FAULT TOLERANCE IN CLOUD COMPUTING- A SURVEY

AAKANKSHA, SANDEEP SHARMA

Department of Computer Science & Engineering

Guru Nanak Dev University

Amritsar, Punjab, India

Abstract- Cloud computing has initiated a revolution in the commercial world by providing computing service to its clients via the internet in a pay-as-you-go manner. Quality of service (QoS) is quite needful to be achieved in cloud computing for customer contentment. Quality of service includes reliability, performance and handiness that can be offered in an application by utilizing the fault tolerance mechanism. Fault tolerance is the ability of an application to go on responding even if there are faults and errors occurring in the application. Since breakdown and errors that usually occurs in the cloud are quite hard to discover and evade then fault tolerance mechanism is must requisite to be implied in the cloud. Fault tolerance improves the performance by rendering the service to guests on demand. This paper address various types of faults, failure and fault tolerance techniques. Fault tolerance assorted methods are given and compared. In this paper, several architectures given by research expertise are discussed with various factors achieved.

Keywords- virtualization,Hypervisor, fault tolerance ,Vms.

I. INTRODUCTION

Cloud computing can be described as the abstraction and encapsulation of web supported computers, origin and aids. It fundamentally allots the computation resources via web services. Cloud computing render various cloud computing services to its all users like data host server, data depositary, database, networks, software. Since the computer network makes it possible to supply all cloud services to its client in an easier manner internet can be entitled as the cloud. It brings up evolution to the conventional grid-based computing. The easiness of using cloud computing are on-demand work, scalability and reliability[1]. Cloud computing expenses can be bear by users as it is supplied in a Pay-As-You-Use manner. Cloud is essentially provided by large dispersed data centres. As the people are becoming more dependable on their gadgets for every assistance, data storage has become everyone priority. The main reason for espousing cloud computing by enormous organizations is cost-effectiveness. Cloud computing has made it convenient for large as well as small scale industries for accumulating data at a diminished cost. Cloud computing eliminates the requirement for the task like hardware set up so that user and service provider can expend more time on succeeding crucial enterprise goals. Users who subscribe to cloud are allocated computing services via the cloud and they are able to gain accession to accelerated and flexible organization level computing services. Client interaction with the cloud through their web applications programme eradicate the requisite for installing numerous software applications on hardware which eventually figure out the retention affiliated issues. It not only simplifies in house computing operations but also change the way outcomes are delivered to its users. The cloud relinquishes the storage and applications as a service but not a product. The three major services that clients are rendered to users are briefly mentioned below.

SaaS or software as a service is a software system ideal in which applications are ready-made reachable to clients through the web (computer network) by service supplier and vendors. Guest can retrieve cloud applications from any place via the web applications programme or by just signing in. Anyone who needs an admittance to a specific part of software program can sign in as a client, either it is one or two consumer or thousands of worker in a firm. Software as a service major reward is the lower cost of use. Examples of Saas Google documents, Google maps.

Platform as a service is termed as PaaS, it furnishes platform to permit developers to develop applications, process and organize cloud services without being hosting it on their hardware. It generally diminishes the complexness of installing and handling the data retention by locating the framework in the data centre and accessed by the guests via computer network services are perpetually updated & brand-new features added. It includes software assistance and administration assistance, increased agility, distributing, collaborating, accommodating and managing applications.

Infrastructure As A Service also termed as Iaas: It is responsible for granting admittance of cloud environment services and numerous means in a virtualized manner via the IP connected networks. The foundation is supplied on users request for data depositary and well-kept by the vendors. Data is mostly kept in various diffused data centres so that in case one component go down the other will take-over inevitably. The client disk space is located on the diffused file system whereas another major constituent is a rule for assets allotment. Cloud computing is an assured diffused surroundings and it intemperately relies upon the coercive algorithm.

