Micro Service Based System for Cost Based Selection of Cloud Provider Services

Nupura Torvekar, Pravin S. Game

Abstract: ***The increase in the amount of data generated on a daily basis coupled with the need to store and manage this data has encouraged the organizations to adopt cloud computing. In order to ensure better availability and reliability of their data as well as resources, most of the organizations make use of one or more cloud service providers .But the use of cloud resources puts forth some challenges as well. One of the challenges is its detailed monitoring. As the number of services utilized by the cloud consumers goes on increasing, the number of logs and metrics generated by them also scales rapidly.The dynamic nature of cloud infrastructure and the variety of services offered by several cloud vendors demands a sophisticated mechanism to calculate and analyze the cost of using different services. The billing reports by the cloud service providers deliver statistics about the usage of resources and the costs associated with them. It contains large amount of data which needs to be processed in order to gain useful information. In this paper, we propose a micro service based architectural framework which gathers the data from two different cloud service providers. This data is not only stored but processed to generate reports to enable optimal use of cloud infrastructure. The use of microservices framework provides benefits and is a preferred framework for the development of cloud applications. The main aim of this work is to provide an integrated mechanism to enable the comparison of cost for using similar cloud services.***

Keywords**: Application Programming Interface(API), cloud computing, cloud service providers, microservices**

I. INTRODUCTION

Cloud computing involves the provision of computing resources such as networks, servers, storage and applications to enable on-demand network access. It also offers the pay-as-you-go facility which means that the user must pay only for the resources used. The cloud services can be deployed by three different ways such as a public cloud, a private cloud or a hybrid cloud [1]. The challenge in addressing the big data combined with the need for massive computational infrastructure to perform large scale and complex computing has led to the growth and popularity of cloud computing.

The most important aspect of cloud computing is its pay per use model so that the consumers are billed according to their usage. But the metering and billing for different resources is immensely heterogeneous. For example, most of the IaaS (Infrastructure as a Service) resources may consist of reserved servers of clustered server resources and the metering may comprise volume based metering or on-demand models based on per-hour usage. The billing on the other hand would be on a per-month basis where the consumer may be charged for its metered usage over the whole month. The cost for PaaS (Platform as a Service) comprises the amount of network bandwidth(incoming as well as outgoing), stored data, CPU time per hour and other similar metrics. Similarly, the billing for SaaS(Software as a Service) consists of the subscription fees for monthly usage and per-user fees for a particular monthly cycle. Some other aspects may also include the billing models for database servers as well as the metering and billing of monitoring infrastructures [2].

The cloud consumers making use of the cloud provider services are charged according to their usage based on their respective pricing models and the details are provided in the form of detailed billing records. The billing mechanism adopted by different cloud service providers is different and hence, cloud consumers face challenges in realizing their cloud spend apart from understanding the difference in pricing models for different varieties of the same resource. Some of the useful insights about cloud billing systems are presented in [3] which focus on three different cloud service providers. The granularity of compute time billing, network billing and storage billing is studied by the authors and the respective results obtained are explained in detail. It also mentions how billing model differs from one cloud provider to other. The complex nature of cloud billing makes it challenging for the users to optimize their cloud costs.

One more important aspect is that, the detailed billing information is usually provided as aggregates for monthly cycle and the corresponding resource usage. It therefore puts forth a challenge for the consumers to determine the amount of usage of particular resource, the time interval for which it is used as well as the specific user account associated with it [4]. Moreover, some of the organizations even deploy their applications on multiple cloud environments and would therefore require techniques to not just integrate but also compare the cost of utilization for similar set of services.

In order to facilitate the cloud consumers to integrate their applications with the cloud provider services, the cloud vendors provide a set of API (Application Programming Interfaces) [5][6].

Revised Manuscript Received on January 15, 2020

Nupura Torvekar, Pursuing Master of Engineering (Computer Engineering) from Pune Institute of Computer Technology, Pune.

Pravin S. Game, Department Computer Engineering, Pune Institute of Computer Technology, Pune.

