

## WORKFLOW SCHEDULING BASED ON WORK RATE COMPARISON

Vignesh.V, Assistant Professor(SI.G), Abinaya.R,HabeebNisha.B, JeniJafrin.S

Information Technology, Sri Ramakrishna Engineering College, India

**Abstract-**Workflow scheduling plays an important role in today's computing world. Many scheduling algorithms have been designed for workflow scheduling process. Instance Intensive workflow scheduling work with multiple workflow instances. Online train ticketing, e-commerce, e-government are the suitable examples for Instance Intensive workflow scheduling. Here we consider QoS parameters based on user's perspective. The most important QoS parameters to be considered are speed, time and cost. To improve the overall execution time least response time algorithm(LRTA) is used. The QoS based Deadline Allocation Scheduling Algorithm can be used to still improve the efficiency among servers.

**Keywords:** Workflow, QoS, LRTA, QDA.

### I. INTRODUCTION

Scheduling has paid much attention as the demand for cloud applications increases. Scheduling is used to allocate machines dynamically in cloud which allows the users to share the resources in cloud. Each and every user assigns the task to cloud, and the task is processed in priority scheduling basis. The tasks are arrived in the queue and executed based on FCFS scheduling. Response is given to the user within the given time thus decreasing the response time.

#### A. CLOUD COMPUTING

The latest trend used to store huge volumes of data and allowing the users to access the resources as pay per use is cloud computing. Cloud computing is a type of Internet-based computing that provides shared resources and data to users and other devices. It is a model for enabling ubiquitous, on-demand access to a shared pool of configurable resources (e.g. networks, servers, storage, applications and services), which can be managed with minimal effort. Cloud computing and storage solutions provide users and enterprises to store and access the data both privately and publicly (Third-party). Cloud computing helps to achieve scalability and efficiency. Cloud has

elastic nature. Cloud provides services such as Software as a service (SaaS), Platform as a service (PaaS) and Infrastructure as a service (IaaS). Cloud computing services can be private, public or hybrid. Private cloud provides services for the internal users. Public cloud provides services for third party users. Hybrid cloud is a combination for both internal and external users.

### B. SCHEDULING TASKS WITHIN THE DEADLINE

Time Cost Tradeoff Problem (TCTP) is one of the most important issues. This problem often comes up when the duration of a project needs to be reduced to compensate for unexpected delays. The great importance of time and cost leads to the simultaneous management of project completion time and project cost. The main objective of this problem is to determine the optimal solution of activity durations and activity costs that leads to a balance the project completion time and the total project cost. Much of the resource planning literature has been presented under the assumption of Quality of Service project parameters; in the real world, however, activity durations and the structure of project networks are affected by considerable uncertainties.

All existing patterns of the TCTP satisfy the QoS parameters by reducing the duration of activities. Several procedures (e.g., additional resource allocation, improvement in the technology level, and increase in the quality of materials) are available for time cost tradeoff problem.

### C. QoS BASED DEADLINE ALLOCATION

QDA algorithm references the main sub-deadline allocation criteria of CTC algorithm, uses QoS utility function value as a service resource selection condition, and takes user preferences into account. A set of workflow instances is provided as input with the same type of tasks and results in a set of scheduling results, which gives the corresponding service resource for each task within the instance set. QDA results in better performance, less execution time, better user satisfaction and improved load balancing rate.

#### D. REQUIREMENTS OF QoS

- Reliability: It can be calculated as follows:

$$\text{Reliability} = \text{MTBF} = \text{MTTF} + \text{MTTR}$$

- Testing time: It can be calculated as follows:

$$\text{Testing Time} = \text{Time to prepare test Environment} + \text{Time to execute Test Suite for a Cloud workload}$$

- Availability: It can be formulated as

$$\text{Availability} = \text{MTTF} / \text{MTBF}$$

- Network bandwidth: The network bandwidth can be calculated as number of bits transferred/received in a particular workload in one second.

$$\text{Network Bandwidth} = \text{Bits/second (b/s)}$$

- Computing capacity: It can be formulated as follows:

$$\text{Computing Capacity} = \frac{\text{Actual Usage time of the Resource}}{\text{Expected Usage time of the Resource}}$$

#### D. INSTANCE INTENSIVE WORKFLOW SCHEDULING

Instance Intensive workflow scheduling is a scheduling mechanism which includes multiple workflow instances. Earlier scheduling algorithms allocate the task to the node which is visited first. This leads in under loading of other nodes, thus decreasing the performance. This scheduling algorithm helps in balancing the load among the loads, thus resulting in quick response and less execution time. This workflow scheduling takes the account of user's QoS preferences. Most of the user's preference is based on high performance. Many cost constrained algorithms are considered to minimize the overall execution cost. This situation can fully meet user's demand.

