RESEARCH REVIEW JOURNALS
Volume-03
Issue-09
September-2018

ISSN: 2455-3085 (Online)
RESEARCH REVIEW International Journal of Multidisciplinary
www.rrjournals.com [UGC Listed Journal]

Implementation Strategy of a Mechanisms to Create a Reliable Cloud Computing System – A Review

[1]Mohd Waseequsheraz Mohammedrafeequ & [2]Dr. Shesharao M. Wanjerkhede

[1]Research Scholar, Deptt of Computer Application , Sri Satya Sai University of Tech. and Medical Science, Sehore, MP (India)
[2]Ex-Professor, Department of Computer Science and Engineering, Guru Nanak Dev Engineering College, Bidar, Karnataka (India)

ARTICLE DETAILS

Article History
Published Online: 07 September 2018

Keywords
Fault tolerance as a Service, System level Fault tolerance, Cloud Computing, Fault tolerance factors, Cloud Reliability, VM reliability, Fault Independence, Fault tolerance properties

ABSTRACT

Cloud Computing, with its great potentials in low cost and on-demand services, is an auspicious computing platform for both commercial and non-commercial computation clients. Cloud Computing aims to provide reliable services within data centres that contain servers, storage and networks. The services are delivered to the users transparently without their need to know the details of the underlying software and hardware. One of the challenge for cloud computing is to ensure that the applications run without a hiatus in the services they provide to the users. With the increasing demand and benefits of cloud computing infrastructure, real time computations are performed on cloud infrastructure. In most of the real-time cloud applications, processing on computing nodes are done remotely. So there are more chances of errors. On the other side, most of the real time systems are also safety critical and should be highly reliable. In this context, applications' running on cloud requires fault tolerance abilities so that they can come over the impact of system failures and perform their functions correctly when failure occurs. This paper provides evaluation of study of various fault tolerance mechanisms, several fault tolerance models and their comparison among them.

1. Introduction

Cloud computing can be defined as a way of using computational resources such as storages, operating systems etc. which are located remotely and are provided as a service over internet [1]. The elementary advantages of cloud computing include low costs, high availability, scalability and elasticity. On the other hand, the reliability of Cloud computing still remains a major concern among users. Due to economic pressures, these computing infrastructures often use commodity components exposing the hardware to scale and conditions for which it was not originally designed [2]. As a result, significantly large number of failures manifest in the system and seemingly impose high implications on the hosted applications, effecting their performance and availability. In this context, applications require fault tolerance abilities so that they can overcome the impact of system failures and perform their functions correctly when failures happen. Fault tolerance bear-on with all the inevitably techniques to enable robustness and dependability. The main benefits of implementing fault tolerance in cloud computing include failure recovery, improving reliability, and enhanced availability[3]. Robustness leads to the property of providing a correct service in an adverse situation arising due to an uncertain system environment [4]. Dependability is related to some QOS aspects provided by the system, including attributes like reliability and availability [5].

The motivation of the survey of existing fault tolerance techniques and models in cloud computing is to encourage researcher to contribute in developing more efficient algorithm. This paper is planned to deliberates about various aspect of faults and the need of fault tolerance in cloud computing.

2. Fault taxonomy and its necessity

Fault tolerance refers to correct and continuous operation even in the presence of faulty components. In most of the real time cloud applications, processing on computing nodes is done remotely. So, there are more chances of errors. So, there is an increased requirement for fault tolerance to achieve reliability for the real time computing on cloud infrastructure. Fault tolerance is carried out by error processing which have two constituent phases. The phases are "effective error processing" which aimed at bringing the effective error back to a dormant state, i.e. before the occurrence of error and "latent error processing" aimed at ensuring that the error does not become effective again.

Based on fault tolerance different fault tolerance techniques and strategies, we classify faults as follows:

Proactive fault tolerance: The Proactive fault tolerance policy is to predict the fault and avoid recovery from fault, errors and failure and proactively replace the suspected component means detect the problem before it actually come. It is a concept that prevents compute node failures from impacting running parallel applications by pre-emptively migrating

Reactive fault tolerance: Reactive fault tolerance policies reduce the effort of failures, when the failure occurs. This technique makes system more robust. In other words, we can say that it an On-demand fault tolerance.