APIs allow the communication between two different web-based programs or entities and consist of a set of programming instructions. A combination of the API, programming tools, and documentation is provided by the companies in the form of a software development kit (SDK) [7]. APIs provide programmatic access and insights about the billing and usage of cloud resources and hence, are a crucial aspect in managing and understanding cloud costs. The proposed system makes use of REST based APIs from the cloud providers to access the billing information from the cloud providers which aids in the analysis of cost and usage management of cloud infrastructure.

The microservices architectural framework forms the backbone of the proposed system. Micro service architecture consists of small independent set of services and each service defines an individual business capability. As the microservices architecture consists of a group of services communicating together, microservices can be designed, implemented and deployed using various strategies [8]. All these strategies have been studied by researchers and they are presented in the form of taxonomies or systematic mapping studies.

A proper understanding of the microservices framework can be achieved by studying the taxonomy provided in [9]. Various aspects including the design approaches, technological stack and interfaces for implementation, the deployment platforms, the security, reliability and availability concerns along with the organizational aspects are presented in this work in the form of taxonomy. The detailed view of the microservices lifecycle aspects helps to compare and choose appropriate tools for designing micro services.

The proposed system is used to design a billing microservice which helps in the cost optimization of cloud infrastructure by referring the visual and analytical reports.

This paper is organized as follows: Section II describes the work related to cloud infrastructure management. Section III explains the goals and objectives. Section IV explains the key components in the proposed system architecture along with the detailed procedure. Section V describes the steps and the analysis carried out for designing the billing microservice. Section VI represents and explains the visual reports generated by the proposed system. The final conclusion is discussed in Section VII.

II. REVIEW OF LITERATURE

The existing work by various researchers includes the study of the cloud billing models, cloud cost tracing frameworks as well as metering for cloud services along with the evaluation of its pricing models.

The design and architecture for the rating and billing solution for cloud service providers is presented in [10]. A microservice approach is used to design the solution along with a rule based engine which enables dynamic rate generation for the cloud services that are utilized. A mathematical model for this CYCLOPS platform is presented and it helps to validate the architecture. The CYCLOPS architecture is a collection of five different microservices out of which three of them are responsible for the prime functionality of rating, charging and billing. The dynamic rating can be determined by user-driven rules given to a rule engine and the corresponding model configured for different cloud providers. The final charge is calculated by combining the resource rate along with the usage records and presented in the form of charge data records (CDR).

A cost tracing system named Costradamus is proposed in [4], which enables cloud based software service users to gain an insight into the cost and usage scenario per API operation for the consumed service. Three different tracing approaches namely log import, response recording and modeling are used for evaluating the capacity usage of four infrastructure services. The main advantage of the proposed system is that is provides detailed information about the cost for individual API operations. The proposed system explores the performance and deployment cost tradeoffs for the smart grid application management context.

A billing model based on smart metering system for the cloud services is presented in [11]. The model is developed such that the charge varies based upon the load of the cloud infrastructure. It is important to map the relation between the load conditions for the cloud at a specific time with that of the pricing structure. It uses the auto regressive integrated moving average (ARIMA) model in order to predict the load for the next time period. The prediction is carried out based on the historic data obtained through the monitoring framework. As there is a mapping between the infrastructure load and the price, the model also predicts the price for the next interval of usage. The final billing for the consumer is calculated based on the predicted price and the amount of service utilization which is similar to the smart metering context.

A novel definition of a cloud supply chain for cloud consumers as well as cloud providers is introduced in [12]. The cloud supply chain is composed of the interconnection of network for provisioning of resources to the consumers. It includes the components such as actors, services or the products, the information flow and the accounting and billing details. The main aspect of its execution is the management and coordination of services to enable proper provisioning, monitoring and management of the various accounting and billing processes. The service providers and the end customers are the main actors of the supply chain. Monitoring the data flow is a crucial part in the supply chain and the key aspects of monitoring along with the usage of dynamic infrastructures is also explained by the authors. The authors also explore the relationship between accounting and billing along with its implications on the information flow and data management.

