ISSN:0975-9646

Comparative Analysis of Fault Tolerance Techniques in Cloud Computing

Deepak K Gaur[#1], Ajitabh Mahalkari[*2]

[#] *Department of Computer Science and Engineering*

Sushila Devi Bansal College of Technology, Indore(M.P)-45333, India

1deepakme153@gmail.com

2ajitabh.mahalkari@sdbct.ac.in

***Abstract*— Cloud computing adaptation is on peaks, organizations rely on the third party infrastructure of cloud service providers. Every business needs high availability and resilience against the faults. Fault tolerance is an approach to tolerate and avoid faults and errors in the system after its development and implementation. This work is presenting, detailed discussion, about the types of faults and various types of fault tolerance techniques and comparison of various fault tolerance models. This paper is indented to develop a comparative analysis of fault tolerance models so that advantages of the models can be leveraged and also highlight the shortcomings of the available models, can be extended for further research in the field of cloud computing.**

***Keywords*— Cloud computing, fault tolerance, failures, proactive, reactive, adaptive**

I. INTRODUCTION

Cloud computing is a model for provide more flexibility, shared pool of resources to get on demand network access to configurable computing resources (e.g., networks, servers, storage, applications, and services), which can be rapidly provisioned and released with minimal management effort or service provider interaction. After cloud computing technology various types of services were introduced so that the consumer can get access over the network. Several characteristics like cost-effectiveness, scalability, reliability and flexibility of cloud computing makes impact on users to adopt this technology.

Cloud Computing is getting popular day by day, and also proved to be useful to organization, due to this there is a tremendous need of fault tolerance in cloud computing. Fault tolerance is group of actions and events that require avoiding the impact of the fault occurred in the system. Fault tolerance includes techniques to make system robust enough to handle the faults

Key motivation of using fault tolerance in cloud computing may includes failure recovery, lower costs, and improvised performance. Cloud computing has become first choice to design, develop and deploy distributed applications because of the features like virtualization, flexibility and elasticity. Quality of service is been managed by cloud computing in view of dependability and availability.

As it is clear that fault tolerance is one of the key business requirements of cloud computing, is the reason of motivation to involve more researchers to develop efficient fault tolerance strategies. Fault tolerance is an important key issue in cloud computing that why amount of research is being carried out on fault tolerance in cloud computing

Fault tolerance basically concerns with the policies and techniques to avoid faults in the system so that business operations do not face the down time. Client need continuity and availability, which can interrupted due to fault in the system.

Faults in the cloud computing may include hardware faults, software faults, network faults, process faults etc. Fault tolerance policies may include reactive fault tolerance, proactive fault tolerance, adaptive fault tolerance etc.

Main goal of the review is to find out various existing fault tolerance techniques and models available in cloud computing. Different approaches will be compared on different aspects, and intended to choose the best suited solution and also to extend the work done by the researchers in the past. This paper is planned to deliberates about various aspect of faults and the need of fault tolerance in cloud computing.

II. OVERVIEW OF FAULT TOLERANCE IN CLOUD COMPUTING

Fault tolerance leads to right and nonstop operation even within the sight of faulty components. It is the art of making computing systems that keep on operating sufficiently within the sight of faults. A fault tolerant system might have the option to tolerate at least one fault types including-transient, intermittent or permanent equipment faults, software and design errors, operator errors, or externally incited upsets or physical damage. In real time cloud applications, handling on computing nodes is done remotely which has a likely events of errors.

These occasions increment the requirement for fault tolerance techniques to accomplish reliability for the real time computing on cloud infrastructure. The connection between faults, error and failure is depicted in the figure 1.1. Each is relied upon one another. Events of fault either in the part (Hardware) or in designing methodologies (Software) inclined to errors leads to failure.

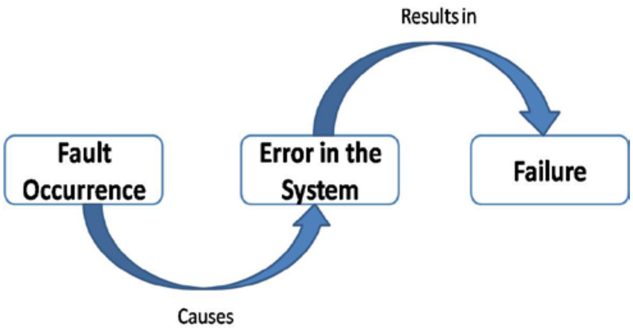


Figure 1.1 depicting relationship between fault, errors and failure.

- ***Types of Faults***

The faults and their causes can be classified based on several factors.

