REVIEW ARTICLE OPEN ACCESS

Achieving High Performance Distributed System: Using Grid, Cluster and Cloud Computing

Sunil Kr Singh[1], Kavneet Kaur[2], Anuj Aggarwal[3], Dharvi Verma[4]
[1, 2,4]CSE Department, Bharati Vidyapeeth's College of Engineering Affiliated to GGSIP University, New Delhi, India
[3]ECE Department, Bharati Vidyapeeth's College of Engineering Affiliated to GGSIP University, New Delhi, India

Abstract
To increase the efficiency of any task, we require a system that would provide high performance along with flexibilities and cost efficiencies for user. Distributed computing, as we are all aware, has become very popular over the past decade. Distributed computing has three major types, namely, cluster, grid and cloud. In order to develop a high performance distributed system, we need to utilize all the above mentioned three types of computing. In this paper, we shall first have an introduction of all the three types of distributed computing. Subsequently examining them we shall explore trends in computing and green sustainable computing to enhance the performance of a distributed system. Finally presenting the future scope, we conclude the paper suggesting a path to achieve a Green high performance distributed system using cluster, grid and cloud computing.

Keywords: Grid Computing, Cluster Computing, Cloud Computing, High Performance Computing, Distributed Computing.

I. INTRODUCTION

The paper basically focuses on the various factors that would provide high performance computing environment in distributed systems. High performance computing is narrowly defined as the development and use of the fastest and most powerful computing systems i.e., potential computing. It covers technological, political and economic features of the distributed computing enterprise. The major findings and recommendations on the subject have been summarized in the later sections of the paper.

1.1 Cluster Computing

All along the archetypal generation of computing, projects prescribing immense estimations and ample processing were dependent on jurisdiction or sprinkling corporative. Such hulking supercomputers and frameworks were exorbitant for singleton. Though expenses of PCs are thumbing down, supercomputers are still out of sight. As a result of which Donald Becker and Thomas Sterling imported Beowulf clustering in1993 [1] which lit off the counter computers, building a cluster that emulated dormant supercomputers. The paramount behind this is to create a computing arrangement for providing the necessary processing power at a nominal cost. As the nodes are repository of processors, security is absolutely airy and thence alertness in confining interconnected networks from external networking. Admitting considerable expedient in computing power, clustering certainly has hitches and hesitancies as a comparably newfangled technology. Distributed computing administers to beset a distended sphere of clustering by permitting the nodes to prevail all over the world and also be multiuse machines. Distributed computing has an analogous notion as clustering, allowing many nodes to work on large problems in parallel after breaking them into smaller units. Innumerably work units are distributed several times to too many nodes, curbing the probabilities of processing lapses and narrate for processing done on tedious CPUs. The client supervises the data resurgence and capitulation laps along with the code essential to order the CPU how to routine the work unit.

1.2 Grid Computing

Amalgamation of computers from different managerial realms to achieve a bourgeois objective, coined as grid computing, can be mapped to late 1980's and early 1990's[2]. Adoption of middleware, software that associates software peripherals and venture utilities, is a preeminent scheme of grid computing to segment and allocate fragments of a program amid numerous computers. It embraces computation in a distributed manner, gathering colossal clusters. Harmonizing applications on grids can be a convoluted job, chiefly while managing stream of instructions across distributed computing assets. Grid workflow systems are refined as a functional form of workflow management system framed distinctively to construct and accomplish an array of ciphering or data handling steps.

1.3 Cloud Computing

Genesis of the term 'cloud computing' is ambiguous, although it sounds to be borrowed from the habit of employing sketches of clouds to symbolize networks. The custom of catering remote connection to computing activity through networks contributed to prevalent usage of this caption. Cloud computing cite to an exemplary of network computing where a program or utility executes on a connected servers instead of confined computing apparatus. Corresponding to the conventional client-server or mainframe model, a node associates with a server to accomplish a job.