The analysis of the cloud monitoring subsystem is carried out by the authors in [13]. The process of collecting the monitoring data for OpenStack is studied in this work. This open source cloud solution consists of a module called Ceilometer which provides the necessary infrastructure to collect the measurements. The main components which include the agents and the services and their collaboration in order to monitor the instances are described in detail.

Some of the problems with cloud monitoring comprise of the issues faced in constant polling frequency, global polling frequency as well as lack of policy based monitoring. In order to overcome these issues, the authors present a policy based monitoring methodology encompassing the strategies such as metric profiling as well as tiered polling frequency. The alternative solutions proposed by the authors were evaluated using R and Python and suggest that the monitoring data size can be significantly reduced due to these techniques.

The proposed system with the help of microservices architectural framework tries to analyze the data from a cost management perspective such that the available data is filtered from a large amount of data to present the cost for three different categories of cloud services.

III. GOALS AND OBJECTIVES

- To collect the information from cloud service providers and present it in the form of visual statistics for better understanding of cost and usage.
- To design an integrated framework to monitor the cost of resources for three different categories compute, storage and management tools with the intent of relative comparison.

IV. SYSTEM ARCHITECTURE

The aim of the proposed system is to design a billing microservice for fetching, filtering, processing and eventually storing the billing data of the cloud services into a data store. The architectural design of the proposed system is described in fig. 1. The proposed system is useful for organizations using the cloud resources from Amazon Web Services and Microsoft Azure.

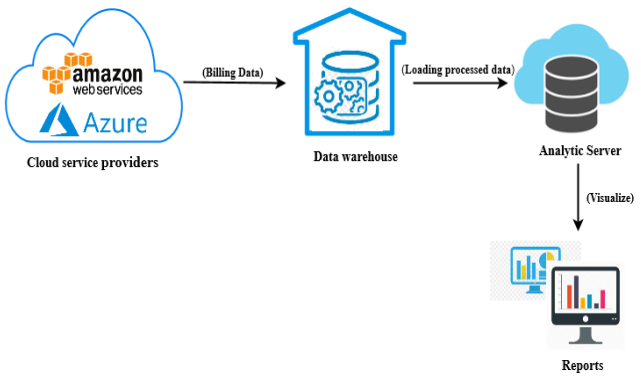

Fig.1. Proposed System Architecture

The key components in the architecture are as follows:

A. Cloud Services

Any kind of resource including compute power, storage, networking, database and so on which is provisioned through the internet by a cloud vendor can be termed as a cloud service.

B. Data Warehouse

The data warehouse acts as the centralized pool for the organization data. It can also be used to identify and analyze the patterns in the data. The data is secured in the data warehouse and is based on the analytics infrastructure platform.

C. Analytic Server

It is a cloud based platform for data storage and analysis.

D. Reports

The graphical representation of the processed and analyzed data to be displayed for comparative purpose can be done using these reports.

The proposed system involves various steps which need to be followed in order to generate the final results. The various steps involved in this procedure can be explained as follows:

1. Authentication and Authorization for Microservice

The distributed nature of microservices requires sophisticated access control mechanisms to ensure its security. Microservices are authenticated to verify their identity against an internal or external system. After its identity has been verified, the next step is to control access to verified users. Authorization mechanisms are used to decide what type of data and service can be accessed by the user. The authorization technique used here is that of a token which provides unique identity for verified users.

2. Data Fetching

After the microservice has been authenticated, the next step is data gathering. Data can be gathered from AWS with the help of APIs like Amazon CloudWatch as well as AWS cost and usage reports.Data from Azure can be gathered with the help of APIs to get the Billing Report files. Billing report files are crucial as they provide detailed information about the type of product used, the account from which the usage has occurred, the cost that is incurred, and also the measure used to specify the price. This data is provided in the form of csv files and help in getting useful information.

3. Data Loading

The next step involves loading the data into the analytical server or the storage system. The data that is fetched from the billing records of the two cloud providers is filtered and relevant information is loaded. This data is loaded into the tables in the server. The data tables are updated as new report files are generated at regular intervals.

