International Journal for Innovative Engineering and Management Research

PEER REVIEWED OPEN ACCESS INTERNATIONAL JOURNAL

www.ijiemr.org

IJIEMR Transactions, online available on 18th Jul 2023. Link :http://www.ijiemr.org/downloads.php?vol=Volume-12&issue=Issue 07

10.48047/IJIEMR/V12/ISSUE 07/08

Title **Index-based dynamic-provision of resources in Cloud Computing Virtualization Model**

Volume 12, ISSUE 07, Pages: 69-75

Paper Authors **S. Santhosh Kumar, Mrs.Raziya Begum, Medishetti Vedavani, Lakavath Rajitha**

International Journal for Innovative Engineering and Management Research
PEER REVIEWED OPEN ACCESS INTERNATIONAL JOURNAL
www.ijiemr.org

Index-based dynamic-provision of resources in Cloud Computing Virtualization Model

S. Santhosh Kumar[1], Mrs.Raziya Begum[2], Medishetti Vedavani[3], Lakavath Rajitha[4]
[1,2,3,4]Assistant Professor,Department of CSE,Balaji Institute of Technology & Science
Narsampet, Warangal - 506331, Telangana, India.
santhu3513@gmail.com[1], raziyabegum66@gmail.com[2],
vedavanimedishetti96@gmail.com[3],
lrajitha91@gmail.com[4]

Abstract
The majority of identity management research in the context of cloud computing has focused on safeguarding user information. However, when users utilize cloud services, they frequently leave a trace, and the resultant user traceability may result in the exposure of private user data. Meanwhile, the usage of data by attackers gives cloud providers a risky situation. One of the main problems with cloud computing is resource management, which uses virtualization technology to hide the complexity of the underlying resources, allowing the vastly dispersed resources to come together to form a single, enormous resource pool. By employing logically implemented resource management methodologies and procedures, efficient resource provisioning can be accomplished. In various applications and mobile contexts, top-k query processing is commonly employed. An index employed in the top-k query processing can be carried out effectively using layer-based indexing techniques and effective query processing. It is challenging to use the existing approaches because of the issue of a long index construction time for multidimensional and huge data. The dimensional layer (DL-Layer), a novel idea for building layer-based indexes, is what we proposed in this work. The current methods wrap the remaining input data around a layer that is constructed as a balanced layer with the outermost data. DL, on the other hand, creates a layer that is imbalanced and does not encompass the remaining data.

Keywords—Role-Based Access Control; Indexing; Task Scheduling; Cloud Computing; Resources

Introduction
A shared pool of reconfigurable computing resources (such as networks, servers, storage, applications, and other services) can be quickly provisioned and released with little management work or interaction with cloud service providers (CSPs) thanks to the cloud computing model [1]. Resource management, one of the main problems with cloud computing, tries to protect the heterogeneity and complexity of the underlying resources by using virtualization technology, which unites the vastly dispersed resources into a single, enormous resource pool. As a result, it can ensure the efficient provision and use of resources by putting resources management concepts and procedures into practice. Consequently, there are several additional hurdles to achieving effective administration of cloud computing resources. Multiple tenants have security risks to the cloud service provider (CSP): By using cloud services, malicious users may seize the chance to attack the CSP through data theft, vulnerability scanning, and denial of service (DoS) attacks, among other activities. Attacks, in which the malicious user can visit the CSP normally after an attack or begin another attack by generating a new pseudonym, are particularly dangerous if hostile users are permitted to log in using a pseudonym. Thankfully, many process apps can now be run in the cloud thanks to the development of clouds. The workflow applications serve as the mechanism for executing complex business processes, which are made up of a series of activities or tasks that transfer information from one activity to another.

