Available Online at www.ijcsmc.com

International Journal of Computer Science and Mobile Computing

A Monthly Journal of Computer Science and Information Technology

ISSN 2320–088X

IJCSMC, Vol. 4, Issue. 2, February 2015, pg.73 – 79

RESEARCH ARTICLE

Database in Cloud Computing - Database-as-a Service (DBaas) with its Challenges

Shweta Dinesh Bijwe
Student of Master of Engineering in (CS & IT)
HVPM's college of Engineering and Technology
Amravati, India
Shweb350@gmail.com

Prof. P. L. Ramteke
Associate Professor and HOD in Department of (IT)
HVPM's College of Engineering and Technology
Amravati, India
pl_ramteke@rediffmail.com

Abstract-
The Cloud computing (CC) has been widely recognized as the next generation's computing infrastructure [6]. CC offers many advantages by allowing users to use infrastructure like servers, networks, and storages, platforms containing middleware services and operating systems, and softwares for application programs. Cloud computing is an emerging style of IT delivery that intends to make the Internet the ultimate home of all computing resources. Cloud computing has emerged as a popular model in computing world to support processing large volumetric data using clusters of commodity computers. The delivery of Services by cloud Service Providers in terms of DBase is important as the cloud environment gives access to centralized shared hardware, software and other information. This paper introduces a new transactional "database-as-a-service" (DBaaS) called Database/Relational Cloud. A DBaaS promises to move much of the operational burden of provisioning, configuration, scaling, performance tuning, backup, privacy, and access control from the database users to the service operator, offering lower overall costs to users. Database as a service has several major issues and concerns, such as data Scalability, Elasticity, Availability, security, expectations, etc, issues. Proposed solutions include risk management, better contractual agreements, database encryption, and authenticity techniques. By bettering these situation the DBaas service in cloud computing is effective to manage today's vast growing datasets.

Keywords- Cloud computing, database-as-a-service (DBaaS), *Service Level Agreement (SLA)*, *Scalability*, Autonomy, multi-tenancy, *Privacy*.

I. Introduction

Cloud computing is an emerging style in the field of IT that intends to make the Internet the ultimate home of all computing resources- storage, computations, and accessibility. It offers on-demand use of third-party infrastructures on the basis of pay-per-use. In Cloud computing multimedia database is based on the current of database development, object-oriented technology and object-oriented fields in the database, which increasing display and its vitality [1]. This paradigm reduces customers' need for hardware while improving the elasticity of computational resources, by which they allows to adapt to business requirements. Because of its robustness, scalability, performance, high availability, least cost and many others, businesses are finding it attractive to adopt the cloud computing paradigm.

Cloud computing offers software developers and vendors the promise of efficient resource utilization and on-demand scalability with minimal capital investment. For consumers, it offers infrastructure-free computing, where users access their "desktop" and data from any location—work, home, on the road, or from within other organizations. For enterprise businesses, cloud computing offers the potential to outsource computing infrastructure to focus on core competencies with higher efficiencies which is much beneficial to them. Cloud computing is primarily based on virtualization, which enables multitenancy and on-demand use of scalable shared resources by all tenants [2].

It is very advantageous to use cloud services for business person, to customer, businesses can eliminate the need for the professionals who maintain and support the underlying complexities for some of the most desirable new IT technologies, such as highly scalable, variably provisioned systems. An obvious benefit to customers is that computing resources, such as virtual servers, data storage, and network capabilities, are all load balancing and automatically expandable. Resources are allocated as needed, and loads can be transferred automatically to better locations, producing a robust, reliable service [4]. The idea is simple enough. A large company with many computing resources, such as large datacenters, reaches an agreement with customers. Customers can run their programs, store their data, and host virtual machines, and so on, using the provider's resources. Customers can terminate their contract, avoid startup and maintenance costs, and benefit from the provider's ability to dynamically allocate their resources. Cloud computing service provides typically following resource types, which create a more common classification of cloud types: infrastructure, platform, or software. In this paper we are mainly focusing on Daas service type.

DaaS is a prime example of SaaS. [5] *SaaS i.e. Software as a service* provides customers with a particular piece of software. The provider runs the software and provides Internet access to it, but the customer feeds it data and instructions. The service provider picks the database management software and installs, runs, and manages it.

In any case, it's still just a software application provided to the customer. For economic reasons, DaaS is particularly well suited to many small- to medium-sized businesses that rely on databases but find their installation and maintenance costs restrictive. Maintaining a database requires trained professionals, so the service is even more valuable because customers don't have to hire, train, and pay them.