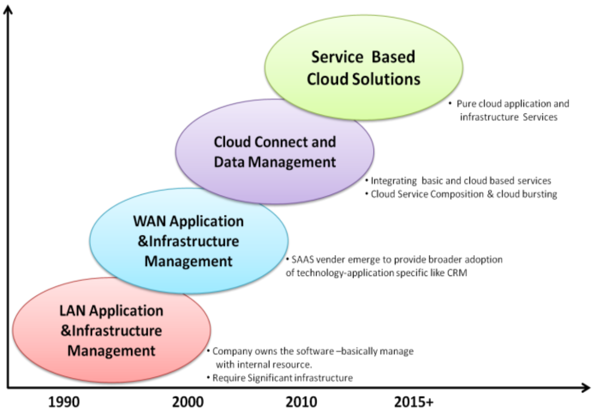

Figure 1:Year wise expansion of Cloud

The digression with cloud computing is that the computation may be executed on a single or many linked nodes at the same instance, applying the notion of virtualization. Virtualization allows multiple servers to be designed and distributed among several autonomous 'virtual' servers, operating separately seeming to the node to be a single device. These virtual servers are core, extensile, mountable and un-mountable, un-influencing the nodes.

II. NIGGLING SCRUTINY

2.1 Cluster Computing

IBM sighted cluster computing as a substitute to coupling thumping mainframes, to produce a more profitable form of economic affinity, in 1960s [3]. IBM's Houston Automatic Spooling Priority (HASP) system and its heritor, Job Entry System (JES), permitted dissemination of effort to a node-manufactured mainframe cluster. IBM still opts clustering of mainframes via parallel systems, permitting hardware, operating systems, middleware and system management software to maintain powerful conduct and cost improvements allowing massive mainframe users to pursue executing their current operations. Howbeit, cluster computing did not achieve strength until the concurrence of three important bents in1980s namely, performance microprocessors, high speed networks and standard tools for high performance distributed computing [4]. New bent is the proliferating demand of computing power for estimation science and monetary utilities, coupled with the high cost and low receptiveness of classical supercomputers. The developments in these technologies and their availability make clusters or networks of computers a tempting solution for cost-efficient parallel computing.

2.1.1 ARCHITECTURE

In Beowulf system, the utilities never see the computing node (slave node) and interaction is with master node only. Master node is a special machine\system which manages the slave nodes. Master typically has only two interfaces: one that communicates with the personal Beowulf networks and other that is for general purpose network. Slave has their own version of the same operation system. It manages its local memory and disk space. However, the private slave networks may humungous files containing global data, and can be accessed by slaves if required [20]. An archetypal Beowulf contour is depicted in figure 2.

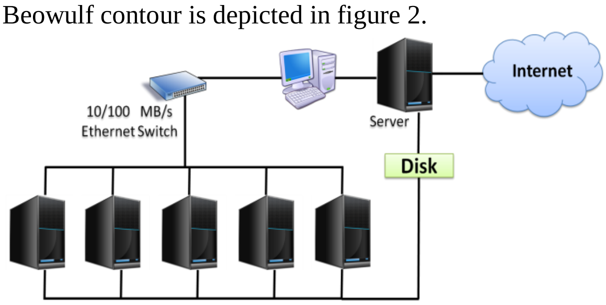

Figure 2: Beowulf Contour

Computer clusters exist on separate physical computers with same operating system. With the onset of virtualization, cluster nodes may be executed on a separate physical computer with separate operating systems designed virtually to look similar. The clusters may be virtualized on various structures as maintenance takes place.

2.1.2 CHARACTERISTICS

The high- availability and load- balancing capability of clusters attracts customers and interests investors. Due to low cost and elasticity they are easier to handle and maintain. Ubiquitous approach maximizes its efficiency and performance. As clusters are simple in design, every node can be granted required attention, thus, avoiding affects of node failures.

2.1.3 IMPLEMENTATION

The Linux world supports various cluster software. Linux Virtual Server, Linux-HA- director-based clusters permit entry for services to be distributed among several cluster nodes. OpenMosix, Kerrighed, OpenSSI are clusters consolidated into kernel that provide for direct process transfers among homogeneous nodes.

Microsoft Windows Computer Cluster Server 2003 dependent on Window server platform, permits parts for High Performance Computing like, Job Scheduler and management tools.

Several middleware technologies have been worked upon for different hardware and software compatibilities.

2.2 Grid Computing

Grid computing is an essence of distributed computing. Just as internet user sees a consolidated instance of content through web, a grid user views a single, large virtual computer.

Grid technologies commit to change the way in which complex computational problems are tackled by organizations. However, the intent of large-scale resource sharing has not yet been accomplished in several areas. Grid computing has developed the field of computing. Started as a project to link US supercomputers, Grid computing has evolved far beyond.