International Journal for Innovative Engineering and Management Research
PEER REVIEWED OPEN ACCESS INTERNATIONAL JOURNAL
www.ijiemr.org

To accomplish a broad objective, another is based on a few technical rules [5]. The tasks in a workflow application are interdependent, with one task's output serving as another task's input. The load balancer must be able to dynamically adjust the server cluster to the current user access because the node servers of load equalizers for load balancing are virtual machines (VMs) of the cloud rather than fixed physical machines. This prevents resource waste and the scenario where the available resources are insufficient to fulfill user requests [1]. To be more specific, we contribute the following inthis work: For the high-dimensional and huge database, we suggest a new Dimensional layer approach concept of the DL-Layer (Dimensional layer). By splitting the input data's dimension, DL-Layer creates a divided dataset first and then builds a divided-convex process in each divided dataset. When compared to the convex process, the suggested method takes less time to establish an index and can create a list of layers in high-dimensional, huge datasets. To increase the accuracy of the DL-Layer, we suggest a technique called DLSelectAttribute that divides a dimension by choosing key attributes. The primary attributes of the supplied data are used by DL-SelectAttributes to partition the dimension. Through numerous trials, we demonstrate the benefits of DL-Layer in terms of performance.

The rest of this paper is organized as: Section 2 explains the previous works and Section 3 illustrates the proposed work. Section 4 explains the performance execution results of the proposed method;section 5 concludes the paper with future scope.

Previous Works:

Many technologies offer anonymity by encrypting the data that is outsourced to ensure security, according to a recent survey [11]. The data will be protected by the encryption method, but by activating security, it is unable to detect changes in the user's behavior. Accordingly, viruses and hacker incursion are the most unaddressed threat to the security of computer information storage. The following factors, along with poor key management and identity authentication systems, contribute to the aforementioned security issues [16]. Numerous studies on the topic of cloud service selection have been conducted, including approaches for rating-oriented collaborative filtering [4, 5],and approaches for ranking-based collaborative filtering [3], among others. To identify similar neighbors and forecast the missing QoS values, the model uses trust. In [5] personalized cloud service selection method uses value distribution and experience usability to find how similar the services are. Their suggested method delays the larger jobs for a longer period while completing the smaller activities first. For fog-cloud networks, the authors of [11] suggested a layered deadline-based resource allocation mechanism to accommodate changing user demands. A resource allocation model for vehicle cloud computing (VCC) networks was put forth by the authors in [12]. With the maximization of the acceptance rate and minimization of cloud provider cost as constraints, they formulate a multi-objective optimization problem. In [15], the authors presented an expansion of the Hungarian algorithm-based fog computing architecture, where the fog layer's function is to manage resource provision things to cut down on latency. In [17], the authors used the mobile cloud paradigm to create a fall detection application for elderly people and addressed the advantages of implementing healthcare apps on the abovementioned architecture. An energy-efficient cloud-based application for the continuous monitoring of patients in a persistent state by sensing their data was provided by the authors in [19]. The cloudlet is a cyber extension proposed that aims to take advantage of nearby public clouds' capabilities [8]. A compute offloading framework [13], is intended for mobile devices running the Android operating system. A framework for.NET code offloading called MAUI maximizes energy savings [15]. By taking advantage of the features of the.NET Common Language Runtime, code portability is accomplished. Another code offloading framework, CloneCloud, was created

International Journal for Innovative Engineering and Management Research
PEER REVIEWED OPEN ACCESS INTERNATIONAL JOURNAL
www.ijiemr.org

using a modified Android branch [16]. The framework must undergo these changes to facilitate dynamic profiling and migration. ThinkAir is another code offloading system that supports on-demand cloud resource allocation to overcome MAUI's scalability difficulties [17]. Distributed shared memory (DSM) is a new method of code offloading that COMET investigates [18].

Proposed Work:

Data leaking is one of the top cloud security concerns for enterprises, according to more than 60% of respondents. The development of technology in areas like network connectivity, distributed computing, and application computing is very helpful for the introduction of cloud computing. The user needs to authenticate to access the cloud-based files by offering user credentials, to the cloud. The user can use cloud services including file access, requests, and uploads after having his credentials verified. Additionally, based on his given position, the user has access to the files [27]. To do this, the cloud keeps track of user behavior in a database of users [28]. According to their positions within the organization, cloud administrators are responsible for determining each cloud user's job and allocating resources accordingly. The user is granted access to the requested files if he is authenticated and has the necessary access permissions [29].

