

A LATEST EVALUATION OF THE PERFORMANCE FOR DISTRIBUTED COMPUTING SYSTEMS ON MONITORING

Mohd Muntjir¹, Mohammad Junedul Haque²

¹*College of Computers and Information Technology Taif university Taif Saudi Arabia*

²*College of Computers and Information Technology Taif university Taif Saudi Arabia*

Author Correspondence: College of Computers and Information Technology Taif University Saudi Arabia, muntjir.m@gmail.com, junedulhaq@gmail.com

Abstract: - At present distributed computing is one of the most exploited computing platforms with the emerging techniques like cloud computing. Latest information technologies such as distributed computing system have brought the incorporation for comprehensive enterprises faster to being an authenticity. This technology and principles unaccompanied, though, are not adequate to concern scalability, flexibility, and usability of incorporated enterprise information systems, without which the adjustment can disintegrate of its own credence. Therefore it is becoming essential to understand all features of the distributed computing; by setting up the hardware to applications for different software's on the distributed systems possibly the most significant factor is "monitoring service". Distributed Computing System (DCS) aims at attaining higher execution speed than the one obtainable with uni-processor system by exploiting the collaboration of multiple computing nodes interconnected in some fashion. Present paper focused on the importance of DCS and its major role in monitoring data, earlier DCS models which has been developed to monitor data along with its advantages and limitations. Finally paper focuses on cost optimization using DCS in general. Monitoring is an important tool for program visualization, debugging, testing, and development. Thus there is a need to develop the generic monitoring service to support all aspects of management in a distributed system.

Keywords: Distributed Computing; Publish Systems; Grid computing; Subscribe Systems; Monitoring

1. Introduction

Distributed Computing System (DCS) is a computing perception that refers to multiple computer systems working on a single problem. In DCS, a single problem is allocated into many parts, and different computers resolve each part. As far as the computers are connected, they can interconnect with each other to resolve the difficulty. If it finished appropriately, the computers execute like a single object. A distributed system incorporating networked heterogeneous processors requires well-organized task-to-processor project to accomplish rapid alternate time [1]. One of the hopes for the Distributed Computing Systems is that enhancing appliances to the communication system would deliver sufficient support delivered to the computer software

system to permit it to function in an effective method. It has typically engaged the problems connected with the inter-connection of hundreds of computer systems [2]. The growth in computer science have caused numerous institutions and companies to assemble massive quantity of data, from the projects pertaining to Astronomy, Climate Modelling, High Energy Physics, Bioinformatics, and Earth Observation and many others [3]. The increasing apprehension among organizations, their administration and even to its end users is the level of observance of systems with the amendable policies of IT in managing important data. Supervising and the observance of the systems is critical in IT due to the processes concerned with process ,data sharing and collection are connected in most of the contemporary communications such as social networks, banking,economics, communication and amusement. It has been enhanced in legislative laws such as US Health Insurance Portability, Accountability Act[1999], and Sarbanes-Oxley Act (2002),in order to require the authentic contract [4].This paper is ordered as the subsequently. In Part I it has emphasized the importance of Distributed Computing and its role in assessing the data. Part II would certainly evaluate the different types of Distributed Computing Systems which have been developed to observe information and then its advantages and restrictions. Part III would concentrates on charge optimization using Distributed Computing Systems, and at last the actual papers would concludes along with future recommendations with essential tips and conclusion with summary.

2. Overview about the Distributed Computing Systems (DCS)

Distributed computing refers to computing that includes multiple loosely coupled processors working together to solve an overall problem [6]. Distributed computing provides a natural method to solve complex data and computation intensive problems that arise in power systems analysis and control. Distributed Computing System (DCS) aims at attaining higher execution speed than the one with uniprocessor system by exploiting the collaboration of multiple computing nodes interconnected in a framework. The best possible speed up will be obviously obtained if the several partitions of the given computational task can run independently in parallel. In DCS a single large problem is fragmented into multiple small problems, processes them and combines all the solutions into one solution for the problem [5]. The purpose of the DCS is to coordinate the use of shared resources and also provide communication services to users. In DCS various set of processors handles multiple tasks and execution of all the requested tasks and modules [7].