Adaptive fault tolerance: The fault-tolerance needs of an application change depending on its current position in its state space and the range of control inputs that can be applied. So, in adaptive fault tolerance all procedures are done

automatically according to the situation. Adaptive Fault Tolerance (AFT) can assure adequate reliability of critical modules, under temporal and resources constraints, by allocating just as much redundancy to less critical modules as can be afforded, thus elegantly reducing their resource requirement.

3. Existing fault tolerance technique in cloud computing

Several fault tolerance techniques that are existing currently in clouds [3], [4], [6], [7], [8] are as follows:

A. Self-Healing

In this method divide and conquer technique is used, in which a huge task is distributed into several parts. This division is done for better performance. In this, various instances of an application are running on various virtual machines and failure of all this individual instance are handled automatically.

B. Job Migration

Sometimes it happens that due to some reason a particular machine fails and cannot execute job. On such a failure, a task is migrated to working machine using HA-Proxy. Also, there are algorithms that automatically determines the fault and migrates batch applications within a cloud of multiple datacenters.

C. Check Pointing

It is a proficient task level fault tolerance technique for large applications. In this method, check pointing is done in system. When a task fails, instead of initiating from beginning it is restarted from the recently checked pointed state. Check pointing is carried out periodically i.e., checkpoints are kept and process is executed from the recent check point, once system governs the fault.

D. Replication

Replication means copying. Several replicas of tasks are created and they are run on different resources, for effective execution and for getting the desired result. Hadoop, HA-Proxy, Amazon EC2 like tools are there on which replication can be implemented. Also, there are mainly three different types of replication schemes such as Active Replication, Semi-Active Replication and Passive Replication.

E. Task Resubmission

Many times, it happens that due to high network traffic or due to heavy work load, a task may fail, whenever such failed task is detected, at runtime the task is resubmitted either to the same or different working resource for execution. For these, certain algorithms are designed, which assigns task to resources on the basis of certain properties.

F. Masking

After occupation of error recovery, the new state needs to be identified as a transformed state. If this process applied systematically even in the absence of effective error provide the user error masking [6].

G. Resource Co-allocation

In refers to the process of allocating resources for further execution of task. Many algorithms are designed, that deals which resource allocation depending on the properties of VM such as workload, type of task, capacity of VM, energy awareness etc.

H. Timing Check

This is deployed with the help of watch dog. It is a simplest technique with time as a critical function [4]. It keeps the track of task execution, whether the task has been completed in required amount of time or not. Depending on which further action for fault tolerance is taken.

I. Rescue Workflow

A workflow consists of a sequence of connected steps where each step follows without delay or gap and ends just before the subsequent step may begin. In this technique, it allows the workflow to carry on until it becomes unimaginable to move forward without catering the failed task.

J. User Specific(defined) Exception Handling

In this case, whenever fault is detected, action is predefined by the user, i.e. user defines the particular treatment for a task on its failure.

Several models that are implemented based on above techniques are as follows:

"AFTRC" – It is an Adaptive Fault Tolerance model in Real time Cloud Computing. In this proposed model system tolerates fault proactively and makes decision on the basis of the reliability of the processing nodes [7].

"LLFT" - is a propose model which contains a low latency fault tolerance (LLFT) middleware for providing fault tolerance for distributed applications deployed with in the cloud computing environment. This middleware replicates application by the using of semi-active replication or semi-passive replication process to protect the application against various types of faults [8].

"FTM"- is a model to overcome the limitation of existing methodologies and achieve the reliability and flexibility, they propose an inventive perspective on creating and managing fault tolerance .By this particular methodology user can specify and apply the desire level of fault tolerance. FTM architecture this can primarily be viewed as an assemblage of several web services components, each with a specific functionality [9].

"FTWS"- is a Fault Tolerant Work flow Scheduling algorithm for providing fault tolerance by using replication and resubmission of tasked based on based on the priority of the task. This model is based on the fact that work flow is a set of tasks processed in some order based on data and control

dependency. Scheduling the workflow along with the task failure consideration in a cloud environment is very challenging. FTWS schedule and replicates the tasks to meet the deadline [10].

