A COMPARISON OF BIG DATA AND CLOUD COMPUTING APPLICATIONS WITH PROSPECTIVE TO SOFTWARE ENGINEERING

Rana Muhammad Nadeem [1], Shabir Ahmad[2*], Rana Muhammad Saleem[3]
[1]Department of Computer Sciences, Government Post Graduate College Burewala, Pakistan
[2]Department of Computer Sciences, Government College of Commerce, Multan, Pakistan
[3] Department of Computer Sciences, Burewala Sub Campus, University of Agriculture, Faisalabad, Pakistan
*Corresponding author: Email: mian_shabbir@hotmail.com

***ABSTRACt**: Cloud computing is used as software development service, operate and maintain systematic process that is an essential element to achieve a disciplined approach in the field of software engineering. Adding security and privacy throughout the engineering process, reliable, accurate, robust and reliable system, as well as adaptable to meet the needs of users and is vital to ensure the development of emerging software services. At this point, developers will be able to assist in providing a more reliable service should research methods and tools.*

Privacy-oriented information systems and cloud services is fully justified. In cloud environment large volumes of data, large volumes of data platforms for the collection of emerging data, management and visualization using a new approach offers advantages to change the software development trends. Organizations have always produced significant amounts of unstructured data from sources such as medical images, blogs, radio-frequency identification (RFID) tags, and locality sensors. Historically, organizations threw away most of the data they could collect to avoid what were once considered excessive costs of managing such a data deluge. In this paper, different aspects of big data along with cloud computing applications applications are compared. In addition, the researchers also suggest the possible solutions of these applications with respect to software engineering. This paper will help the new researchers to compare the cloud computing and big data applications and will lead to develop new solutions for the existing inadequacies.

Keywords: Cloud Computing, Big Data, Requirement Engineering, Software Process, Modeling, Implementation

1. INTRODUCTION

Today, the most popular applications is an Internet service to millions of users. Google, Yahoo! websites as face book and get millions of clicks a day. This generates terabytes of valuable data that can be used to improve user satisfaction and online advertising strategy. This real-time capture such as data storage and analysis, the top level are all common needs of online applications. In recent years to solve a series of problems have emerged the technology of cloud computing.

Developers with innovative ideas for Internet services, does not require a large capital investment for the team is no longer deploy the service; This paradigm shift is transforming the IT industry. The operation of computer data centers in large databases, data centers benefit from economies of scale, costs, electricity, bandwidth and hardware information processing in the cloud to enable the operation was a key factor in the descent way.

2. BACKGROUND

Today, the most popular applications is an Internet service to millions of users. Google, Yahoo! websites as face book and get millions of clicks a day. This generates terabytes of valuable data that can be used to improve user satisfaction and online advertising strategy. This real-time capture such as data storage and analysis, the top level are all common needs of online applications. In recent years to solve a series of problems have emerged the technology of cloud computing.

This article provides a comprehensive background study for scalable data management and analysis. In addition, the series focuses on a system designed to support applications to update their large workload against the Internet. In this work, the development of new applications and systems and determines some of the design challenges facing the distributed in the design and application designers and the implementation of the transition of traditional enterprise infrastructure other to extend most Challenges that need to be addressed.

2.1 Models of Software Process Life Cycle

In this section, the cloud and the software life cycle process of large data and model are discussed in relation to software engineering.

2.1.1 Life Cycle Models and Cloud Software Process

Especially from the perspective of government and industry, the independence of employing a security framework in the cloud has been proposed [2] (Cloud SSDLC). Cloud SSDLC, secure systems development lifecycle (SSDLC), cloud security manual for critical domains and brings together risk concerns. Cloud launch SSDLC, development, implementation, there are five main phases, including operation and layout. Also cloud critical areas for safety and the corresponding risks are integrated into each stage. Industry and government when the terms are used in different cases suggested Cloud SSDLC to demonstrate practical use and legal problems.

