International Journal of Application or Innovation in Engineering & Management (IJAIEM)
Web Site: www.ijaiem.org Email: editor@ijaiem.org
Volume 3, Issue 7, July 2014 **ISSN 2319 - 4847**

Management of data intensive application workflow in cloud computing

Jaswinder Kaur[1], Sahil Vashist[2]

[1]Department of CSE, CEC, Landran(Pb),India

[2]Department of CSE, CEC, Landran(Pb),India

ABSTRACT

Cloud computing means different things to different people i.e. it is the one stop shop which provides scalability, pay-per-use utility model and virtualization. In this work we propose next generation cloud deployment model which are best fit for data intensive applications. In our proposal we are working for strength of application workflow in cloud computing and create the required network and computing resources provisioning and job allocation.

Keywords: workflow, scheduling, deployment.

1. INTRODUCTION

Cloud computing provides the web services to the user through internet connection and by using hardware and software in the data centers. This statement provides the new paradigm, in which the infrastructure of cloud computing and software applications are moved towards the data center and it is accessible by using internet connection. Cloud computing data center consists of distributed and parallel system which is collection of virtualized and interconnected computers. These computer systems are dynamically presented and provisioned as computing resources based on SLA (Service Level Agreements). The above statement puts cloud computing into market oriented perspective and stresses the economic nature of this phenomenon. Cloud's aspire to control the data centers of next generation by developing them as a virtual services of a network (user-interface, application logic, hardware, database) which enable users to deploy and access applications on demand at competitive costs depending upon users Quality of Service (QoS) requirements from anywhere in the world. It provides important advantage to IT companies by releasing them from the lower level tasks of setting up basic physical components (servers) and software applications and thus enabling them to focus on creation and innovation of business value for their services [1].

2. DATA INTENSIVE APPLICATION WORKFLOW

In cloud computing, the data-intensive computing environment consists of applications that analyze, produce or manipulate data in the range of hundreds of MB (megabytes) to PB (petabytes). A data-intensive application workflow deals with the high workloads of data to control than its computations of the data. The another meaning of data intensive deals with the transferring the huge amount of data but computational part deals with the processing of the tasks. The transfer of data consumes more time, and also store that data, than the processing of the data. For characterizing the difference between the data-intensive and computer-intensive one aspect is used which is the CCR (Computation to Communication Ratio). The applications with the lower values of this ratio are data intensive applications in nature [2].

2.1. Components of workflow

The architecture of the workflow management system includes component that control data with the processing tasks. The run time border provides the functionality of design the execution of tasks and build time is used for functionality of design to define tasks. At run time, various components are used for processing the tasks and data equally, which is different from the previous model. The scheduler does not control the data but manages the tasks. The scheduler is the main component which handles the scheduling. The scheduler always providing the separate scheduling policy for data transfer tasks [3].

2.2. Scientific workflow applications

Scientific applications like earthquake science, astronomy, gravitational-wave and others have embraced workflow activities to do large scale science. Workflow enables researchers to collaboratively manage, design and obtain that follow the thousands of steps, access the petabytes of data and generate some amount of intermediate and finalize data products.

We know that every lifecycle consists of number of phases. The workflow lifecycle is also consists of number of generation phases in which firstly analysis is defined, then the workflow planning phase begins where the needed resources are selected, the execution phase begins in which actual computations are performed and after getting result it is stored. During this workflow lifecycle, particular input data and the components are needed. During this process, the data need to be stage-in and stage-out of the computational resources. As results are produced, they are combined with the metadata and information so that they can be processed and shared with other collaborators [4].