Figure 1: Distributed Computing

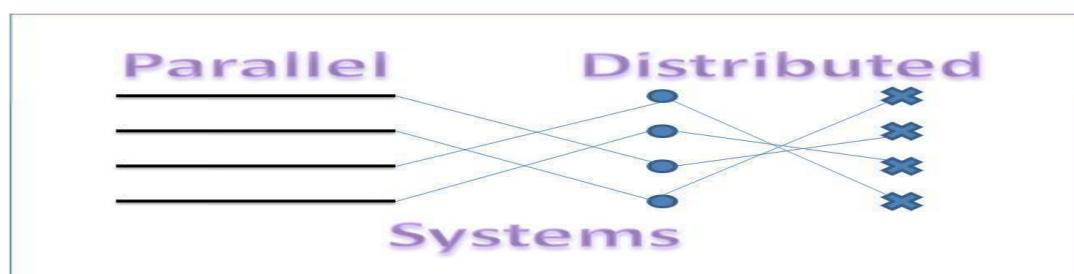


Figure 2: Difference between Distributed and Parallel Computing

In earlier days, when CPU power and/or memory were limited the main driving force for Distributed System (DS) was to develop the system with larger computing power and as well as significant amount of (shared) memory to tackle much computer-intensive task (mainly scientific and engineering problems). This also led to

cluster and supercomputer architecture. With the advent of hardware technologies, not only the CPU computing (Moore's law still driving the increase rate for CPU power), but also other hardware like memory, GPU computing and cluster computing, is attaining pet scale computing with a modest size cluster [10]. Dong has designed taxonomy for the classification of scheduling algorithms in distributed systems [8]. Furthermore, Dong has broadly categorized scheduling algorithms as static vs. dynamic, optimal vs. suboptimal, local vs. global, application centric vs. resource centric, and distributed vs. centralized [9]. Additionally, internet and extra technologies such as mobile computing, has beguninnovativenotions which comprises real time computing allocatedclassification, physically distributed computing, cloud computing etc. and this has concerned both academia as well as industry [11; 12; 13]. While associated to integratedobserving, a computer monitoring system is interesting than observing sequential computing system due to:

(1) Numerous asynchronous processes: Asynchronous parallel process cannot be reproduced and has changeable behaviour repetitionsoccurring from reasonableenvironments in real-time distributed computing system, and visualizing the implementation sequence of the system's various processes in unachievable.

(2) Crucial time constraints: The correctness of the real-time Distributed Computing System is supported on its performance pattern in orientation with the time phase. Integrally in convinced domains like those of a chemical plant or weapon system, the real-time Distributed Computing Systems are more comparable to implementation of the real world and the correctnessvaries on the output of the processors involved, comprehensive of the time constrictionsforced by the processes in the real domain. Though any modificationsperceived in the monitoring activity is insupportable in real time distributed background.

(3) Interruption in significant communication: The physicallyextensive nodes of real-time Distributed Computing Systemoutcomes in a extensivewait in interactionprominent to inappropriate synchronization thereby deliveringimproper location and timing throughout the world. The real-time Distributed Computing Systemimplementation is constrained by time imprecisions, untrustworthy system performance and indecisiveness, thatacquireexperiments in monitoring the implementationconstructing it almostincredible for situations across the world. The two essentials, universal state and timing, that are illustrative of the system performance during implementation, cannot be observed. Additionally, thewholepresent monitoring systems are expressed to distribute the properties in computing system [14; 15; 16; 17; 18; 19; 20; 21]. The subsequentsegment would investigate some of the representations and methodologies used for observing and cost optimization simulations in Distributed Computing Systems.

