# Efficient Scheduling of Scientific Workflows using Hot Metadata in a Heterogeneous Multisite Cloud

[1]Kore Anjineyulu, [2]Dr.A.Sureshbabu
[1]M.Tech Scholar(CSE),[2]Professor,JNTUACEA
*#Department of CSE, JNTUACEA, Ananthapuramu*
[1]koreanjineyulu@gmail.com
[2]asureshjntu@gmail.com

***ABSTRACT***

**Scientific applications that are large-scale and data-intensive are often articulated as science workflows. In this document, we consider the issue of scheduling a big SWf efficiently in a multisite cloud, i.e. a cloud with geo-distributed cloud information centers. The reasons for using multiple cloud sites to run a SWf are that data is already being distributed, the resources needed to run a single site exceed the limits, or the monetary cost is lower. Metadata management in a multi-site cloud has a critical effect on SWf planning effectiveness as it offers a worldwide perspective of information place and allows for task monitoring during execution. However, it introduces different difficulties that need to be resolved in order to be used effectively for apps for workflow allocation, scheduling and processing. Although there has been extensive study of the workflow planning issue, there are very few initiatives adapted to cloud settings. Project offers load balancing with energy-aware operating model using various servers to provide cloud-based scaling activity. Project provides detailed workflow scheduling with load balancing and scaling techniques and also takes advantage of some of the most desirable features of server consolidation mechanisms. Load balancing with Workflow Scheduling is one of the major cloud computing problems that is needed to spread the dynamic workload across various nodes to guarantee that no single node is overwhelmed. It helps to optimize resource usage and helps to improve system efficiency. The load balancing objective is to minimize the use of resources, which will further decrease the consumption of resources.**

***Key words:*** **Metadata handling, multisite cloud, scientific workflows**

## I. INTRODUCTION

In the order of petabytes, many scientific applications process large amounts of data today. As the information size rises, so do the computing resources demands. It is possible to express such data-intensive apps as Scientific Workflows (SWfs) [3]. A SWf is an assembly of scientific data processing tasks such as input data loading, data processing, data analysis, and output information aggregation. The application is modeled as a graph, with vertices representing processing employment and edging their data dependencies. Since a job's input information can be allocated or partitioned, in a number of executable assignments a job can be further decomposed. Such SWf structuring offers a clear perspective of the application flow and promotes application execution in a geo-distributed setting. Many Scientific Workflow Management Systems (SWfMSs) are currently accessible, for example. Pegasus[7] as well as Swift. Some of them, for instance. Chiron, distributed support, execution of multiple sites.The cloud stands out as a convenient SWfs [3] handling infrastructure, as it provides the opportunity to lease resources on a very big scale and comparatively low price that is a cloud with cloud information centers (sites) geo-distributed. Let us point out, however, that it is not necessarily better to use various cloud locations than a single cloud site. However, there are significant instances where a good choice is to use a multisite cloud. All common government clouds now support this choice well, e.g. Azure, EC2, and Google Cloud [5], which enable various locations to use a single cloud account, thereby avoiding the burden of various accounts.Multiple cloud sites are used for three primary purposes: 1.Already distributed information, 2.Resource boundaries at a single cloud site, and 3.Monetaring cost.

First, the data to be processed by the SWf may already be distributed at different sites because it comes from different experiments, sensing devices or laboratories, e.g ALICE LHC [5] collaboration. In this case, it may be difficult to move the data to a single site, because the data is too large or not allowed to leave the hosting site. Second, the resource requirements for performing the SWf may well exceed the boundaries placed on a single site by the cloud provider.. For example, for both standard and premium accounts, Microsoft Azure imposes a maximum number of virtual CPUs in the VMs per site. Additionally, there are other limitations on storage, network bandwidth and nearly all the resources per site. Other cloud suppliers have comparable constraints, e.g Google cloud, EC2. Third, using VMs on various locations may be less cost. Cloud supplier at separate locations, EC2, Azure and Google Cloud have distinct VM rates. They also pay for the inter-site transfer of information. Using suitable algorithms for scheduling.

Metadata has a critical effect on an SWfMS effectiveness as it offers a global perspective of the place of the information and allows for job monitoring during execution. It should therefore be easily accessible at any specified moment to the scheme. While it has been shown that effective handling of metadata plays a main role in performance, little study in the multisite cloud has targeted this problem. It is even necessary to persist some SWf metadata as provenance information to allow traceability and reproducibility of the employment of the SWf. In particular, some metadata are accessed more frequently than others. We denote such metadata by warm metadata and argue that it should be recognized and treated dynamically in a particular, faster available manner than the remainder of the metadata. Inter-site network latency is much greater in a multi-site infrastructure than intra-site latency. This consideration must remain at the heart of a multi-site metadata management system design. It is necessary to take into consideration several design principles.