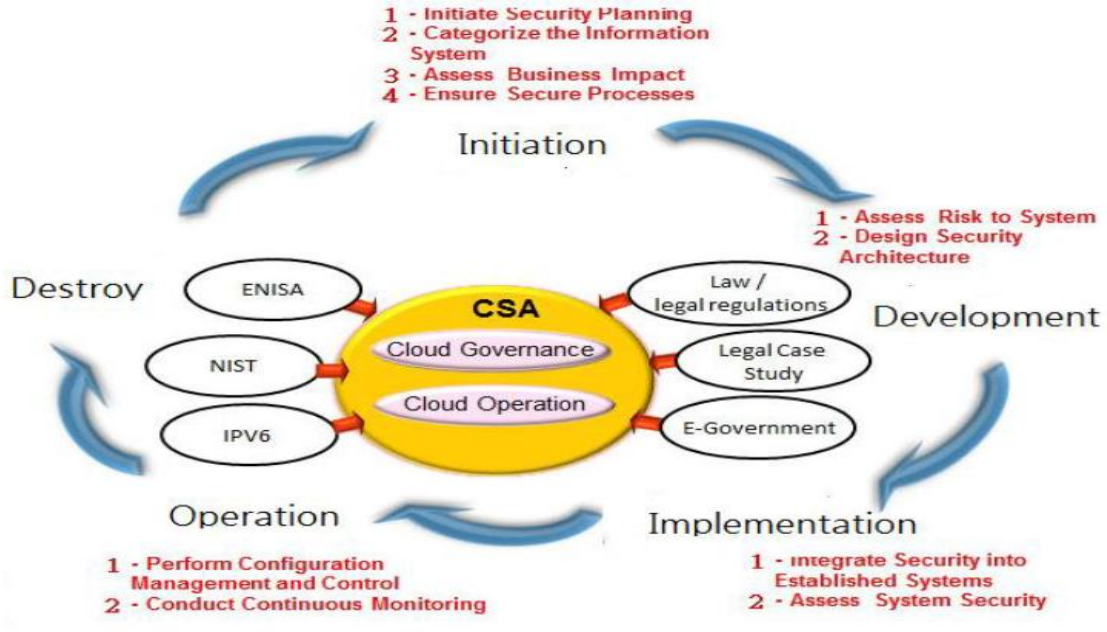


Fig-1: SSDLC Cloud Computing Model

2.1.2 Life Cycle Models and Big Data Software Process

With the dissemination of all areas of activity of the company's digital technology, industry and business, market forecasting, to obtain desired information data, such as customer service and mining in a new field of play are Entering into behavioral predictions must use scientific solutions to take advantage of new opportunities, social groups and forecasts of activity, etc.

Large data technologies, repetitive improvement and improved model number data model data collected, such as improved methods of scientific discovery reuse that you think you should adopted to blog articles [3].

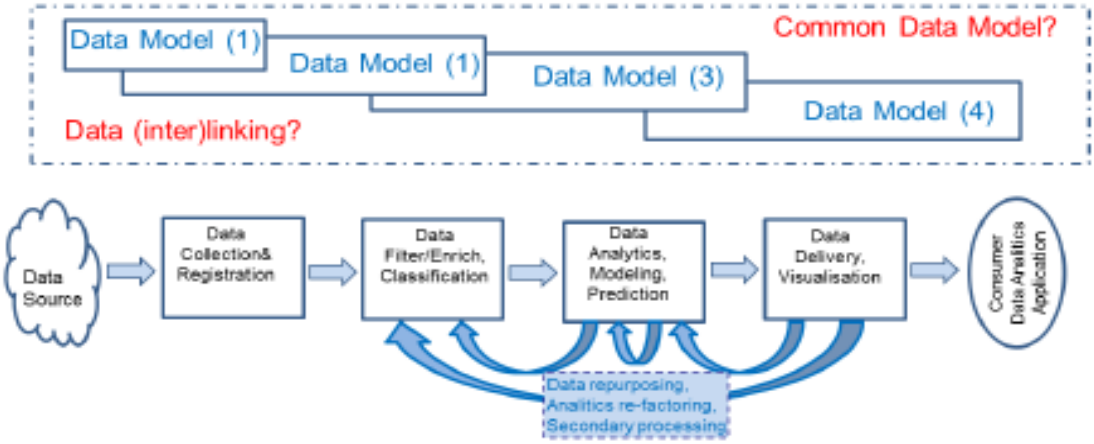


Fig-2: Big Data Lifecycle in Big Data Ecosystem

Another study reflects the complex and iterative procedure for scientific research: research in several stages: research project or planning experiment; data collection; data processing; The publication of the study results; Discussion, feedback; archived.

