation via virtual machines, which are partitioned sections of the available cluster of physical servers. Additionally, both AWS and Azure offer data storage in databases named DynamoDB and DocumentDB, respectively [2], [3]. These databases offer methods of storing, organizing, and retrieving data for use in computation and analytics. Finally, as part of the data processing framework, both cloud providers offer data analytics applications which leverage the underlying virtual machines and databases to perform computation and categorization on stored data. AWS and Azure both offer Hadoop analytics platforms to characterize large groups of data [1].

Cost

AWS pricing for virtual machines ranges from $0.0065 per hour to $1.68 per hour [4]. Similarly, Azure virtual machines cost between $0.018 per hour and $1.025 per hour [5]. Both companies' price ranges are based on CPU and memory specifications. AWS pricing for DynamoDB is $0.0065 per hour per 36,000 writes and $0.0065 per hour per 180,000 reads [6], while Azure pricing for DocumentDB is $0.008 per hour per 360,000 reads or writes [7]. Both AWS and Azure leverage virtual machines when performing analytics, and pricing is defined by the cost of the respective virtual machines [2], [3].

Technology of Data Storage and Processing

Virtual Machines

Cloud providers offer virtual machines as computational space to host services for clients. Virtual machines are subsets of physical servers residing in clusters around the world, and they can be configured with varying CPU and memory amounts to ensure clients are able to select efficient computational capacity for their specific applications [1]. Configurations vary from one CPU with 512 megabytes of RAM to 32 CPUs with 60 gigabytes of RAM [4], [5].

Key-Value Databases

AWS DynamoDB and Azure DocumentDB are both non-relational key-value databases. This technology associates data in tables where each row is a data entry and each data entry contains keys and their associated values. For example, a sample data entry could be a reading from a weather sensor, where the keys are timestamp and temperature. The corresponding values would be the timestamp and the numerical temperature reading; this storage method allows for the data to be organized for direct access [8]. The throughput of a key-value database corresponds to the amount of reads and writes (stores and accesses) available per second. For AWS, the throughput is limited to 40,000 reads or writes per second per table, while Azure limits the throughput to 250,000 reads or writes per second per table. Both cloud providers offer increased limits upon special request [9], [10].

Data Processing and Analytics

Data processing and analytics frameworks read organized data as inputs and write categorized conclusions as outputs. Due to this relationship with the key-value database tables, the throughput of the data processing unit is constrained by the read and write throughputs of the databases used for input and output data. The analytics are also constrained by the processing capacity of the underlying virtual machines, and CPU and memory specifications must be chosen with respect to the intended data and stream or batch size which will be analyzed [1]. Another constraint for analytics is the algorithm used to process the data. Several Hadoop technologies exist, such as AWS Elastic MapReduce and Apache Storm, which allow for specialized processing methods. For streamed data, Apache Storm is suitable due to its ability to process continuous streams of data unlike EMR's required batch processing [11].

Implementation

The end-user implementation of the data processing and storage frameworks is entirely software-based, while the underlying infrastructure for cloud services is entirely hardware-based [1]. Both AWS and Azure offer API endpoints and libraries to manage their respective cloud components, such as virtual machines and databases. The analytics and processing frameworks leverage software development in their respective languages or in libraries for existing languages, such as Python [12], [13].

[1] C. Ji, Y. Li, W. Qiu, U. Awada and K. Li, "Big Data Processing in Cloud Computing Environments," *2012 12th International Symposium on Pervasive Systems, Algorithms and Networks*, San Marcos, TX, 2012, pp. 17-23.

[2] Amazon Web Services, "Big Data on AWS," *Amazon Web Services*. [Online]. Available: https://aws.amazon.com/big-data/. [Accessed: Oct. 21, 2016].

[3] Microsoft, "Azure Products," *Microsoft*. [Online]. Available: https://azure.microsoft.com/en-us/services/. [Accessed: Oct. 21, 2016].

[4] Amazon Web Services, "Amazon EC2 Pricing," *Amazon Web Services*. [Online]. Available: https://aws.amazon.com/ec2/pricing/on-demand/. [Accessed: Oct. 21, 2016].

[5] Microsoft, "Linux Virtual Machines Pricing," *Microsoft*. [Online]. Available: https://azure.microsoft.com/en-us/pricing/details/virtual-machines/linux/. [Accessed: Oct. 21, 2016].

[6] Amazon Web Services, "Amazon DynamoDB Pricing," *Amazon Web Services*. [Online]. Available: https://aws.amazon.com/dynamodb/pricing/. [Accessed: Oct. 21, 2016].

[7] Microsoft, "DocumentDB Pricing," *Microsoft*. [Online]. Available: https://azure.microsoft.com/en-us/pricing/details/documentdb/. [Accessed: Oct. 21, 2016].

[8] Amazon Web Services, *Amazon DynamoDB Developer Guide*, Amazon Web Services, 2012.

[9] Amazon Web Services, "Limits in DynamoDB," *Amazon Web Services*. [Online]. Available: http://docs.aws.amazon.com/amazondynamodb/latest/developerguide/Limits.html#limits-capacity-units-provisioned-throughput. [Accessed: Oct. 21, 2016].

[10] Microsoft, "Performance levels in DocumentDB," *Microsoft*. [Online]. Available: https://azure.microsoft.com/en-us/documentation/articles/documentdb-performance-levels/. [Accessed: Oct. 21, 2016].

[11] Hortonworks, “Processing Streaming Data In Hadoop with Apache Storm,” *Hortonworks*. [Online]. Available: http://hortonworks.com/hadoop-tutorial/processing-streaming-data-near-real-time-apache-storm/. [Accessed: Oct. 21, 2016].

[12] Amazon Web Services, “AWS SDK for Python,” *Amazon Web Services*. [Online]. Available: https://aws.amazon.com/sdk-for-python/. [Accessed: Oct. 21, 2016].

[13] Microsoft, “How to use Service Management from Python,” *Microsoft*. [Online]. Available: https://azure.microsoft.com/en-us/documentation/articles/cloud-services-python-how-to-use-service-management/. [Accessed: Oct. 21, 2016].