International Journal of Innovative Technology and Exploring Engineering (IJITEE)
ISSN: 2278-3075 (Online), Volume-9 Issue-1, November 2019

Performance Enhancement of Cloud Computing: Methodology & Tool

Abey Jacob, Cyril Raj

Abstract— This paper describes the testing process employed for testing the in-house developed cloud by using the Google open source tool PerfKit and employing techniques for increasing the performance. Though new tools for testing cloud are emerging into the market, there are aspects which are suited for manual testing and some which can be speeded up using automatic testing tools for various cloud environments for Infrastructure, Platform and Software services. This paper brings out the techniques best suited to test different features of Cloud computing environment and to figure out the lacuna in performance of cloud services. The authors also try to bring out solutions to improve the performance of cloud (recommend) by using various tools to figure out the debugging and analysis process guidelines to follow while fine tuning the performance of private clouds.

Index Terms—Cloud Computing, Testing tools.

I. INTRODUCTION

High-Performance Computing (HPC) helps researchers, engineers and academicians to unravel advanced scientific and business problems that need terribly high figure capacity, storage, large bandwidth and highly optimized interconnectivity. Scientists and researchers has to anticipate for longer to get access from a pooled cluster resources or has to procure dearly-won physical systems (clusters) at own research centre.

Computing facility over cloud, a reliable method to access from a common set of reconfigurable highly intensify computing facilities (e.g., servers, storage, networks, applications, software, and services), which is comfortably accessible by on demand basis. For application and analysis teams, computing over the cloud can facilitate on the fly access to this stable, highly intensified computing resources and storage, by not procuring and retain the innovatory computing nodes. This private cloud is an associate in nursing IaaS Cloud that can facilitate virtual instruments and cluster on the fly for researchers and scientists. IaaS Web access portal links you with the GUI to various offerings for Creating, Destroying and saving the virtual instruments. It conjointly provides single-click cluster with MPI or HADOOP platform change thereon. If wish to try, it's just to pick out the required environment (i.e. MPI) from the given list and Machine Size (i.e. Small/Medium/Large), along with all the nodes of a cluster with the required duration for running the virtual cluster.

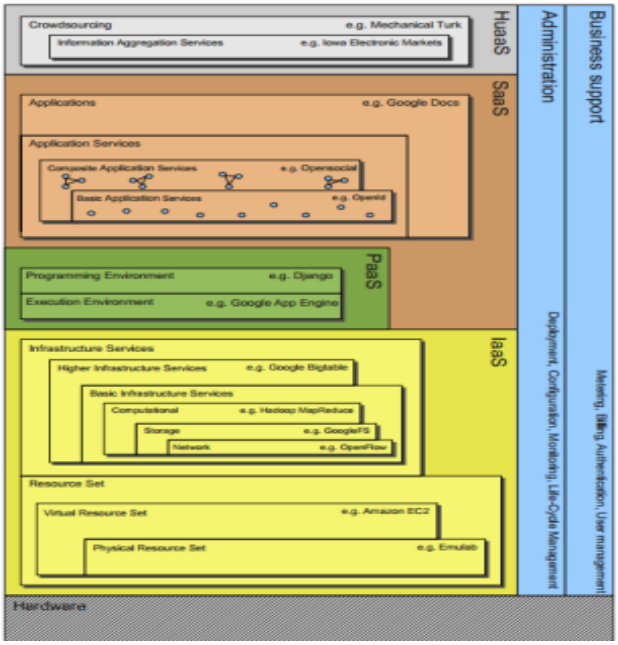
Revised Manuscript Received on November 30, 2019.
***** Correspondence Author
Abey Jacob*, Principal technical Officer with Centre for Development of Advanced computing (CDAC), Bangalore - India.
Dr. V cyril raj, Join Registrar (E&T) at Dr.M.G.R University, Chennai - India.

This cloud can facilitate for Software Services within a variety of the "application acceptance gateway" and alternative numerous usability depended "Problem finding setting Portals".

This web access hosted service facilitates online access for keying serial and parallel assignments/executables to HPC cloud computing setting that's obtainable as Infrastructure Service of cloud. This conjointly furnish observance for task standing and viewing/downloading the out comings/faults knowledge. Web access portal is made for the engineering application developers in the cloud for accessing the virtual machines and for executing their tasks, wherein not moving to the cumbersome of scripting on command lines.

