

Introduction to Cloud Computing and Examples of Cloud Databases

Sofia Salen

Master's in Data Science student at
Goergen Institute of Data Science at
the University of Rochester
ssalen2@ur.rochester.edu

Jing Wang

Master's in Data Science student at
Goergen Institute of Data Science at
the University of Rochester
Jwang163@ur.rochestere.edu

ABSTRACT

The goal of this paper is to study databases used with the three different types of cloud computing: Infrastructure as a Service (IaaS), Software as a Service (SaaS), and Platform as a Service (PaaS). In particular, this paper will give examples of cloud service models and also discuss cloud databases that are provided by cloud computing companies. Since databases provided by cloud computing platforms are quite different, this paper will look into two cloud databases, Amazon's SimpleDB, provided by Amazon Web Services (an example of IaaS) and Google's Bigtable, provided by Google App Engine (an example of PaaS) in terms of their data model, advantages, and limitations.

1. INTRODUCTION TO CLOUD COMPUTING

Currently, cloud computing is a popular topic regarding manipulating and storing data. "Cloud computing is a way of referring to the use of shared computing resources, and it is an alternative to having local servers handle applications." [1]

With cloud computing, users can customize software and applications that are scalable, secure, and reliable. Cloud computing saves companies money by no longer needing large amounts of people and facilities to run and update applications. There are three types of cloud computing, and this paper will describe each in detail.

As databases are part of almost every computer application, cloud computing providers and platforms like Amazon Cloud Service and Google Cloud Platform offer database services too. Amazon Cloud Service provides support for relational databases including MySQL, Oracle, and SQL Server Instances [9]. In addition, Amazon Cloud Service also provides support for NoSQL databases such as Amazon SimpleDB [9]. Google Cloud Platform also provides support for both relational databases called Google Cloud SQL and NoSQL databases called Google's Bigtable [5]. Since relational databases are more familiar to us, this paper will explore more on NoSQL databases by studying the data models of Amazon SimpleDB and Google's Bigtable.

In this paper, types of cloud computing, examples of cloud based platforms, and cloud databases are analyzed and discussed in detail.

2. TYPES OF CLOUD COMPUTING AND EXAMPLES OF CLOUD SERVICE MODELS

There are three types of cloud computing, and they are Infrastructure-as-a-Service (IaaS), Software-as-a-Service (SaaS), and Platform-as-a-Service (PaaS). In addition to defining the three types of cloud computing, this paper will provide examples of platforms and services to show benefits of using each type of cloud computing.

For a visual representation of the differences between PaaS, IaaS, and SaaS, one can refer to Figure 1. It displays the services that providers manage versus users manage. In particular, for Software as a Service, the provider supplies the client/user with all services including applications, data, runtime, middleware, operating system O/S, virtualization, servers, storage, and networking hardware. The chart shows that for Infrastructure as a Service, all hardware is supplied, but the user needs to manage everything else. Platform as a service provides all of the same services as IaaS, but unlike SaaS it does not supply applications and data. If a company chooses to not use IaaS, PaaS, or SaaS, then they must manage and provide all services on their own which is very expensive and inefficient.

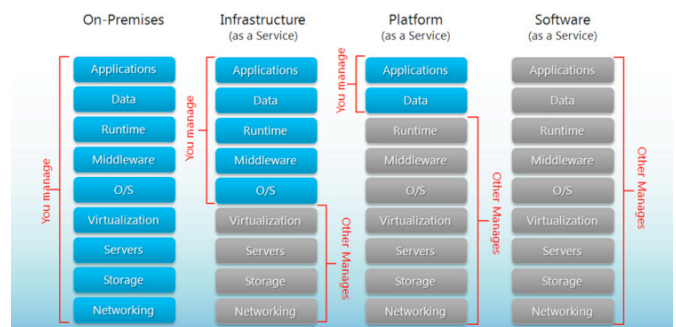


Figure 1. IaaS, PaaS, and SaaS what provider (other) manages vs. user (you) manages [12]

2.1 Infrastructure-as-a-Service and Amazon Web Services

Infrastructure-as-a-Service (IaaS) provides “server, storage, and network hardware” [1] for storing applications, and a platform for running applications. By using IaaS, clients can minimize costs by using resources as needed rather than paying for and maintaining a data center. Amazon Web Services is an example of Infrastructure-as-a-Service, and it provides clients with the resources to build their own scalable applications in the form of a variety of cloud-based products. [1] With Amazon Web Services, companies do not need to maintain or pay for storage, servers, or networking hardware. This enables organizations to move faster and lower IT costs because clients only pay for what they use. [9]