2.2 Software Requirements Engineering

This section presents the cloud computing and big data software requirement engineering.

2.2.1 Requirements Engineering for Cloud Computing

Data-intensive systems like cloud computing encompass huge amount of data in terabytes to petabytes (online http://wgiss.ceos.org/dsig/whitepapers/Data%20Lifecycle%20Models%20 and%20Concepts%20v8.docx). So systems need require very big storage and exhaustive computational power to execute queries quickly. To analyzes the extensive requirements the state-of-art paper [4, 5], describes various challenges associated with extensive requirements and suggest numerous solutions in meeting these requirements. Figure 3 illustrates the 2 architectural models for this type of system.

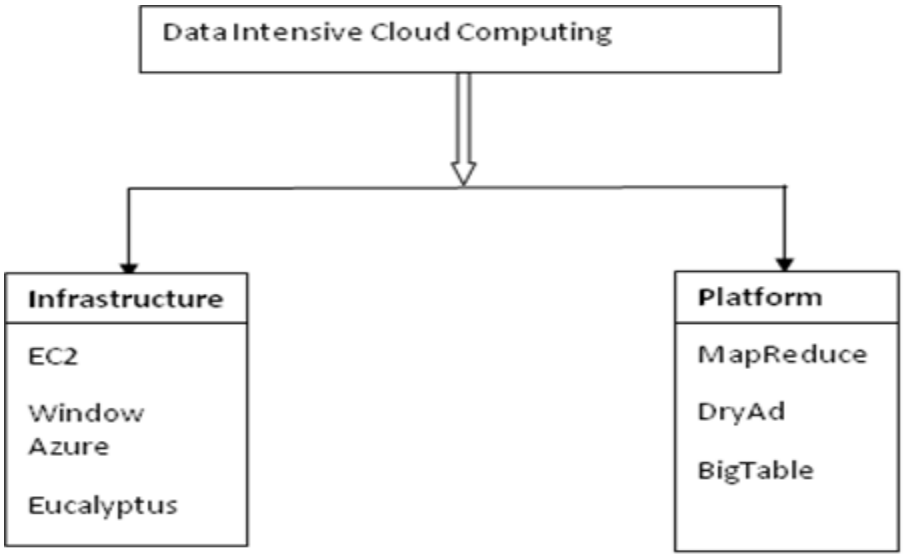


Fig-3: Data Intensive Cloud Computing

Table-1: The following table, enlist the cloud computing requirements, challenges and their possible solutions.

S#	Requirements	Challenges	Solutions
1	Scalability: Cloud should be able to support large no. of users with efficiency.	A cloud must have effective management, proper utilization of resources and mechanism for task-mapping.	Distributed file systems such as (GFS), Hadoop Distributed File Systems (HDFS) [6] Programming Platforms like **MapReduce,** Distributed Storage and Database Systems BigTable are good options
2	Availability, Fault Detection, and Fault Tolerance	In big data clouds, faults may lead to failure or crashes. There is need of mechanism to deal with these in timely manner.	Kahuna [7] is a fault detection tool. HiTune [8] is used for data flow performance
3	Flexible and Efficient User Access	In MapReduce strict any task can be converted to map and reduce tasks.	Dryad [9] supports thousands of nodes for large operations. Sawzall [10], is a high performance computing System suitable for MapReduce.
4	Elasticity	Adjustments are required in clouds for low and high load which is supported by VMs. During physical migration an elastic load balancing Challenges arise.	ElasTras [11] provides elasticity that is on demand provision of resources or de-allocation of resources. Zephyr [12] is used for live migration with Elastree capabilities.
5	Multiple Platform based Clusters sharing	Information for usage of resources for Dryad and Hadoop.	Otsu [13] is a tool for monitoring applications in a cluster those are data intense in nature. Mesos [14], provides sharing capability for multiple frameworks on the same cluster.
6	Scheduling of Disk Head	The disk I/O speed may reduce in a shared environment during multiple workloads.	A disk-scheduling scheme [15], which co-schedules data across all servers in the cluster.
7	Heterogeneous Environment	The execution speed of tasks varies in In heterogeneous systems	LATE [16], a scheduling algorithm which identifies slow tasks and prioritizes them according to their expected completion time
8	Data Handling, Locality, and Placement	Data-analysis servers For data-intensive applications is also critical which may affect the application performance.	Volley [17] is an automatic data placement tool.
9	Effective Storage	For Data-intensive systems leads	DiskReduce [18] is focused to reduce this