II. CLOUD OFFERINGS

The three main types of offerings by our private Cloud are IaaS, PaaS and SaaS. The different layers of stack are as shown in fig1.



Individual components at each layer need to be tested, and then Integration Testing should be conducted to ensure the quality of the Cloud computing system. Specifically, with respect to Cloud computing environment, testing should be carried out for cloud-specific factors such as elasticity. The Standard Performance Evaluation Corporation has been investigating application workloads suitable for benchmarking. It lists four key metrics relevant in the context of cloud benchmarking based on the report published by Steering Committee, SPEC OSG Cloud Computing Working Group in 2012.

The main aspects to be tested in the Cloud Computing environment from the user's viewpoint include the following. The main aspects to be tested in Cloud Computing environment from the users view point include the following

- Elasticity
- Provisioning interval, or the lag between when a resource is requested and when it is actually available.
- Agility or the ability of the provider to track the needs of the workload.
- Scale-up or the improvement in response times with an increased amount of resources.
- Elastic speed-up or the development in turnout as a further resource is supplementary on the fly.
- Throughput
- Response time
- Variability

III. CLOUD STACK COMPONENTS

Cloud Stack is conceived in such a manner to cater to the needs of scientific environments and application ecosystems. The stack contains most reliable versions of most appropriated Cloud elements needed to make the non-public scientific cloud with Hypervisor, Middleware and a Portal. Images of CentOS, MPI & HADOOP along with facilities for submitting executables through a portal to virtual HPC clusters.

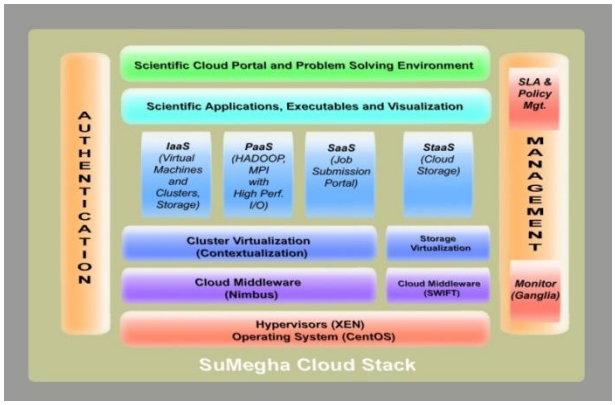


Figure 2: Cloud Stack

The architecture of our private cloud introduces the concept of two nodes, akin to Client-Server architecture: Virtual Machine Manager (VMM) Node & the Service Node (SN). The minimum requirement for installation is a desktop machine, where both VMM and Service Node can be installed. The VMM Node has to be setup initially and then the Service Node and finally configure both VMM and Service node.

This Cloud can have a multi-node installation. There shall be only one Service Node, which acts as a controller. VMM Nodes can be installed on multiple machines and configured with a Service Node. Cloud Lab kit then installs the Cloud components automatically in an interactive mode. The administrator should be aware of the network details (ex: Hostname, IP address, DNS, Gateway) which has to be provided during installation.

In order to program a Virtual Machine Manager (VMM) node and Service Node for Cloud installation, it should possess the following minimum requirements.

IV. VIRTUAL MACHINE MANAGER (VMM) NODE

Operating System	CentOS Version > 6.2
CPU	One or more 64-bit x86 CPU(s), 1.5 GHz or above, 2 GHz or faster multi-core CPU recommended
RAM	Minimum 4 GB
Disk Space	Minimum 60 GB; Minimum 2GB for /boot partition
Network	Internet Connectivity

SERVICE NODE

Operating System	CentOS Version > 6.2
RAM	4 GB
Disk Space	Minimum 100 GB
Network	Internet Connectivity
Software	Oracle JAVA 1.7, Python (2.6 – 3.0)

V. SCIENTIFIC CLOUD FOR HPC

High-Performance Computing (HPC) permits researchers and academicians to resolve sophisticated scientific problems, algorithms and business problems which need really large cipher capacity, immense storage, bigger system of measurement and least delayed network. Researchers and academicians has to stay prolonged to get connected with pooled cluster machines or have to procure these high-end devices at a higher cost for them.

