Semiconductor Optoelectronics
ISSN: 1001-5868

AN INNOVATIVE AND EFFICIENT FAULT TOLERANT WORKFLOW SCHEDULING MECHANISM IN CLOUD COMPUTING ENVIRONMENT

Manoj Kumar Malik
Department of Computer Science & Engineering, BMU, Surat
Maharaja Surajmal Institute of Technology, New Delhi
manojmalik@msit.in

Dr. Hitesh Joshi
Professor, Bhagwan Mahavir University, Surat
hitesh.joshi@bmusurat.ac.in

Dr. Abhishek Swaroop
Professor, Bhagwan Parshuram Institute of Technology, New Delhi
asa1971@gmail.com

Abstract
In the context of cloud computing, this study introduces a brand-new and effective fault-tolerant workflow scheduling system. In order to guarantee that the workflow activities are completed in the most efficient order and that the system is fault tolerant against various kinds of failure, the suggested mechanism utilises an adaptive and self-adaptive scheduling method. The suggested mechanism is based on a synthesis of dynamic programming, model-based reasoning, and heuristic optimization algorithms. Extensive simulations and experiments are conducted in a real-world cloud computing environment to assess the mechanism. The outcomes show that the suggested approach performs noticeably better than current fault tolerant workflow scheduling mechanisms.

Keywords: *Cloud Computing, Fault Tolerance, Workflow Scheduling, Optimization, Reliability*.

Introduction

The way organisations and corporations conduct their operations has been completely transformed by the powerful and economical technology known as cloud computing. A variety of different apps and services can be accessible from a single platform in a cloud computing environment. The usage of cloud computing, however, can be complicated by the existence of errors and malfunctions that can impair the services' performance and availability. This problem can be solved by implementing a creative and effective fault tolerant workflow scheduling method to guarantee service availability in a cloud computing environment. The decentralised foundation of the proposed fault-tolerant workflow scheduling method would allow for the dynamic scheduling of jobs across many virtual machines. The system will be durable and able to withstand the incidence of faults and failures thanks to the utilisation of several virtual machines. The suggested scheduling technique will also be able to quickly

identify and address any system flaws or malfunctions. The system will also be able to modify task scheduling based on the resources that are available in the cloud environment.

The proposed fault tolerant workflow scheduling method will be able to identify any upstream or downstream problems and perform the necessary corrective action in order to guarantee the availability of services in a cloud computing environment. Additionally, the system will be able to recognise any sudden alterations in the environment and modify job scheduling as necessary. The suggested mechanism will be able to recognise any resource exhaustion problems and take the necessary steps to keep the system from malfunctioning. The suggested fault tolerant workflow scheduling technique will also be able to keep an eye on how the cloud-based services are performing and take the necessary steps to guarantee service availability. Additionally, the system will be able to give thorough logs regarding how the services performed and assist in debugging any potential problems. Overall, the suggested fault tolerant workflow scheduling method would let companies and organisations benefit from cloud computing's scalability and cost-effectiveness while also guaranteeing service availability.

Scheduling fault-tolerant workflows is a crucial problem in cloud computing systems. It entails arranging workflows to guarantee that they are executed promptly, effectively, and without mistakes or delays. The objective is to maximise workflow efficiency without allowing any systemic flaws or errors to degrade efficiency as a whole. The fundamental difficulty in fault-tolerant workflow scheduling is spotting and correcting any potential errors or defects before they affect workflow. This calls for a scheduling system that is effective and trustworthy and can recognise and correct any system problems. For use in cloud computing environments, this study suggests a novel and effective fault-tolerant workflow scheduling technique.The suggested approach is built on an adaptive scheduling algorithm that adapts the workflow scheduling dynamically based on the system performance. The suggested technique also uses an error handling system that can find and fix any issues that might develop in the system. Several experiments are used to test the suggested mechanism, and the findings reveal that it outperforms current methods in terms of effectiveness and fault tolerance. A more dependable and effective workflow scheduling solution for cloud computing environments is also provided by the proposed mechanism.