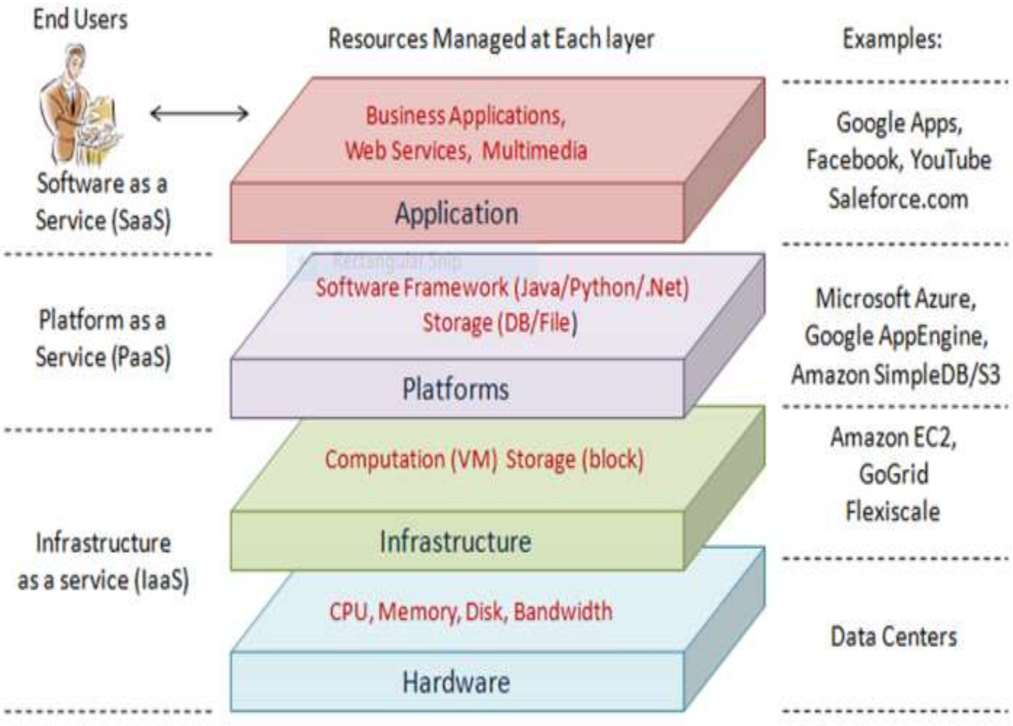
End Users
Resources Managed at Each layer
Examples:
Software as a Service (SaaS)
Business Applications, Web Services, Multimedia
Application
Google Apps, Facebook, YouTube Saleforce.com
Platform as a Service (PaaS)
Software Framework (Java/Python/.Net) Storage (DB/File)
Platforms
Microsoft Azure, Google AppEngine, Amazon SimpleDB/S3
Infrastructure as a service (IaaS)
Computation (VM) Storage (block)
Infrastructure
Amazon EC2, GoGrid Flexiscale
CPU, Memory, Disk, Bandwidth
Hardware
Data Centers

Fig.services of clouds[2]

Quality of Service (QoS) has a vital role in the cloud environment. In accordance to Quality of Service, reliability and performance are obligatory characteristics to be stipulated in the cloud computing paradigm. As the customers access the services remotely and data is situated on data centres through the internet instead locally due to the risk of errors and breakdown amplified. The assistance provided to customers should attain reliability and faster execution. These factors can be attained through fault tolerance.[3] It is the competency of a system to carry on its procedure even in the beingness of breakdown. Fault tolerance is a major reason for anxiety for both customers and aid clients. A fault tolerant system should be capable of managing faults in respective hardware or software or power breakdown or any kind of bizarre behaviour and still carry on its processing. The basic difficulty is as the complications of system is increment, its reliability fall off if necessary measures are not taken. But even in the case when the system is configured and implemented perfectly faults can be still be expected. Fault tolerance has been noticed for many years, there are several methods of fault tolerance which can be implied according to the user requirements and their applications. The primary thought of using fault b tolerance is to center on masking of faults instead of completely removing or fixing them[4]. Fault tolerant should be used for defaulted constituent and in spite of implementing fault tolerant mechanism in all components as it will diminish the cost of installation. A system capable of fault tolerance should attain parameters including reliability, availability and performance. Reliability and performance play a vital role in fault tolerance as low reliability and low performance in cloud computing can cause breakdown and become a cause of discontentment to customers. By providing high availability and reliability leads to robust fault tolerance during development time. High reliability is quite difficult to achieve as there is a larger number of physical servers and virtual machines in the cloud whose risk of failure is high. The focus on availability and reliability concern can improve system safety[5]. The primary effort of this paper is to give a general survey of fault tolerance techniques. Section II gives an summary of fault tolerance and discusses kind of faults and failure and section III gives literature assessment of earlier work done on fault tolerance whereas section IV identifies general parameters involved in fault tolerance and in section IV have compared parameters achieved in techniques by the research expertise.