4. Report Generation

The final and the major step is report generation. The data tables loaded into the data store are used to generate these reports. A comparison report showing the total cost incurred for compute, storage and monitoring tools for AWS and Azure is provided by the system.

V. SYSTEM ANALYSIS

The billing microservice designed in the proposed system comprises the following major tasks:

- Fetching the billing details from AWS and Azure
- Filtering the data to extract the relevant attributes
- Processing the data to convert it into a standard format
- Loading the data into a persistent store
- Generation of analytical reports for visualizing the relevant information.

The proposed system gets the data for two cloud service providers: Amazon Web Services and Microsoft Azure. The cost and usage report files for the consumed resources for a particular time period are used for the analysis purpose. The cost and usage files are in the csv format and consist of the detailed record of the type of usage, the product used, time period for which it is being utilized, the corresponding accounts using a resource and all other related information.

The billing files may contain some additional attributes which may not be useful for analysis and these are filtered out to maintain the list of relevant attributes. The raw data fetched with the help of the APIs needs to be transformed by converting it into a standard format before it is loaded into the database.

The data is loaded into a persistent storage which enables the addition of new data into the system and can facilitate consolidated analysis of the data over a specific time period.

The consolidated data is then used to generate analytical reports which may prove to be useful for better understanding of the cloud resources.

The cost and usage report files consist of resource usage details pertaining to all the utilized services for a particular organization. The proposed system takes into consideration three categories compute, storage and management tools for Amazon Web Services as well as Microsoft Azure [14] [15]. The aggregated cost of utilizing these services is provided to the user.

VI. RESULTS AND DISCUSSION

The billing microservice gathers the data and performs the necessary operations and analysis and presents the results on the dashboard.

A dashboard displaying the total cost of utilizing the compute, storage and monitoring tools can be seen is presented. The screenshot for the same can be seen in Fig. 2. It not only provides an overview of the consolidated expenditure, but helps the user in comparing the relative charges for both the cloud providers. The most important aspect to be noted here is that the total cost and the results may vary depending upon the number of instances that are hosted on the cloud and the type of service that is chosen.

The total compute cost for AWS comprises of the products such as amazon EC2, Lambda, Container registry and container service. The total compute cost for Azure comprises of the cost of virtual machines, container instances as well as container registry.

The total storage cost for AWS consists of the cost for products including Simple Storage Service, Elastic File system, storage gateway. Similarly, the Azure cost consists of the cost for block storage.

The cost of AWS monitoring comprises the cost incurred by Amazon CloudWatch service. Similarly, the cost for Azure Monitor is taken into consideration for Azure.