Cloud computing is a practical replica for accessing the resources from a distributed pool of reconfigurable HPC hardware machines along with relevant software packages (e.g., servers, storing space, interconnectivity, running suits, software, and services) on the fly. For analysis teams, cloud computing is capable to present a simple one-shot opportunity to a stable, high-performance clusters and storing space, without procuring and maintain dearly-won hardware (clusters) at their organization. This cloud is associate in nursing IaaS that facilitates virtual machine and virtual cluster on the fly as and when needed. Scientific Cloud Web access portal has the Graphical User Interface, which simplifies the Creation, liquidation and preserving the virtual machines. It conjointly serves the platform to make a simple cluster with MPI or HADOOP thereon.

Highlights

- User-friendly GUI to the scientific Cloud.
- On the fly Creation and liquidation of Virtual Machine and Cluster, as and when required.
- Option to preserve the Virtual Machine.
- Executing the Virtual Machine Images.
- Facility to capture record the Virtual machines and Cluster.
- SSH-based key authenticated access to the Virtual machines and Cluster.

VI. FEATURES TO BE TESTED

This section describes the various items that required to be tested in our cloud computing. The network interconnectivity infrastructure level comprises of the routers, switches, gateways and firewalls. Suggesting and executing a suitable solution for "Benchmarking and Performance Monitoring of Scientific Cloud."

- Methodology for Performance Evaluation
- Finding Optimal Performance
- Benchmarking and Standardization

VII. TESTING METHODS & TOOLS

For each of the above features describe, what are the testing method and tools available? Many standard non-cloud testing tools can be re-used for performance, stress and benchmarking of Cloud Computing Environment.

AVAILABLE BENCHMARKING SUITS

- PerfKit Benchmarker
- LAMMPS (Large-scale Atomic/Molecular Massively Parallel Simulator)
- GEAGUL-C v1.0 (Grid Software- Enabling Applications for Grid Computing Using GLobus and C-Language)
- GOPAEAG-v1.0 (Globus and Object oriented Programming Approach to Enable Applications for Grid Computing)
- SPAGMOS-v1.0 (Software Probes for Assessment of Grid Middleware Overheads)

SUGGESTED SOLUTION: PERFKIT BENCHMARKER

PerfKit Benchmarker is associated in nursing benchmarking tool accustomed live and to equate the cloud performances. PerfKit Benchmarker is legalized underneath the Apache 2 legal terms & conditions. The goal is to form associate in nursing an active benchmarking structure which it constitutes. However, Cloud creators square measure developing applications, assessing Cloud options and nursing how to build use cases for each cloud.

PerfKit Benchmarker calculates the starting to finish time to provide resources within the cloud, additionally to reportage on the foremost customary metrics of top accomplishment, e.g.: Delay, amount of data passing, execution time, IOPS. PerfKit Benchmarker minimizes the complication in handling benchmarks on assisting cloud service vendors by combined and basic commands. It's conceived in such a manner to handle via servicer enabled command-line scripts.

PERFKIT INSTALLATION AND CONFIGURATION

PerfKit Benchmarker is a free to use publically available tool. For Benchmarking we have used a trial version for 60 days and it can be updated later can be downloaded from the GitHub. Before we run the Benchmarker we are supposed to create an account either in Google, Amazon, Microsoft Azure, Alicloud, Digital Ocean, Rackspace. For Benchmarking the account was created on Google Cloud.

After the installation and configuration of PerfKit, instances are created. User can either select the default images which are available or can create the image and upload. User can also select the machine size for the image. Machine-size can be from 1vCPU to 32vCPUs. With the selected Machine-size the memory of the image is also noted. Once the instances are created it is possible to know the detailed description of the instance.

Benchmarking is done by capturing and analyzing factors like how much CPU is utilized, Disk Bytes, type of Disk Operations, Network Bytes and Network Packets used. The screenshots based on these parameters are depicted for the instance created. PerfKit Benchmarker is an easy and simplified way for benchmarking the cloud.