#### II. LITERATURE SURVEY

Hadi Mokhtari, Reza Baradaran and Ali Salmasnia proposed a new model Discrete Time Cost Tradeoff problem for the management of project. Time and Cost plays a major role in the development of project. The budget should be less over the completion time of the project. The project should be completed within the given duration of time. The DTCTP model is used as a zero one problem which increases the

probability of completing the project within the deadline and within the budget.

Wei-Neng Chen, Student Member, IEEE, and Jun Zhang, Senior Member, IEEE proposed an approach called Ant Colony Optimization system to schedule large scale workflow by considering various QoS parameters. Here various parameters like makespan, cost, reliability, time were considered. The main objective of the ACO algorithm is to find a feasible solution that satisfies all the user's QoS parameters.

Wanchun Dou, Xuyun Zhang, Jianxun Liu, and Jinjun Chen, Senior Member, IEEE proposed a privacy aware cross cloud service composition method called HistoryRecord based Service Optimization method (Hiresome-2). This method is used to ensure security and privacy for the cloud which provide their service transaction details. It is highly needed to ensure the privacy of the cloud services. The k-means algorithm is used here to filter the representative service.

#### III. PROPOSED SYSTEM

In Proposed system, the task is sent to the cloud which is serviced within a suitable facility node. A facility node may contain different computing resources such as web servers, database servers, directory servers. Here user's QoS factors like response time, speed and cost is considered. The task will be processed in corresponding cloud server based on user category where scaling depend on it. We model a cloud server system which indicates the arrival time of requests and process those requests by the comparison of work rates among the servers. The system under consideration contains servers which render service in order of task request arrivals (FCFS). As the population size of a typical cloud is relatively high while the probability that a given user will request service is relatively small.

#### A. PERFORMANCE ANALYSIS OF SERVERS

In the world of today, servers are accessed by many users as well as high performance machines. So, getting the most out of those machines and keeping them to provide response are critical to maximizing their value. However, maintaining high levels of server performance and compare work rates of servers. Tracking performance and user requests are response by best server. There are many ways for IT professionals who can simplify server management while maintaining top performance. Benchmark testing can help administrators monitor server resources, manage utilization, optimize performance. Simple server

configuration can squeeze more performance out of both physical and virtual machines.

All the organizations handle different servers and different network speeds. Servers handle various problems which affects the system like more number of load balance, data transferring speed, etc.

Benchmark testing can help establish baseline performance analysis and track the system's performance over the time and cost. Benchmarks can also be used by the administrators to manage server resources utilization and performance. This can allow data center cloud to identify potential performance problems and plan for future capacity needs.

All the metric testing will produce data, that data may not be relevant. Following a structured approach to benchmark testing will ensure the correct metrics are being measured, variables are accounted for and the results are accurate.

## B.COMPARATIVE ANALYSIS

Server selection plays an important role in replication networks, such as peer-to-peer (P2P) and content delivery networks (CDNs). An analytical investigation of the strengths and weaknesses of existing server selection policies, based initially on an Processor Sharing (PS) theoretical model. A theoretical benchmark to evaluate the performance of two general server selection policies, referred to as EQ\_DELAY and EQ\_LOAD, which implies a wide range of existing server selection algorithms. The EQ\_LOAD achieves an average delay always higher than or equal to that of EQ\_DELAY. A key theoretical is-server system, the worst case ratio between the average delay of EQ\_DELAY or EQ\_LOAD and the minimal average delay (obtained from the benchmark) is precise. This shows, how this worst case scenario can arise in highly heterogeneous systems. This result, when interpreted in the context of routing, means that the price of anarchy in unbounded delay networks depends on the various topologies, and can potentially be very large. Our analytical findings are extended in asymptotic regimes to the First-Come First-Serve and PS models and supported by simulations for various arrival and service processes, scheduling disciplines, and workload delivers temporal locality. These results indicate that our analysis is applicable to realistic scenarios. distributed systems, game theory, load balancing, peer-to-peer networks, price of anarchy.