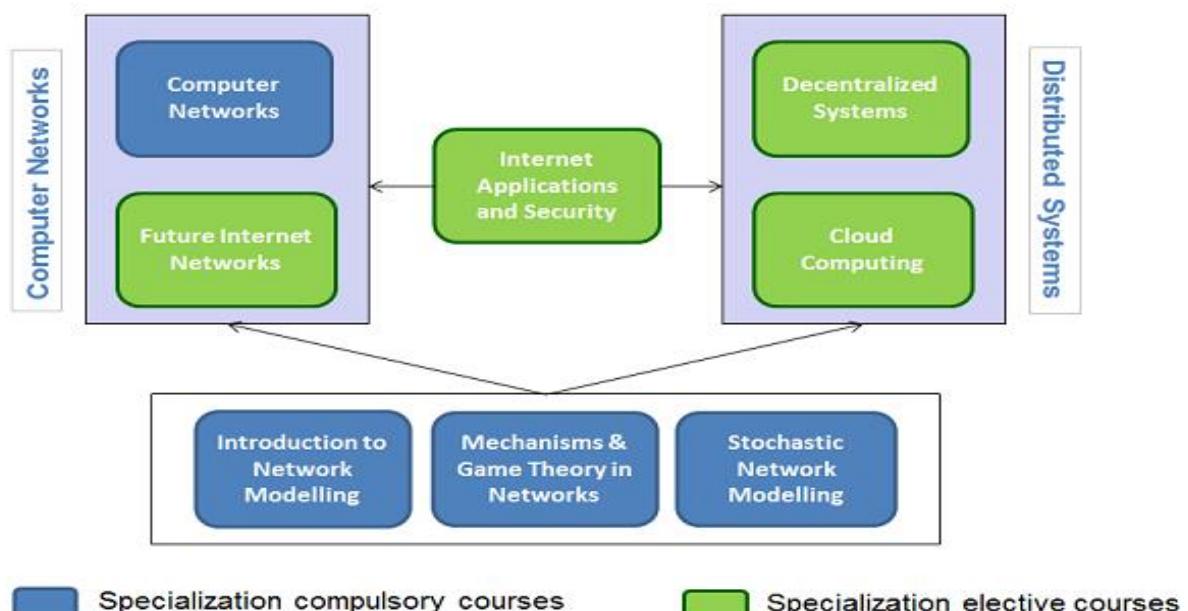


Figure 3:Distributed Computing With Computer Networks

3. Some Related Works about the Monitoring on DCS

Over the past years, a lot of methods to distributed computing system techniques are actually formulated. These compriseRemote Procedure Calls, Socket Programming, object-oriented Distributed Computing Environment, Message-Oriented Middleware, and Java Remote Method Invocation, etc.[22] and adiversity of monitoring and detection services exists,VMware,INCA, vCloud, ZenOSS, MonALISA, etc. As far as concern, there are only

few published results that have been studied quantitatively for the enactment and monitoring of present distributed system. [25] The prospective and reserves of Globus Toolkit's Monitoring and Discovery Service was studied by Aloisio restraining their explanations by testing only on GIIS [23], although Laszewski, Dinda and Plale (2002) explored more on the positive and negative aspects of structuring a Grid Information Service on relational and hierarchical representations [24]. The assessment in these demonstrations is concentrating on the elements in the notions and facilities. ; An artificial assignment was identical by Plale [2003] in his analysis, which was inventeddeveloping a prearrangedessential operators such as updates queries, in contradiction of the hypothetical data attained from two different databases such as MySQL and Xindice, there after calculate the consequence. Though, there are no evidencefacilities, which retain either of the two methodologies. Ganglia are a scalable distributed monitoring system for excessiveimplementation computing systems such as grids with clusters [14; 15; 16; 17; 18; 19; 20; 21] is. The proficienciesacquiredacross real world implementations on organizations of unreliablearrangements, scales, and objectpresentationareas with more than five hundred clusters across the world were pertained to cultivate, design, appliance and estimate Ganglia [26].