Fault Type	Cause of occurrence
Network fault	Network partition, Congestion, packet loss, packet corruption, destination failure, link failure
Physical faults	Fault in CPUs, Fault in memory, Fault in storage, Power failure
Media faults	Storage media hardware faults, Storage to compute communication faults
Process faults	Shortage of resource, software bugs, inefficient processing capabilities
Service expiry fault	Timing failure, omission failure, response failure and crash failure. Failures may be permanent, intermittent and transient.

- ***Types of Failure***

Failures that occur during computation on system resources can be classified as timing failure, omission failure, response failure and crash failure. Failures may be permanent, Intermittent and transient.

Failure Type	Cause of occurrence
Permanent failure	Accidental power breakdowns, natural disasters
Intermittent failure	These failures happen while the system performs its operations, execution errors
Transient failure	These failures are caused by some inherent fault in the system and they can be easily resolved.

Need of Fault Tolerance

Increase in interest for computing resources has prompted an expansion in the categories of services, and lead to create large-scale data centres. Earlier, high performance was considered as the principle standards in the design of data centres.

Today, with the improvement of cloud computing-based data focuses and increase in interest for the use of cloud services, failure is basic in the present data centres, which can be harmful to business by large scales of data losses.The motivation behind fault tolerance is to accomplish robustness and constancy in each system. Based on the policies and procedures of fault tolerance, the methods can be separated into two categories: proactive and reactive.

Proactive fault tolerance policy focuses on recovery of fault, error, and failure with the assistance of prediction, and proactive fault tolerance distinguishes the suspicious items and replaces it with the right data, which implies that finding the issue before it really happens.

Reactive fault-tolerance policy attempts to handle failures when they happen. It very well may be partitioned into error processing and fault-treatment techniques. The reason for error processing is to eliminate errors from the computation. Error treatment additionally expects the reactivation of errors. Fault-tolerance by error processing, which is made out of two phases, is performed as following: The viable error processing stages, that expects to conceal viable fault before event of failure, and hidden error processing that means to guarantee that the faults won't be activated again.

III. Existing Approaches in Fault Tolerance

Fault tolerance is indeed a key component for any cloud service provider and required a potential amount of attention on fault tolerant techniques. Fault tolerant techniques include the strategy to handle the fault in the system. There are number of fault tolerance techniques already existing with pros and cons. broadly techniques can be classified in three categories: Proactive, Reactive and Adaptive fault tolerance techniques

Proactive fault tolerance: The essence of Proactive fault tolerance polices is to predict the fault and stay away from recovery from fault, errors and failures proactively supplant the presumed segment (i.e) it identifies the issue before it really comes. It prevents compute hub failures from affecting running equal applications by pre-emptively migrating pieces of an application (task, process, or virtual machine) away from hubs that are going to fail. A portion of the techniques based on these policies are Pre-emptive migration, Software Rejuvenation and utilizing self-healing.

Proactive Fault Tolerance Techniques	
Software Rejuvenations	• Technique that designs the system for periodic reboots • Restarts the system with clean state and helps to fresh start.
Pre-emptive Migration	• Count on a feedback-loop control mechanism. • Application is constantly monitored and analyzed.
Self-Healing	• Task can be divided into parts • Multiplication is done for better performance. • It automatically handles failure of application instances.

***Reactive Fault Tolerance*:** Reactive fault tolerance policies lessen the impact of failures on application execution when the failure effectively occurs. It is also called as on-demand fault tolerance. There are various techniques based on these policies like Checkpoint/Restart, Replay-retry, task resubmission, recue workflow, user defined exception handling, retry, S-Guard, job migration and so on.

Reactive Fault Tolerance Techniques	
Check pointing	• an efficient task level fault tolerance technique for long running and big applications • method after doing every change in system a check pointing is done • it is allowed to be restarted from the recently checked pointed state.
Job Migration	• Count on a feedback-loop control mechanism. • Application is constantly monitored and analyzed.
Replication	• Task can be divided into parts • Multiplication is done for better performance. • It automatically handles failure of application instances.
Safety-bag checks	• Blocking of commands is done which are not meeting the safety properties.
S-Guard	• Less turbulent to normal stream processing • Based on rollback recovery
Resubmission	• Job may fail whenever a failed task is detected • Runtime the task is resubmitted either to the same or to a different resource for execution.

Adaptive fault tolerance: The fault-tolerance of an application should be changed relying upon scope of control inputs and the current position in its state space. Adaptive fault tolerance consequently invokes the procedures to control the situation and they assure adequate reliability of critical modules under any resources and temporal constraints by apportioning as much redundancy resources and modules.