3. RELATED WORK

Task scheduling is very important to scientific workflows and task scheduling is challenge problems too. It has been research before in conventional distributed computing systems. Reference [5] is a scheduler in the Grid that guarantees that task scheduling activities can be queued, monitored, programmed and managed in a fault tolerant manner. Reference [6] proposed a task scheduling strategy for urgent computing environments to guarantee the data's robustness. Reference [7] proposed an energy-aware strategy for task scheduling in RAID structured storage systems. Reference [8] studies multicore computational accelerators and the MapReduce programming model for high performance computing at scale in cloud computing. They evaluated system design alternatives and capabilities aware task scheduling for large-scale data processing on accelerator-based circulated systems. They improves the MapReduce programming model with runtime support for utilizing multiple types of computational accelerators via runtime workload adaptation and for adaptively mapping MapReduce workloads to accelerators in virtualized execution environments. but, none of them focuses on falling the processing cost and transmitting time between data centers on the Internet. because cloud computing has become more and more important, new data managing systems have intended, such as Google's GFS (Google File System) and Hadoop. Their data covered in the infrastructures and the users can't control them. The GFS is calculated essentially for Web search applications. Some researchers are based on cloud computing. The Cumulus project [9] introduced scientific cloud architecture for a data centre. And the Nimbus [10] toolkit can directly turn a cluster into a cloud and it has already been used to build a cloud for scientific applications. Within a undersized cluster, data movement is not a big difficulty, because there are high-speed connections between nodes, that is, the Ethernet and the processing time is not longer. However, the systematic cloud workflow system is distributed applications which need to be executed across several data centers on the internet. In recent studies, Reference [11] from the cost aspect studied the compute-intensive and data-intensive application. They formulate a non-liner programming model to minimize the data retrieval and executing cost of data-intensive workflows in clouds. Reference [12] investigated the effectiveness of rescheduling using cloud resources to increase the reliability of job conclusion. Particularly, schedules are initially generated using grid resources while cloud resources are used only for rescheduling to deal with delays in job conclusion. A job in their study refers to a bag-of-tasks application that consists of a large number of independent tasks; this job model is common in many science and engineering applications. They have devised a novel rescheduling method, called rescheduling using clouds for consistent completion and applied it to three well-known existing heuristics. Reference [13] proposed matrix based k-means clustering strategy to reduce the data movement in cloud computing. Though, the falling of data movement and cost do not mean that the processing cost and transmitting time reduce. In this work, we attempt to schedule the application data based on PSO algorithm in order to reduce the data transmitting time and process cost. Reference [14] study the deployment selection challenge from two different and usually conflicting angles, that is from the user's and the system provider's perspective. Users want to optimize the completion of their specific requests without worrying about the consequences for the whole system. The provider's purpose however is to optimize the system throughput and allow a fair usage of the resources, or a tradition mode as distinct by the decision makers. Whereas the users are most expected pursuing the same strategy for each request, the system dependable may face a active environment, including changing necessities, varying usage patterns and changing decisions in terms of business objectives. To address this issue, they propose a multi-objective optimization framework for selecting distributed deployments in a heterogeneous environment based on Genetic Algorithm (GA). In fact, task assignment has been found to be NP-complete [15]. Since task assignment is NP-Complete problem, Genetic Algorithm (GA) has been used for task assignment [16].

4. SYSTEM MODEL

Our Framework treat Workflow as to a number of requests coming towards cloud for processing with one result. Therefore we can say if more than one input comes they are connected to a graph in such a way that it seems there is no loss of time or cost due to multiple inputs. This can be represented in a graph theory. Therefore this allows model to allow as a one output. In figure.1 we can notice that t_1 is being allotted to VM_1 as workflow application defined by ten tasks which are represented as nodes. Since these tasks depend upon each other, therefore dependencies of tasks are shown in this example as an arrow. To execute this type of tasks our framework uses a set of resources (VMs) on demand. But in this case the transfer time of data between VMs are playing an important role in keeping the workflow in a optimal level. In this framework we draw a similar graph of virtual machine as the workflow to map the existing flow on the basis of transfer time between these virtual machine. For example we draw a matrix which shows the transfer time between each virtual machine and with this we can evaluate the right virtual machine configuration combination.

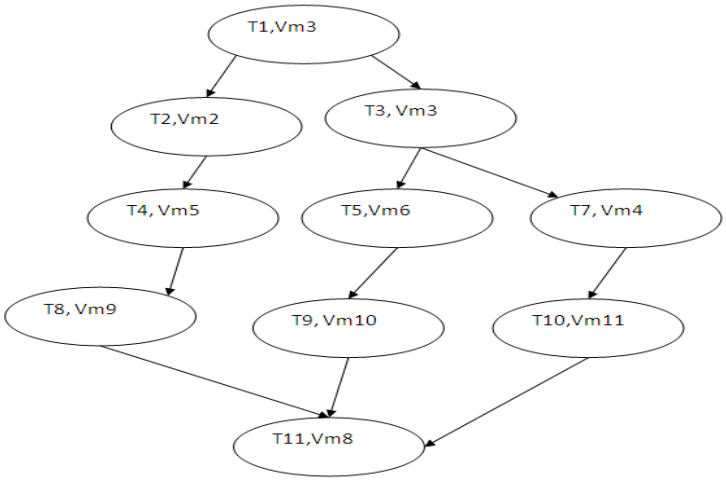

Figure 1: A view of a workflow

4.1.Algorithm for workflow

1. Initialize request position vector randomly as the request dimension equal to the size of the special tasks.
2. Calculate each particle's fitness value as in divide it into different tasks.
3. If one one tasks depends upon other link it directly.
4. Selecting the best particle from all the particle as the best .
5. Update all the vms with latency between them.
6. map the Vms to the taks as per latency
7. Execute the workload to vms

5. SIMULATION AND RESULTS

In this section we are discussing the simulation results in which the setup is done by using cloud simulator. In the figure below we can see that the over a workflow of 100 jobs TT based is able to perform 83 percent of jobs under desired response time which non-TT based approach handled 63 percent of jobs.