2.2.1 ARCHITECTURE

Grid computing is based on an open set of standards and protocols which enable communication across geographically dispersed and heterogeneous environment. Example of grid architecture is Open Grid Services Architecture (OGSA).

Like all other computing, grid architecture is also defined in layers. There are four layers in grid architecture, lowest being the network layer which connects the grid resources and highest being the application layer which includes applications in science, engineering, business, finances and more as well as portal and development toolkits to support the applications. This is the layer that a grid user can view and interact with. The application layer often includes the service-ware which performs the general management functions like tracking, who is provided with grid resources and who is using them. The intermediate layer known as middleware layer provides the tools that enable various elements to participate in grid. The middleware layer is sometimes the "brain" behind a computing grid [19]. Resource layer is sandwiched between the network and middleware layer. This layer contains the actual grid resources that are connected to the network. This has been depicted pictorially in fig 3.

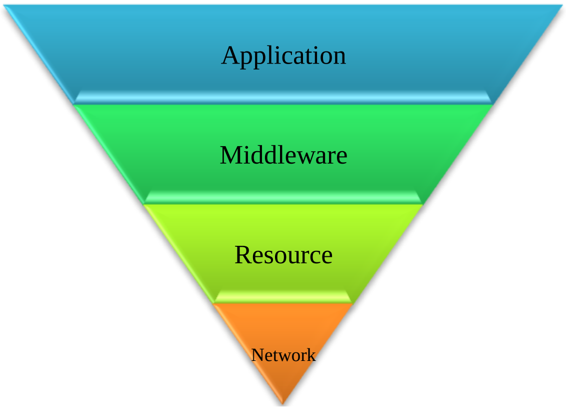

Figure 3: Grid Architecture

2.2.2 CHARACTERISTICS

Grid system governs resources that are not subjected to centralized control by integrating resources and users in different control domains. Built from multipurpose protocols and interfaces that address issues like authentication, authorization, resource discovery and resource access, grid uses standard, open, general-purpose protocols and interfaces. Grid permits its constituent resources to be used in a managed way to deliver various nontrivial qualities of service[12]. Grid applications have common requirements like, partitioning of applications to break the problem into discrete pieces, task and workflow discovery and scheduling, distributing the data where and when it is required, provisioning and distributing application codes to specific system nodes, result management to assist in the decision processes of the environment, autonomic features like self configuration, self recovery, management and optimization and many other.

2.2.3 IMPLEMENTATION

Many applications like collaborative engineering, data exploration, high throughput computing, distributed supercomputing have benefited from grid framework. Some of the more visible areas of grid implementation are:

Schedulers- types of applications responsible for management of jobs- form a hierarchical structure with Meta schedulers and other lower level schedulers. The job acknowledged by grid schedulers are analyzed and allocated to respective resources for execution on the basis of service level requirements.

ResourceBroker- responsible for pairing links between the client and server- enables selection of best available resources for executing a job. This pairing involves allocation of apt resources for task execution and supports nodes' deadline and financial constraints for optimized scheduling.

Load balancing feature enables distribution of workload among the resources in grid environment and requires integration into any system in order to prevent connections with schedulers and resource managers

Grid portals are similar to web portals as they provide uniform access to grid resources. They have the capability of querying database, transferring files, checking job feedback, status, security management and providing personalized solution availability.

Integrated solutions are a combination of existing advanced middleware and application functionalities, to provide coherent and high performance results across the grid environment. This advancement has been witnessed by several global industry sectors

2.3 Cloud Computing

Cloud computing which provides shared resources, software and information to computers and other devices on demand is coined as "Internet Based computing"[16]. Technical definition is 'a computing capability that provides an abstraction between the computing resource and its underlying technical architecture (servers, storage, network), enabling convenient on demand network access to a shared pool of configurable computing resources that can be rapidly provisioned and released with minimal management effort or service provider interaction'[17]. Cloud technologies have created a new trend in parallel programming [18]. In this section, we shall discuss more about the cloud computing.

2.3.1 ARCHITECTURE

Figure 4 gives a general idea of the architecture of cloud computing for smarter planet and layered view of IT infrastructure, services and applications that constitute cloud computing. It is possible to differentiate four layers that progressively shift the point of view from the node to the end-user.