	Mechanism	to high disk usage. This leads to 200% extra utilization of disk space.	overhead. It reduces the disk usage between 10 and 25 %..
10	Privacy and Access Control	Large storage systems require interactions between multiple users. This requirement introduces additional challenges of procurement and management of access controls between the users.	In [19], the authors proposed a model which provides dynamic delegation of rights with capabilities for accounting and access confinement.
11	Billing	Incorporating accurate billing mechanism is significant and challenging for data intensive cloud systems. With massive requirements to access and compute huge amount of data, appropriate and methods are needed to compute billing.	A fair billing system for data intensive computing entails three components [20]. These include: • Cost of Data Storage • Cost of Data Access • Cost of Computation Of these three components, cost related to computation is normally billed in CPU hours.
12	Power Efficiency	For scalable systems, power requirements are likely to be increased due to addition of resources. Multicore systems are also being utilized for clouds. Reducing power usage for such systems is also desirable.	FAWN [21] is a flash-memory based system, which is designed to promote low power for data intensive applications requiring random access. Advancements in multi-core technology have lead to their utilization in cloud systems. Shang and Wang [22] have proposed power saving strategies for multi-core data intensive systems.
13	Network Problems	With low cost switches and top of the rack setup, the buffer of the switch may become full and packet loss may result. Barrier synchronized scenarios could encounter TCP Incast problem due to which long delays might occur.	To solve the TCP in cast problem, [23] proposed that TCP Retransmission Time Out (RTO) be reduced. Through real experiments, the authors observed that microsecond timeouts allowed servers to scale up to 47 in barrier synchronized communication environment.

2.2.2 Requirements Engineering for Big Data

The key factors that are taken into account when engineering large data requirements are the following:

Scalability: To ensure compliance with very large and growing data warehouses, including the ability to easily add additional storage resources if necessary.

High performance: maintaining a low response time and data entry time, has to keep pace with business.

High efficiency: to reduce data center storage and related costs.

Operational simplicity: additional IT environments to simplify data management without tremendous staffing.

Business Data Protection: In case of a disaster for business users and provide high availability for business continuity.

Interoperability: Very integration of complex environments and to provide Çevik bir infrastructure that supports a wide variety of business applications and analytical platforms such as Hadoop.

2.3 Software Design and Implementation

This section presents the design and implementation of large-volume data and cloud computing software.

2.3.1 A virtual cloud-based Design and Implementation of 10-Gigabit NIC

Platform and cloud services, has become a classic of supercomputing services for the period [28]. Changes in cloud technology, industrial technology and industrial competitiveness will greatly influence and strengthen regional scientific and technological innovation and accelerate the process of regional economic restructuring. The core technology of virtualized cloud computing resources. For the development of virtualization, server performance is more powerful, we need the number of virtual machines can run more and more number of NIC ports. The introduction of multi-core server platforms, and the popularity of cloud computing applications, providing a rapid increase in the number of applications for the Ethernet port. Five years ago there was a single processor core, most servers, virtual machines could lead to a very limited number of realistic and do not apply to the virtual machine integration services, which is not a very high demand for The NIC ports. Cloud computing, the application of quad-core processors and even the eight-core processor can even provide 64 cores

in a bidirectional server executable greatly improved from the number of servers to virtual machines with popularity popularity , We need more NIC ports [29, 30].

.