This paper introduces a new transactional "database-as-a-service" (DBaaS) called Database/Relational Cloud. A DBaaS promises to move much of the operational burden of provisioning, configuration, scaling, performance tuning, backup, privacy, and access control from the database users to the service operator, offering lower overall costs to users. Early DBaaS efforts include Amazon RDS and Microsoft SQL Azure, which are promising in terms of establishing the market need for such a service. In the cloud market, dozens of companies offer secure storage, and many have added functionality to the storage and called it database services. This is common because this setup doesn't face the same issues as a scalable, secure, relational database run in the cloud [3]. Nor is it as attractive. In fact, few cloud concepts face as many problems, which is why we focus on DaaS; if it's usable, other cloud services should be as well.

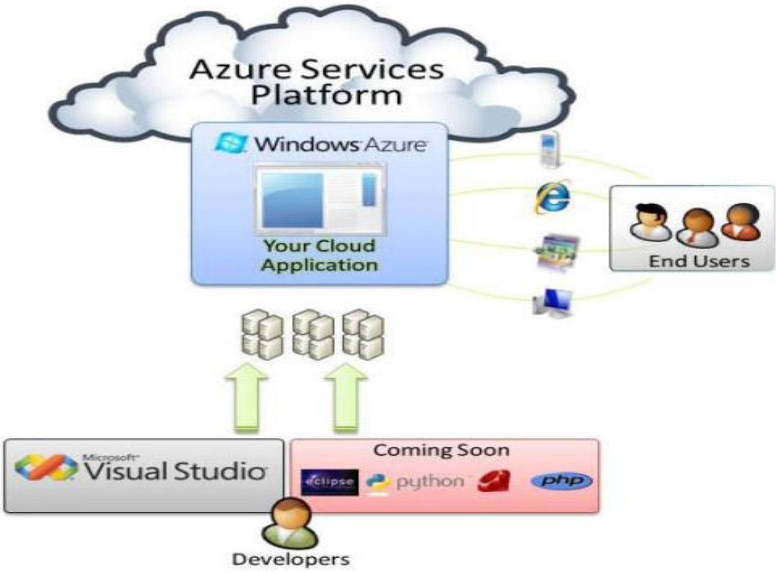

Fig. Example of Daas servise provided by Microsoft Azure [9]

II. Database-as-a-Service in Cloud

Database-as-a-Service (DBaaS) is a service that is managed by a cloud operator (public or private) that supports applications, without the application team assuming responsibility for traditional database administration functions. With a DBaaS, the application developers should not need to be database experts, nor should they have to hire a database administrator (DBA) to maintain the database [7]. True DBaaS will be achieved when application developers can simply call a database service and it works without even having to consider the database. This would mean that the database would seamlessly scale and it would be maintained, upgraded, backed-up and handle server failure, all without impacting the developer. Database as a service (DaaS) is a prime example of a service that's both exciting and full of difficult security issues.

Cloud providers want to offer the DBaaS service described above. In order to provide a complete DBaaS solution across large numbers of customers, the cloud providers need a high-degree of automation. Unction's that have a regular time-based interval, like backups, can be scheduled and batched. Many other functions, such as elastic scale-out can be automated based on certain business rules. For example, providing a certain quality of service (QoS) according to the service level agreement (SLA) might require limiting databases to a certain number of connections or a peak level of CPU utilization, or some other criteria. When this criterion is exceeded, the DBaaS might automatically add a new database instance to share the load. The cloud provider also needs the ability to automate the creation and configuration of database instances [8]. Much of the database administration process can be automated in this fashion, but in order to achieve this level of automation, the database management system underlying the DBaaS must expose these functions via an application programming interface.

Cloud operators must have to work on hundreds, thousands or even tens of thousands of databases at the same time. This requires automation. In order to automate these functions in a flexible manner, the DBaaS solution must provide an API to the cloud operator [6]. The ultimate goal of a DBaaS is that the customer doesn't have to think about the database. Today, cloud users don't have to think about server instances, storage and networking, they just work. Virtualization enables clouds to provide these services to customers while automating much of the traditional pain of buying, installing, configuring and managing these capabilities. Now database virtualization is doing the same thing for the cloud database and it is being provided as Database as a Service (DBaaS). The DBaaS can substantially reduce operational costs and perform well. It's important as well as simple thing that the goal of DaaS is to make things easier.
Cloud Control Database as a Service (DBaaS) provides:

- A shared, consolidated platform on which to provision database services
- A self-service model for provisioning those resources
- Elasticity to scale out and scale back database resources
- Chargeback based on database usage

The aggressive consolidation of information technology (IT) infrastructure and deployment of Database as a Service (DBaaS) on public or private clouds is a strategy that many enterprises are pursuing to accomplish these objectives. Both initiatives have substantial implications when designing and implementing architectures for high availability and data protection. Database consolidation and DBaaS also drive standardization of I.T. infrastructure and processes. Standardization is essential for reducing cost and operational complexity. Databases deployed in the Bronze tier include development and test databases and databases supporting smaller work group and departmental applications that are often the first candidates for database consolidation and for deployment as Database as a Service (DBaaS).