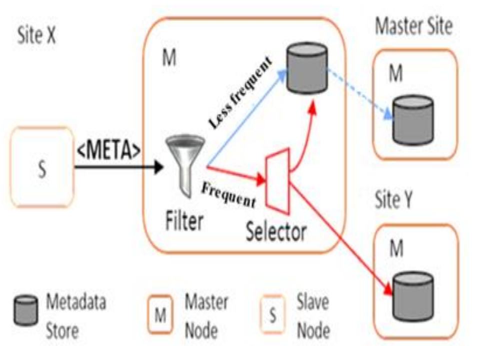


Fig.1.1: System Architecture

## II. LITERATURESURVEY

In [1], Hot and cold information identification method where warm records stay in memory while cold records are candidates for secondary storage migration. During ordinary system runtime, we suggest to sample record accesses and record the accesses on a consolidated log. A transaction copies its record access data into big (shared) buffers which are only flushed asynchronously when complete; the transaction is not waiting for log flushes. Access to sampling and logging reduces overhead on the critical path of the system. It also enables us, if required, to transfer classification to a distinct machine (or CPU core).Estimated frequencies of record access are then calculated from the accesses logged, and documents with the largest approximately frequency form the hot set.

In [2], Provide detailed characteristics of five science workflows that include massively parallel workflows that process big quantities of information, pipeline applications that divide input datasets and work in parallel on separate pieces, and workflows that have a fairly fixed structure and conduct identical analyzes on various input datasets. Workflows are taken from various fields of application such as astronomy, biology, gravitational physics, and science of earthquake. While these workflows are not claimed to depict the complete spectrum of science workflows.

In [3], Describe the design of a distributed file system that differs significantly from any of the file systems mentioned above. The metadata management layer that horizontally partitions and replicates metadata of the file system across a shared-nothing server cluster, spanning various geographic areas, most distinguishes our system. File system activities that possibly span various files or directories are converted into distributed transactions and processed through a transaction planning and replication management layer of an extensible distributed database system to guarantee adequate coordination of linearizable updates.

In [4], Exploring the feasibility of a particular metadata storage and management layer for parallel file systems, where metadata involves both file activities and metadata of provenance. In specific, we are investigating the design optimality experimentally-whether provenance metadata should be loosely coupled or closely integrated with file metadata storage schemes. In order to perform this experimental assessment, we consider two systems that applied comparable distributed ideas to the management of metadata, but focused specifically on metadata type. FusionFS [9], which implements metadata management of distributed files based on distributed hash tables.SPADE [9], which utilizes a database of graphs to store audited provenance information and offers a distributed provenance query module.

We focus on the workflow mapping aspect of the issue in this document. We assume that the application is already represented in an abstract workflow form that identifies the components of the application and their dependencies, as well as the data they use and produce, but does not specify specific resources to be used. The issue of workflow mapping can be described as discovering a mapping of activities to assets that minimizes the total execution time of the workflow. The execution of the workflow consists of the runtime of the assignments and the tasks of information transfer which process information in and out of the computation.

In [5], To implement SWfs in a multi-site cloud, we suggest a general multi-objective planning solution. A multi-objective cost model, a Single Site VM Provisioning (SSVP) approach, and ActGreedy [5], a multi-site planning approach, are included. The cost model involves two goals under stored information limitations, namely lowering execution time and financial expenses, which specifies that some information should not be moved because it is either too large or for proprietary purposes. While these limitations are helpful for solving certain operations, they do not greatly decrease the complexity of scheduling operations. We find a cloud environment homogeneous, i.e. from a single supplier. The situation of federated clouds (with various cloud suppliers) is beyond the scope of this document and not a reality. To assess the cost of implementing SWfs in a multi-site cloud, design a multi-objective cost model that involves implementation time and financial expenses. A single site VM supplying strategy (SSVP) to produce VM supply plans to implement fragments at each site. ActGreedy[3] multisite planning algorithm that utilizes the price model and SSVP to plan and implement multi-site cloud SWfs.