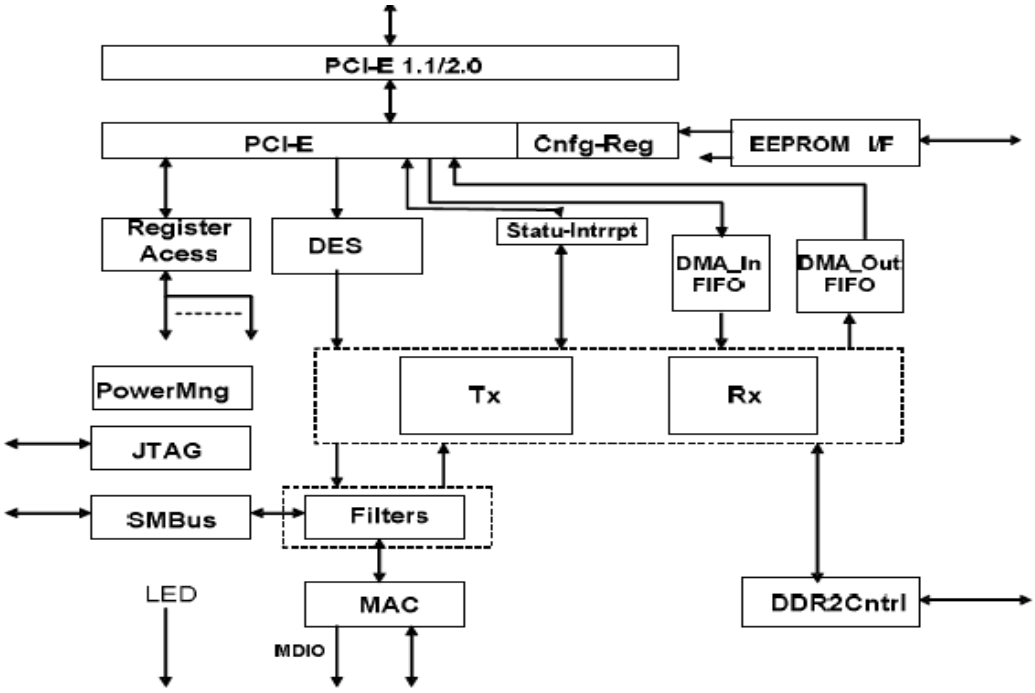


Fig-4: Cloud Computing-Oriented virtual 10-Gigabit NIC hardware architecture

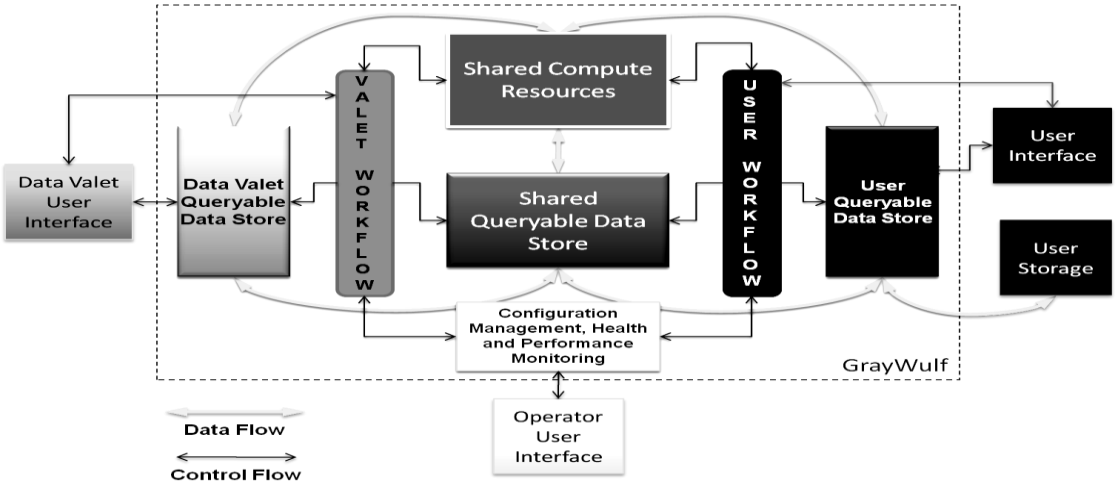


Fig-5: Big Data Software Design

The data resources are accessed by three different groups:

Users who perform analyses on the shared database.

Data valets who maintain the contents of the shared databases on behalf of the users.

Operators who maintain the compute and storage resources on behalf of both the users and data valets.

2.4 Software Testing

This section provides Big Data and cloud computing.