II. Fault Tolerance : an overview

Fault tolerance is the competence of the system carry out its actions even in the existence of breakdown and to prohibit service breakdown by decreasing their outcome on the cloud. A system is said to fail when it do not operate the way it is intended to.[3]

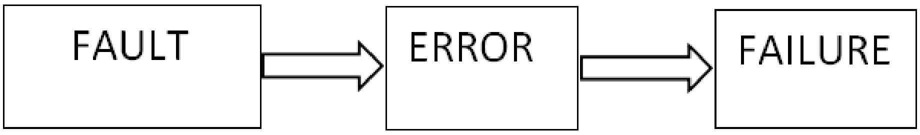

fig. Classification of fault tolerance

A. FAULT

Fault can drive the system incompetent of doing its needed task due to the beingness of bug or error in one or more than one portion of the system. The main reason behind the presence of an error is a fault. Fault can be a physical shortcoming or defect in the software component or hardware component example a computer code bug. Faults can originate errors and errors can further lead to cause breakdown. Faults can be categorized on different elements such as network, processor, process, media and physical faults. Faults can be originated due to the beingness of software bug or external behaviour or hardware faults.

- ✓ *Physical faults*: The Fault which takes place in hardware like fault in CPUs, Fault in memory, Fault in disk,virtual machine etc.
- ✓ *Intentional faults:* faults that are caused to system on purpose by inserting malicious code.
- ✓ *Internal faults:* faults which happen due to presence of software bug or internal design.
- ✓ External faults: faults which can cause serious outcomes due to some external behaviour.
- ✓ *Design faults* : manufacturing defects in caused by fuzzy human behaviour can cause various failures.
- ✓ *Operational faults:* are also caused in phase of creation which can cause operational fault like application failure or Vms failure.
- ✓ *Software faults* occur due to presence of bug or design fault or due to fuzzy human factors which are harder to prevent.
- ✓ *Permanent faults* are the faults that once occurred then continues to exist until replaced or repair. They are caused due to Power failure and they can cause major disruptions.Permanent ,Intermittent and transient comes under the category of timing faults.

✓ *Intermittent fault* are the one that occurs occasionally, they sometime vanish and then reappears so it become quite difficult to detect them.

✓ *Transient faults* are the faults caused by some inherent fault in system, if operation is repeated several times then fault goes away and corrected by rollback to previous state or by restarting the system. Faults are also classified on their behaviour into Byzantine faults and Crash faults.

✓ *Byzantine faults* are the faults which cause system to behave arbitrarily at arbitrary times such that a system do no stop execution even after failure but gives wrong output.

✓ *Crash faults* are the faults which once occurred make the system halt completely[6].

B. ERROR

The existence of fault can become the reason of the system component behaving erroneously. Error can be defined as foreseen difference between its observed value and literal value . Faults can cause errors and errors can further lead to cause failures. Errors are related with inaccurate values in the system state . There are various types of errors that can be caused like network errors ,software errors and miscellaneous errors.

C. FAILURES

The wrongful conduct of system when system reaches an fallacious statement observed by human or by system itself. Breakdown takes place due to an error which leads to arrival of an fallacious state. There are many classification of breakdownthat have been formulated. Failure various types includes crash failure,omission failure ,timing failure ,omission failure ,response failure and many other following discussed .