Relatedly, additional studies conducted on the monitoring of cultures are Supermom [27] which is a classified cluster monitoring system that engages a statically composedorder of elaborateinfluences for the gathering and composing of cluster data acquired through client kernel segmentsimplemented on distinct nodes. However another statistically assembled hierarchy, which is of the relational database to acquire index and gather cluster data, was established [28]. Additional server/client cluster monitoring system is the PARAMON thatretains the server to send anindomitable amount of node databases, and the clients acquire the information from the servers and decode the information obtained [29]. The popular client server used in profitablecharacteristics is the Big Brother for data distribution in various systems and also to observe them. Wang hassuggested a combined monitoring framework for dispersedbackground with heterogonous monitoring system using a case study based on Enterprise Service Bus [30]. The background minimised the costs of software development, preservation and enlarged the scalability,robustness, flexibility, and availability has to reviewed several distributed computing system using ATLAS grid resources [31]. The monitoring tools joined in terms of data communication and data visualization delivered by a backend in the lightweight data-interchange format of Javascript Object Notation that is based on the JavaScript library jQuery and plug ins. Conceptionstructures are the xbrowse background, and the hBrowse framework, whereas the structure for strategizing libraries are High charts, flot, graph tools, and Google charts. Rennese,Birman,Vogels and Astrolabe had generated to resolvegeneralassortment of ascendantleimpediments. The technique used a peer-to-peer protocol based on the SQL Query language for combination [32].

Zhang, Freschl, &Schopf [33] demonstrated the scalability and presentation of three monitoring and information services:

- (1) The Globus Monitoring and Discovery Service (MDS) [34];
- (2) Monitoring & Discovery System (MDS) [35], the Relational-Grid Monitoring Architecture (R-GMA) DataGrid [36] used in the European DataGrid [37], Hawkeye [38]. Everysystem was used in construction or near-production grid test layers. The resultspresented that all the three services presented good scalability and recognized higher settlements for caching or pre-fetching the data. Furthermore, the systems were also requisite to have principalconstituents at well-connected sites because of high load perceived in every system. Although, in order to observe the data, numerous algorithms have been recommended. Hence, the study suggested a resolution based on the insolvability of monitoring asubjective linear-time temporal logic formula on partially ordered logs and also to monitor the usage of data in synchronized distributed systems using a case study. In this study,Authorsilluminated the achievability and benefits of monitoring the usage of complexdata. Furthermore few studies have inspected the pricing matters in distributive cloud computing framework. Resource utilization based pricing is exceptionallycomplex to the method in which the system is proposed, improved, organized, observed, and restrained. Accordingly, in the study accompanied by Wang [30], it was recommended that decoupling users from cloud providers with anestimatingarrangement would act as a connection in such circumstances. Cloud computing has mainlyimproved the setting of system proposal and optimization using Amazon EC2 cloud service

4. Conclusion

Monitoring is the systematic and routine collection of information from projects and programmers for program visualisation, debugging, testing, and development. It's very needed to establish general monitoring services to handle all facets of management in a distributed computing system. Although many researches has developed the several tools for monitoring the structures, still they are not practical for monitoring real-time distributed computing systems because of monitor's insensitivelandscape. The fundamental objective of Distributed Computing System (DCS) is to increase presentation by concerning users and IT possessions in a cost-effective and reliable method. It also confirms accountability lenience and assists resource convenience in the event that one of the components fails. The particular system as well as OS assist may very well be charge too high to be able to end-users. However; it is not an easy level to programming to end-users. Grid as well as cloud

computing processing shows up offering one of the three HPC types and many study has been carried out inside each classification. The main goal of upcoming distributed computing systems is to reduce the functioning expense of data centres and increase the flexibility in compliance, breakdown, and agile recovery.