2.2 Software-as-a-Service and Salesforce

Software-as-a-Service (SaaS) is an application based software model offered to clients to use over the internet. SaaS providers lease application software to clients and rent data center resources like servers, storage and networking hardware from IaaS providers. [4] **Salesforce** provides a plethora of products to assist company clients in using data to solve business problems and connect to their customers. As Salesforce is an example of Software-as-a-Service, functionalities of the platform are drag and drop making it easier for companies to innovate. [10]

2.3 Platform-as-a-Service and Microsoft Azure

Platform-as-a-Service (PaaS) offers a combination of SaaS and IaaS, meaning that providers offer hardware and some application software to clients so they can develop and deploy applications over the internet. [1] Since Microsoft Azure is an example of Platform-as-a-Service, it is a comprehensive set of cloud services that developers and IT professionals use to build, deploy, and manage applications through a global network of data-centers. Integrated tools support an organization’s ability to build anything from simple mobile apps to internet-scale solutions. [11]

3. EXAMPLES OF CLOUD DATABASES

A cloud database is the database that typically runs on cloud computing platform. Users can choose from two types of methods to run their database in the cloud [8]. The first method is cloud platforms allow users to install and maintain their own databases for a limited time [8]. That is, users can purchase or maintain a database from a third party and use other services provided by cloud computing platforms. For example, users can use Oracle Database 11g Enterprise Edition provided by Oracle on Microsoft Azure (which is PaaS as explained previously). The second method is cloud platforms are responsible for installing and maintaining the databases and users pay for these parts of services [8]. This method is called Database-as-a-service (DBaaS) [8]. For

This work is licensed under the Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License. To view a copy of this license, visit <http://creativecommons.org/licenses/by-nc-nd/4.0/>. For any use beyond those covered by this license, obtain permission by emailing info@vlldb.org.

example, Amazon Web Services offers three kinds of databases: SimpleDB (NoSQL key-value store), Amazon RDS (relational database supporting MySQL) and DynamoDB. Microsoft offers Azure SQL Database service on Microsoft Azure cloud computing platform. Google offers a NoSQL cloud bigtable that is NoSQL big data database services on Google cloud platform. In this paper, we will look at Amazon’s simpleDB and Google’s bigtable in detail in terms of their data model and compare their advantages and limitations.

3.1 Amazon SimpleDB

3.1.1 Basic Information about Amazon SimpleDB

Amazon simpleDB is a NoSQL distributed database written in Erlang by Amazon.com [9]. It is used as a part of Amazon Web Services (AWS). Developers only need to store and query data through web services and Amazon simpleDB does the rest. In this way, the service allows developers to focus more on their application development and save time by giving the time-consuming database administration job to Amazon simpleDB. Also, though users are not doing database administration jobs on their own machines, Amazon simpleDB provides core database features: speedy, in time query, and it provides high availability, and durability [9].

3.1.2 Data Model

Amazon SimpleDB’s data model makes it easy to store, manage, and query the structured data. When using Amazon SimpleDB, developers organize their dataset in domains within which they can put data, get data, or run queries. Figure 2. shows what Amazon SimpleDB’s domains look like [7]. There are n spreadsheets, each one represents one domain [7]. In each spreadsheet there are one to n items (rows) and one to n attributes (columns). Items contain name:value pairs that are associated with each attribute, meaning that there is a data value where each item attribute pair meets.

	A	B	C	D	E	F	G	H
1		Attribute 1	Attribute 2	Attribute 3	...	Attribute <n>		
2	Item 1	value	value	value	value	value		
3	Item 2	value	value	value	value	value		
4	Item 3	value	value	value	value	value		
5	...	value	value	value	value	value		
6	Item <n>	value	value	value	value	value		
7								
8								
9								
10								
11								
12								

Figure 2. Domains for Amazon SimpleDB platform. [7]

3.1.2.1 Multiple domains with one item example

Amazon’s SimpleDB allows users to easily add, update, and delete domains and attributes [6]. Here is an example of how a user can add a domain using Amazon’s SimpleDB platform [6]. In this case, there are multiple domains and one item per domain. Figure 3. shows two domains each with one item, CustomerID:

123 and 456. To add domains to the table, use the PUT function with attribute name-value pairs for each CustomerID. This is shown below:

PUT (item, 123), (First name, Bob), (Last name, Smith), (Street address, 123 Main St.), (City, Springfield), (State, MO), (Zip, 65801), (Telephone, 222-333-4444)

PUT (item, 456), (First name, James), (Last name, Johnson), (Street address, 456 Front St.), (City, Seattle), (State, WA), (Zip, 98104), (Telephone, 333-444-5555)

CustomerID	First name	Last name	Street address	City	State	Zip	Telephone
123	Bob	Smith	123 Main St	Springfield	MO	65801	222-333-4444
456	James	Johnson	456 Front St	Seattle	WA	98104	333-444-5555

Figure 3. Multiple domains with one item per domain, shows output after completing PUT function. [6]