1. *Crash failure* are the breakdownin which system halts but work accurately until it stopped working. Once the system crashes, it do not perform any further results. It comes under the category of permanent omission failure. For example an operating system comes to halt situation and the only solution left is to reboot it.

2. *Omission failure* when the server lacks in getting responses or fails to respond to a request for example not getting response or failed messages due to overloaded server and When the server is not getting enough responses then the current server is not affected directly as the server is not known about the messages are not received yet. Some other types of omission faults in software can make the system hang caused by infinite loops .

3. *Timing breakdown*is one of classification of which occurs when response do not reach the server at the specified time interval that it intended to reach which include delay faults and early faults.

4. *Response breakdown*are the one in which the response received is incorrect completely for example a search engine that do return web pages not associated to searched data or incorrect web pages are returned [7].

5. *Arbitrary breakdown*: The most serious breakdown are Byzantine breakdown or arbitrary breakdown which deviates from intended processing according to an algorithm due to bug or malicious attack.In particular, it may happen that a server is producing output it should never have but which cannot be detected as being incorrect. It is quite clear that byzantine or arbitrary breakdownare the worst kind of faults. Breakdown that cause halting of server can be classified into several types.

6. *Fail-stop breakdown*are the one of crash failure which makes the server halt and can be easily detected and results in permanent failures.

7. *Fail silent breakdown*are the breakdownin which the process cannot variousiate between crash breakdownand omission failure.

8. *Fail-safe breakdown*deal with arbitrary breakdownand do not cause any kind of harm. It is quite clear that byzantine or arbitrary breakdownare the worst kind of faults.