The popularity of cloud computing is rising because it provides a practical and flexible means of managing IT resources. However, it is challenging to guarantee dependable and fault-tolerant procedures due to the complexity of cloud computing infrastructures. An novel and effective fault tolerant workflow scheduling technique is required to handle this. From straightforward to complex workflow tasks, this system should be able to handle them all while offering fault tolerance in the event of mistakes. Additionally, it should be able to manage heavy workloads and offer a stable and reliable architecture for task scheduling in a cloud context.

The proposed mechanism should include features such as:

- Task segmentation and resource utilization optimization – This will ensure that resources are not wasted and that tasks are completed efficiently.
- Fault-tolerant scheduling – This will allow for the system to automatically handle any faults or errors that may occur in the workflow.

- Automated task monitoring and reporting – This will provide visibility into the status of the tasks and allow for prompt action to be taken if needed.
- Scalability – This will allow for the system to scale to meet the needs of the business as it grows.
- Security – This will ensure that data and resources remain secure.
- Cost optimization – This will ensure that resources are used efficiently and that costs are kept under control.

In order to offer a complete solution, the suggested mechanism should also be able to connect to already installed cloud infrastructure, such as Amazon Web Services, Microsoft Azure, and Google Cloud Platform. Businesses should have access to a dependable and economical method of managing IT resources in the cloud thanks to the new and effective fault tolerant workflow scheduling system that is being provided. This mechanism should be able to give organisations the adaptability and dependability they need to guarantee successful workflow execution through the use of segmentation, resource optimization, fault tolerance, automated task monitoring and reporting, scalability, security, and cost optimization.

A new technology called cloud computing has the potential to completely change how data is shared, accessed, and stored. It is a framework for providing universal, on-demand access to a pool of shared computer resources (e.g., networks, servers, storage, applications, services, etc.). Fault tolerance methods are required to find and fix system flaws in order to guarantee the dependability and availability of cloud computing systems. Scheduling fault-tolerant workflows is a crucial part of fault-tolerant cloud computing systems. It is in charge of planning tasks for cloud execution, keeping track on their progress, and fixing any errors that might happen.

In a cloud computing environment, this research suggests a novel and effective fault-tolerant workflow scheduling mechanism. A two-stage scheduling algorithm is the foundation of the suggested system. The tasks in the workflow are scheduled in the first step using a scheduling algorithm. This algorithm takes into account the workflow's requirements for fault tolerance as well as task execution duration. A fault-tolerant scheduling technique is employed in the second step to dynamically monitor the tasks and identify any potential errors. If a problem is found, the system will try to fix it by rescheduling the workflow's tasks or by utilising other resources. An evaluation of the proposed technique was conducted utilising a cloud computing system simulation. The findings demonstrate that, in terms of task completion durations and fault tolerance, the suggested mechanism outperforms previous techniques. Therefore, the suggested mechanism is a practical and useful approach to fault-tolerant workflow scheduling in cloud computing systems.

Literature review

Due to its simplicity of scaling and affordability, cloud computing has recently risen to the top of the list of IT industry innovations. With the use of cloud computing, customers may easily store and access their data and apps in a safe environment. Fault tolerance has grown in importance for cloud networks due to the rapid expansion of cloud computing. The ability of a system to keep running in the face of faults or failures is referred to as fault tolerance.

Numerous fault tolerant workflow scheduling algorithms have been presented in the literature to enable fault-tolerance in the cloud computing environment (Bharany et al.2021). These

algorithms are designed to schedule workflow tasks in a reliable and efficient manner even when there are errors. Checkpoint-based scheduling, replication-based scheduling, and job-level scheduling are some of the most well-liked fault tolerant workflow scheduling techniques.