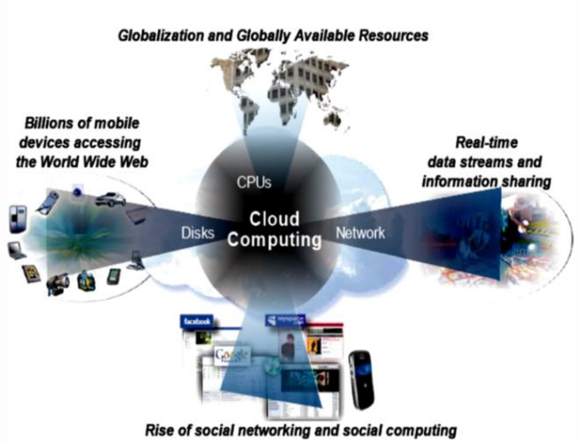

Figure 4(i). Cloud Computing model for smarter planet

Figure 4(ii):.Layered view of Cloud architecture

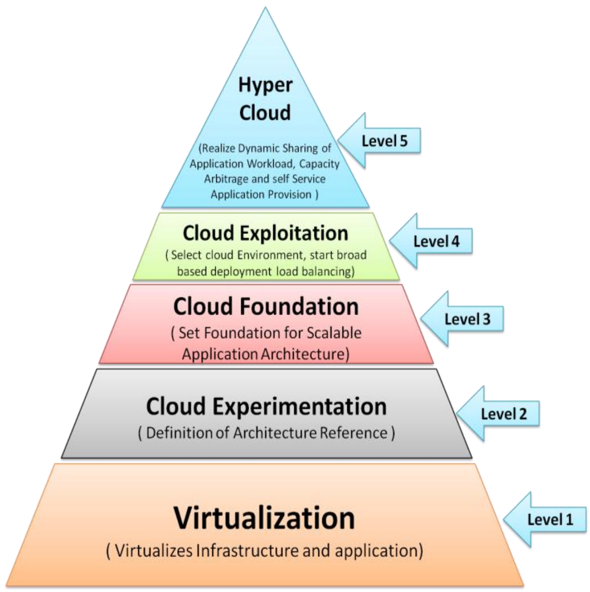

Figure 4(iii): Cloud Computing Adoption Structure

The lowest level is characterized by physical resources with infrastructure on top of it. Cluster data centers and spare desktop machines are resources of different nature. Infrastructures support commercial cloud and are composed of data centers hosting hundreds of nodes, while private cloud provides a more heterogeneous environment. This level provides the "horse power" [14] of the cloud.

The physical infrastructure whose aim is to provide an appropriate execution environment for applications and to exploit physical resources, is managed by the main middleware layer. The core middleware rely on virtualization techniques to provide advance services like QoS, sandboxing, application independence. Hardware level virtualization and programming level virtualization are most popular amongst various alternatives available. Application independence and partitioning of physical resources like memory and CPU is guaranteed virtually by hardware level virtualization. On the other hand, sandboxing and execution management for applications developed via specific technology or programming language is provided by programming level virtualization. On top of this, the central middleware provides a wide spectrum of services that help service providers for delivering a professional or commercial service to the end user. Negotiation of QoS, admission control management of execution, monitoring, accounting and billing are example of such services. Core middleware along with physical infrastructure represents the platform on top of which the applications are employed in cloud. Direct user level access to this layer is very rare and thereby core middleware are accessed through a user level middleware to deliver these services. This provides environment and simplifying tools to develop and employ applications in cloud. The user level middleware consists of the access point to applications in cloud.

2.3.2 CHARACTERISTICS

Cloud computing plays a major role in enhancing technological infrastructure resources for computing. It facilitates interaction between humans and machines at a nominal cost.

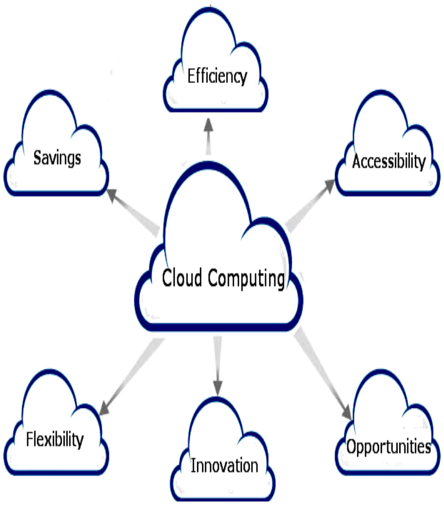

Figure 5: Features of cloud computing

Easy maintenance and sharing of resources enables the node to access the data regardless of its device or location. Performance, productivity, reliability and scalability are not an issue while using distributed high performance cloud computing. Security due to centralization of data is not at par with the expectations. Virtualization enables sharing of resources without actually possessing them.