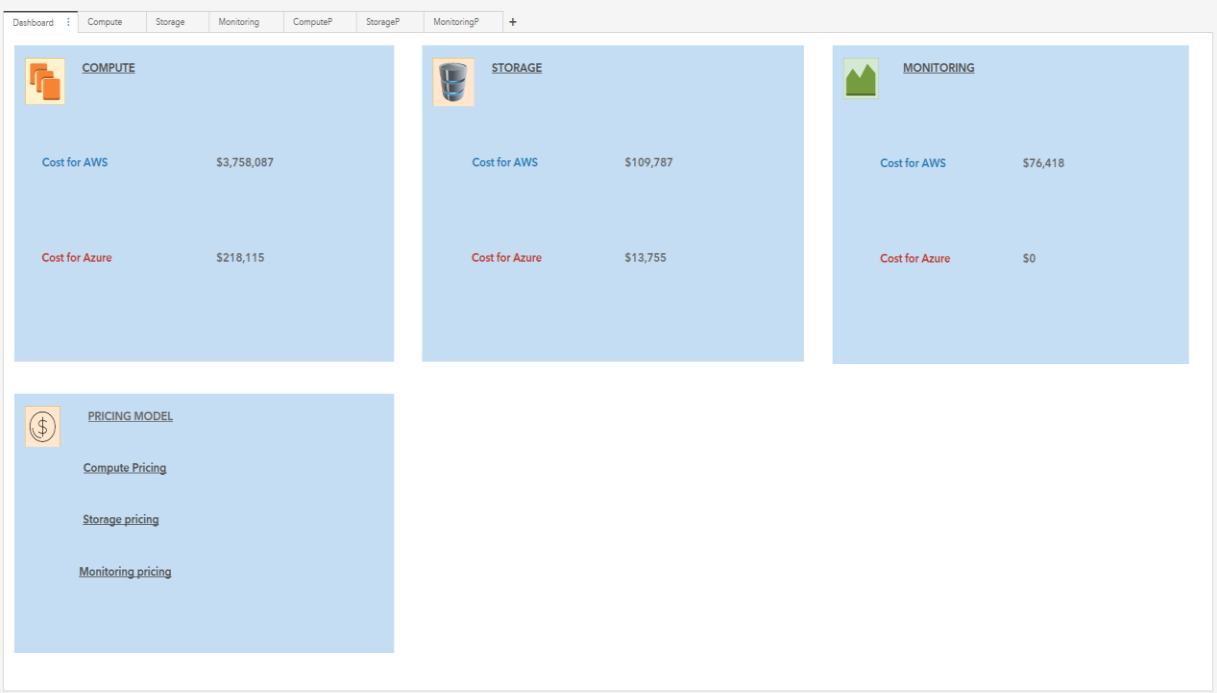

Fig.2. Dashboard for selecting the type of service

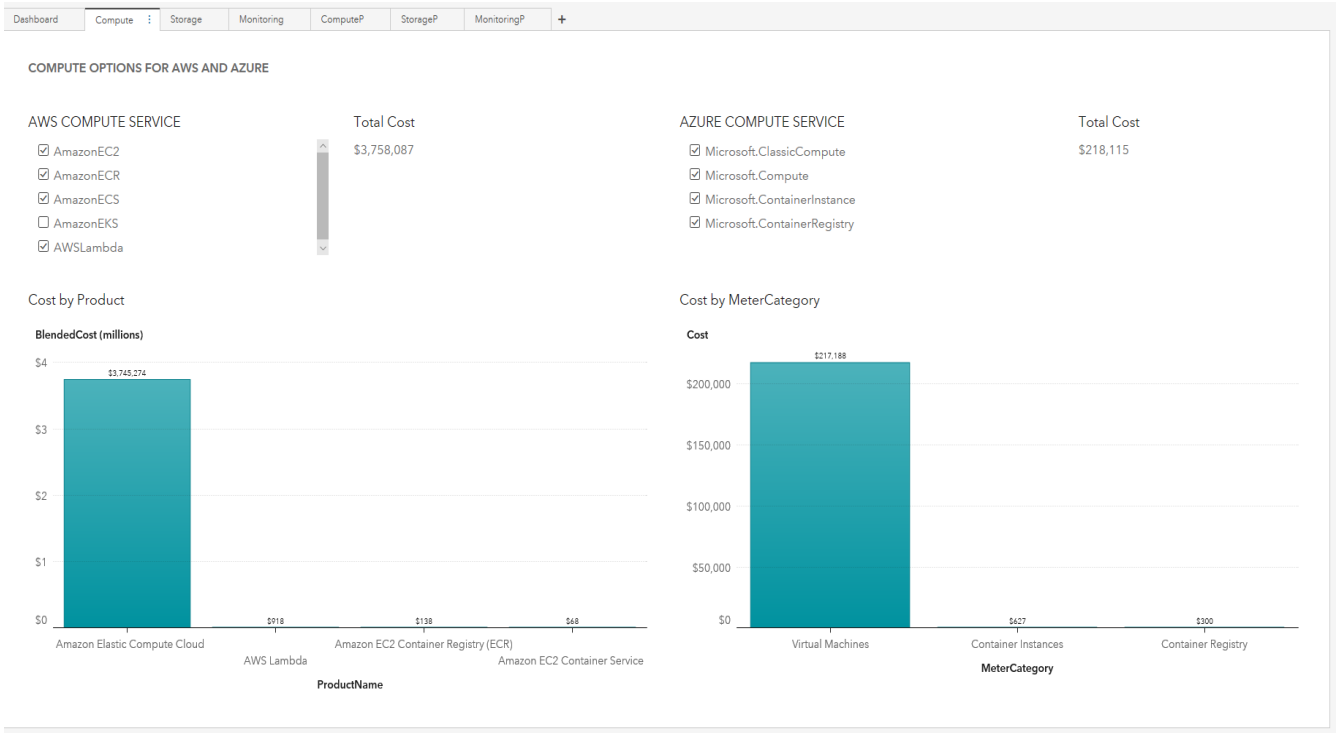

Fig.3. Cost for Compute resources

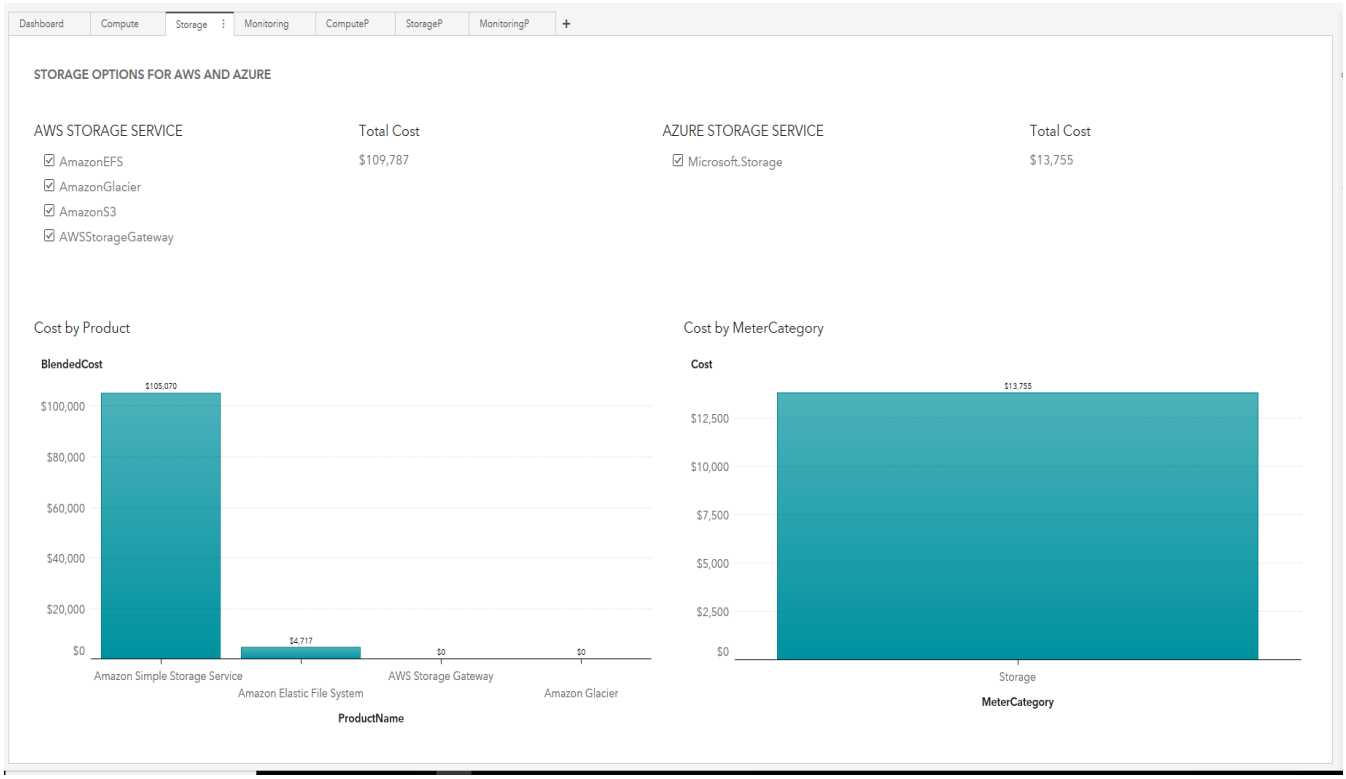

Fig.4. Cost for Storage resources

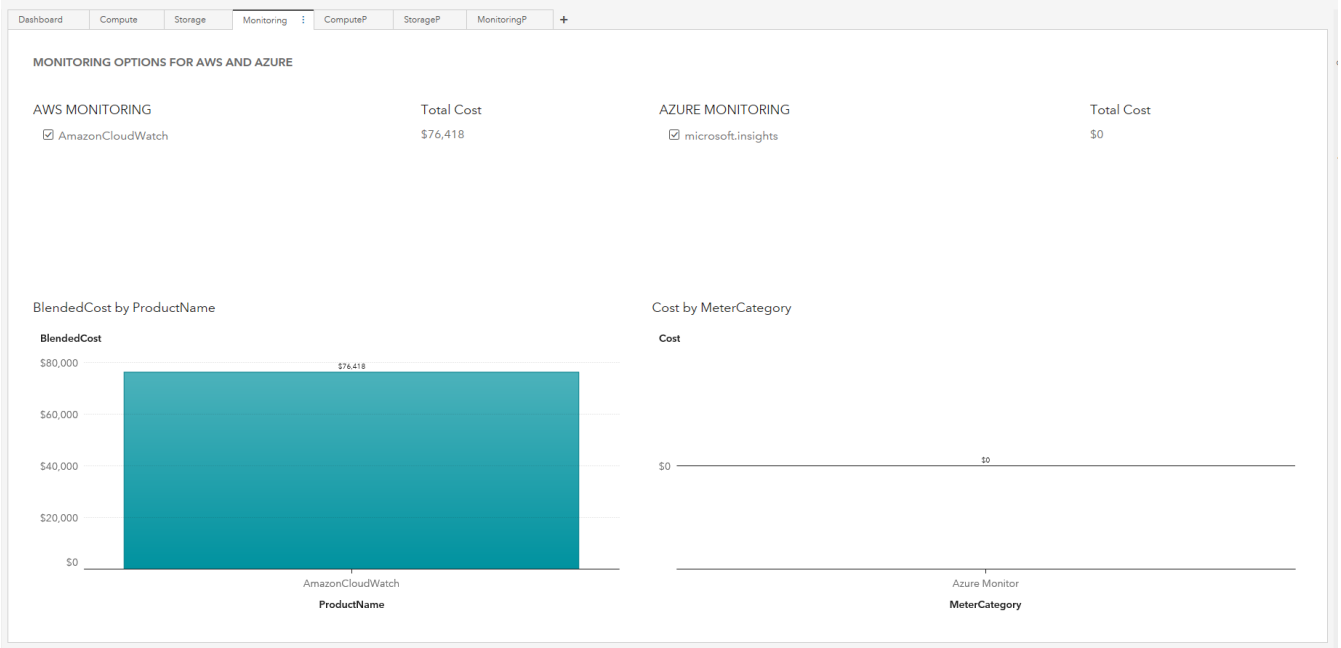

Fig.5. Cost for Monitoring resources

VII. CONCLUSION

A micro service based approach for fetching the data from cloud providers is presented in this work. The proposed system fetches the data from two cloud service providers Amazon Web Services and Microsoft Azure. This data is processed and stored in a persistent storage to generate analytical reports for further analysis. The main challenge in this case is the filtering and processing of data on a continuous basis. The use of microservices architecture in the proposed system is beneficial for secure data handling as well as efficient processing and loading of the data. The main aspect of this system is the unified approach for two different cloud providers AWS and Microsoft Azure. The cost of using comparative services from these cloud providers for three different categories compute, storage and management tools is provided in the form of visual statistics. This integrated report simplifies the task of organizations using these resources by enabling them to compare the cost of resources at one place and eliminating the need to use two different tools for getting the information relevant to a particular use case. It will facilitate improved transparency into the cost of operating these services and in turn help the administrators in efficient cloud infrastructure management.

The proposed system gets the data from two different cloud vendors Amazon Web Services and Microsoft Azure. In the future, it can be enhanced to incorporate several other cloud vendors. Moreover, the inclusion of several other services into the system will prove to be an added advantage. The users will then be able to have a comparison of all their resources at one place.