When both Small and Medium VMs are Running

Iteration	25000	30000	35000	40000	45000	50000
VM(S)	0.26	0.23	0.27	0.3	0.37	0.4
VM(M)	0.3	0.14	0.01	0.01	0.01	0.01
Console	0.07	0.09	0.1	0.11	0.12	0.14

When both Medium and Large VMs are Running

Iterations	25000	30000	35000	40000	45000	50000
VM(M)	0.26	0.0	0.03	0.02	0.02	0.12
VM(L)	0.06	0.0	0.04	0.04	0.0	0.02
Console	0.07	0.09	0.1	0.11	0.11	0.14

When both Large and Small VMs are Running

Iteration	25000	30000	35000	40000	45000	50000
VM(L)	0.26	0.0	0.02	0.0	0.01	0.0
VM(S)	0.19	0.22	0.39	0.31	0.31	0.36
Console	0.07	0.07	0.11	0.11	0.12	0.14

When only Small Size VMs are Running

Iteration	25000	30000	35000	40000	45000	50000
VM(S)	0.25	0.34	0.21	0.42	0.38	0.52
VM(S)	0.25	0.25	0.28	0.36	0.35	0.36
Console	0.07	0.08	0.1	0.1	0.12	0.14

When only Medium Size VMs are Running

Iteration	25000	30000	35000	40000	45000	50000
VM(M)	0.04	0.02	0.02	0.02	0.0	0.01
VM(M)	0.09	0.13	0.12	0.22	0.11	0.19
Console	0.07	0.08	0.09	0.11	0.13	0.14

When only Large Size VMs are Running

Iteration	25000	30000	35000	40000	45000	50000
VM(L)	0.03	0.05	0.06	0.06	0.02	0.04
VM(L)	0.01	0.0	0.0	0.0	0.0	0.0
Console	0.06	0.08	0.09	0.10	0.12	0.14

The below Figure depicts the Screenshot of the CPU Utilization for the instance of 2vCPU

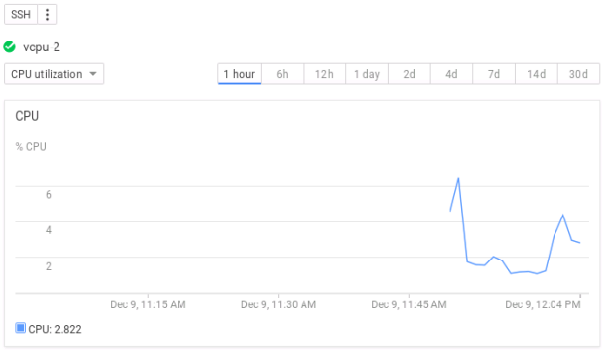


Figure: Screenshot of CPU Utilization for 2vCPU

The below depicts the Screenshot of the Network Packets for the instance of 2vCPU

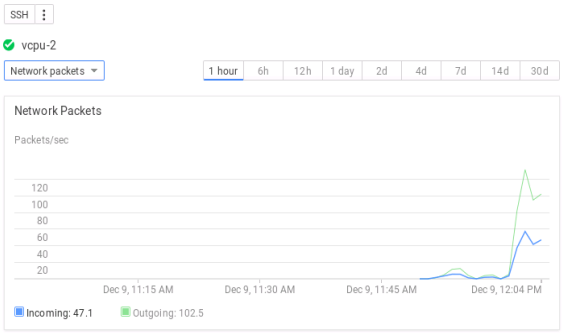


Figure: Screenshot of Network Packets for 2vCPU

The below Figure depicts the Screenshot of the CPU Utilization for the instance of 8vCPU

Figure: Screenshot of CPU Utilization for 8vCPU

The below Figure depicts the Screenshot of the Network Bytes for the instance of 8vCPU

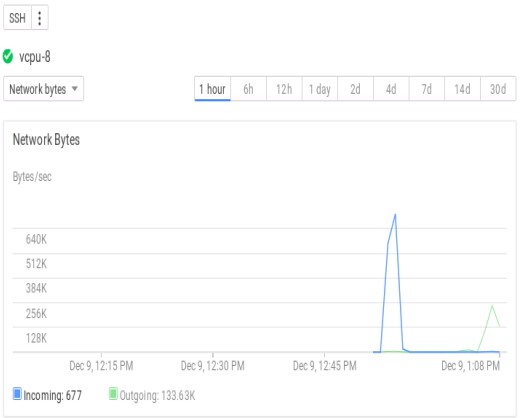


Figure: Screenshot of Network Bytes for instance of 8vCPU

DEBUGGING, VISUALIZER AND ANALYZING

This Debugger having inbuilt Visualizer and Analyzer is a sophisticated computing & adaptable simultaneous error detection ecosystem. This tool comprise of intelligent set of tools that can aid developers/testers in both accuracy and fulfilling the decided capabilities. Apart from traditional removal of bugs from parallel programming, it also helps for visualize error fixing. The behavioral analyzer figures out the network signaling obstructions and enhances the through put for both computation and in signal transmission to help users for accurately fixing it in parallel applications. This is a whole error identifying ecosystem for developers, who wants create multi threaded implementations by message-passing standards. Its software framework equip with a unwavering link to the message passing scripts coded on diverse graded communication and makes it freely access on the processor architecture.