III. Some Challenges to Database-as-a-Service for the Cloud

The key ingredients to this success are due to many features DBMSs offer: overall functionality, consistency, performance, and reliability. In spite of this success, during the past decade Database Scalability, Elasticity, Availability and Autonomy in the Cloud there has been a growing concern that DBMSs and RDBMSs are not *cloud-friendly*. This is because, unlike other technology components for cloud service such as the webservers and application servers, which can easily scale from a few machines to hundreds or even thousands of machines, DBMSs cannot be scaled very easily. There are three challenges that drive the design of Relational Cloud: efficient multi-tenancy to minimize the hardware footprint required for a given (or predicted) workload, elastic scale-out to handle growing workloads, and database privacy.

In fact, past DBMS technology fails to provide adequate tools and guidance if an existing database deployment needs to scale-out from a few machines to a large number of machines [10]. Cloud computing and the notion of large-scale data-centers will become a pervasive technology in the coming years. There are some technology hurdles that we confront in deploying applications on cloud computing infrastructures: DBMS scalability and DBMS security. In this paper, we will focus on the problem of making DBMS technology cloud-friendly. In fact, we will argue that the success of cloud computing is critically contingent on making DBMSs scalable, elastic, available, secure and autonomic, which is in addition to the other well-known properties of database management technologies like high-level functionality, consistency, performance, and reliability. In this section we are addressing some of the issues related to DBaas in cloud as follows:

A. Database Scalability in the Cloud

Scalability is a desirable property of a system, which indicates its ability to either handle growing amounts of work in a graceful manner or its ability to improve throughput when additional resources (typically hardware) are added. In the context of cloud-computing paradigms notion of scalability, there are two options for scaling the data management layer. The requirement of making web-based applications *scalable* in cloud-computing platforms arises primarily to support virtually unlimited number of end-users. Another challenge in the cloud that is closely tied to the issue of scalability is to develop mechanism to respond to sudden load fluctuations on an application or a service due to demand surges or troughs from the end-users. Scalability of a system only provides us a guarantee that a system can be scaled up from a few machines to a larger number of machines. In cloud computing environments, we need to support additional property that such scalability can be provisioned dynamically without causing any interruption in the service.

The first option is to start with key-value stores, which have almost limitless scalability, and explore ways in which such systems can be enriched to provide higher-level database functionality especially when it comes to providing transactional access to multiple data and informational entities. The other option is to start with a conventional DBMS architecture and leverage from key-value store architectural design features to make the DBMS highly scalable.

A good DBaaS must support database and workloads of different sizes. The challenge arise when a database workload exceeds the capacity of a single machine. A DBaaS must therefore support scale-out, where the responsibility for query processing is partitioned amongst multiple nodes to achieve higher throughput. But what is the best way to partition databases for scale-out? The answer depends on the way in which transactions and data items relate to one another. In Relational Cloud, we use a recently developed workload-aware partitioner
[5], which uses graph partitioning to automatically analyze complex query workloads and map data items to nodes to minimize the number of multi-node transactions/statements which incurs significant overhead, and are the main limiting factor to linear scalability in practice.

B. Database Elasticity in the Cloud

This type of dynamic provisioning where a system can be scaled-up dynamically by adding more nodes or can be scaled-down by removing nodes is referred to as *elasticity*. One of the major factors for the success of the cloud as an IT infrastructure is its *pay-per-use* pricing model and *elasticity*. For a DBMS deployed on a pay-per-use cloud infrastructure, an added goal is to optimize the system's operating cost. *Elasticity*, i.e. the ability to deal with load variations by adding more resources during high load or consolidating the tenants to fewer nodes when the load decreases, all in a live system without service disruption, is therefore critical for these systems.