REFERENCES

1. D. Milojicic, V. Kalogeraki, R. Lukose, Nagaraja, K., et al, Peer to peer computing, HP laboratories Palo Alto, *Technical Report HPL-2002-57*, 2002.
2. Sarbanes-Oxley Act, "107th Congress", *Public Law*, pp. 107-204, 2002.
3. J. Nabrzyski, J. Schopf, J. Węglarz, 2004, *Grid Resource Management: State of the Art and Future Trends*, Kluwer Academic Publishers, pp. 15-23
4. The Health Insurance Portability and Accountability Act (HIPAA), "104th Congress", *Public Law*, pp. 104-191, 1996.
5. A. Keren, A. Barak, "opportunity cost algorithms for reduction of I/O and inter process communication overhead in a computing cluster", *IEEE transaction on parallel and distributed systems*, Vol. 14, No. 1), pp. 39-50, 2003.
6. Directive 95/46/EC of the European Parliament and of the Council, "On the Protection of Individuals with Regard to the Processing of Personal Data and on the Free Movement of Such Data", *Official Journal L 281*, 1995.
7. B. Dong, Q. Zheng, J. Yang, H. Li, M. Qiao, M, An e-learning ecosystem based on cloud computing infrastructure, In: *9th IEEE International Conference on Advanced Learning Technologies*, pp. 125–127, 2009.
8. H. Hussain, S.R. Malik, R. Hameedb, S.U. Khanb, G. Bickler et al. "A survey on resource allocation in high performance distributed computing systems", *Parallel Computing*, Vol. 39, pp. 709–736, 2013.
9. TeraGrid, TeraGrid Archives, <https://www.xsede.org/tg-archives>, 2010.
10. N.F. Khan, K. Govil, "Cost Optimization Technique of Task Allocation in Heterogeneous Distributed Computing System", *Int. J. Advanced Networking and Applications*, Vol. 5, No. 3, pp. 1913-1916, 2013.
11. Y. Wei, M. Brian Blake, "Service-Oriented Computing and Cloud Computing: Challenges and Opportunities", *IEEE Internet Computing*, Vol. 14, No. 6, pp. 72-75, 2010..
12. Microsoft Azure, The cloud for modern business, <http://azure.microsoft.com/en-us/>, 2014.
13. Amazon cloud service. (2014) *Amazon cloud service*, Available at: <http://aws.amazon.com/ec2/>
14. H. Garcia-Molina, F. Germano, W. H. Kohler, "Debugging a Distributed Computing System", *IEEE Trans. Software Engineering*, Vol. 2, pp. 210-219, 1984.
15. Haban, D. DTM - A Method for Testing Distributed Systems, *Proc. 6th Symp. Re- liability in Distributed Software and Data- base Systems*, CS Press, Los Alamitos, Calif, Order No. 737, pp. 66-73, 1987.
16. R.J. LeBlanc, A.D. Robbins, Event- Driven Monitoring of Distributed Programs, *Proc. 5th Int'l Conf Distributed Computer Systems*, pp. 515-522, 1985.
17. B. Plattner, J. Nievergelt, "Monitoring Program Execution: A Survey", *IEEE Trans. on Computers*, pp. 76-93, 1981.
18. B. Plattner, "Real-Time Execution Monitoring", *IEEE Trans. Software Engi- neering*, Vol. 6, pp. 756-764, 1984.
19. R. Snodgrass, "A Relational Approach to Monitoring Complex Systems", *ACM Trans. Computer Systems*, Vol. 6, No. 2, pp. 157-196, 1988.
20. H. Tokuda, M. Kotera, C. W. Mercer, "A Real-Time Monitor for a Distributed Real- Time Operating System", *ACM SIGPLAN Notices*, Vol. 24, No.1, pp. 68- 77, 1989.
21. M. Van Steen, A. Tanenbaum, *DistributedSystems: Principles and Paradigms*. Englewood Cliffs, NJ: Prentice Hall, 2002.
22. G. Aloisio, M. Cafaro, I. Epicoco, S. Fiore, S. "Analysis of the Globus Toolkit Grid Information Service", <http://www.gridlab.org/Resources/Deliverables/D10.1.pdf>, n.d.
23. B. Plale, P. Dinda, G. Laszewski, Key Concepts and Services of a Grid Information Service, *ISCA 15th International Parallel and Distributed Computing Systems (PDCS)*, 2002.
24. B. Plale, Whitepaper on Synthetic Workload for Grid Information Services/Registries, *Data Workshop 2003* held in conjunction with *Globus World 2003*, San Diego, 2003.
25. M.L. Massie, B.N. Chun, D.E. "Culler, "The ganglia distributed monitoring system: Design, implementation, and experience", *Parallel Computing*, vol. 30, no. 5-6, pp. 817-840, 2004.