This error fixing tool has:

- Conformity Debuggers – Parallel processing Debugger
- Behavioral Debuggers – on the fly Communication obstructive detector & Profile Visualizer

VIII. CONFORMITY DEBUGGING

Conformity debug in scripts includes finding and correcting the reasons (errors), which resulted in unpredicted performance of the script. Apart from the problems associated with sequential scripts based problems, the message passing (MP) parallel programs got counterfeit in fine tuning among allied related function and accuracy of messages/signals - in terms of both signal inputs & order and contention for shared resources if any.

Normal reason for imperfect Message Passing script response is interconnected multiplex scenario and identifying primary reason needs filter all other reasons. Provides facility for clubbing the events using with multiple events can be visualized as a single view and back to back debugging execution order for other associated duty within that faction. This type facility in debugger made it perfect for Message Passing scripts.

PARALLEL PROCESSING DEBUGGER

- Distributed and parallel application task debugging capability through a single point.
- Support to generate dynamic type function.
- Status display facility for communication & accomplishment aligned activity.
- Option for choosy debugger manger for jobs generated in dynamic.
- Each individual task has a symbolic, interactive source code debugger.
- Customized operation through pushbuttons & scripts.

BEHAVIORAL DEBUGGING

Fine calibration of performance for MP based parallel program is an exciting job. For fine tuning the application, this tool is useful in finding the communication and computation congestions. Computational congestion can be easily identified with profile Visualizer by hierarchically examine stack of data created by parallel program in a systematic manner.

Cost of communication is a reason for degradation in performance. Identifying and analyzing the bottleneck for communication needs, browsing through all the events of the parallel application. Tracking this with tools and visualizing the same mayn't be feasible in all multi thread programming. This tool can facilitate automatic identification of communication congestion with a two-pass trace collection and the filtration method point to the congested areas. This tool also helps in graphically visualize these areas and to correlate the reasons for degradation in performance to their source.

PROFILE VISUALIZER

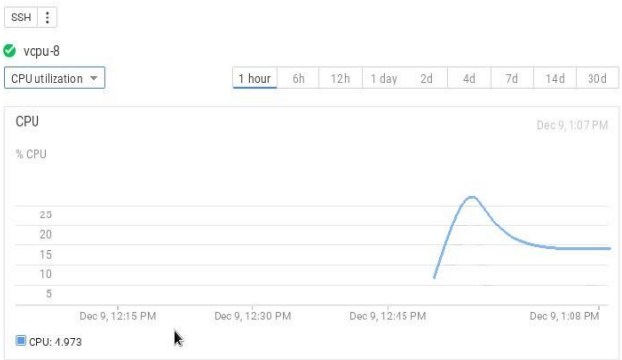
- Order of data illustration.
- Various Task execution analysis using Source Code View, Gantt chart and Task Graph.
- Provision for users for masking no relevant facts.

IX. COMMUNICATION BOTTLENECK DETECTOR

- Provision for identifying Communication congestion Automatically
- Facility to point out regions of congestion in hierarchically manner.
- Lower anxiety to application developer due to generation of selected task and selected regions.
- Details of each and every task's communication are visible through the Task Graph.
- Communication events in bottleneck region can be seen through the Space Time Diagram.
- Can view the Source Code corresponding to the entire task causing to the bottleneck region.
- Short time analysis by bucket generation conviction.
- Minimal trace files even for huge user programs.
- Event traces are collected by static instrumentation method.
- Separate trace sample selection and analysis

X. CONCLUSION

The paper discussed the testing methodology used for testing various aspects of IaaS, PaaS and SaaS. It describes the popular kinds of bugs that appear in the Cloud Ecosystem and the recommends some guidelines to reduce the number of bugs in the system.



Popular testing tool PerfKit used for performance testing and in-house developed debugger is used for debugging and visualizing the performance bottleneck. Based on the analysis results from this debugging tool, load on cloud can be distributed for a better performance. The resultant graph shows that CPU utilization and performances are increased drastically on multiple parallel load structure. The effort required to build new tools for testing the cloud .vs. the practical implication on testing an operational cloud is yet quite debatable.

