A Novel Probabilistic Framework For High Temporal Qos In Scientific Cloud Workflow Systems

Prof. Jasobanta Laha[1st]
Associate Professor
SERC, Bhubaneswar

Prof. (Dr.) R. N. Satpathy[2nd]
Principal, HIT, Bhubaneswar

Prof. (Dr.) C. R. Panda[3rd]
Principal, SERC, Bhubaneswar

Abstract:

Cloud Computing is an emerging area in the field of information technology (IT). Its technology can provide virtually unlimited scalable performance computing resources. In the real world, scientific applications need to be time constrained and required to be completed by satisfying a set of temporal constraints. The task execution time or activity duration is one of the basic measurements for system performance, often needs to be monitored and controlled by specific system management mechanisms. To ensure satisfactory temporal correctness (high temporal QoS), to guarantee on-time completion of most constraints, workflow applications is a critical issue for enhancing the overall performance and usability of scientific cloud workflow systems. The novel comprehensive probabilistic temporal framework is a cost-effective solution to deliver high temporal QoS in scientific cloud workflow systems. The temporal framework is designed to support the lifecycle of workflow instances from build time modeling stage to workflow runtime execution stage until successful completion. The probability based temporal consistency model and its capability is realized by three components to support workflow instances throughout their lifecycles including temporal constraints setting, temporal consistency monitoring and temporal violation handling.

Keywords: TIME CONSTRAINED, HIGH TEMPORAL QUALITY OF SERVICE (QOS), SCIENTIFIC CLOUD WORKFLOW, TEMPORAL CONSTRAINTS SETTING, TEMPORAL CONSISTENCY MONITORING, TEMPORAL VIOLATION HANDLING

Introduction:

Cloud computing adopts a market oriented resource model like cloud resource computing, storage and network are charged as per the usage. The cost for supporting temporal QoS (including both time overheads and monetary cost) can be managed effectively in Scientific Cloud Workflow Systems. This article proposes a novel probabilistic temporal framework for high temporal QoS in scientific cloud workflow systems. The temporal framework can provide a systematic and cost-effective support for time-constrained scientific cloud workflow applications over their whole lifecycles compared with conventional temporal QoS. With a probability based temporal consistency model, there are three major components in the temporal framework:

i) Component 1 – temporal constraint setting

ii) Component 2 – temporal consistency monitoring

iii) Component 3 – temporal violation handling

Framework:

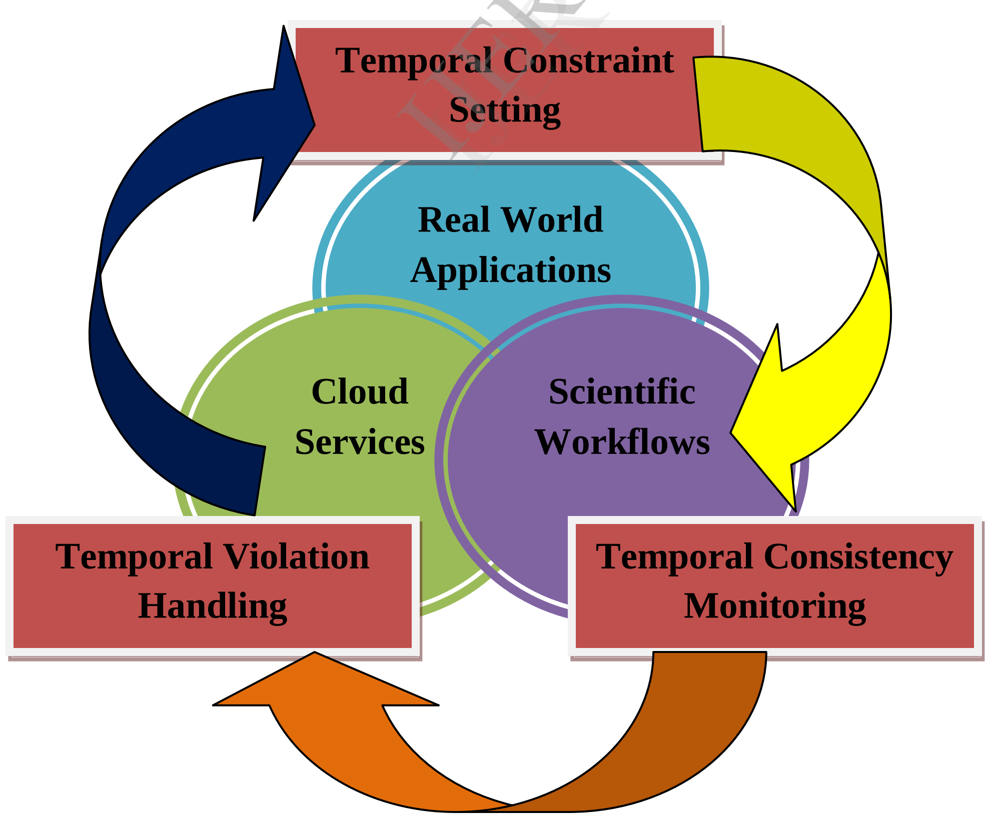

Figure: The Probabilistic Framework for Temporal QoS

The novel temporal framework is a cost-effective solution to deliver high temporal QoS in scientific cloud workflow systems. It is designed to support the whole lifecycle of workflow instances, from the build-time modeling stage to the workflow run-time execution stage till successful completion. The foundation of this framework is the probability-based temporal consistency model, and its capability is realized by three components that support workflow instances throughout their lifecycles, including temporal constraints setting, temporal consistency monitoring and temporal violation handling.

The probabilistic framework consists of three components to provide lifecycle support for high temporal QoS in scientific cloud workflow systems. The three inner cycles stand for three important factors involved in the scientific cloud workflows like:

i) Real World Applications

ii) Scientific Workflows

iii) Cloud Services

The basic requirements for scientific cloud workflows come from real world applications and are abstracted by service users with the support of workflow modeling tools then create all specifications. These specifications in the form of visualized workflow templates submitted by the service users and scientific workflows are executed by the cloud workflow systems with the underlying cloud computing infrastructures.

Cloud workflow systems are a type of platform that can utilize many other cloud services based on software or computing principle. The execution of scientific cloud workflows is complex and has large-scale sophisticated processes containing many data and computation intensive activities. The performance of scientific cloud workflow systems is uncertain since cloud services are generally with highly dynamic performance in the cloud computing environments. The process complexity and performance uncertainty causes, high temporal QoS cannot be easily achieved without comprehensively designed framework to support the lifecycle of scientific cloud workflows. This is the main objective of this article.

Three components are organized in a cycle which denotes that they are working in such an order to support the whole lifecycle of scientific cloud workflows. The Components are:

i) Component – I: Temporal Constraint Setting
ii) Component – II: Temporal Consistency Monitoring
iii) Component – III: Temporal Violation Handling

Component – I: Temporal Constraint Setting

Temporal constraint setting assigns both global temporal constraints (temporal constraints for entire workflow instances) and local temporal constraints (temporal constraints for local workflow segments and individual workflow activities) in scientific cloud workflow specifications at workflow build time and temporal constraints as a type of QoS requirements, are to be specified in cloud workflow definitions. With other QoS constraints such as cost and security, these temporal constraints serve as critical criteria for the selection of cloud services and the SLA (Service Level Agreement) management [4]. During cloud workflow runtime, service providers are obligated to complete workflow processes within the assigned temporal constraints, otherwise penalty may be enforced according to the service contracts signed. The setting of high quality temporal constraints is essential to the successful completion of scientific cloud workflows.

In scientific workflow systems, temporal consistency is critical to ensure the timely completion of workflow instances. The correctness of temporal consistency, temporal constraints are often set and then verified. The current work adopts, user specified temporal constraints without considering system performance, and may result in frequent temporal violations that deteriorate the overall workflow execution effectiveness. The task of setting temporal constraints described in this paper is to assign a set of coarse-grained and fine-grained upper bound temporal constraints to scientific workflows. Coarse-grained constraints refer to those assigned to the entire workflow or workflow segments, while fine-grained constraints refer to those assigned to individual activities. Though coarse-grained constraints can be deemed as collection of fine-grained constraints, they are not in a simple relationship of linear culmination and decomposition. To ensure on-time fulfillment of workflow instances, both coarse-grained and fine-grained temporal constraints are required. When scientific workflows are deployed on dynamic computing infrastructures like grid the performance of the underlying resources is highly uncertain [6]. The quality of temporal constraints can be measured by two criteria:

i) Well balanced between user requirements and system performance

ii) Well supported for both overall coarse-grained control and local fine-grained control