26. M. Sottile, R. Minnich, Supermon: a high-speed cluster monitoring system, In: *Proceedings of Cluster*, 2002.
27. E. Anderson, D. Patterson, Extensible, scalable monitoring for clusters of computers, In: *Proceedings of the 11th Systems Administration Conference*, 1997.
28. R. Buyya, ‘Parmon: a portable and scalable monitoring system for clusters’, *Software—Practice and Experience*, Vol. 30, No. 7, pp. 723–739, 2000.
29. X. Wang, H. Wang, Y. Wang, ‘A Unified Monitoring Framework for Distributed Environment’, *Intelligent Information Management*, Vol. 2, pp. 398-405, 2010.
30. J. Schovancova, “ATLAS Distributed Computing Monitoring tools after full 2 years of LHC data taking”, *Journal of Physics: Conference Series*, pp. 396, 2012.
31. R.V. Renesse, K.P. Birman, W. Vogels, W. Astrolabe: *A Robust and Scalable Technology For Distributed System Monitoring, Management, and Data Mining*, Department of Computer Science, Cornell University, Ithaca, n.d.
32. A. Zhang, H. Freschl, P. Schopf, “A Performance Study of Monitoring and Information Services for Distributed Systems”, <http://arxiv.org/ftp/cs/papers/0304/0304015.pdf>, n.d.
33. K. Czajkowski, S. Fitzgerald, I. Foster, C. Kesselman, Grid Information Services for Distributed Resource Sharing, In: *Proc. 10th IEEE International Symposium on High Performance Distributed Computing (HPDC-10)*, IEEE Press, 2001.
34. Monitoring & Discovery System (MDS), “GT Information Services: Monitoring & Discovery System (MDS)”, <http://toolkit.globus.org/toolkit/mds/>, 2001.
35. Data Grid, Data Grid Information and Monitoring Services Architecture: Design, Requirements and Evaluation Criteria, *Technical Report*, 2002.
36. Data Grid, “Data Grid Tutorial”, <http://eu-datagrid.web.cern.ch/eu-datagrid/Tutorial/tutorial.htm>, 2013.
37. Hawkeye, “Hawkeye”, <http://www.cs.wisc.edu/condor/hawkeye>, 2013.
38. M. Litzkow, M. Livny, M. Mutka, M. Condor - A Hunter of Idle Workstations, In: *Proceedings of the 8th International Conference of Distributed Computing Systems*, pp. 104-111, 1988.

A Brief Author Biography

1st Mohd Muntjir has received the B.Sc. in Mathematics from Choudhary Charan Singh University Meerut U.P. in 2002 and M.C.A. degree in Computer Science from Hemawati Nandan Bahuguna University Srinagar Garhwal Uttarakhand in 2007. Since 2011, he is working with College of Computers and Information Technology Taif University Saudi Arabia. His research interests are DBMS, Cloud Computing, Image Processing, Sensor Networks, E-Commerce and Multimedia Technology.

2nd Mohd Junedul Haque has received M.tech. Degree in Computer Science from Indraprastha University New Delhi. Since 2011, he is working with College of Computers and Information Technology Taif University Saudi Arabia. His research interests are Image Processing, DBMS, Multimedia Technology and Cloud Computing.