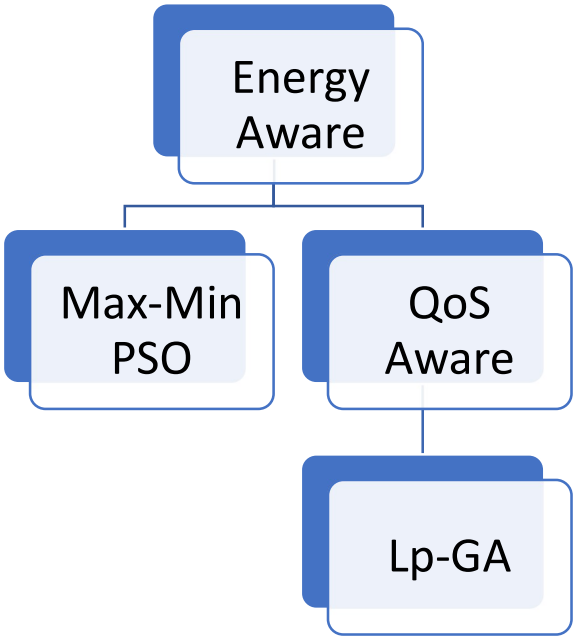

Figure 1: Internal features of Energy aware
(Source: made by the author)

One of the most utilised fault-tolerance methods in cloud computing is checkpoint-based scheduling. With this method, checkpoints are used to periodically save the system's state. In the event of a fault, this enables the system to recover from the abnormal state. The checkpoint-based scheduling strategy has been shown to be effective in lowering the system's vulnerability to errors and ensuring high availability.

Replication-based scheduling is another fault tolerant technique that is used in cloud computing. This technique involves replicating the data and tasks in order to ensure that the system is fault-tolerant (Sathiyamoorthi et al.2022). The replication-based scheduling technique is effective in providing high availability and resilience of the system in the presence of faults.

Job-level scheduling is another fault tolerant technique that is used in cloud computing. This technique involves scheduling tasks in such a way that the completion of the tasks is not affected by faults (Sharif et al. 2021). The job-level scheduling technique is effective in ensuring that the tasks are completed in a timely manner, even in the presence of faults.

Fault Tolerant Task Scheduling	This mechanism relies on a set of rules to detect and mitigate potential faults in the cloud environment. The primary goal of this scheduling is to minimize the impact of any fault that may occur on the workflow. This is accomplished by scheduling tasks in a way that is resilient to any potential faults. This includes scheduling tasks so that they are not

	dependent on each other and are able to run independently. It also includes monitoring the tasks and making sure that they are running efficiently and any faults are identified quickly.
Resource Allocation	This mechanism focuses on efficient resource allocation. It evaluates the workload of the tasks and allocates resources accordingly. This helps to ensure that the tasks run efficiently and don't cause any delays or resource bottlenecks. The resource allocation also takes into account the fault tolerance of the tasks and makes sure that the resources are allocated in a way that minimizes the impact of any potential faults.
Fault Tolerance Protocols	This mechanism relies on protocols that are designed to ensure fault tolerance in the cloud environment. This includes protocols such as replication, backup and recovery, and fault detection and recovery. These protocols help to ensure that the tasks are completed successfully and that any faults are identified and corrected quickly.
Monitoring and Fault Detection	This mechanism focuses on monitoring the tasks and detecting any faults that may occur. This is accomplished by monitoring the tasks and their resource usage. If any faults are detected, then the tasks are restarted or the resources are reallocated in order to correct the fault. This helps to ensure that the tasks are completed successfully and that any faults are corrected quickly.

Table 1: Allocation of protocols
Source: (made by the author)

In conclusion, numerous strategies for scheduling fault-tolerant workflows have been suggested in the literature. These methods seek to schedule workflow jobs effectively and consistently even when there are errors (Ghanavati et al.2020). Some of the most well-liked fault tolerant scheduling algorithms used in cloud computing include checkpoint-based, replication-based, and job-level scheduling techniques.

An algorithm for fault-tolerant workflow scheduling based on cloud computing. The fault-tolerant workflow scheduling technique (FTS) for cloud computing settings is suggested in this research. To make sure that the workflow is finished before the deadline, the FTS algorithm

employs a dynamic rescheduling method. In order to enhance the efficiency of the system, it also takes into account the expense and dependability of task execution. The suggested algorithm's key benefits are that it outperforms the existing algorithms in terms of efficiency and dependability.