On-demand self service, broad network access, resource pooling, rapid elasticity and measured services are five essential characteristics of cloud computing [15].

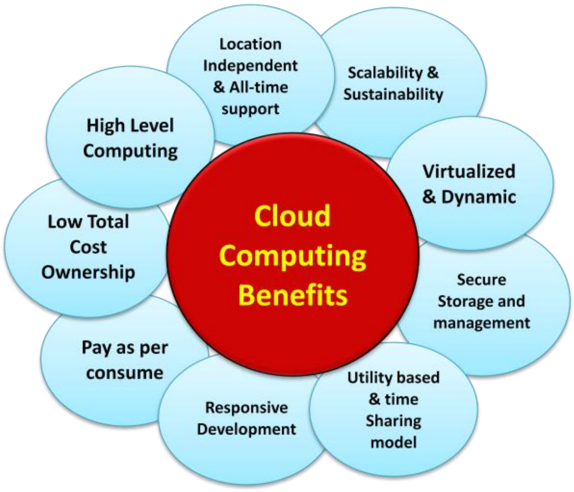

Figure 6: Benefits of Cloud Computing

2.3.3 IMPLEMENTATION

The wide spectrum of services exposed by cloud are classified and organized into three main offerings that are available to nodes: scientific, institution and enterprises via software, platform and infrastructure as a service.

Infrastructure as a service (IaaS) or hardware as a service delivers IT infrastructure based on virtual or physical resources as a commodity to customers. These resources meet the node requirements in terms of memory, CPU type and power storage.

Platform as a service (PaaS) provides a development platform where users can develop their own applications and execute them on cloud. Google AppEngine is an example of such a service.

Software as a service (SaaS) enables end user integrated services comprising hardware development and applications. Users are not permitted to customize these services but can access those services hosted in the cloud. An example of SaaS is a Google Document.

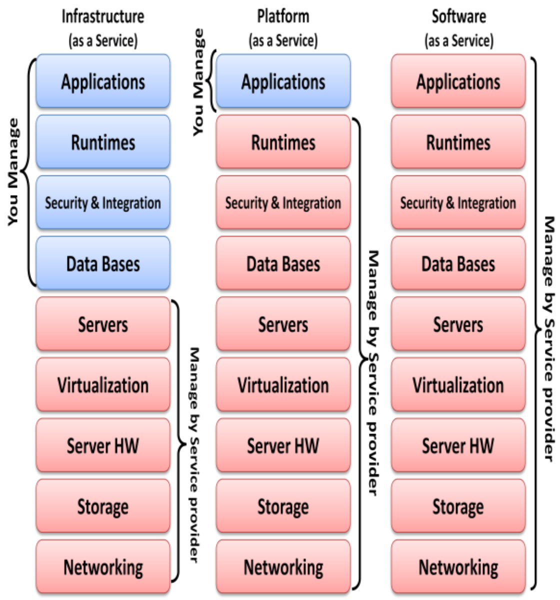

Figure 7. IaaS, PaaS and SaaS model

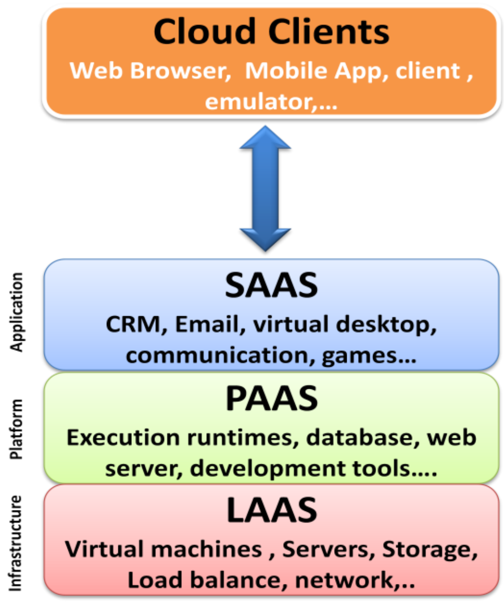

Figure 8: Layered Architecture

III. TRENDS IN HIGH PERFORMANCE COMPUTING

Since 60 years, the field of computing has undergone rapid transformations. Despite this, the long term evolution of performance seems to be stagnant. Massively Parallel Processor (MPP) systems are being accepted for engineering as well as for new commercial applications.