IV. Fault Tolerance Models

After studying the available literature in the field of fault tolerance in cloud computing, we could find some popular fault tolerant models. Different models are available based on above mentioned techniques, popular models are as follows:

AFTRC (Adaptive fault tolerance Real-time computing) - Real time application needs high processing capabilities using cloud computing. AFTRC is based on proactive fault tolerance and is the notion of high reliability of the system

LLFT (Low latency fault tolerance) – LLFT acts as middle ware to tolerate the fault for the system. This middleware provide fault tolerance by replication .The application uses semi-active replication or semi-passive replication process to protect against various types of faults.

FTM(Fault tolerance model): FTM is proposed to beat the limitation of existing methodologies of on-demand service. This model ensures reliability and resilience by using innovative philosophy through which the user can specify and apply the desired level of fault tolerance without requiring any information about its implementation. FTM can be seen as an assembly of several web services components, each with a specific functionality.

FTWS (Fault tolerance workflow scheduling): FTWS model contains a fault tolerant work flow scheduling calculation to give fault tolerance by using replication and re submission of tasks based on the priority of the tasks in a heuristic matrix. This model is based on workflow - a set of tasks processed in some request based on data and control dependency. The scheduling of workflow also considers the task failure in the cloud environment. FTWS replicates and schedule the tasks to meet their deadlines.

Candy - Candy is a component based availability modelling frame work which constructs a comprehensive availability model semi automatically from system specification described by systems modelling language. This model is based on the fact that high availability assurance of cloud service is one of the main characteristic of cloud service and also one of the main critical and challenging issues for cloud service provider .

FT-Cloud is a component ranking based frame work and its architecture is used for building cloud application. FT-Cloud employs the component invocation structure and frequency to identify the component. There is an algorithm to automatically determine fault tolerance stately.

Magi-Cube: is highly reliable and low redundancy storage architecture for cloud computing. They build the system on top of HDFS and use it as a storage system for file read /write and metadata management. They also built a file scripting and repair component to work in the back ground independently which provides high reliability and performance at low cost.

V. Comparative analysis of Fault Tolerant Models

In this section of work, we are going to compare the different fault tolerant models mentioned above. Models can be compared based on various aspects like types of fault they handle, approach to handle the fault and also on the basis of different criterion. The primary objective of the comparison is to choose the right approach towards the fault tolerance and also to find out short coming of the existing approaches which can be further extended as research objectives.

Below tables (1),(2) and (3) will show the comparative study/analysis of available fault tolerance models.

Table (1) shows the comparison of popular fault tolerance models based on the approach.

Models / Nature	AFTRC	LLFT	FTM	FTWS	Candy	FT-Cloud	Magi Cube
Proactive	✓	✗	✗	✗	✗	✓	✗
Reactive	✗	✓	✓	✓	✓	✗	✓
Adaptive	✓	✗	✗	✗	✓	✓	✓

Table (2) showcasing the comparison of popular models based on types of fault they handle and significant strategies against the fault tolerance.

Fault Tolerant Models	Protection against fault	FT policies and techniques				
		Proactive Techniques		Reactive Techniques		
		Faulty component self repair	Migration - Preventive	Fault effect Inspection	Replication/ Duplicate	Migration-Reactive
AFTRC	Reliability	no	no	yes	yes	yes
LLFT	Crash-cost, trimming fault	no	no	no	yes	no
FTM	Reliability, availability, on demand service	no	no	yes	yes	yes
FTWS	Dead line of work fL	no	no	yes	yes	no
Candy	Availability	no	no	no	yes	no
FT-Cloud	Reliability, crash and value fault	no	no	no	yes	no
Magi Cube	Performance, reliability, L storage cost	no	no	no	yes	yes

Table (3) Displaying comparison of fault tolerant models based on the different matrices

Models / Criteria	AFTRC	LLFT	FTM	FTWS	Candy	FT-Cloud	Magi Cube
Performance	***H***	***H***	***Avg***	***Avg***	***Avg***	***H***	***H***
Response time	***Avg***	***Avg***	***Avg***	***Avg***	***Avg***	***Avg***	***Avg***
Scalability	***H***	***H***	***L***	***L***	***H***	***H***	***H***
Throughput	***H***	***Avg***	***Avg***	***L***	***H***	***Avg***	***H***
Reliability	***H***	***H***	***Avg***	***Avg***	***H***	***H***	***H***
Availability	***H***	***H***	***H***	***Avg***	***H***	***Avg***	***Avg***
Usability	***H***	***Avg***	***Avg***	***Avg***	***Avg***	***H***	***H***
Overhead	***Avg***	***L***	***L***	***H***	***L***	***H***	***Avg***
Impact on cost	***Avg***	***L***	***L***	***H***	***L***	***H***	***H***

H-High, L-Low, Avg-Average

VI. METRICS OF FAULT TOLERANCE

The existing fault-tolerance method in cloud computing consider various parameter. The parameters resemble there kind of fault-tolerance (proactive, reactive and adaptive), *performance, response-time, scalability, throughput, reliability, availability, usability, security and associated over-head.* Table (3) summarized the Comparison among various models based on the metrics component.