ACKNOWLEDGMENT

We would like to thank SAS Research and Development (India) Pvt. Ltd. for providing the necessary resources for carrying out this work. We wish to express our heartfelt gratitude towards Mr. Manish Limaye and Mr. Suresh Koukuntla whose expertise and assistance were paramount in carrying out this work. Their invaluable insights were very helpful. We are also immensely thankful to everyone who have helped us in carrying out this research.

REFERENCES

1. "The NIST Definition of Cloud Computing," [Online]. Available: https://csrc.nist.gov/publications/detail/sp/800-145/final [Accessed: Nov. 5, 2019].
2. "Cloud metering and billing," [Online]. Available: http://www.ibm.com/developerworks/cloud/library/cl-cloudmetering/ [Accessed: Nov. 6, 2019].
3. R. Jellinek, Y. Zhai, T. Ristenpart and M. Swift, "A Day Late and a Dollar Short: The Case for Research on Cloud Billing Systems," HotCloud (2014).
4. J. Kuhlenkamp & M. Klems, "Costradamus: A Cost-Tracing System for Cloud-Based Software Services," ICSOC 2017.
5. "AWS Billing and Cost Management," [Online]. Available: https://docs.aws.amazon.com/awsaccountbilling/latest/aboutv2/billing-what-is.html [Accessed: Nov. 6, 2019].
6. "Use Azure Billing APIs to programmatically get insight into your Azure usage," [Online]. Available: https://docs.microsoft.com/en-us/azure/billing/billing-usage-rate-card-overview [Accessed: Nov. 6, 2019].
7. T. Velte, A. Velte and R. Elsenpeter, *Cloud Computing: A Practical Approach*, McGraw-Hill, Inc., New York, NY, 2009 .
8. M. Fowler, "Microservices," [Online]. Available: https://martinfowler\\.com/articles/microservices.html. [Accessed:Apr. 20, 2019].
9. M. Garriga, "Towards a Taxonomy of Microservices Architectures," In Software Engineering and Formal Methods, Lecture Notes in Computer Science, 2018, vol. 10729, pp. 203-218.
10. S. Patanjali, B. Truninger, P. Harsh, "CYCLOPS: A micro service-based approach for dynamic rating, charging \& billing for cloud," In Proc. 13th International Conference on Telecommunications (ConTEL), 2015, pp. 1-8.
11. A. Narayan, S. Rao, G. Ranjan, K. Dheenadayalan, "Smart metering of cloud services," 6th Annual IEEE International Systems Conference (IEEE SysCon 2012), Mar. 2012.
12. M. Lindner, F. Galán, C. Chapman, S. Clayman, D. Bergström and E. Elmroth,"The Cloud Supply Chain: A Framework for Information, Monitoring, Accounting and Billing," CloudCom (2011) 2011.
13. A. Anwar, A. Sailer, A. Kochut and A. Butt,"Anatomy of Cloud Monitoring and Metering: A case study and open problems," Proceedings of the 6th Asia-Pacific Workshop on Systems, Tokyo, Japan, Jul. 2015.

14. “Amazon." Amazon Web Services [Online]. Available: https://aws.amazon.com/ [Accessed: Nov. 6, 2019].
15. “Microsoft Azure,” [Online]. Available: https://azure.microsoft.com/en-us/ [Accessed: Nov. 6, 2019].

AUTHORS PROFILE

Nupura Torvekar, born 31 Mar, 1994 graduated Bachelor of Engineering (Computer Engineering) from Savitribai Phule Pune University in 2016, currently pursuing Master of Engineering (Computer Engineering) from Pune Institute of Computer Technology, Pune.

Pravin S. Game, is working at Pune Institute of Computer Technology, Pune. He is a Ph.D. scholar in Computer Engineering at Shri JJT University, Jhunjhunu, Rajasthan. He received his Master of Engineering in Computer Engineering from Savitribai Phule Pune University and Bachelor of Engineering in Computer Science & Engineering from Sant Gadge Baba Amravati University. His research interests include data mining, big data analysis, machine learning.