At the onset of 1990s, MPP systems came into the market claiming vector multiprocessors as pushovers. Top500 list, to provide a more reliable statistics on high performance computers [10], in June 1993 declared that 156 MPP systems were employed already.

Based on the present Top500 data and assumption that the current rate of performance improvement would continue in future, we can hypothesize the observed conduct and analyze these values with the intention of government programs such as the department of Computing and Communications, and PetaOps initiative. Considering that in 2005, no small system made it to the Top500 ranking. First PetaFlop/s were available around 2009 and rapid changes were adopted in technology used in high performance computing devices, but still there is currently no reasonable image possible for architecture of the PetaFlops systems at the end of the decade [11].

IV. DESIGN TECHNIQUES FOR GREEN SUSTAINABLE COMPUTING

Many advanced methodologies for enhancing energy efficiency of IT(Information Technology) and making it more feasible involve the need to dynamically accustom computation to the appropriate energy profile. Complex distributed computing environments provide an array of opportunities for managed adoption among multiple nodes and at multiple levels.

Cluster EAC[5] (Energy Adaptive Computing) is a model that requires a significant computation employing multiple servers before response for a request submitted by a client can be returned. This implies client's role is minimal and adaption of energy is of prime concern in data center infrastructure. In EAC cluster, the adaption of energy happens at multiple levels, with power limit that a level needs to adapt.

Client-server EAC requires to be handled with a well-managed end-to-end adaptation including the client, server and the interceding network. The motive of management is to elevate the client experience within the energy limits. As the client becomes more mobile and demand richer capabilities, the limited battery capacity gets in the way causing hindrance to energy adaptations

Cloud techniques have been proposed to outsource mobile computation to cloud platform that can make the required resources on demand available [6] [7] [8].

Energy adaption in P2P (Peer-to-Peer) environment requires cooperation among peers. This issue has been examined and the solution proposed is an energy adaptive version of Bit Torrent protocol [9].

The various issues for realization of EAC are: Hierarchical Power Control, Demand side adaption, supply side adaption, and QoS (Quality of Service) aware scheduler.

V. PROJECT ANALYSIS

In order to understand and achieve desired high performance system, we must be aware of features of specified computing techniques. To make study and analysis of these systems more understandable and communicable, this section describes and examines some famous projects in these fields.

As per the framework of the paper, the discussion begins with the exploration of THE LATTICE PROJECT [22].

The Lattice project being a research project in Grid computing, targets unification of computing resources into a grid computing system to make the resources systematically accessible and recognizable.

To achieve something high we must recognize the reason as to why that height is required. In case of Lattice, the ever increasing complexity and size of data calls for increase in computational efficiency. To attend this call, lattice project requires amalgamated computational resources. Since divergent computational resources are used, software development is in a dire need of this project which it slakes by employing open source equipments. The development of user interface along with increased scalability engrossed humongous effort. Lattice project engaged features of grid for accessing large-scale resources and modeled it over the scope of personal computers.

Architecture born out of fusion of Globus and BOINIC was adopted in various applications and projects like, BLAST (Basic Local Alignment Search Tool), ClustalW, IM (Isolation with Migration) Et. Al.

Researchers in Asia –Pacific region have been trying to solve numerous issues and problems linked to cluster computing, like employment of multithreaded DSM runtime systems; reduction in network overheads and communication patterns; development of realist communication models; distributed and parallel file systems. Various researches and papers presented in this regard have been recorded marvelously by Mark Baker, RajkumarBuyya, Ken Hawick, Health James and Hai Jin in their paper Cluster Computing R&D in Australia [24].

Google App Engine and Amazon Web Service are two leading cloud platforms. The comparative study [23] of these platforms by Chao He under the guidance of Prof. Raj Jain revealed that neither of them was significant in successful round trip time and throughput facets of cloud computing. Also it was concluded that none of these two was superior to the other. Cloud computing can get judicious performance in comparison to the traditional web servers depending upon the service delivered. The experiments conducted by them provided a better insight on the methodology to construct the cloud computing infrastructure and platforms.