3.1.2.2 Multiple domains with multiple items example

Now, take a more complex example where we have several domains each with multiple items. After we input data into Figure 2., we compile all spreadsheets into one table with all items of all domains displayed [7]. As shown in Figure 4., each item shows one thing purchased along with its category, subcategory, name, color, size, make, and model. Unlike relational databases, one cell in a domain can have multiple values. For instance, in this example, item_03 lists three values in the Color attribute: blue, yellow, and pink. In addition, Amazon SimpleDB's data model does not require that all domains have the same attributes [7]. Different domains can have completely different attributes. In this example, one spreadsheet contains items from the Clothing Category along with attributes Subcat., Name, Color, and Size, while another spreadsheet contains items from Car Parts Category with attributes Subcat., Name, Make, and Model, and so on. Therefore, Amazon's SimpleDB offers more flexibility than relational databases since multiple values are allowed for each cell. Also, instead of updating tables each time in MySQL, with Amazon's SimpleDB we can easily add, update and delete attributes that apply to certain records in each domain [3].

ID	Category	Subcat.	Name	Color	Size	Make	Model
Item_01	Clothes	Sweater	Cathair Sweater	Siamese	Small, Medium, Large		
Item_02	Clothes	Pants	Designer Jeans	Paisley Acid Wash	30x32, 32x32, 32x34		
Item_03	Clothes	Pants	Sweatpants	Blue, Yellow, Pink	Large		
Item_04	Car Parts	Engine	Turbos			Audi	S4
Item_05	Car Parts	Emissions	O2 Sensor			Audi	S4
Item_06	Motorcycle Parts	Bodywork	Fender Eliminator	Blue		Yamaha	R1
Item_07	Motorcycle Parts, Clothing	Clothing	Leather Pants	Black	Small, Medium, Large		

Figure 4. Populated spreadsheet of all domains with all items [7]

3.1.3 Advantages

Amazon's SimpleDB is inexpensive to run, simple to use, and offers stability in accessing data. With Amazon's SimpleDB, users only pay for resources that they consume. [9] Amazon's SimpleDB is easy to use because it does not need pre-defined schemas, data will be automatically indexed by the system whenever there is a change in attributes [3]. Amazon has data centers around the globe. Thus in case one center is down, clients can always access their data. [5]

3.1.4 Limitations

Amazon's SimpleDB also has some limitations. It cannot deal with large queries and complicated joins, and cannot guarantee data integrity and consistency. Though, SimpleDB covers 80% of all database necessities, it is only significant in certain contexts [3]. It cannot deal with lengthy queries and complicated joins [3]. In addition, since data is dispersed among several data centers, data integrity and consistency is not guaranteed, this will make the user experience bad [3].

3.2 Google's Bigtable

3.2.1 Basic Information

Google's Bigtable is a type of NoSQL database provided by Google App Engine, which is an example of Platform-as-a-Service. Bigtable has been described as "a sparsely populated table that can scale to billions of rows and thousands of columns, allowing you to store terabytes or even petabytes of data." [2] "Many Google projects including web indexing, Google Earth, and Google Finance store data in Bigtable." [2] This database successfully provides a flexible, high-performance solution for all of these Google products based on their different needs [2].

3.2.2 Data Model

The data model of "Bigtable is a sparse, distributed, persistent multidimensional sorted map." [3] Figure 5. shows the Bigtable data model. This includes a map where each value is "an uninterpreted array of bytes" [2] that outputs a string. "The map is indexed by a row key, column families, and a timestamp." [2] "Each row is indexed by a single row key" [2] and "each row key is atomic." [2] A column qualifier is the name you assign to your data values within a family so you can uniquely identify each member of the column family. Timestamp is a way of indexing different versions of the same data. [2]

(row:string, column:string, time:int64) → string

Figure 5. Google Bigtable data model [2]

3.2.2.1 Example of Google Bigtable data model

To further explain the data model, we will give an example that refers to Figure 6. [2]. We want to keep a copy of a large collection of web pages and related information that could be used by many different projects [2]. For this example, the table created by the model in Figure 6. is called Weetable and there are two column families: anchor and contents. In particular, the row key is the reversed URL "com.cnn.www", "contents:" is the column

family, and the cell contents are “<html>...”. The “contents:” column has three versions, at timestamps t_3 , t_5 , and t_6 . The anchor column family has qualifiers “cnnsi.com” and “my.look.ca.” (using syntax: family:qualifier). The “contents” column family contains the main website “<html>...” while the “anchor” column family contains two websites “CNN” and “CNN.com” that take you to the original website “com.cnn.www” [2].