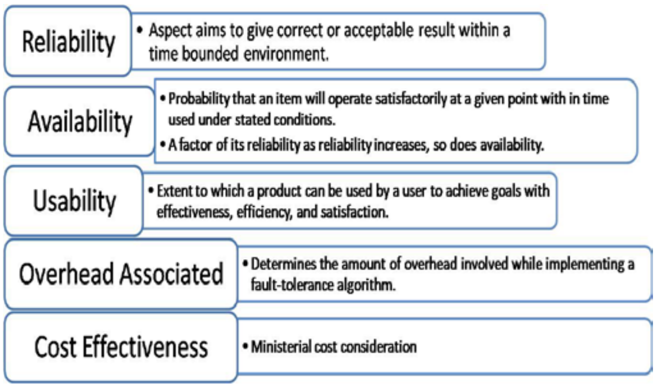
Various parameters including fault-tolerance type (proactive, reactive, and adaptive), performance, time response, scalability, through put, reliability, availability, usability, security, cost effectiveness and overhead associated are considered when managing the existing fault-tolerant methods in the Cloud Computation.

Additional metrics parameters are,

Response Time - is the measure of time taken to respond by a specific algorithm. This parameter should be limited.

Scalability–This is the capacity of an algorithm to perform fault-tolerance for a system with any limited number of nodes. This metric should be improved.

Throughput–This is used to ascertain the no. of tasks whose execution has been finished. It should be high to improve the performance of the system.



VII. CONCLUSIONS

Fault-tolerance methods work when a fault enters the limit of a system. Accordingly, in principle, fault-tolerance methods are used to predict the fault and perform a fitting action, before the faults really happen. This article discusses the classification of faults and the need to cover the fault-tolerance with various implementation techniques. Distinctive fault-tolerance models are presented, and thought about in terms of fault tolerance in the cloud.

In the present scenario, there are tolerance error models that acquaint diverse fault-tolerance mechanisms with improve the system. Be that as it may, there are still challenges that should be considered for any framework or model. There are weaknesses that can't finish all aspects of the faults. Thus, it is possible to overcome the weaknesses of every previous model.

REFERENCES

[1] Peter Mell, Timothy Grance, "The NIST Definition of Cloud Computing", *Recommendations of the National Institute of Standards and Technology*, Special Publication 800-145, 2010

[2] Yashpalsinh Jadeja, Kirit Modi, "Cloud Computing - Concepts, Architecture and Challenges", *International Conference on Computing, Electronics and Electrical Technologies [ICCEET]* 2012

[3] Jayadivya S K, Jaya Nirmala S. "Fault tolerant workflow scheduling based on replication and resubmission of tasks in Cloud Computing", *International Journal on Computer Science and Engineering (IJCSE)* 2012

[4] Anju Bala, Inderveer Chana, "Fault Tolerance- Challenges, Techniques and Implementation in Cloud Computing" *IJCSI International Journal of Computer Science Issues, Vol. 9, Issue 1, No 1*, January 2012

[5] Deepak Poola, Kotagiri Ramamohanarao, and Rajkumar Buyya,"Fault-Tolerant Workflow Scheduling Using Spot Instances on Clouds" *ICCS 2014. 14th International Conference on Computational Science Volume 29, 2014, Pages 523–533*

[6] Ajitabh Mahalkari, Ritu Tondon "A Replica Distribution Based Fault Tolerance Management For Cloud Computing" *(IJCSIT) International Journal of Computer Science and Information Technologies*, Vol. 5 (5), 2014, 6880-6887

[7] Deepak Poola, Kotagiri Ramamohanarao, and Rajkumar Buyya,"Fault-Tolerant Workflow Scheduling Using Spot Instances on Clouds" *ICCS 2014. 14th International Conference on Computational Science Volume 29, Pages 523–533*, 2014

[8] R. Raju,J. Amudhavel, M. Pavithara et al. "A Heuristic Fault Tolerant MapReduce Framework for Minimizing Makespan in Hybrid Cloud Environment"