VI. FUTURE PROSPECTS OF DISTRIBUTED COMPUTING

Increasing power and speed data centre is not always efficient and sometimes leads to an additional cost, so one should not expect to increase the efficiency more than a required limit. Distribution of data centers and use of closest data centers is a better and a far more optimal choice. It has been predicted that storage and computing on personal computers will be forgotten and transferred into distributed clouds. Therefore, architecture and evaluation of data centers should be performed for future of computing through suitable prediction. According to review and evaluation performed in the field of high performance computing, high performance distributed computing through grid, cluster and cloud still has a shortage in performance evaluation and special measures are required for this work. It is better to consider delay in evaluations or implement a criterion for evaluation of service level agreement because these agreements are most important for the users and one can present more accurate evaluation in future by specifying type of user's requests or specifying and distinguishing all users. High performance embedded computing (HPEC) systems are amongst the most challenging systems in the world to build. The primary sources of these difficulties are the large number of constraints on an HPEC implementation:

- **Performance:** latency and throughput.
- **Efficiency:** processing, bandwidth, and memory.
- **Form Factor:** size, weight, and power.
- **Software Cost:** code size and portability.

Thus in future, we hope to achieve high performance distributed system by combining best features of grid, cluster and cloud computing as well as reconfigurable computing[21].

Besides the aforementioned requirements, the emergence of Jungle Computing has given a boost to the field of Distributed Computing. It uses a system which is distributed, is highly diverse and provides computing at very high speeds [25] [26]. But the fact that it is highly non-uniform is viewed as a hindrance if not handled properly. There is an urgent need for easy and efficient Jungle Computing in scientific practice, by exploring a set of state-of-the-art application domains [26]. Thus, the need of an hour is a system which not only combines the features of grid, cloud and cluster computing but goes beyond it to incorporate efficient jungle computing, thus providing an easier and faster system.

VII. CONCLUSION

We discussed origin of cluster, grid and cloud computing and studied their architecture, characteristic features and discussed their current applications and fields of implementation.

Further we had an overview on trends of computing and glimpse of green sustainable computing which allowed us to create intent of developing a high performance distributed system which would meet the aim of green sustainable computing and would combine best features of all the available computing models, especially the most popular ones as per trends in computing. In the nutshell, we conclude that by extrapolating trends in high performance computing we draw the conclusions that parallel computing is the core mechanism by which computer performance can cope up with the predictions of Moore's law in the face of increasing influence of performance and the architecture of HPC will continue to develop at quick rates. Thus, it would be increasingly important to find paths to motivate scalable parallel programming without compromising with transportability. Such a challenge could be defeated by evolution of software systems and algorithms that support portability besides relaxing burden of program design and implementation. Table 1 enable us to compare and achieve this high performance distributed system using grid, cluster and cloud computing.

Table 1: High Performance Grid v/s Cluster v/s Cloud Computing

Category	Grid	Cluster	Cloud
Size	Large	Small to medium	Small to large
Resoruce type	Heterogeneous	Homogeneous	Heterogeneous
Initial Capital Cost	High	Very High	Very low
Typical ROI	Medium	Very High	High
Network type	Private Ethernet based	Private IB or proprietary	Public Internet Ethernet Based
Typical Hardware	Expensive	Very expensive- top of the line	Usually VMs atop of hardware
If I didn't know any better:	Faster workstations	supercomputers	Bunch of VMs
SLA requirement	High	Strict	Low
Security Requirement	High	Very low- but typically high	Low

REFERENCES

[1] Michael Hill, "*Distributed Computing: An Unstoppable Brute Force*", SANS Institute Info Sec Reading Room, pp 1-13, November 13, 2003.

[2] Manish Parashar and Craig A. Lee, *"Grid Computing: Introduction and Overview*", pp 1-6, 2005, Available at: www.researchgate.net

[3] R. Buyya, "*High Performance Cluster Computing: Architectures and Systems*", ISBN-10: 0130137847, Prentice Hall; 1 edition, 21 May 1999.

[4] Albert Y. Zomaya, "*Handbook of Nature-Inspired and Innovative Computing integrating classical models with emerging technologies*", EBook Springer, chapter 16, pp 521 – 530, 2006.

[5] Krishna Kant, Muthu kumar Murugan, and David Hung Chang Du, "*Energy Adaptive Computing for a Sustainable ICT Ecosystem"*,Design Technologies for Green and Sustainable Computing Systems, Springer, pp 59-91, 2013.