## C.NETWORK LOAD BALANCING

Network load balancing is the ability to balance traffic across two links without using complex routing protocols. This capability balances sessions like

Web, email, etc. over multiple connections in order to spread out the amount of bandwidth used by each user, thus increasing the total amount of bandwidth. For example, a user has a single WAN connection to the Internet operating at 1.5Mbit/s. The second broadband (cable, DSL, wireless, etc.) connection operating at 2.5Mbit/s. This would provide 4Mbit/s of bandwidth when balancing sessions it balances sessions across each WAN link. When the browsers connect to the Internet, they commonly open multiple sessions, one for the text, another for an image, another for some other image, etc. Each of these sessions can be balanced across the available network connections. An FTP application only uses a single session, thus it is not balanced. If a secondary FTP connection is made, then it may be balanced so that on the whole, traffic is distributed across the various connections and thus provides an overall increase in throughput. Additionally, network load balancing is commonly used to provide network redundancy so that in the event of a link outage, access to network resources is still available via the secondary link(s). Redundancy is a key requirement for business continuity plans and generally used in conjunction with critical applications.

Most network load balancing systems also incorporate the ability to balance traffic. Inbound load balancing is generally performed through dynamic DNS which can either be built into the system, or provided by an external service or system. Having the dynamic DNS service within the system is generally thought to be better from a cost savings and delay.

Multiple nodes are joined to create a cluster. Clusters can use network load balancing whereby simultaneous cluster request are distributed between cluster servers. Round-robin DNS records is one form of cluster load balancing. It works by creating multiple host records for one machine. As clients make requests, DNS rotates through its list of records.

In addition to the before mentioned, to connect a terminal server cluster, one needs a load-balancing technology such as Network Load Balancing (NLB). A load balancing solution will distribute connections to each of the terminal servers.

Terminal Session Directory is a one that allows users to automatically reconnect to a disconnected session in a load balanced Terminal Server farm. The directory keeps a list of sessions indexed by username and server name. This enables a user to reconnect to the Terminal Server where the disconnected session resides in order to resume working in that server. This reconnection will



work even if the user connects from a different computer.

Load balancing can improve the distribution of workloads across multiple computing resources, such as computers, computer server, network links, central processing units and disk drives. Load balancing helps to optimize resource use, maximize throughput, minimize response time, and avoid overload of any single node. Using multiple components with load balancing instead of a single component may increase reliability through redundancy. Network Load Balancing Services (NLBS) is a clustering and load balancing that is intended to provide high availability and high reliability, as well as high scalability. NLBS is intended for applications with relatively small data sets that rarely change and do not have long-running in-memory states. These types of applications are called stateless applications.

#### D. LEAST RESPONSE TIME ALGORITHM

The least response time algorithm is a kind of dynamic work allocation algorithm. When a new task arrives it first searches for the server which is free of task i.e. if the work rate of the server is zero then that server is allocated to perform the task. If any of the servers does not have zero work rates then the work rate of the servers are compared and the server with least work rate is assigned to do the job. The less work rate denotes that the server has completed the work within less amount of time. So the execution time of the server is less thereby resulting in fast response. The server with less response time is made to do the job. The least response time algorithm works well for servers with equal capacity or servers with different capacities. Least Response Time allows you to distribute client requests across multiple servers. It helps to improve server fault tolerance and end-user response time. This algorithm distributes client requests across multiple servers to optimize resource utilization. In a scenario with a limited number of servers providing service to a large number of clients, a server can become overloaded and degrade server performance. Load balancing is used to prevent bottlenecks by forwarding the client requests to the servers best suited to handle them.