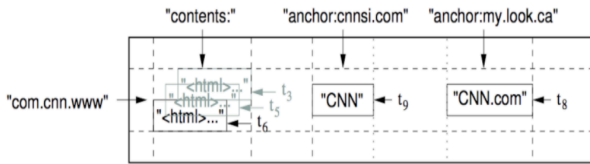


Figure 6. Insert caption to place caption below figure. [2]

3.2.3 Advantages and Limitations

Using Google’s Bigtable has advantages and limitations. Google’s bigtable has a quicker response time than Relational Database Management System (RDBMS) [3]. Conventional approaches such as join and normalization in relational database are not needed here [3]. Administering the database is simple with Bigtable because it is an example of Platform-as-a-Service [5]. Therefore, users do not need to worry about purchasing and maintaining hardware. Instead users only need to design table schemas. However, even if a task needs to be completed in one data center, Bigtable does not allow it because their system uses a large number of nodes to complete any task. Additionally, Bigtable does not have support for ACID transactions as used in RDBMS [3].

4. CONCLUSION

Cloud computing provides users with the opportunity to customize reliable software and applications, and save money by only paying for resources they use. There are three types of cloud computing: Infrastructure-as-a-Service (IaaS), Software-as-a-Service (SaaS), and Platform-as-a-Service (PaaS). For each type of cloud computing there are several cloud based platforms that companies can use to store and manage their data. These include Amazon Web Services (IaaS), Salesforce (SaaS), and Microsoft Azure (PaaS). These could platforms also offer support for relational databases and NoSQL databases. Amazon SimpleDB and Google Bigtable are two NoSQL cloud databases running on Amazon Web Services (IaaS) and Google App Engine (PaaS).

5. REFERENCES

- [1] Bhardwaj, Sushil, Leena Jain, Sandeep Jain. “Cloud Computing: A Study of Infrastructure as a Service (IAAS)”. *International Journal of Engineering and Information Technology*, vol. 2, no. 1, 2010. pp. 60 – 63.
- [2] Chang, Fay, Jeffrey Dean, Sanjay Ghemawat, Wilson C. Hsieh, Deborah A. Wallach Mike Burrows, Tushar Chandra, Andrew Fikes, Robert E. Gruber. “Bigtable: A Distributed Storage System for Structured Data”. *Google Inc.*
- [3] Ramanathan, Shalini, Savita Goel, Subramanian Alagumalai. “Comparison of Cloud database: Amazon’s SimpleDB and Google’s Bigtable.” *Recent Trends in Information Systems (ReTIS), 2011 International Conference on*, 21-23 Dec. 2011.
- [4] Wu, Linlin, Saurabh Kumar Garg, Rajkumar Buyya. “SLA-based Admission Control for Software-as-a-Service provider in Cloud Computing Environments”. *Journal of Computer and System Sciences*, 23 Dec. 2011.
- [5] “Overview of Cloud Bigtable.” *cloud.google.com*. Google Cloud Platform, 8 November 2017 Published. Web. 5 December 2017. Accessed. <<https://cloud.google.com/bigtable/docs/overview>>
- [6] “Amazon SimpleDB Product Details.” *aws.amazon.com/simplydb/details/*. Amazon Web Services. Web. 5 December 2017. Accessed. <<https://aws.amazon.com/simplydb/details/>>
- [7] “Data Model.” *docs.aws.amazon.com/AmazonSimpleDB/latest/DeveloperGuide/DataModel*. Amazon Web Services. Web. 5 December 2017. Accessed. <<http://docs.aws.amazon.com/AmazonSimpleDB/latest/DeveloperGuide/DataModel.html>>
- [8] Wikipedia Contributors. “Cloud database.” *Wikipedia, The Free Encyclopedia*. Wikipedia, The Free Encyclopedia, 4 December 2017. Web. 7 December 2017. Accessed. <https://en.wikipedia.org/wiki/Cloud_database>
- [9] “Amazon SimpleDB.” *aws.amazon.com/simplydb/*. Amazon Web Services. Web. 5 December 2017. Accessed. <<https://aws.amazon.com/simplydb/>>
- [10] “What is cloud computing?.” *Salesforce.com/cloudcomputing/*. Salesforce. Web. 3 December 2017. Accessed. <<https://www.salesforce.com/cloudcomputing/>>
- [11] “What is Azure?.” *Azure.microsoft.com/en-us/overview/what-is-azure*. Microsoft Azure. Web. 6 December 2017. Accessed. <<https://azure.microsoft.com/en-us/overview/what-is-azure/>>
- [12] Stamey, Laura. “IaaS vs PaaS vs. SaaS Cloud Models (Differences & Examples).” *Hostingadvice.com/how-to/iaas-vs-paas-vs-saas/*. HostingAdvice.com. 30 May 2017. Web. 29 November 2017. Accessed. <<http://www.hostingadvice.com/how-to/iaas-vs-paas-vs-saas/>>