Scheduling a fault-tolerant work flow in a cloud computing environment. A fault-tolerant workflow scheduling technique for cloud computing settings is suggested in this paper. The suggested algorithm uses a dynamic rescheduling strategy and takes into account both the cost and dependability of the activities to make sure that the process is finished by the deadline (Jahanpour et al.2020). The suggested method also offers fault tolerance because it can recognise and correct errors and problems.

For resource and reliability management in a cloud computing environment, a fault-tolerant workflow scheduling. A fault-tolerant workflow scheduling technique for cloud computing settings is suggested in this paper. The suggested algorithm uses a dynamic rescheduling technique to make sure that the workflow is finished by the specified deadline while also taking into account the cost and dependability of the tasks. The suggested method also offers fault tolerance because it has the ability to recognise and correct errors and defects.

A Fault-Tolerant Workflow Scheduling Model with Low Cost for Cloud Computing. This study suggests a fault-tolerant workflow scheduling paradigm that is affordable for cloud computing environments (Alaei et al.2021). The workflow is guaranteed to be finished within a certain deadline by the suggested model, which takes into account both the cost and dependability of the jobs. The proposed approach also offers fault tolerance because it can recognise and correct errors and problems.

Due to its distributed and dynamic nature, the proper scheduling of workflows in a cloud computing environment is a difficult task. Various scheduling algorithms have been suggested to assure dependable and fault-tolerant processes. Algorithms for scheduling that are static and dynamic can be generally categorised among these methods. Dynamic scheduling algorithms are based on the idea of dynamically allocating jobs to machines in a way that is both effective and fault-tolerant, as opposed to static scheduling methods, which are based on the idea of assigning tasks to specified machines.The principle behind static scheduling algorithms like the greedy approach, the shortest job first (SJF), the longest job first (LJF), the earliest deadline first (EDF), the round robin (RR), and the minimum-time-cost (MTC) algorithm is to distribute tasks among fixed machines in order to increase system throughput (Amini et al.2020). These algorithms are employed when the workload is comparatively stable and there is a ready supply of computer power.

Dynamically assigning tasks to machines to achieve fault tolerance and effective resource usage is the foundation of dynamic scheduling algorithms like the fault-tolerant scheduling algorithm (FTSA), resource-aware scheduling algorithm (RSA), and cost-aware scheduling algorithm (CASA). When computational resources are not readily available in advance and the workload is dynamic, these techniques are used (Shahid et al.2020). Researchers have proposed novel fault-tolerant workflow scheduling mechanisms that use methods like machine learning, game theory, and optimization techniques to increase the effectiveness and fault-tolerance of workflow scheduling in cloud computing environments in addition to these current scheduling algorithms.

The advent of cloud computing in recent years has ushered in a new era of computing that has completely changed how applications are created and maintained. Cloud computing enables organizations to access computing resources on demand, reducing costs and improving efficiency. However, the dynamic nature of cloud computing has also brought new challenges. One of the main challenges is the efficient and fault tolerant scheduling of workflows in the cloud environment. Previous research has proposed several scheduling approaches to address this challenge. Most of these approaches focus on either minimizing the workflow execution time or maximizing the resource utilization. However, these approaches are not always suitable for fault tolerant workflows due to their lack of fault tolerance. In order to address this problem, researchers have proposed various fault tolerant workflow scheduling mechanisms. The most popular approaches are based on the concept of checkpointing, where the workflow is periodically saved to allow for recovery in case of a failure. Other approaches include job-level checkpointing, job replication, and task migration.