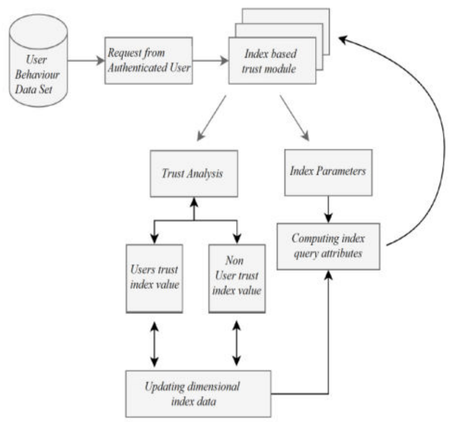

Figure 1: Architecture of Proposed Work

The proposed system's architecture is depicted in Figure 1, which accepts user requests that have been authenticated as input and process them using the trust module. The trust computing module continuously tracks users to determine how trustworthy they are. The amount of correctly attempted versus erroneously attempted activities is used to compute the trust value. Additionally, four criteria are used to evaluate trust computation: role violation, repeated fake requests, malicious program uploads, and duplicate uploads. Every user's computed trust value is updated. Considering the value of trust, the system determines if a user is allowed or prohibited access to the cloud. The system selects the user as malevolent if the user's computed trust exceeds a certain amount (threshold), however, it may infrequently do so due to the user's erroneous unsure actions. The user must then approach the client to cancel the authorization. Values for the trust computing parameters are cleaned from user behavior and described. In this work, we offer DL-Layer (Dimensional layer), a novel layer-based index creation technique for high-dimensional databases. As we indicated in the introduction, as the dimension of the data grows, the convex construction time grows exponentially. The convex is truly created as the number of layers decreases.

Algorithm 1:

Input:

I_x: a set of i-index objects

I: the number of query attributes

Output:

I_{dim}: a set of dimensional-index data

Algorithm:

If I < 8 then

Begin the process for I_x

Else

Begin the process for I_{dim} to generate I_{dim}:I_x

International Journal for Innovative Engineering and Management Research
PEER REVIEWED OPEN ACCESS INTERNATIONAL JOURNAL
www.ijiemr.org

While $I_x \neq \{0\}$, do

Index the I_{dim} into the number of I_x in I

Store the Ix objects in I_{dim}:I_x

Return I

By reducing the input data's dimension, the DL increases the number of layers, speeds up computation, and enhances the convex process. DL-Layer is created in three steps: Dimension division step one, divided-convex process construction step two, and merging step three. To construct data with a divided dimension at a divided dimension level, we first divide the input data's dimension. Based on the data with divided dimensions, we secondly generate a divided-convex process. To merge each divided-convex process at the merging level, we develop the final DL-Layer. The DL-Basic technique for dimension division is shown in algorithm 1.

Comparative Results and Discussions:
In this section, several test scenarios are created to assess the viability of the proposed work. We compare the proposed work with the user-based method (UBM) and the non-user-based method (NUBM) to examine the performance of the prediction. The rating-oriented approaches proposed work, NUBM, and UBM score the cloud services according to the expected QoS values. We use the cloud computing tool to create the experimental dataset because there aren't any better data sources that are suitable for enabling mobile cloud computing simulation. At first, a cloud service open-source package built on Java, we developed cloud service invocation codes.

Table 1: Dataset values of parameters considered for the analysis of the proposed work

Parameter	Dataset Value
Battery	0.5 KVA – 1 KVA
Throughput Value – Minimum	100 bps
Throughput Value – Maximum	1 Mbps
Bandwidth	2 M – 10 M
No. of cloud service users	100
No. of mobile cloud users	200
Memory	1 GB – 10 GB
Response Time – Minimum	0.005s
Response Time – Maximum	25s

It was simpler to create servers, mobile users, and service models withsupplementary network simulators. The alterations in the user context information are simulated to carry out our tests realistically. Information on the context and descriptions of the practical cloud service QoS values are shown in Table 1.