2.4.1 Software testing based on cloud computing

Cloud test, cloud computing technology is based on the software testing method [32, 33]. The article explains, the definition of cloud testing is derived from the concept of cloud computing. Cloud testing is another type of programming that tries to mimic the movement of real customers and anxiety as a method to perform complex testing of web applications that benefit from distributed test sites computing situations. You have an unlimited number of assets with cloud tests, when you spend when you spend, do what time passes.

2.4.2 Large Data Software Testing

Testing large volumes of data, and faced what is called the organization as a result of insufficient information about how

2.3.2 Big Data Software Design and implementation

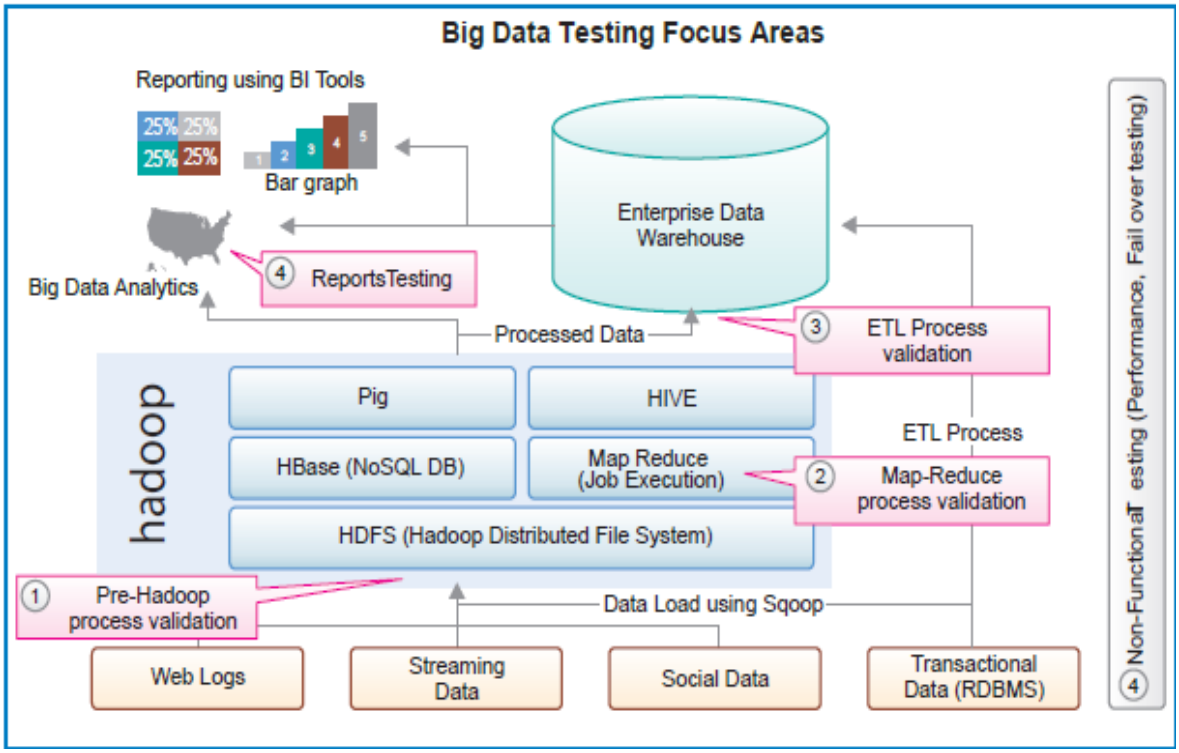
GrayWulf [31] cluster, which is high level system is shown in Figure. In this computing farm is separated from computer resources and shared query able data stores

the data will be tested are among the biggest challenges. Organizations, the strategy of validating structured and unstructured data for the start of the tests, the maximum test environment for creating databases, affordable and non-relational to those who already face difficulties to do non-functional tests . These difficulties and delays in the administration to the low quality of the data on the production and causes an increase in the cost. The robust test approach rejection for the verification of structured and unstructured data, and the total cost, and the first tests must be initiated in order to reduce the life span of the application to identify potential failures in the beginning And to drive the market [34, 35].

It requires different types of tests such as strong test data and test environment with functional testing and non-functional management to ensure good quality reliably processing data from various sources and analysis. Map process of structured validation and validation of unstructured data, functional verification test events, such as validity of data storage, it is important to ensure that the data are accurate and qualified.