"Candy"- is a component based availability model. It is based on the 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 [11].

"FT-Cloud"- is a component ranking based frame work and its architecture for building cloud application. FT- Cloud occupies the component invocation structure and frequency for identify the component. Also, there is an algorithm to automatically govern fault tolerance stately [12].

4. Comparison of various models based on protection against type of fault and procedure

TABLE I

Model Name	Regular Protection against type of fault	Application procedure for tolerate the fault
AFRTC	Reliability	1. Delete node depending on theirreliability 2. Back word recovery with the help of check pointing
LLFT	Crash-cost, trimming fault	Replication
FTM	Reliability, availability, on demand service	Replication users application and in the case of replica failure use algorithm like gossip based protocol.
FTWS	Dead line of work flow	Replication and resubmission of jobs
Candy	Availability	1.It assembles the model components generated from IBD and STM according to allocationnotation. 2.Then activity SNR is synchronized to system SRN by identifying the relationship between action in activity SNR and state transitioninsystem SRN.
FT-Cloud	Reliability, crash and valuefault	1. Significant component is determined based on theranking. Optimal ft technique isdetermined.

5. Conclusion

Fault tolerance is a critical issue and comes into picture the moment fault enters system. This paper discusses about fault taxonomy, its need along with various techniques for implementing fault tolerance. Several fault tolerance models are discussed and compared on the basis of protection against type of fault and procedure. Presently, there are several mechanism for fault tolerance but still there are number of challenges which needs to be considered. Also, there are some drawbacks no one of them can fulfil all the aspects of faults. So, there is likelihood to overcome these drawbacks and try to make a solid model that may cover maximum fault tolerance aspect.

References

1. Sun Microsystems, Inc. "Introduction to Cloud Computing Architecture" White Paper 1st Edition, June 2009
2. K. V. Vishwanath and N. Nagappan, "Characterizing cloud computing hardware reliability," in Proceedings of the 1st ACM symposium on Cloud computing, ser. SoCC '10. New York, NY, USA: ACM, 2010, pp. 193–204.
3. AnjuBala, 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 ISSN (Online): 1694- 0814 www.IJCSI.org
4. Benjamin Lussier, Alexandre Lampe, Raja Chatila, JérémieGuiochet, Félix Ingrand, Marc-Olivier Killijian, David Powell, "Fault Tolerance in Autonomous Systems: How and How Much?" LAAS-CNRS 7 Avenue du Colonel Roche, F-31077 Toulouse Cedex 04, France
5. P.Latchoumy and P.SheikAbdulKhader,"Survey on fault tolerance in grid computing" IJCSI International Journal of Computer Science Issues, Vol 2, No 4, November 2011.
6. Jean-clandeLaprie "Dependable computing and fault tolerance: concepts and terminology" LAAS-CNRS 7 Avenue du Colonel Roche, 31400 Toulouse, France
7. Sheheryar Malik and FabriceHuet "Adaptive Fault Tolerance in Real Time Cloud Computing" 2011 IEEE World Congress on Service
8. Wenbing Zhao, P.M. Melliar and L.E. Mose" Fault Tolerance Middleware for Cloud Computing" 2010 IEEE 3rd International Conference on Cloud Computing.
9. Ravi Jhawar, Vincenzo Piuriand MarcoSantambrogio"A Comprehensive Conceptual System level Approach to Fault Tolerance in Cloud Computing" IEEE
10. Jayadivya S K, Jaya Nirmala S, Mary SairaBhanus"Fault Tolerance Workflow Scheduling Based on Replication and Resubmission of Tasks in Cloud Computing" International Journal on Computer Science and Engineering (IJCSE)
11. Fumio Machida, Ermeson Andrade, Dong Seong Kim and Kishor S. Trived "Candy: Component-based Availability Modeling Framework for Cloud Service Management Using Sys-ML" 2011 30th IEEE International Symposium on Reliable Distributed Systems.
12. ZibinZheng, Tom Chao Zhou, Michel R. Lyu, and Irwin king "FT-Cloud: A Component Ranking Framework for Fault-Tolerant Cloud Applications "2010 IEEE 21st International Symposium on Software Reliability Engineering.