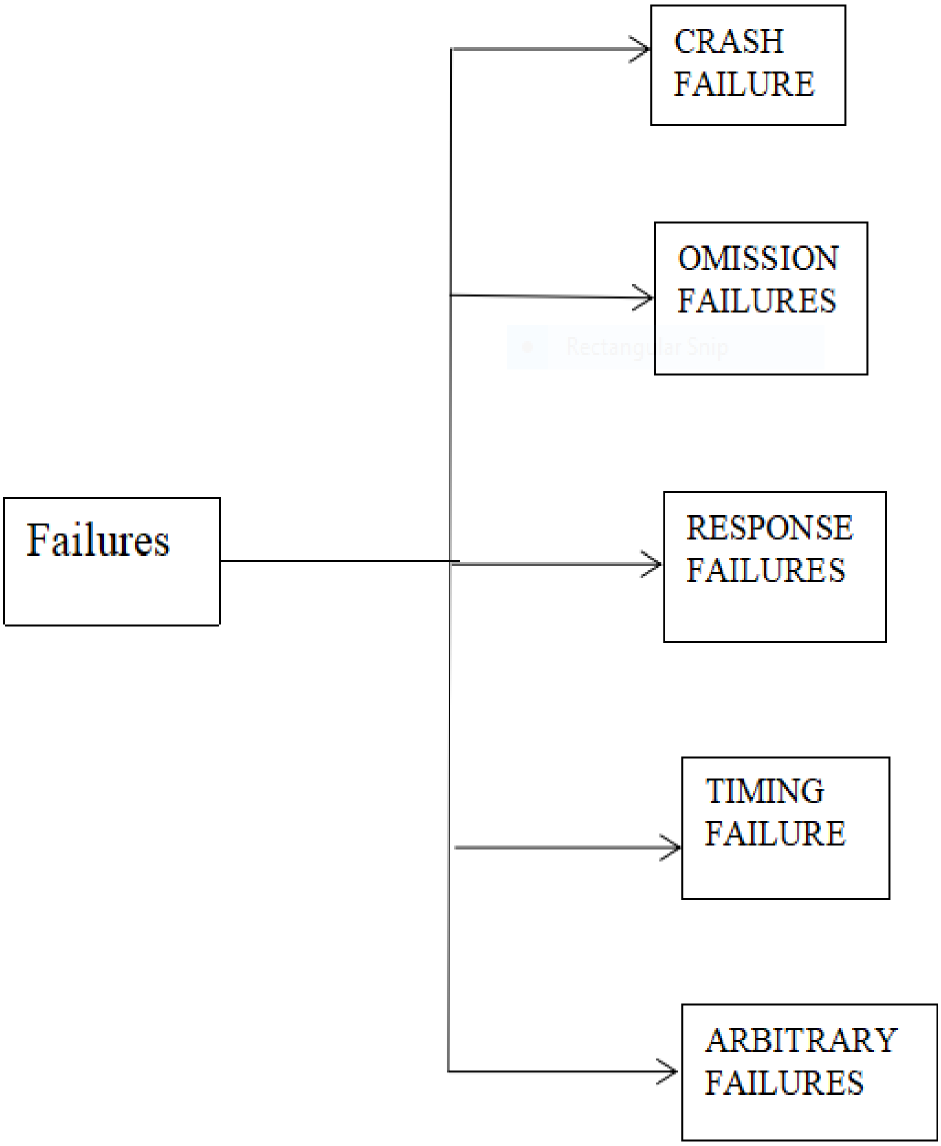

Fig. Types of failures under mixed category

9. *Fail silent breakdown*are the breakdownin which the process cannot variousiate between crash breakdownand omission failure.

10. *Fail-safe breakdown*deal with arbitrary breakdownand do not cause any kind of harm. It is quite clear that byzantine or arbitrary breakdownare the worst kind of faults.

FAILURE MEASURE:

There are common measures of breakdown includes: Failure Rate, Mean time to Failure, Mean time to Repair and Mean Time between Failure and defined as follows:

Failure rate can be defined as predicted number of breakdown in per unit time and it alter with time proportionally.

Mean time to failure (MTTF) used for checking stability of the system that cannot be fixed. It can be defined as predicted time until the first breakdown of system occurs. It is a statistical value which compute mean of large number of units. For reliability it can be defined as

$$\text{MTTF} = \frac{1}{\lambda}$$

Mean time to repair (MTTR) can be defined as the average time requisite to amend the system. It can be nominate mathematically as predicted number of fixations per unit of time. Fixing can also be seen as restoring or switching failed system.

$$\text{MTTR} = \frac{MTTF}{MTTF + MTTR}$$

Mean time between breakdown(MTBF) is a basic estimate of average time between various failures. It used to render the failure quantity per million hours for a product. MTBF can be used for both repairable and non repairable systems.[7]

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

various fault tolerance methods **:**

There are several methods that helps in handling faults which comes below Reactive , Proactive and Adaptational fault tolerance techniques which are discussed further below. Many authors have represented assorted methods of fault tolerance and parameters achieved with a specific fault tolerance technique including reliability and performance.

1. Reactive fault tolerance

The impact of breakdowncan be attenuate on the system that lead to cost and endeavor effectiveness which can be accomplished through reactive fault endurance. It is used after fault takes place. It helps in rising from unstable to stable state so that expected results can be achieved. It helps in plying system which means the system can handle variance and remain effective. It includes checkpointing, replication, self healing which are defined below.

1) *Checkpointing:* It is a method in which whenever a system confronts a breakdown, it is brought back to preceding correct state using up-to-minute checkpoint. The technique work on-basis