When we enter when it comes to large data and multiple nodes, data and poor data quality at all stages of the process will have the opportunity to encounter problems. Evidence of data functionality, as a result of coding errors or node configuration errors are made to detect these data problems. Data, to ensure the error-free process, the test must be performed in three phases, each of the large data processing. Functional tests include: (i) pre-processed Hadoop verification; (II) The verification of output of Hadoop MapReduce process data; And (iii) validate the data and load it into edw'y. In addition to this functional checks, including performance testing and nonfunctional test failover testing must be performed. Figure 7 shows a diagram of large normal data architecture and highlights the areas you should focus on the test.

Fig-6: Big Data Testing Focus Area



3. CONCLUSION

The cloud provides the infrastructure needed to deliver services directly to customers via the Internet and use data based on the high-profile form of revenue data. This comparison of different stages of large data / frames, cloud computing and to examine different aspects of software engineering practices.

A limitation of this study, researchers only architecture, design, implementation and security are analyzed in terms of their stage. Most come with a solution to this framework, even though they are pure and are not being very effective, there are many challenges. Security, which prevents companies from entering the cloud remains the biggest obstacle. On the other hand, dynamic programming in heterogeneous environments is still a problem in this context.

REFERENCES

[1] Armbrust, M., Fox, A., Griffith, R., Joseph, A. D., Katz, R., Konwinski, A., & Zaharia, M. (2010). A view of cloud computing. *Communications of the ACM*,*53*(4), 50-58.

[2] J. B. Rothnie Jr., P. A. Bernstein, S. Fox, N. Goodman, M. Hammer, T. A. Landers, C. L. Reeve, D. W. Shipman, and E. Wong. Introduction to a System for Distributed Databases (SDD-1). *ACM Trans. Database Syst.*, 5(1):1–17, 1980.

[3] D. J. Dewitt, S. Ghandeharizadeh, D. A. Schneider, A. Bricker, H. I. Hsiao, and R. Rasmussen. The Gamma Database Machine Project. *IEEE Trans. on Knowl. and Data Eng.*, 2(1):44–62, 1990.

[4] Jawwad Shamsi, Muhammad Ali Khojaye, Mohammad Ali Qasmi; Data-Intensive Cloud Computing: Requirements, Expectations, Challenges, and Solutions. In: Springer (2013).

[5] SIDDIQ, Shahida, et al. "Implementation Issues of Agile Methodologies in Pakistan Software Industry." *International Journal of Natural and Engineering Sciences* 8.3 (2014): 43-47.

[6] Grossman, R., Gu, Y.: On the varieties of clouds for data intensive computing. In: IEEE Data Engineering (2009)

[7] Bu, Y., Howe B., Balazinska, M., Ernst, M.: Hadoop: efficient iterative data processing on large clusters. J. Proceedings VLDB Endowment **3**(1–2), 285–296 (2010).

[8] Tan, J., Pan, X., Kavulya, S., E. Marinelli, E., Kavulya, S., Gandhi, R., Narasimhan, P.: Kahuna: Problem diagnosis for MapReduce-based cloud computing environments. In: 12th IEEE/IFIP NOMS (2010)

[9] Dai, J., Huang, J., Huang, S., Bo Huang, B., Liu, Y.: HiTune: dataflow-based performance analysis for big data cloud. In: Usenix HotCloud (2011)

[10] Isard, M., Budiu, M., Yu, Y., Birrell, A., Fetterly, D.: Dryad: distributed data-parallel programs from sequential building blocks. In: ACM SIGOPS/Eurosys (2007)

[11] Pike, R., Dorward, S., Griesemer, R., Quinla, S.: Interpreting the data: parallel analysis with Sawzall. Sci. Program. J. (Special Issue on Grids and Worldwide Computing Programming Models and Infrastructure) **13**(4), 227–298

[12] Das, S., Agrawal, D., Abbadi, A.: ElasTras: an elastic transactional data store in the cloud. In: Usenix Hotclud (2009)

[13] Elmore,A., Das, S., Agrawal, D.,Abbadi,A.: Zephyr: live migration in shared nothing databases for elastic cloud platforms. In: ACM SIGMOD (2011)