## III. EXISTING SYSTEM

An entire application as a service that can be used straight without modification. In the current scheme, Workflow Services with Time Slot and Resource Utilization are accessible for single server, which causes Server Hang and not all server resources to be correctly used. Because the server services are combined with user data storage web services and many other security operations that are controlled by first-in-first-out methods that make the server activity into Deadline length queues. Only homogeneous multi-site environment where each site has the same amount of VMs of same type.

- The workflow deadlines of applications cannot fulfill the requirement.
- Unreserved time slots are not structured from the point of view of service suppliers as vital for resource use.
- In order to prevent renting fresh resources, the use of time slots at reserved intervals should be enhanced.

## IV. PROPOSED WORK

The multi-site planning method suggests using distributed strategy to define and utilize the hot metadata for effective SWf in a multi-site cloud. Efficient handling of metadata is critical in multisite cloud efficiency of large-scale SWf execution. The term warm metadata refers to frequently accessed information.

This concept is introduced to the leadership of SWf and defines warm metadata as the metadata that is often accessed during execution of SWf. The less frequently accessed metadata, on the other hand, is called cold metadata. In cloud computing, service capabilities are generally considered unlimited, which can be used at any moment. Detailed control of the energy-aware operating model used for load balancing with cloud work flow scheduling using multiple servers for application scaling and other cloud activities is included in our Proposed System. Extending multi-server scheduling activities offers Energy Saving, Huge Load Controlling, Un Used Sharing, or Idle servers. To save energy, idle and lightly loaded servers are moved to one of the sleeping states. Some of the most desirable characteristics of server consolidation systems mentioned in the literature are also exploited by the load balancing and scaling algorithms.

- Cloud servers based on separate working administrations with varying degrees of effectiveness in processing.
- Load balancing and scaling of applications to maximize the amount of servers running on numerous idle servers.
- Provides an Effective Use of Server Resource.

**Load balancing:**

The load balancing idea dates back to the moment of implementation of the first distributed computing systems. It means precisely what the name suggests, distributing the workload uniformly to a server set to maximize throughput, minimize reaction time, and boost system resilience to flaws by preventing system overload.

**Idle servers:**

Idle and underused servers contribute considerably to waste energy; see Section Survey reports that idle servers contribute 11 million tones of unnecessary CO2 emissions each year and that the complete annual cost of idle servers is one billion tones. No energy is consumed by an energy-proportional scheme when idle, very little energy under a light charge and gradually more energy as the load rise.

**Server consolidation:**

The word server consolidation is used to define changing idle and lightly loaded systems to a sleeping state workload migration to avoid overloading of systems any optimization of cloud performance and energy efficiency by redistributing the workload mentioned in Section For instance, when choosing to migrate some of the VMs operating on a server or change a server to a sleeping state.
we can use the word server consolidation.Predictive policies, such as those discussed to allow a server to operate under a sub-optimal scheme when historical data on its workload indicate that it is likely to return to the optimal scheme in the near future.

**Energy proportional systems:**

A system's energy efficiency is measured by the power ratio per Watt. Computing systems ' performance has raised much quicker over the past two decades than their energy efficiency proportional energy systems. The energy consumed by an idle system should be close to zero in an ideal world and grow linearly with the system load. Even systems with a linear scale of energy demands when idle use more than half of the energy they use at complete load in actual life. Data gathered over a lengthy period of time demonstrates that the typical data center server working scheme is far from an optimal energy consumption scheme. As a function of the load imposed on the system, the dynamic range I is the deference between the upper and lower limits of a system's energy consumption. A big dynamic range implies that when its load is small, a device can function at a reduced percentage of its maximum energy.

## V. EXPERIMENTAL RESULTS

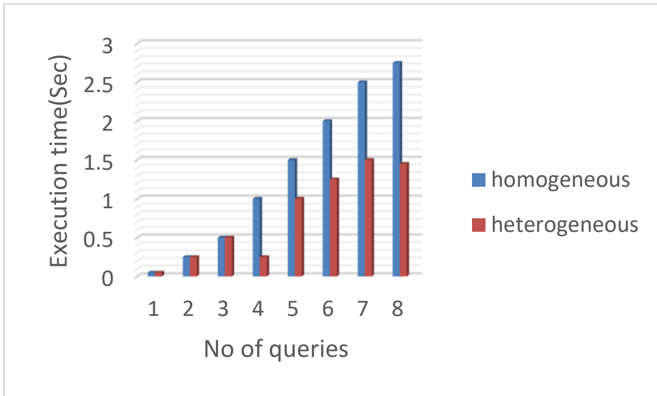


Fig .2: Comparison between execution time of heterogeneous and homogeneous multisite cloud.