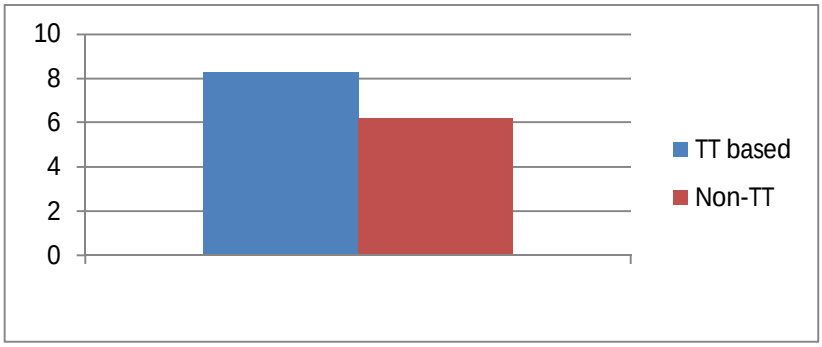

Figure: 2 Comparisons of Approaches

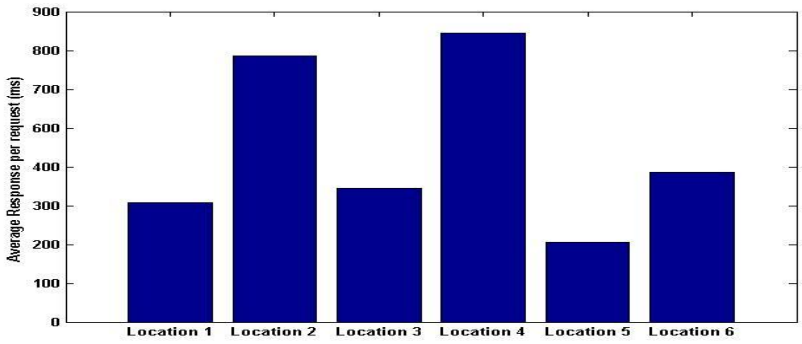

Fig.3 .Results

In Figure 3: we can see that the six location in which vms are located and the response time between the user and the vm therefore these response time is taken into consideration for handling the requests and linking tasks to its.

6. CONCLUSION AND FUTURE WORK

In order to make full use of the resources these different requests related to a single workflow needs to be handled. In this work we have find the dependability of the workflow to each other and linked these workflow to the total time taken between virtual machines. In future we will try to link these workflow management to the cost of the virtual machine and the requests coming as workflow.

REFERENCES

[1.] A. Beloglazov, J. Abawajy, R.Buyya, "Energy-aware resource allocation heuristics for efficient management of data centers for cloud computing", Future Generation Computer Systems, pp. 755-768, 2012.
[2.] R. Buyya, S. Pandey, "Data Intensive Distributed computing: Challenges and solutions for large-scale information system", Information science reference, 2012.
[3.] S. Pandey, "Scheduling and Management of Data Intensive application workflows in Grid and cloud computing environments", Dec. 2010.
[4.] E. Deelman, A. Chervenak, "Data management challenges of data intensive scientific workflow", 2008.
[5.] T. Kosar, M. Livny, Stork: Making data placement a first class citizen in the grid, in: Proceedings of 24th International Conference on Distributed Computing Systems, ICDCS 2004, (2004) 342–349.
[6.] J.M. Cope, N. Trebon, H.M. Tufo, P. Beckman, Robust data placement in urgent computing environments, in: IEEE International Symposium on Parallel &Distributed Processing, IPDPS 2009, (2009)1–13.
[7.] T. Xie, SEA: A striping-based energy-aware strategy for data placement in RAID-structured storage systems, IEEE Transactions on Computers 57 (2008) 748–761.
[8.] M. Mustafa Rafique a, Ali R. Butt a, Dimitrios S. Nikolopoulos b,c, A capabilities-aware framework for using computational accelerators in data-intensive computing, J. Parallel Distrib. Comput. 71 (2011) 185–197.
[9.] L. Wang, J. Tao, M. Kunze, A.C. Castellanos, D. Kramer, W. Karl, Scientific cloud computing: Early definition and experience, in: 10th IEEE International Conference on High Performance Computing and Communications, HPCC'08,(2008) 825–830.
[10.] K. Keahey, R. Figueiredo, J. Fortes, T. Freeman, M.Tsugawa, Science clouds:Early experiences in cloud computing for scientific applications, in: First Workshop on Cloud Computing and its Applications, CCA'08,(2008) 1–6.
[11.] S. Pandey, A. Barker, K. K. Gupta, R. Buyya ,Minimizing Execution costs when using globally distributed cloud services,2010 24th IEEE International Conference on Advanced Information Networking and Applications .
[12.] Y. C. Lee, A. Y. Zomaya, Rescheduling for reliable job completion with the support of clouds, Future Generation Computer Systems 26 (2010) 1192_1199.
[13.] D Yuan, Y Yang, X Liu, A data placement strategy in scientific cloud workflows, Future Generation Computer Systems(2010)1200-1214.
[14.] E. Vineka, P. P. Beranb, E. Schikutab, A dynamic multiobjective optimization framework for selecting distributed deployments in a heterogeneous environment , Procedia Computer Science 4 (2011) 166–175.
[15.] V.M. Lo, "Task assignment in distributed systems", PhD dissertation, Dep. Comput. Sci., Univ. Illinois, Oct. 1983.
[16.] G. Gharooni-fard, F. Moein-darbari, H. Deldari and A. Morvaridi, Procedia Computer Science, Volume 1, Issue 1, May 2010,Pages1445-1454 ,ICCS 2010.