[14] Ren, K., López, J., Gibson, G.: Otus: resource attribution in data-intensive clusters. In: Mapreduce (2011)

[15] Hindman, B., Konwinski, A., Zaharia,M., AliGhodsi, A., Joseph, A., Katz, R., Scott Shenker, S., Stoica, I.: Mesos: a platform for fine-grained resource sharing in the data center. In: Usenix NSDI (2011)

[16] Wachs, M., Ganager, G.: Co-Scheduling of disk head time in cluster-based storage. In: IEEE SRDS (2009)

[17] Zaharia, M., Konwinski, A., Joseph, A., Katz, R., Stoica, I.: ImprovingMapReduce performance in heterogeneous environments. In: Usnix OSDI (2008)

[18] Agrawal, S., Dunagan, J., Jain,N., Saroiu, S., Wolman, A., Bhogan, H.: Volley: Automated data placement for geo-distributed cloud services. In: Usenix NSDI (2010)

[19] Fan, B., Tantisiriroj, W., Xiao, L., Gibson, G.: DiskReduce: RAID for data-intensive scalable computing. In: PDSW Super Computing (2009)

[20] Harnik, D., Kolodner, E., Ronen, S., Satran, J. Shulman-Peleg, A., Tal, S.: Secure access mechanisms for cloud storage. In: 2nd Workshop on Software Services: Cloud Computing and Services: Cloud Computing and Applications based on Software Services (2011)

[21] Ahmad, Shabir, Shafiq Hussain, and Muhammad Farooq Iqbal. "A FORMAL MODEL PROPOSAL FOR WIRELESS NETWORK SECURITY PROTOCOLS." *Science International* 27.3 (2015).

[22] Andersen, D., Franklin, J., Kaminsky, M., Phanishayee, A., Tan, L., Vasudevan, V.: FAWN: a fast array of wimpy nodes. In: Communications of the ACM (2011)

[23] Shang, P., Wang, J.: A novel power management for CMP systems in data-intensive environment. In: Parallel & Distributed Processing Symposium (IPDPS) (2011)

[24] Vasudevan, V., Amar Phanishayee, A., Shah, H., Krevat, E., Andersen, D., Ganger, G., Gibson, G., Mueller, B.: Safe and effective fine-grained TCP retransmissions for datacenter communication. In: ACM SIGCOMM (2009).

[25] Khan Kamran et al. "Evaluation of PMI's Risk Management Framework and Major Causes of Software Development Failure in Software Industry". IJSTR 3 (11): 120-124, 2014.

[26] Ahmad, Shabir, et al. " FORMAL METHODS AND NETWORK SECURITY PROTOCOLS: A SURVEY." Sci.Int.(Lahore),29(3), 581-585, 2017.

[27] Azam, Farooq, et al. "Framework Of Software Cost Estimation By Using Object Orientated Design Approach." *IJSTR* **3**(11): 97-100, 2014.

[28] Liang-Jie Zhang and Qun Zhou, *IBM T.J. Watson Research Center, New York, USA*, IEEE International Conference on Web Services (2009)

[29] B. Hayes. Cloud Computing. Commun. ACM,51(7):9{11, 2008. Available at http://doi.acm.org/10.1145/1364782.1364786.

[30] Baloch, Muhammad Perbat, et al. "Comparative Study Of Risk Management In Centralized And Distributed Software Development Environment." Sci.Int.(Lahore),26(4),1523-1528, 2014.

[31] Wang Jun and Fanpeng Meng, 2011 International Conference on Internet Computing and Information Services (IEEE 2011).

[32] Afaq Salman et al. "Software Risk Management In Virtual Team Environment**"**. *IJSTR* 3 (12): 270-274, 2014.

[33] Hussain, Shafiq, et al. "Threat Modelling Methodologies: A Survey." *Sci.Int.(Lahore)*,26(4),1607-1609, 2014.

[34] Ahmad, Shabir, and Bilal Ehsan. "The Cloud Computing Security Secure User Authentication Technique (Multi Level Authentication)." *IJSER* 4(12): 2166-2171 (2013).

[35] Siddique, Abu Buker, et al. "Integration of Requirement Engineering with UML in Software Engineering Practices" *Sci.Int.(Lahore),* 26(5), 2157-2162, 2014.