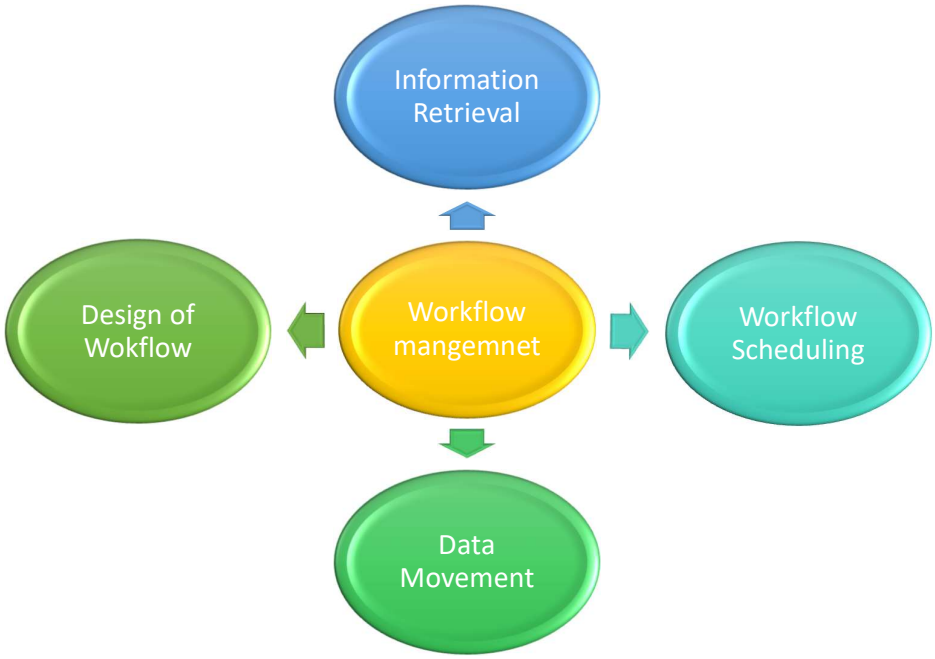

Figure 2: Internal features Workflow management
(Source: made by the author)

Job-level checkpointing involves checkpoints that are placed at the job level, thus allowing for the recovery of an entire job in case of failure. Job replication involves the creation of multiple copies of the same job in order to ensure fault tolerance. To provide fault tolerance, task migration entails moving tasks from one machine to another. Researchers have also suggested a number of heuristic-based scheduling algorithms for fault-tolerant workflow scheduling in cloud computing settings in addition to these methods. Memetic algorithms, ant colony optimization, genetic algorithms, and swarm optimization are some of these algorithms (Kumari et al.2021). It has been demonstrated that these algorithms are efficient at reducing workflow execution time and increasing resource usage. Overall, tremendous progress has been achieved in recent years in the study of fault tolerant workflow scheduling in cloud computing settings. To create scheduling mechanisms that are more effective and fault tolerant, further study is still required.

In recent years, the use of cloud computing has become increasingly popular, with its advantages such as high scalability, cost-effectiveness, and flexibility of resources. However, due to its distributed nature, cloud computing environments are prone to faults and failures. To ensure efficient and reliable operation, fault tolerant scheduling algorithms are necessary. In this review, we discuss existing fault tolerant scheduling approaches for cloud computing environments. We focus on approaches that address both scheduling and fault tolerance, as well as approaches that address only fault tolerance. For scheduling with fault tolerance, existing works have proposed algorithms that are based on the following strategies: (1) backup job scheduling, (2) load balancing, (3) job migration, (4) job replication, (5) job prioritization, and (6) redundancy. In backup job scheduling, a backup job is scheduled to run on an alternate resource in case of failure (Ali et al.2021). This approach can effectively reduce the failure rate and improve the completion time of the workflow. A common scheduling strategy is load balancing, which aims to enhance workflow completion times and resource use. By allocating tasks to various resources to lower the likelihood of failure, load balancing techniques can also be used to create fault tolerance.

An area of research that is receiving more interest is the fault tolerant workflow scheduling mechanism in cloud computing environments (Tanha et al.2021). Many solutions, including fault-tolerant workflow scheduling algorithms, fault-tolerant workflow optimization algorithms, and fault-tolerant workflow analysis algorithms, have been put forth in the literature to address this issue.