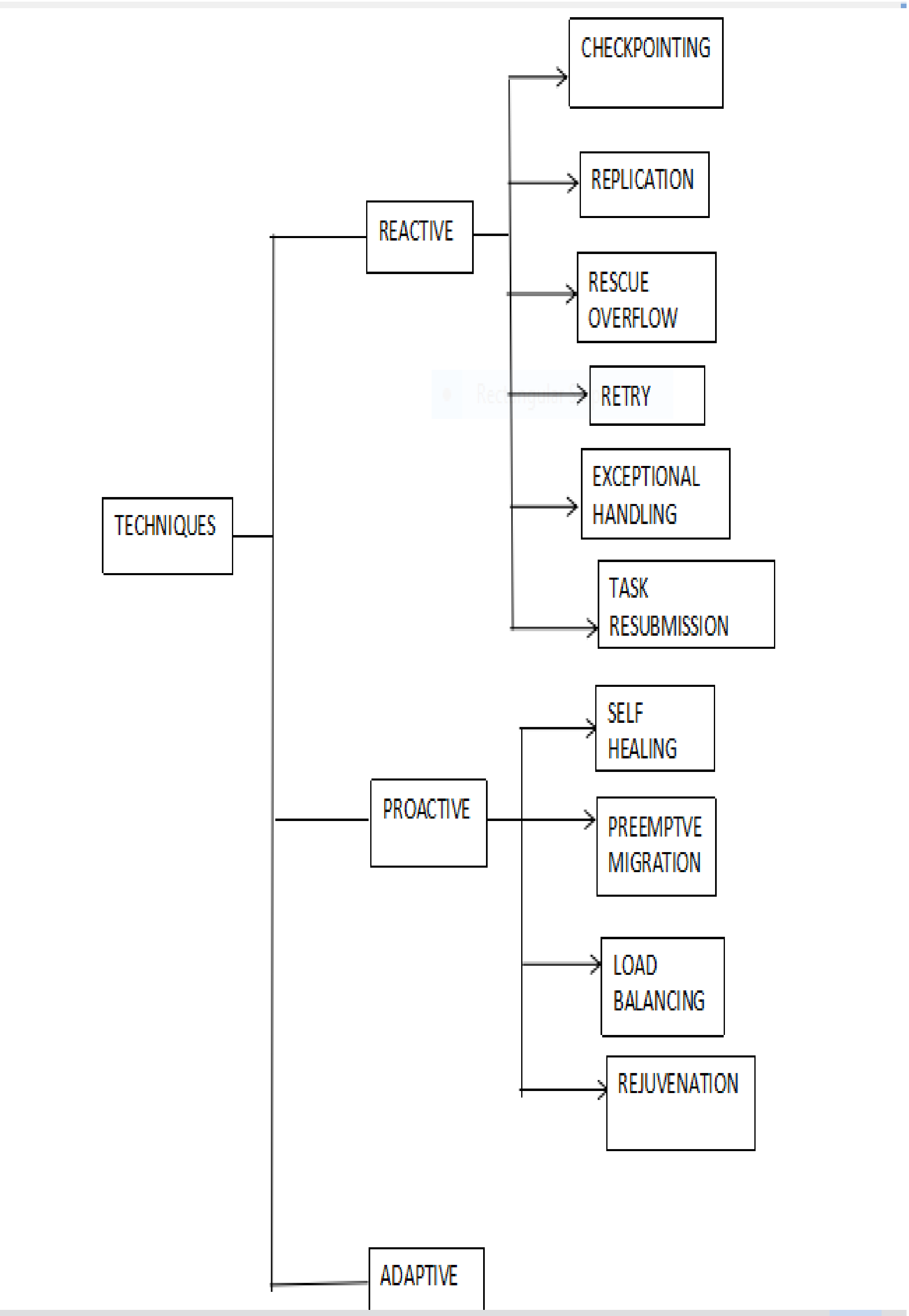

Fig. Techniques of fault tolerance

of saved state by system at various time periods in cache memory and when system undergoes a failure while the execution it can be redeemed from past saved state instead of restarting or rebooting the system. But saving state of the system very often at several time period can cause carrying out time overhead which can be considered as its stain. So checkpoint overhead should be diminishd.

- ✓ Incremental Checkpointing: This system helps in decreasing the checkpoint overhead as the modified pages are only saved onto disk inspite of saving the entire memory to disk. It stores the checkpoint from start to latest increment such that it could help in rebuild the system when failure takes place.

- ✓ Non incremental checkpointing- whole of the memory is saved to disk at each duration but it lead to difficulty in case of limited disk space and is found to be more cost effective.