Even though elasticity is often associated with the scale of the system, but Scalability is a static property of the system that specifies its behavior on a static configuration [13]. Elasticity is a desirable and important property of large scale systems. For a system deployed on a pay-per-use cloud service, such as the Infrastructure as a Service (IaaS) abstraction, elasticity is critical to minimize operating cost while ensuring good performance during high loads. It allows consolidation of the system to consume less resources and thus minimize the operating cost during periods of low load while allowing it to dynamically scale up its size as the load decreases. One must also consider the impact of powering down on availability. For instance, consolidating the system to a set of servers all within a single point of failure as in a switch or a power supply unit. That can result in an entire service outage resulting from a single failure. Furthermore, bringing up powered down servers is more expensive, so the penalty for a miss-predicted power down operation is higher.

C. DBaaS Availability

It is not necessary that all services require the same level of availability which is why DBaaS service definitions include quality of service characteristics. Specifying what levels of availability are needed will enable the provisioning, monitoring and management processes to ensure that instances are deployed on the right platforms and in the correct configuration. The delivery of high availability database services developed with the Exadata Database Machine and the Oracle Database12c software can be achieved through a variety of deployment options utilizing existing and long tested best practices. Exadata [11] and Oracle RAC Cluster ware allow for the deployment of database instances with multiple processing instances spanning multiple servers within and across Exadata frames over a high-speed, low-latency InfiniBand network.

Oracle Automatic Storage Management in conjunction with Exadata Storage Software provides storage cell level high-availability through multiple levels of disk redundancy allowing for the development of tiered storage based upon level and cost of availability. Oracle Exadata, Oracle Database, and Oracle Grid Control work in conjunction to provide the deployment of frame level service availability through the implementation of various stand-by configurations. The combination of Oracle RAC HA deployments utilizing Oracle Active Data Guard capabilities allows for the implementation of database instance, server, and storage high availability configurations supporting frame and site level failover capabilities. The DBaaS Deployment Platform is of course only one of the sets of capabilities to be defined within the physical architecture. Any physical architecture must account for all of the technology and technology services supporting the operations, management, service delivery, service accounting and service development processes as well.

D. Database Autonomy and Efficient multi-tenancy in the Cloud

Managing large systems poses significant challenges in monitoring, management, and system operation. Moreover, to reduce the operating cost, considerable autonomy is needed in the administration of such systems. In the context of database systems, the responsibilities of this autonomic controller include monitoring the behavior and performance of the system, Scalability, elasticity and load balancing based on dynamic usage patterns, modeling behavior to forecast workload changes and take pro-active measures to handle such changes. An autonomous and intelligent system controller is essential to properly manage such large systems.

As the set of databases and workloads are growing, there should be some kind of mechanisms to serve them from a given set of machines. The goal is to minimize the number of machines required, while meeting application-level query performance goals. To achieve this, our system must understand the resource requirements of individual workloads, how they combine when co-located on one machine, and how to take advantage of the temporal variations of each workload to maximize hardware utilization while avoiding over commitment. One of the approach uses a single database server on each machine, which hosts multiple logical databases [12]. Relational Cloud periodically determines which databases should be placed on which machines using a novel non-linear optimization formulation, By combining with a cost model that estimates the combined resource utilization of multiple databases running on a machine.

Modeling the behavior of a database system and performance tuning has been an active area of research over the last couple of decades. A large body of work focuses on tuning the appropriate parameters for optimizing database performance, primarily in the context of a single database server. Another line of work has focused on resource prediction, provisioning, and placement in large distributed systems. To enable autonomy in a cloud database, an intelligent system controller must also consider various additional aspects, specifically in the case when the database system is deployed on a Daas or as pay-per-use cloud infrastructure while serving multiple application tenant instances. In such a multitenant system, each tenant pays for the service provided and different tenants in the system can have competing goals. On the other hand, the service provider must share resources amongst the tenants, wherever possible, to minimize the operating cost to maximize profits. A controller for such a system must be able to model the dynamic characteristics and resource requirements of the different application tenants to allow elastic scaling while ensuring good tenant performance and ensuring that the tenants' service level agreements (SLAs) are met. The current work uses a combination of machine learning Techniques to classify tenant behavior followed by tenant placement algorithms to determine optimal tenant co-location and consolidation.

E. DBaas Privacy

Cloud computing poses privacy concerns because the service provider can access the data that is on the cloud at any time. It could accidentally or deliberately alter or even delete information. Many cloud providers can share information with third parties if necessary for purposes of law and order even without a warrant. That is permitted in their privacy policies which users have to agree to before they start using cloud services [15]. Solutions to privacy include policy and legislation as well as end users' choices for how data is stored. Users can encrypt data that is processed or stored within the cloud to prevent unauthorized access.