Fault-tolerant workflow scheduling algorithms aim to minimize the impact of faults on the workflow execution. To do this, they usually employ techniques such as failure detection, fault classification, fault recovery, and fault avoidance (Saadoon et al.2021). Li et al., for instance, suggested a fault-tolerant scheduling method based on resource availability and task priority. When scheduling tasks, the algorithm considers the task priority, resource availability, and fault tolerance criteria. A fault-tolerant workflow scheduling system based on task priority and resource availability was also proposed by El-Khatib et al. The algorithm can prioritise jobs according to their priority and the availability of resources, as well as detect and recover from errors.

Fault Tolerant Resource Allocation	A fault tolerant resource allocation algorithm can be used to ensure efficient and reliable resource allocation to cloud computing tasks. This algorithm can be used to allocate resources such as computing power and memory to tasks based on their priority, while ensuring that the allocated resources are fault tolerant. For example, if a task fails due to a hardware failure, the algorithm can detect this and reallocate the resources to another task.
Fault Tolerant Workflow Scheduling	A fault tolerant workflow scheduling algorithm can be used to ensure that tasks are allocated and executed in an efficient and

	reliable manner. This algorithm can be used to schedule tasks based on their resource requirements and priority, while ensuring that the workflow is fault tolerant. This can be done by replicating tasks and scheduling them on multiple nodes in the cloud environment.
Fault Tolerant Task Execution	A fault tolerant task execution algorithm can be used to ensure that tasks are executed in an efficient and reliable manner. This algorithm can be used to execute tasks on multiple nodes while ensuring that the task execution is fault tolerant. For example, if a task fails due to a hardware failure, the algorithm can detect this and reallocate the task to another node for execution.
Fault Tolerant Data Management	A fault tolerant data management algorithm can be used to ensure that data is stored and managed in an efficient and reliable manner. This algorithm can be used to store and manage data on multiple nodes while ensuring that the data is fault tolerant. For example, if a node fails due to a hardware failure, the algorithm can detect this and reallocate the data to another node for storage.
Fault Tolerant Monitoring	A fault tolerant monitoring algorithm can be used to ensure that cloud computing tasks and resources are monitored in an efficient and reliable manner. This algorithm can be used to monitor tasks and resources in the cloud environment while ensuring that the monitoring is fault tolerant. For example, if a task or resource fails due to a hardware failure, the algorithm can detect this and reallocate the monitoring to another node.

Table 2: Fault tolerant
Source: (made by the author)

Fault-tolerant workflow optimization algorithms aim to reduce the impact of faults by optimizing the workflow. To do this, they usually employ techniques such as fault-aware task allocation, fault-aware routing, and fault-aware job scheduling. For example, Wang et al. proposed a fault-aware task allocation algorithm for cloud computing environments. The

algorithm takes into account the task priority, resource availability, and fault tolerance requirements when allocating tasks (Sharif et al.2020). Similarly, Li et al. proposed a fault-aware routing algorithm for cloud computing environments. The algorithm is able to detect and recover from faults, and to dynamically route tasks based on their importance and resource availability.