2) *Replication:* Replication is methodin which job or data is copied in various machines such that if one fails then other copies can be perform the processing in similar manner. This process confirms consistency between redundant resources Replication can be divided into passive and active replication. *Passive* replication is one in which only one primary clone is there which performs functionality while the others are backup clone which save the state and only take over the job if primary fails. *Active* replication can leads to concurrently invoking of all replicas and process same request. Replication example tools includes Zerto which enables replication of hypervisor across range of machines. Replication helps to succeed in achieving fault tolerance , reliability and accessibility

3) *Rescue workflow*: this method helps the flow of work to not to stop and continue execution even if the tasks fails until it become impossible to move further. It permit processes to continue its workflow with any interruption or breakdown.[11]

4) *Self healing* : this technique enables the cloud device capable of detecting errors and breakdown and localizing the parts which are flawed one and healing those constituents with recovering algorithm. Multiple instance of application are installed and carrr out its processing on assorted virtual machines and if breakdown occurs it can be controlled automatically through many machines.

5) *Retry:* this technique is in which task is recomplied by the user in easiest manner with the request for running the task again.

6) *Exceptional handling* :In this method the user decided how to handle breakdownin the system and what steps to need to be taken.

7) *Task Re-submission*: whenever the breakdown happens, the task is submitted to same or various machine.[9]

2. **Proactive fault tolerance**

This type of fault tolerance helps to prevent faults before the occurrence of faults. In this technique the happening of faults is foreseen before they take place and supersede the faulty part with non faulty elements. It avoids betterment from faults. It includes preemptive migration, Load balancing and software rejuvenation discussed briefly below .

1) *Preemptive migration:* whenever any kind of task breakdown occurs, it can be moved from one to another machine cautiously which can be put it into action using tools like HAProxy, Hadoop.

2) *Rejuvenation :* It is quite analogous to reboot of the system and the rebooting is done after periodic intervals. Here virtual machines running in cloud environment are restarted and operating system is restored with clean state .[10]

3) *Load balancing :* when the CPU and memory load go beyond in extent for example if 75% utilization of CPU is the extent and if it exceeds then the load of CPU with exceeded limit is relocated to other CPU[8]

III. .EARLIER WORK

Tiancheng.C et al.[12] have focused on the problem occurred due to byzantine breakdownincluding malicious behaviour or threats in the publisher and subscriber model. They have proposed a model which aims to provide safety against byzantine breakdownby hiding or removing Byzantine faulty nodes eventually. In BFT (byzantine fault tolerance) cloning and overlaying of brokers are used in order to concealing the effects of byzantine failures.

Eman.A et al.[13] have projected a model which uses reactive technique of replication and resubmission of tasks both for enduring faults which jointly take in less resources and the rescheduling the task to node with the highest reliability. They have done their experiment on cloud simulating model and have compared their model with other model using replication and rescheduling their tasks without any betterment. The Data Centre Broker (DCB) test the

Vms and reschedules the task to highest reliability node if it do not succeed while the carrying out its execution. The reliability of a peculiar node is calculated on the foundation of several factors such as reliability factor, resubmission factor, maximum reliability and minimum reliability .

Ravi.J et al. [21] have focused on rigidity and non-transparent cloud environment due to abstraction behaviour which have put a burden on users and developers who knows a very little about the inner details of implementation. A various aspect is presented to create and come through fault tolerance by masking the complex implementation of replication technique details by addition of service layer. It permit users to use fault tolerance proficiency accordant to their requirement without deep expertise knowledge of its implementation.

Arvind.K et al.[6] have discussed several types of fault occurring and approaches for fault tolerance in real time distributed system .The approach is utlized in real time distributed system must accomplish the parameters including feasibility ,reliability and scalability.The feasibility of every task is quite essential as there is defined time bound for each task and task should be executed before the deadline.

Bashir.M et al. have given a model with adapted fault tolerance in real time in cloud infrastructure by using virtualization and checkpoints-replay scheme. The model is configured in such a way that nodes are elected based on their preceding pass rate and task assistance time. The algorithm used in the paper tolerates faults on the foundation of high pass rates of computing nodes and fault is tried to repair before the time limit of task finishing which increases the system availability.

Vincenzo.P et al.[4] have demonstrate a new dimension where users are provided fault tolerance as assistance either by third party or aid provider.The paper build a model on the principles of FTM, where virtualzation technique is used to provide high availability and reliability by replicating and migrating task to Vms. The reliability and availability of application at various levels using Markov models and fault trees.