Component – II: Temporal Consistency Monitoring

Temporal consistency monitoring deals with monitoring of temporal consistency state against temporal violations. Based on this model, temporal consistency states of scientific cloud workflows to be under constant monitoring in order to detect potential temporal violations in a timely fashion. The cost of temporal verification is large due to the complexity and uncertainty in cloud workflow system environments. Therefore, cost-effective strategies need to be designed to detect potential temporal violations in an efficient fashion. In scientific cloud workflow runtime, the state of cloud workflow execution towards specific temporal constraints is monitored constantly by the checkpoint selection and temporal verification component. The two major limitations of conventional checkpoint selection and temporal verification are:

i) The selection of multiple types of checkpoints and the verification of multiple types of temporal consistency states cause huge cost but most of them are actually unnecessary

ii) Though the violations of multiple temporal consistency states can be verified

There is no indication for the level of temporal violations and does not support the quantitative measurement of temporal violations. To address the above issues the probability based temporal consistency model, the probability range for statistically recoverable and non-recoverable temporal violations are defined in the first place. The monitoring of cloud workflow executions can be done by:

i) Minimum probability time redundancy based temporal checkpoint selection strategy determines the activity points where potential temporal violations take place

ii) Probability temporal consistency based temporal verification is conducted on a checkpoint to check the current temporal consistency state and the types of temporal violations like recoverable or non-recoverable

Component – III: Temporal Violation Handling -

Temporal violation handling which deals with recovery of temporal violations based on the results of previous component for monitoring temporal consistency, a necessary and sufficient checkpoint is selected where a potential temporal violation is detected.

Previous conventional temporal verification work believes in the philosophy that temporal violation handling should be executed at all necessary and sufficient checkpoints, with the probability of self recovery that philosophy is not necessarily ideal.

A temporal violation handling point selection process can be designed to further determine whether temporal violation handling is necessary at each checkpoint. The necessity for temporal violation handling points means that the probability of self-recovery is a threshold and the probability of temporal violations is high enough so that temporal violation handling is necessary. If a temporal violation handling point is selected, then temporal violation handling strategies may be executed. This temporal framework mainly focuses on those statistically recoverable temporal violations which can be recovered by light-weight temporal violation handling strategies. For such a purpose, representative met heuristics based workflow rescheduling strategies are investigated, adapted and implemented under a novel general two-stage local workflow rescheduling strategy to handle temporal violations. This temporal violation handling strategy is fully automatic and only utilizes existing system resources without recruiting additional ones, the cost of temporal violation handling can be significantly reduced compared with many conventional heavy-weight temporal violation handling strategies.

Conclusion & Future Work:

This model significantly reduces the cost for detection and handling of temporal violations whilst delivering high temporal QoS in scientific cloud workflow systems. It would eventually improve the overall performance and usability of cloud workflow systems because a temporal framework can be viewed as a software service for cloud workflow systems. Consequently, by deploying the new concepts, innovative strategies, scientific cloud workflow systems would be able to better support large-scale sophisticated e-science applications in the context of cloud economy. The future work will focus on how to improve the overall performance of the framework and further reduce the cost of temporal violation handling.

References:

[1] E. Deelman, D. Gannon, M. Shields, and I. Taylor, "Workflows and e- Science: An Overview of Workflow System Features and Capabilities", *Future Generation Computer Systems*, vol. 25, no. 6, pp. 528-540, 2008.

[2] J. Chen and Y. Yang, "Localising Temporal Constraints in Scientific Workflows", *Journal of Computer and System Sciences*, vol. 76, No. 6, pp. 464-474, 2010.

[3] T. Erl, *SOA: Principles of Service Design*: Prentice Hall, 2008.

[4] X. Liu, J. Chen, and Y. Yang, "A Probabilistic Strategy for Setting Temporal Constraints in Scientific Workflows", *Proc. 6th International Conference on Business Process Management (BPM08)*, pp. 180-195, Milan, Italy, Sept. 2008.

[5] I. J. Taylor, E. Deelman, D. B. Gannon, and M. Shields, *Workflows for e-Science: Scientific Workflows for Grids*: Springer, 2007.

[6] J. Chen and Y. Yang, "A Taxonomy of Grid Workflow Verification and Validation", *Concurrency and Computation: Practice and Experience*, vol. 20, no. 4, pp. 347-360, 2008.