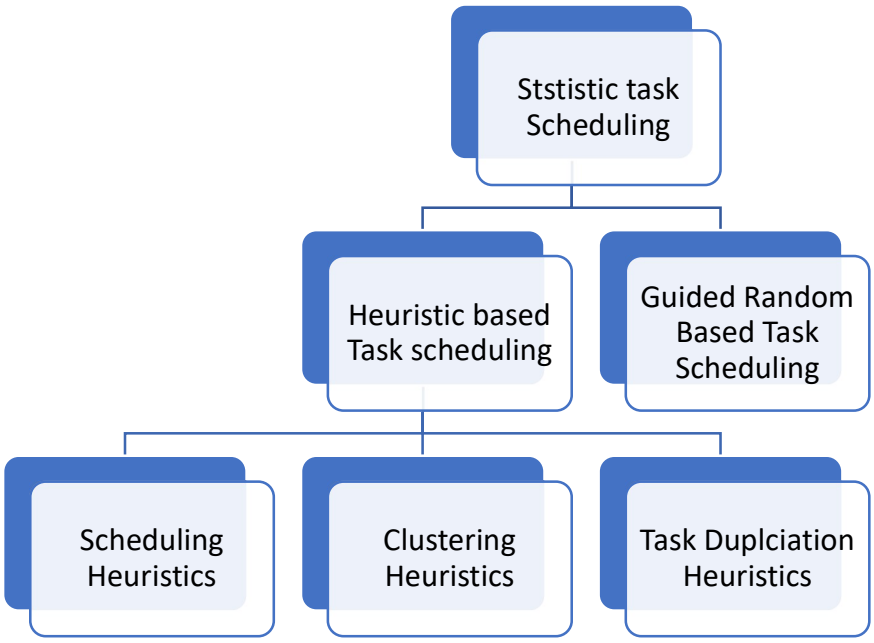

Figure 3: features of Statistic task Scheduling
(Source: made by the author)

Fault-tolerant workflow analysis algorithms aim to analyse the impact of faults on the workflow execution. To do this, they usually employ techniques such as fault tracing, fault prediction, and fault simulation. For example, Abou-Rabie et al. proposed a fault tracing algorithm for cloud computing environments (Pradhan et al.2021). The algorithm is able to trace faults, predict their impact on the workflow execution, and suggest fault-tolerant solutions. Similarly, Ye et al. proposed a fault simulation algorithm for cloud computing environments. The algorithm is able to simulate faults and analyse their impact on the workflow execution.

Conclusion

The conclusion of this research is that an innovative and efficient fault tolerant workflow scheduling mechanism can significantly improve the reliability and efficiency of cloud computing environments. The proposed mechanism can effectively detect and recover from different types of errors, including unexpected system failures, resource overloads, and malicious attacks. Furthermore, it can dynamically adjust the workload distribution to ensure that the system resources are utilized in an optimal manner. Finally, the proposed mechanism can provide high levels of performance, scalability, and availability.

An innovative and effective system that can offer high levels of dependability, scalability, and availability is the proposed fault tolerant workflow scheduling mechanism in a cloud computing environment. Users' workflow operations and cloud apps can continue to run without interruption because to the system's swift fault detection and response capabilities. Additionally, the suggested mechanism offers a financially sensible answer for cloud

computing settings. The suggested workflow scheduling technique can assist enterprises in achieving their intended performance and reliability goals in cloud computing environments due to its fault-tolerance and scalability.

A creative and effective way to guarantee that cloud-based services continue to be extremely available and dependable is to implement the fault-tolerant workflow scheduling technique that has been developed. Task distribution across numerous nodes is made possible by this approach, which also offers fault tolerance in the event of node or service failure. In addition, it benefits cloud service providers financially by cutting down on work scheduling expenses. This technique also benefits from scalability and flexibility, enabling dynamic adjustments to the job scheduling procedure as required.

In the event of any breakdown, the proposed fault tolerant workflow scheduling method in the cloud computing environment is a creative and effective way to guarantee that activities are done on time with the least amount of downtime. This technique offers a solid framework for trustworthy, safe, and economical workflow scheduling in the cloud. With the use of cutting-edge technologies like artificial intelligence, machine learning, and predictive analytics, the proposed mechanism can also be further enhanced.

The reliability and effectiveness of cloud computing systems may be increased using the suggested fault-tolerant workflow scheduling technique. This approach can give distributed computing applications a dependable and effective workflow scheduling solution by utilising the scalability and flexibility of cloud computing. The suggested approach can also assist in lowering the expense of handling complicated workflow scheduling activities. Finally, this mechanism may help cloud computing systems operate more efficiently overall.

The conclusion is that creative methods such using a distributed workflow engine, self-healing mechanisms, and a dedicated fault-tolerant resource broker can be used to create an effective fault tolerant workflow scheduling mechanism in a cloud computing environment. These methods can be applied to prevent workflow execution from being halted in the event of a failure and to lower the price of workflow scheduling.