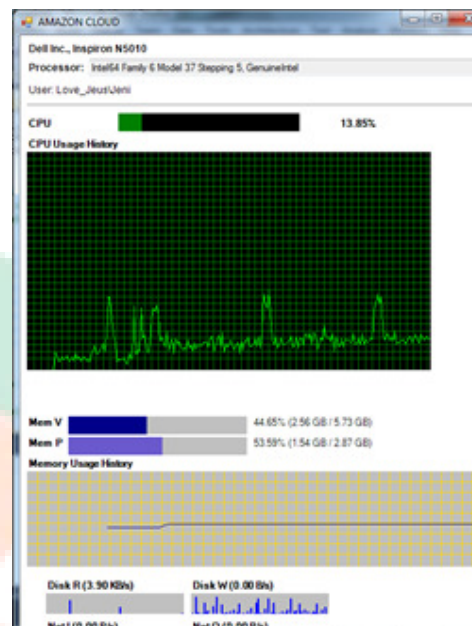


Fig.1 PERFORMANCE OF SERVER

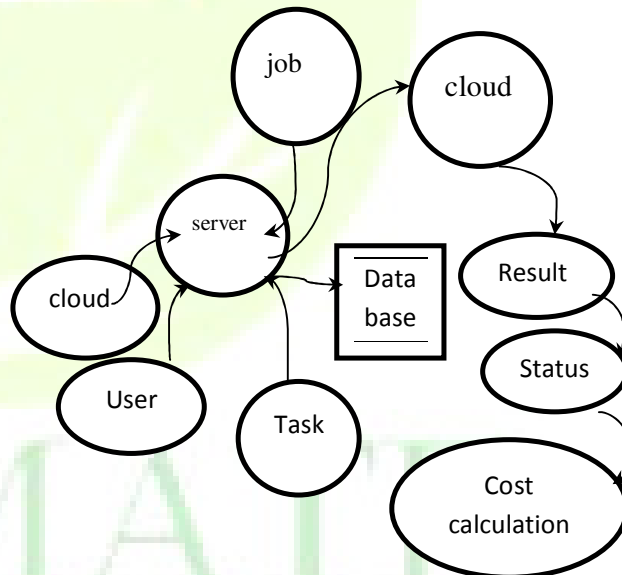


Fig 2 CLOUD WORKFLOW

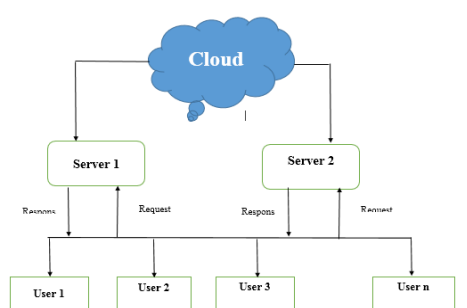


Fig.3 SYSTEM ARCHITECTURE

[7] Ke Liu, Hai Jin, Jinjun Chen, Xiao Liu, DongYuan, Yun Yang, "A Compromised-Time-Cost Scheduling Algorithm in SwinDeW-C for Instance-Intensive Cost-Constrained Workflows"

#### IV. CONCLUSION

Thus multiple workflows can be scheduled using LRT algorithm. Further QDA which considers user's QOS parameter to meet their demands in providing service. The services are provided to users within the deadline with increased speed and performance. The working rate of different servers are compared and the suitable server is scheduled for the process.

#### REFERENCES

- [1] Wei-neng Chen, member, IEEE, Jun Zhang, Senior Member, IEEE, "An Ant Colony Optimization Approach to a Grid Workflow Scheduling Problem With Various QOS requirements", 2009
- [2] Wanchun Dou, Xuyun Zhang, Jianxun Liu, Jinjun Chen, Senior member, IEEE, "Towards Privacy-Aware Cross-Cloud Service Composition for Big Data Applications", 2013
- [3] Hadi Mokhtari, Reza Baradarankazemzadeh, Ali Salmasnia "Time-Cost Tradeoff Analysis in Project Management An ant system approach", 2011
- [4] S. Mukute, G. Hapanyengwi, B. Mapako and M. Nyambo, "Scheduling in Instance Intensive cost constrained workflow in a cloud", 2013
- [5] Ke Liu, Hai Jin, Xiao Liu, and Yun Yang, "A compromised Time Cost Scheduling Algorithm for Instance Intensive Workflow", 2012
- [6] C. Liu, B. T. Loo, Y. Mao, "A scheduling Algorithm for Multi tenants Instance Intensive Workflows", 2013.