A significant barrier to deploying databases in the cloud is the perceived lack of privacy, which in turn reduces the degree of trust users are willing to place in the system. If clients were to encrypt all the data stored in the DBaaS, [14] then the privacy concerns would largely be eliminated. In Relational Cloud, one technique is developed i.e. CryptDB, a set of techniques designed to provide privacy. Database administrators can continue to manage and tune the databases, and users are guaranteed data privacy. The key notion is that of adjustable security: CryptDB employs different encryption levels for different types of data, based on the types of queries that users run. Queries are evaluated on the encrypted data, and sent back to the client for final decryption; no query processing runs on the client.

A unifying theme in our approach to these three big challenges is workload-awareness. Our main design principle is to monitor the actual query patterns and data accesses, and then employ mechanisms that use these observations to perform various optimization and security functions.

IV. Conclusion

In this study, we see that the Cloud computing has been widely recognized as the widely growing computing infrastructure. CC offers many advantages by allowing users to use infrastructure like servers, networks, and data storages, without impacting to the owner's organization. In this paper we are introducing DataBase-as-a-Service (DBaas) which promises to move much of the operational burden of provisioning, configuration, scaling, performance tuning, backup, privacy, of the user's data. Database as a Service architectures offer organizations new and unique ways to offer, use and manage database services. The fundamental differences related to service-orientation and discrete consumer-provider roles challenge conventional models yet offer the potential for significant cost savings, improved service levels and greater leverage of information across the business. As discussed in this paper, there are a variety of issues, considerations that are must to understand for effectively using DBaaS in all organizations. We introduced Relational/Database Cloud, a scalable relational databases-as-a-service for cloud computing environments. Database systems deployed on a cloud computing infrastructure face many new challenges such as dealing with large scalability of operations, elasticity, and autonomic control to minimize the operating cost, Continuous availability, Datasets privacy. These challenges are in addition to making the systems fault-tolerant and highly available**.** Relational Cloud overcomes three significant challenges: efficient multi-tenancy, elastic scalability, and database privacy

References

[1] Lee, D. Patterson, A. Rabkin, I. Stoica, and M.Zaharia (2009, Feb. 10). Above the clouds: A Berkeley view of cloud computing. EECS Dept. University of California, Berkeley, No.UCB/EECS-2009-28[Online].Available:
http://radlab.cs.berkeley.edu/

[2] H. Takabi, J.B.D. Joshi, and G-J. Ahn, "Security and Privacy Challenges in Cloud Computing Environments," IEEE Security & Privacy, vol. 8, no. 6, 2010.

[3] people.cs.uchicago.edu/~aelmore/papers/dasfaa.pdf

[4] M. Armbrust et al., “A View of Cloud Computing,” Comm. ACM, vol. 53, no. 4, 2010, pp. 50–58.

[5] E. Ferrari, “Database as a Service: Challenges and So- lutions for Privacy and Security,” IEEE Asia-Pacifi c Services Computing Conf. (APSCC 09), IEEE CS Press, 2009, pp. 46–51.

[6] http://en.wikipedia.org/wiki/Cloud Computing

[7] http://www.scaledb.com/dbaas-database-as-a-service.php

[8] http://en.wikipedia.org/wiki/Cloud_database

[9] www.jiahenglu.net/course/advancedDataManagement/slides/Lec01.ppt

[10]“Database Scalability, Elasticity, and Autonomy in the Cloud” Divyakant Agrawal, Amr El Abbadi, Sudipto Das, and Aaron J. Elmore Department of Computer Science University of California at Santa Barbara Santa Barbara, CA 93106, USA
http://www.cs.ucsb.edu/˜dsl

[11] www.oracle.com/technetwork/topics/entarch/oracle-wp-cloud-ref-arch-1883533.pdf “An Oracle White Paper/November 2012 Oracle Enterprise Transformation Solutions Series
Cloud Reference Architecture”

[12] www.oracle.com/technetwork/topics/entarch/oes-refarch-dbaas-508111.pdf

[13] people.csail.mit.edu/nickolai/papers/curino-relcloud-cidr.pdf (Relational Cloud: A Database-as-a-Service for the Cloud)

[14] Enhancing Data Integrity and Privacy in the Cloud: An Agenda, IEEE 2013.
David S.L. Wei, Fordham University, San Murugesan, BRITE Professional Services, Australia
Sy-Yen Kuo, National Taiwan University, Kshirasagar Naik, University of Waterloo, Canada.

[15] Securing Database as a Service Issues and Compromises; IEEE conference on, 2012.