Table 2: Different Index Characteristics

Index	Resources	Roots	Data size (MB)
One	15	50	1.71
Two	30	110	1.62
Three	60	220	1.42
Four	100	2000	1.12

Four index-based test scenarios utilizing the workflow program were employed for this, each with a different number of resources, roots, and data quantities. Table 2 provides a summary of the features of the workflow program that was used in the tests. On the other side, Scenario Four illustrates the second case, where a lot of activities are added to the workflow to increase the search space.The huge search space makes it difficult for the optimization algorithm to discover the best answer. The PSO method, in contrast to the GA algorithm with Scenario Two and Three, exhibits a considerable difference in the results.

Table 3: Proposed Work result analysis

Method	Execution Time (sec)	Amount of Analysis (sec)	Bit Rate (bits/sec)
Index: One			
PSO	91	8	8
GA	187	42	16
Proposed Work	85	6	8
Index: Two			
PSO	145	52	16
GA	240	76	32
Proposed Work	106	39	16
Index: Three			
PSO	243	123	32
GA	335	127	64
Proposed Work	223	117	32
Index: Four			
PSO	1700	1100	64
GA	2300	1420	128
Proposed Work	1485	921	64

The GA and PSO algorithms also produce slightly different results for Scenarios Two and Three. In situation 4, the numerous resources cause the search space to be expanded to its maximum size, which is the worst-case situation.Comparing the GA algorithm to the PSO algorithm, a better outcome is still obtained. This outcome is attributable to the quick solution convergence that prevents unneeded solution diversity. Because the PSO algorithm typically gets in the local optimal solution, the PSO technique also showed a significant improvement over the PSO algorithm.Table 3 summarizes the results of averaging the execution times of each algorithm using a variety of tasks. The GA algorithm uses more CPU time than the other.

Conclusions and Future Scope:

We provide a clear understanding of the resource provisioning optimization problem in the cloud computing platform in this paper, and we then suggest a novel resource provisioning strategy based on DL. As a result of employing the DL algorithm, the following conclusions can also be made. (i) The number of CPUs in use, SLA conflicts, and costs are three areas where the DL algorithm surpasses the comparison method. (ii) The DL algorithm's performance is improved by using the library to reduce the complexity. (iii) The number of VCPU/CPU in the experiment is the state space; a smaller space has a faster convergence rate and a higher degree of adaptation. (iv) DL provisions in two circumstances: when the SLA is violated three times in a row, or when it's either too high or too low a utilization rate. On artificial datasets that varied in data size and dimension, we ran trials. According to experimental findings, the convex process was unable to generate an index across high-dimensional data as quickly as the proposed technique does. And because DL-Layer has more layers than a convex process, it is also more effective for handling top-k queries. We intend to expand our schemes in the future to reduce the proposed method's index-building time by hierarchically partitioning the dimension.

References:

1. Perumal, K., Mohan, S., Frnda, J. et al. Dynamic resource provisioning and secured file sharing using virtualization in cloud azure. J Cloud Comp 11, 46 (2022). https://doi.org/10.1186/s13677-022-00326-1
2. Tarafdar A, Debnath M, Khatua S, Das RK (2020) Energy and quality of service-aware virtual machine consolidation in a cloud data center. J Supercomput 76(11):9095–9126
3. Stanly Jayaprakash J, Balasubramanian K, Sulaiman R, Hasan MK, Parameshachari BD, Iwendi C (2022) Cloud data encryption and authentication based on enhanced Merkle hash tree method. Computers, Materials Continua 72(1):519–534.
4. Sayadnavard MH, Haghighat AT, Rahmani AM (2019) A reliable energy-aware approach for dynamic virtual machine consolidation in cloud data centers. J Supercomput 75(4):2126–2147
5. Mao B, Yang Y, Wu S, Jiang H, Li KC (2019) IOFollow: improving the performance of VM live storage migration with IO following in the cloud. FuturGenerComputSyst 91:167–176.
6. Agarwal A, Duong TN (2019) Secure virtual machine placement in cloud data centers. FuturGenerComputSyst 100:210–222
7. Reena Panwar, M. Supriya, "Dynamic resource provisioning for service-based cloud applications: A Bayesian learning approach", Journal of Parallel and Distributed Computing, Volume 168, 2022, Pages 90-107
8. D. Singh and M. Chitkara, "Virtualized Secure Cloud Computing Using Targetive Dynamic Secured Resource Allocation Algorithm," 2023 International Conference on Distributed Computing and Electrical Circuits and Electronics (ICDCECE), Ballar, India, 2023, pp. 1-4, doi: 10.1109/ICDCECE57866.2023.10150471.
9. L. -H. Hung, C. -H. Wu, C. -H. Tsai and H. -C. Huang, "Migration-Based Load Balance of Virtual Machine Servers in Cloud Computing by Load Prediction Using Genetic-Based Methods," in IEEE Access, vol. 9, pp. 49760-49773, 2021, doi: 10.1109/ACCESS.2021.3065170.
10. Y. -L. Chen, S. -Y. Huang, Y. -C. Chang and H. -C. Chao, "Resource Allocation based on Genetic Algorithm for Cloud Computing," 2021 30th Wireless and Optical Communications Conference (WOCC), Taipei, Taiwan, 2021, pp. 211-212, doi: 10.1109/WOCC53213.2021.9603125
11. A. Khodar, H. A. F. Al-Afare and I. Alkhayat, "New Scheduling Approach for Virtual Machine Resources in Cloud Computing based on Genetic Algorithm," 2019 International Russian Automation Conference (RusAutoCon), Sochi, Russia, 2019, pp. 1-5, doi: 10.1109/RUSAUTOCON.2019.8867638.
12. A. A. Nayak and S. Shetty, "A Systematic Analysis on Task Scheduling Algorithms for Resource Allocation of Virtual Machines on Cloud Computing Environments," 2023 International Conference on Recent Trends in Electronics and Communication (ICRTEC), Mysore, India, 2023, pp. 1-6, doi: 10.1109/ICRTEC56977.2023.10111894.
13. A. Belgacem, K. Beghdad-Bey and S. Mahmoudi, "New virtual machine placement approach based on the micro genetic algorithm in cloud computing," 2021 8th International Conference on Future Internet of Things and Cloud (FiCloud), Rome, Italy, 2021, pp. 66-72, doi: 10.1109/FiCloud49777.2021.00017.
14. V. A. Lepakshi and C. S. R. Prashanth, "Efficient Resource Allocation with Score for Reliable Task Scheduling in Cloud Computing Systems," 2020 2nd International Conference on Innovative Mechanisms for Industry Applications

(ICIMIA), Bangalore, India, 2020, pp. 6-12, doi: 10.1109/ICIMIA48430.2020.9074914

15. H. Khan et al., "An Efficient Scheduling based cloud computing technique using virtual Machine Resource Allocation for efficient resource utilization of Servers," 2020 International Conference on Engineering and Emerging Technologies (ICEET), Lahore, Pakistan, 2020, pp. 1-7, doi: 10.1109/ICEET48479.2020.9048233
16. A. Belgacem, K. Beghdad-Bey and S. Mahmoudi, "New virtual machine placement approach based on the micro genetic algorithm in cloud computing," 2021 8th International Conference on Future Internet of Things and Cloud (FiCloud), Rome, Italy, 2021, pp. 66-72, doi: 10.1109/FiCloud49777.2021.00017.
17. S. A. Shamaa, W. Shi and G. Ankenmann, "Probability-Based Crossover Genetic Algorithm for Task Scheduling in Cloud Computing," 2023 6th Conference on Cloud and Internet of Things (CIoT), Lisbon, Portugal, 2023, pp. 231-238, doi: 10.1109/CIoT57267.2023.10084877.
18. Kumar, V. & Ramana, T.. (2022). Fully scheduled decomposition channel estimation based MIMO-POMA structured LTE. International Journal of Communication Systems. 35. 10.1002/dac.4263.