Our result shows fig.2the execution time of heterogeneous multisite cloud is efficient than homogeneous multisite cloud, when the number of queries is increased.

## VI. CONCLUSION

Efficient handling of metadata in a multi-site cloud is critical to the efficiency of large-scale SWf execution. We suggested using a distributed approach to dynamically define and exploit hot metadata for effective SWf scheduling in a multisite cloud. Our solution involves assigned architecture with dynamic hot metadata identification and three hot metadata management approaches: two strategies tailored from associated job (DHT and REP) and a new strategy (LOC) that stores the hot metadata at the site where it is generated. We also adjusted three algorithms for scheduling, i.e. OLB, MCT and DIM, conduct multi-site planning through metadata provisioning. Our strategy offers effective access to hot Meta data, hides latencies in the inter-site network, and remains non-intrusive and simple to implement. Furthermore, as future job, our experiments demonstrate that while no single decentralized approach can suit all SWf structures well, our suggested approach for warm metadata, i.e. LOC, in terms of general SWf execution time, always outperforms other strategies. We validated our solution for generosity in a multisite cloud, which is the most convenient way for researchers to carry out experiments on a scale.

## REFERENCES

[1] Levandoski, J. J., Larson, P.-A., &Stoica, R. (2013). *Identifying hot and cold data in main-memory databases. 2013 IEEE 29th International Conference on Data Engineering (ICDE).*doi:10.1109/icde.2013.6544811

[2] Bharathi, S., Chervenak, A., Deelman, E., Mehta, G., Su, M.-H., &Vahi, K. (2008). *Characterization of scientific workflows. 2008 Third Workshop on Workflows in Support of Large-Scale Science*.doi:10.1109/works.2008.4723958

[3] Pineda-Morales, L., Liu, J., Costan, A., Pacitti, E., Antoniu, G., Valduriez, P., &Mattoso, M. (2016). *Managing hot metadata for scientific workflows on multisite clouds. 2016 IEEE International Conference on Big Data (Big Data).*

[4] Zhao, D., Shou, C., Maliky, T., &Raicu, I. (2013). Distributed data provenance for large-scale data-intensive computing. 2013 IEEE International Conference on Cluster Computing (CLUSTER). doi:10.1109/cluster.2013.6702685

[5] Jin, Shudong; Bestavros, Azer. "GreedyDual* Web Caching Algorithm: Exploiting the Two Sources of Temporal Locality in Web Request Streams", Technical Report BUCS-2000-011, Computer Science Department, Boston University, April 4, 2000.

[6] Liu, J., Pacitti, E., Valduriez, P., de Oliveira, D., &Mattoso, M. (2016). *Multi-objective scheduling of Scientific Workflows in multisite clouds. Future Generation Computer Systems, 63, 76–95*.doi:10.1016/j.future.2016.04.014

[7] Deelman, E., Singh, G., Su, M.-H., Blythe, J., Gil, Y., Kesselman, C., … Katz, D. S. (2005). *Pegasus: A Framework for Mapping Complex Scientific Workflows onto Distributed Systems. Scientific Programming, 13(3), 219–237*
.

[8] D. Zhao et aI., "Distributed File Systems for Exascale Computing," ACM/IEEE Supercomputing (Doctoral Showcase), 2012. [l6] A. Gehani et aI., "SPADE: Support for Provenance Auditing in Distributed Environments," ACM/USENIX Middleware, pp. 101-120,2012

[9] Zhao, D., Zhang, Z., Zhou, X., Li, T., Wang, K., Kimpe, D., …Raicu, I. (2014). *FusionFS: Toward supporting data-intensive scientific applications on extreme-scale high-performance computing systems. 2014 IEEE International Conference on Big Data (Big Data).*doi:10.1109/bigdata.2014.7004214

[10] Liu, J., Pacitti, E., Valduriez, P., &Mattoso, M. (2015). A Survey of Data-Intensive Scientific Workflow Management. Journal of Grid Computing, 13(4), 457–493. doi:10.1007/s10723-015-9329-8

[11] Ogasawara, E., Dias, J., Silva, V., Chirigati, F., de Oliveira, D., Porto, F., …Mattoso, M. (2013). Chiron: a parallel engine for algebraic scientific workflows. Concurrency and Computation: Practice and Experience, 25(16), 2327–2341.doi:10.1002/cpe.3032