ACKNOWLEDGMENT

We are extremely thankful to Executive Director – CDAC Bangalore, Group Heads, Senior Director Ms. Mangala & SUMEGHA/DIViA/PARALLEL DEBUGGER & CLOUD Development team & Validation team for their valuable guidance, support and encouragement.

REFERENCES

1. http://www.sumegha.in/
2. SuMegha Cloud Kit: Create Your Own Private Scientific Cloud by Vineeth S Arackal, Arunachalam B, Kalasagar, Sumit Kumar, Mangala N, Sarat Chandra Babu, Prahlada Rao, Sukeshini. https://pos.sissa.it /270/024/pdf
3. https://pdfs.semanticscholar.org/533f/d669ab5e0fa925dab55bca2d43c936f9a9b3.pdf?_ga=2.204569424.1593676386.1561018071-691968923.1558962402
4. Yatendra Singh Pundhir , "Cloud Computing Applications And Their Testing Methodology", Bookman International Journal of Software Engineering, Vol. 2 No. 1, Jan-Feb-Mar 2013
5. Michael Lynch, Thomas Cerqueus, Christina Thorpe, "Testing a cloud application: IBM SmartCloud inotes: methodologies and tools", International Workshop on Testing the Cloud, IISTA, July 2013, Switzerland.
6. William Jenkins, Sergiy Vilkomir, Puneet Sharma, "Framework for Testing Cloud Platforms and Infrastructures ", 2011 International Conference on Cloud and Service Computing
7. Sue Poremba, "Open Source Cloud Computing Testing Tools", Cloud Expo, 25 April 2013, http://www.cloudcomputingexpo.com/node/2625872
8. Alexandru Iosup,Simon Ostermann,Nezih Yigitbasi,Radu Prodan, Thomas Fahringer,Member and Dick Epema "Performance Analysis of Cloud Computing Services for Many-Tasks Scientific Computing" IEEE published in 2010
9. Dhivya, Padmaveni "Dynamic Resource Allocation using Virtual Machines for cloud computing Environment" IJREAT (International Journal of Research in Engineering & Advanced Technology), Volume 2,2014.
10. Rupali Shelke, Rakesh Rajani "Dynamic resource allocation in Cloud Computing" International Journal of Engineering Research & Technology (IJERT) Vol. 2 Issue 10, October – 2013
11. www.Google Perfkit Benchmarker
12. Pillutla Harikrishna ; A. Amuthan, "A survey of testing as a service in cloud computing", 2016 International Conference on Computer Communication and Informatics (ICCCI) 2016, Pages: 1 – 5
13. Wei Wang ; Ningjing Tian; Sunzhou Huang; Sen He; Abhijeet Srivastava; Mary Lou Soffa; Lori Pollock, "Testing Cloud Applications under Cloud-Uncertainty Performance Effects", IEEE 11th International Conference on Software Testing, Verification and Validation (ICST) 2018, Pages: 81 - 92
14. HongHui Li; XiuRu Li; Huan Wang; Jie Zhang; ZhouXian Jiang, "Research on Cloud Performance Testing Model", IEEE 19th International Symposium on High Assurance Systems Engineering (HASE) 2019, Pages: 179 - 183
15. Charles McDowell and David Helmbold. - Debugging Concurrent Programs. ACM Computing Surveys 21(4):593 { 622 December 1989.
16. Globus project 〈http://www.globus.org〉
17. Karl Czajkowski Ian Foster Nicholas Karonis Carl Kesselman Stuart Martin Warren Smith Steven Tuecke - A Resource Management Architecture for Metacomputing Systems - GRAM - Globus Resource Allocation and Manager.
18. http://clds.sdsc.edu/sites/clds.sdsc.edu/files/wbdb2012/papers/WBDB2012 Paper40Jost.pdf.
19. Net-dbx-G: A Web-Based Debugger of MPI Programs Over Grid Environments. www.cs.pitt.edu/~panickos/publications/conference/Net-dbx-G.pdf