[6] Miettinen AP, Nurminen JK, "*Energy efficiency of mobile clients in cloud computing"*, ACM HotCloud'10 Proceedings of the 2nd USENIX conference on Hot topics in cloud computing, pp 4-11,2010.

[7] Chun B-G, Ihm S, Maniatis P, "*Clone: elastic execution between mobile device and cloud*", in the ACM proc of the sixth conference on Computer systems (EuroSys'11), pp 301-314, 2011.

[8] Curevo E, Balasubramanian A, Cho D-K, Wolman A, Saroiu S, Chandra R, Bahl P, "*Maui: making smart phones last longer with code offload*", ACM MobiSys'10, Proceedings of the 8th international conference on Mobile systems, applications, and services, pp 49-62, 2010

[9] Mayank Raj, Krishna Kant, Sajal K. Das, "*Energy Adaptive Mechanism for P2P File Sharing Protocols*", Parallel Processing Workshops: Euro-Par 2012, Springer Lecture Notes in Computer Science, Volume 7640, pp 89-99, 2013.

[10] Top500 Super Computer Report, Available at: www.top500.org.

[11] Volodymyr Kindratenko, Pedro Trancoso, "*Trends In High Performance Computing*", IEEE Computing, vol. 13 no. 3, pp. 92-95, 2011.

[12] I. Foster and C. Kesselman, "*Computational Grids: Blueprint for a new computing infrastructure*", ISBN:1-55860-475-8, ACM book,1998.

[13] Joshy Joseph, Craig Fellenstein, "*Grid computing*", report published by IBM Press on December 30, 2003.
[14] Christian Vecchiola, Suraj Pandey and Rajkumar Buyya, "*High-Performance Cloud Computing: A view of Scientific Applications*", International Symposium on Pervasive Systems, Algorithms, and Networks, IEEE Computer Society, pp 4-16,2009.
[15] Peter Mell, Timothy Grance, "*The NIST Definition of Cloud Computing*", National Institute of Standards and Technology Special Publication no. 800-145, 7 pages report, September 2011.
[16] Ye Xiaotao, LvAili, Zhao Lin, "*Research of High Performance Computing with Clouds*", ISBN 978-952-5726-10-7, pp 567-577, 2010.
[17] Cloud Computing Definition, National Institute of Standards and Technology, Version 15, available at:www.csrc.nist.gov.
[18] Jaliya Ekanayake and Geoffrey Fox, "*High Performance Parallel Computing with Clouds and Cloud Technologies*", 1st International Conference on Cloud Computing, October, 2010.
[19] High-Energy Grid Development, Available at:www.gridcafe.org.
[20] Michel Dayed, Jack Dongarra, "*High Performance Computing for Computational Science*", ISBN 978-3-642-19327-9, Springer, pp 120-121, 2010.
[21] Sunil Kr. Singh et.al, "*Performance Evaluation of Hybrid Reconfigurable Computing Architecture over Symmetrical FPGA*" Int. Journal of Embedded System & Appl., doi:10.5121/ijesa.2012.2312, Vol.2, No.3, pp 107-116, 2012.
[22] Adam L. Bazinet and Michael P. Cummings, "*The Lattice Project: A Grid Research and Production Environment Combining Multiple Grid Computing Models*"
[23] Chao He under guidance of Prof. Raj Jain, "*Performance Analysis based on two Leading Cloud Computing Platform: Google App Engine and Amazon Web Services*" available at http://www1.cse.wustl.edu/~jain/cse567-11/ftp/cloud/index.html.
[24] Mark Baker, Rajkumar Buyya, Ken Hawick, Health James and Hai Jin, "*Cluster Computing R&D in Australia*" Asian Technology Program, Australia.
[25] Jason Maassen, et al Towards jungle computing with Ibis/Constellation in Proceedings of the 2011 workshop on Dynamic distributed data-intensive applications, programming abstractions, and systems, ACM New York, ISBN 978-1-4503-0705-5.
[26] Jungle Computing: Distributed Supercomputing Beyond Clusters, Grids, and Clouds by Frank Seinstra et al in "*Grids, Clouds and Virtualization, Computer Communications and Networks*", ISBN 978-0-85729-048-9. Springer-Verlag London Limited, 2011, p.