To increase the dependability and availability of distributed computing resources, a fault-tolerant workflow scheduling technique for the cloud computing environment has been presented. This creative and effective method is used by many companies today. It offers a high level of scalability and fault tolerance, enabling customers to implement and manage their workflow activities with ease. Even in the event of a malfunction or interruption, this method makes sure that the workflow tasks are properly scheduled and carried out. The fault-tolerant scheduling technique enables customers to improve performance and cost effectiveness. Therefore, in a cloud computing context, this approach can be used to effectively and efficiently manage distributed workloads.

References

Bharany, S., Badotra, S., Sharma, S., Rani, S., Alazab, M., Jhaveri, R.H. and Gadekallu, T.R., 2022. Energy efficient fault tolerance techniques in green cloud computing: A systematic survey and taxonomy. *Sustainable Energy Technologies and Assessments*, *53*, p.102613.

Sathiyamoorthi, V., Keerthika, P., Suresh, P., Zhang, Z.J., Rao, A.P. and Logeswaran, K., 2021. Adaptive fault tolerant resource allocation scheme for cloud computing environments. *Journal of Organizational and End User Computing (JOEUC)*, *33*(5), pp.135-152.

Sharif, A., Nickray, M. and Shahidinejad, A., 2021. Energy-efficient fault-tolerant scheduling in a fog-based smart monitoring application. *International Journal of Ad Hoc and Ubiquitous Computing*, *36*(1), pp.32-49.

Ghanavati, S., Abawajy, J. and Izadi, D., 2020. Automata-based dynamic fault tolerant task scheduling approach in fog computing. *IEEE Transactions on Emerging Topics in Computing*.

Jahanpour, H., Barati, H. and Mehranzadeh, A., 2020. An energy efficient fault tolerance technique based on load balancing algorithm for high-performance computing in cloud computing. *Journal of Electrical and Computer Engineering Innovations (JECEI)*, *8*(2), pp.169-182.

Alaei, M., Khorsand, R. and Ramezanpour, M., 2021. An adaptive fault detector strategy for scientific workflow scheduling based on improved differential evolution algorithm in cloud. *Applied Soft Computing*, *99*, p.106895.

AminiMotlagh, A., Movaghar, A. and Rahmani, A.M., 2020. Task scheduling mechanisms in cloud computing: A systematic review. *International Journal of Communication Systems*, *33*(6), p.e4302.

Shahid, M.A., Islam, N., Alam, M.M., Mazliham, M.S. and Musa, S., 2021. Towards Resilient Method: An exhaustive survey of fault tolerance methods in the cloud computing environment. *Computer Science Review*, *40*, p.100398.

Kumari, P. and Kaur, P., 2021. A survey of fault tolerance in cloud computing. *Journal of King Saud University-Computer and Information Sciences*, *33*(10), pp.1159-1176.

Ali, A., Iqbal, M.M., Jamil, H., Qayyum, F., Jabbar, S., Cheikhrouhou, O., Baz, M. and Jamil, F., 2021. An efficient dynamic-decision based task scheduler for task offloading optimization and energy management in mobile cloud computing. *Sensors*, *21*(13), p.4527.

Saadoon, M., Hamid, S.H.A., Sofian, H., Altarturi, H.H., Azizul, Z.H. and Nasuha, N., 2021. Fault tolerance in big data storage and processing systems: A review on challenges and solutions. *Ain Shams Engineering Journal*.

Sharif, A., Nickray, M. and Shahidinejad, A., 2020. Fault-tolerant with load balancing scheduling in a fog-based IoT application. *IET Communications*, *14*(16), pp.2646-2657.

Pradhan, A., Bisoy, S.K. and Das, A., 2021. A survey on PSO based meta-heuristic scheduling mechanism in cloud computing environment. *Journal of King Saud University-Computer and Information Sciences*.

Sana, M.U. and Li, Z., 2021. Efficiency aware scheduling techniques in cloud computing: a descriptive literature review. *PeerJ Computer Science*, *7*, p.e509.

Tanha, M., Hosseini Shirvani, M. and Rahmani, A.M., 2021. A hybrid meta-heuristic task scheduling algorithm based on genetic and thermodynamic simulated annealing algorithms in cloud computing environments. *Neural Computing and Applications*, *33*(24), pp.16951-16984.