20. R Hood G Jost - A debugger for computational grid applications - p2d2 - Heterogeneous Computing Workshop 2000.(HCW 2000)
21. N. Karonis B. Toonen and I. Foster - MPICH-G2: A Grid-Enabled Implementation of the Message Passing Interface - Journal of Parallel and Distributed Computing (JPDC) Vol. 63 No. 5 pp. 551-563 May 2003
22. MohanRam N Prahlada Rao B. B Mangala N Sridharan R Asvija B Shamjith K.V A method and system for Debugging and/or run time analyzing of heterogeneous computational grids and/or geographically distributed systems Indian Patent Office-02610/CHE/2007 November 09 2007.
23. Honghui Li, Xiuru Li, Huan Wang, Jie Zhang and Zhouxian Jiang: Research on Cloud Performance Testing Model", 19th IEEE International Symposium on High Assurance Systems Engineering (HASE), 2019.
24. What's Inside the Cloud? An Architectural Map of the Cloud Landscape, http://www.di.ufpe.br/~redis/intranet/bibliography/middleware/lenk-what-2009.pdf
25. B. Asvija, K. Kalaiselvan, M. Ateet, R. Sridharan, S.R.Krishnamurthy. "A Performance Based QoS Aware Resource Brokering Framework for the Grid", 2012 Ninth International Conference on Information Technology – New Generations, 2012
26. Cyril Raj, Abey Jacob: Performance Evaluation of SUMEGHA Cloud Computing Environment: Methodologies & Tool, International Journal of Innovative Technology and Exploring Engineering (IJITEE) ISSN: 2278-3075, Volume-8 Issue-12, October 2019.
27. Christoph Laaber. "Continuous software performance assessment: detecting performance problems of software libraries on every build", Proceedings of the 28th ACM SIGSOFT International Symposium on Software Testing and Analysis - ISSTA 2019, 2019
28. Sonam Srivastava, Sarv Pal Singh. "A Survey on Latency Reduction Approaches for Performance Optimization in Cloud Computing", 2016 Second International Conference on Computational Intelligence & Communication Technology (CICT), 2016
29. Akshay R. Rele, Sudhir N. Dhage. "Towards the performance measurements of private cloud", 2015 International Conference on Communication, Information & Computing Technology (ICCICT), 2015
30. Harikrishna Pillutla, A. Amuthan. "Recursive Self Organizing Maps and Software Defined Networking Cloud - A Survey", 2019 International Conference on Computer Communication and Informatics (ICCCI), 2019
31. S. Jagannatha, N. S. Shravan, S Kavya. "Cost performance analysis: Usage of resources in cloud using Markov-chain model", 2017 4th International Conference on Advanced Computing and Communication Systems (ICACCS), 2017
32. W. K. Daniel. "Challenges on privacy and reliability in cloud computing security", 2014 International Conference on Information Science, Electronics and Electrical Engineering, 2014
33. D.J. Miller. "Integrating Web service and grid enabling technologies to provide desktop access to high-performance cluster-based components for large-scale data services", 36th Annual Simulation Symposium 2003 SIMSYM-03, 2003
34. Alam, Md. Imran, Manjusha Pandey, and Siddharth S Rautaray. "A Comprehensive Survey on Cloud Computing", International Journal of Information Technology and Computer Science, 2015.
35. William Jenkins, Sergiy Vilkomir, Puneet Sharma, George Pirocanac. "Framework for testing cloud platforms and infrastructures", 2011 International Conference on Cloud and Service Computing, 2011
36. Mohamed Firdhous, Osman Ghazali, Suhaidi Hassan. "Trust Management in Cloud Computing: A Critical Review", International Journal on Advances in ICT for Emerging Regions (ICTer), 2012

AUTHORS PROFILE

ABEY JACOB had completed B.E (ECE) and M.Tech (CSE) and at present working as a Principal technical Officer with Centre for Development of Advanced computing (CDAC), Bangalore - India. He is currently pursuing PhD in cloud computing, at CSE Department of Dr.M.G.R University, Chennai - India. His areas of interest are Grid Computing, Cyber Forensic and Cryptography. He has published couple of papers in National and International Journals.

Dr. V CYRIL RAJ had completed M.E, PhD and presently working as Join Registrar (E&T) at Dr.M.G.R University, Chennai - India. Formerly he is the Dean (E&T) & HoD for Computer science department. He has guided more than 20 PhD students and his areas of interest are Bioinformatics Data Mining, Grid Computing, Cloud & Mobile Computing, Robotics etc. He had published a lot of papers in International and National level journals and also authored many books