[9] Ji Wang, Weidong Bao, et al. "FESTAL: Fault-Tolerant Elastic Scheduling Algorithm for Real-Time Tasks in Virtualized Clouds" *IEEE TRANSACTIONS ON COMPUTERS*, 2014 DOI: 10.1109/TC.2014.2366751

[10] Ajitabh Mahalkari, Ritu Tondon "Performance Evaluation Of Quality Metrics Based Fault Prediction model For Replica Management In Cloud Computing" *(IJCSIT) International Journal Of Computer Science And Information Technologies*, Vol. 6 (3) , 2015, 2512-2518.

[11] J.Soniya et al. "Dynamic Fault Tolerant Scheduling Mechanism for Real Time Tasks in Cloud Computing" *International Conference on Electrical, Electronics, and Optimization Techniques (ICEEOT)* – 2016

[12] Xiaomin Zhu et al. "Fault-Tolerant Scheduling for Real-Time Scientific Workflows with Elastic Resource Provisioning in Virtualized Clouds" *IEEE TRANSACTIONS ON PARALLEL AND DISTRIBUTED SYSTEMS*, 2015 DOI 10.1109/TPDS.2016.2543731

[13] Muhammad Abdulhamid et al. "Fault tolerance aware scheduling technique for cloud computing environment using dynamic clustering algorithm" *Neural Computing and Applications* · July 2016 DOI: 10.1007/s00521-016-2448-8

[14] Guoqi Xie, et al. "Quantitative Fault-tolerance for ReliableWorkflows on Heterogeneous IaaS Clouds", *IEEE TRANSACTIONS ON CLOUD COMPUTING*, NOV 2016 DOI 10.1109/TCC.2017.2780098

[15] SatjotKaur, Gurbhej Singh "Review on Fault Tolerance Techniques in Cloud Computing" *International Journal of Engineering and Management Research ISSN (ONLINE): 2250-0758* Volume-7, Issue-3, May-June 2017

[16] Yanchun Wang, et al. "Formulating Criticality-Based Cost-Effective Fault Tolerance Strategies for Multi-Tenant Service-Based Systems" *IEEE TRANSACTIONS ON SOFTWARE ENGINEERING* DOI 10.1109/TSE.2017.2681667

[17] Xiwei Qiu et al. "Correlation Modeling and Resource Optimization for Cloud Service With Fault Recovery" *IEEE TRANSACTIONS ON CLOUD COMPUTING* DOI 10.1109/TCC.2017.2691323

[18] Priti Kumari, Parmeet Kaur "A survey of fault tolerance in cloud computing" *Journal of King Saud University – Computer and Information Sciences* (2018)

[19] Bashir Mohammed et al. "Failure Analysis Modelling in an Infrastructure as a Service (Iaas) Environment" *Electronic Notes in Theoretical Computer Science* 340 (2018) 41–54 Elsevier

[20] Moin Hasan, Major Singh Goraya "Fault tolerance in cloud computing environment: A systematic survey*" Elsevier Computers in Industry* 99 (2018) 156–172

[21] Mao-Lun Chiang et al. "An Agreement under Early Stopping and Fault Diagnosis Protocol in a Cloud Computing Environment" *IEEE Translations and content mining are permitted for academic research only* DOI:10.1109/ACCESS.2018.2859351

[22] Haoran Han et al. "Fault-Tolerant Scheduling for Hybrid Real-TimeTasks Based on CPB Model in Cloud" *IEEE ACCESS 2017* DOI 10.1109/ACCESS.2018.2810214

[23] PeiYun Zhang et al. "An Online Fault Detection Model and Strategies Based on SVM-Grid in Clouds" *IEEE/CAA JOURNAL OF AUTOMATICA SINICA*, VOL. 5, NO. 2, MARCH 2018

[24] Bhupesh Kumar Dewangan et al. "Self-characteristics based Energy-Efficient Resource Scheduling for Cloud" *Elsevier International Conference on Pervasive Computing Advances and Applications* 152 (2019) 204–211

[25] CHUN GAN et al. "An Overview of Fault-Diagnosis and Fault-Tolerance Techniques for Switched Reluctance Machine Systems" IEEE *Digital Object Identifier 10.1109/ACCESS.2019.2956552*

[26] Lei Wang et al. "Architecture-based Reliability-sensitive Criticality Measure for Fault-Tolerance Cloud Applications", IEEE *Transactions on Parallel and Distributed Systems* 2019, DOI 10.1109/TPDS.2019.2917900