Min.fe et al.[14] have given a taxonomy which classify various cloud recovery methods from normal activities to sproadic activities. It is analyzed which recovery method designed for which particular cloud activity and employed under which phase whether run-time or design time.The taxonomy is obtained from classifying cloud recovery methods into four groups and it can help by providing guidelines for literature reviews of cloud recovery techniques .

Pranesh.D et al.[15] have presented a virtualization and Fault tolerance (VFT) done by checkpointing, redundancy and fault handler for data centres. Here Hypervisor only distribute task to those virtual nodes has good performance and remove faulty nodes from recoverable nodes.The success rate is used as measure of performance of virtual node's. The model aims not only to diminish the possibility of fault but also its impact by not assigning task to virtual node's physical server with poor performance.

Jing.Liu et al.[16] render a scheme in which proactive fault tolerance as service termed as PFTaas for cloud application which provide attainble service. Proactive fault tolerance is provided to make precise tolerance of faults. It render feasible betterment towards convenience of cloud system. PFTaas is rendered in an efficient way in which reboot is done when breakdown happen and entire cloud service re-execution is avoided.

Alain.T et al[17] have discussed two FT policies where organization fault tolerance is obligation of cloud service provider or cooperative work by both customers and cloud provider. It utilizes a checkpointing method on each data center(Infrastructure as a service), which prevent the state of each VM every 7 seconds. The latest checkpoint is returned to a latest image of the unsuccessful VM when task is not accomplished successfully. Hardware breakdown can only be sensed and fixed by the cloud provider whereas VM breakdown can be sensed by the two participants but only fixed by the cloud provider and application breakdown can only be noticed by the client but can be fixed by the two participants.

IV. PARAMETER IDENTIFIED AND ANALYSIS

1. *Reliability* : It measure how accurate system execute its task without any failure. The system with high reliability is expected to execute without any kind of interruptions including errors or breakdown. It is defined in terms of time such that system continues to execute in the absence of occurrence of any breakdown for long interval for example a system would not be considered as reliable if it stopped its operation for even a milliseconds .

2. *Availability* is described as the characteristic in which a system is made accessible instantly. The estimation of time in which the system is in its executuon state is known as availability. A system is considered to be ready for use if the possibility of its working state is high at any instant of time, for example web users cannot wait for the websites to load for more than 7 seconds. It differs from reliability interms of its high tolerance degree of interruptions for short period of time whereaas reliability have low .

3. *Safety* refers to the situation that when a system temporarily fails to operate accurately. It can be defined as the likeliness of system of either performing accurately or discontinue its operation. For example, many process-control systems, such as those used for controlling nuclear power plants or alarm system which are needeed to render a high degree of safety. In case the system get down for small duration can cause disastrous results.

4. *Maintainability* refers to the probability of repair in a given period of time. Highly sustainable system may also show large intensity accessibilty, specifically if breakdown can be detected and fixed certainly.

5. *Response time*: the elapsed time to inquire a service request known as response time. Less response time is beneficial for efficient cloud computing.

6. *Scalability :* It is potential of a system to handle large work or its potential to work on large number of nodes without affecting the performance.

7. *Performance:* the more the system is reliable ,better is the execution.

S.NO	Paper title	Author	Context	Parameters attained
1.	A Reactive Fault Tolerance Approach For Cloud Computing	Eman AbdElfattah , Mohamed Elkawkagy and Ashraf El-Sisi	They have projected a framework which utilize reactive technique of duplication and resubmission of tasks both for tolerating faults which together consume less assets and the rescheduling the task to node with the highest reliability.	Reliability assessment improved
2	Comprehensive conceptual system level approach to fault tolerance in cloud computing.	Ravi Jhawar and Vincenzo Piuri	A model is presented to create and manage fault tolerance by masking the complex implementation of replication technique details.	Service flexibility is achieved with reliability and availability

3	Fault tolerance in real time distributed system	Arvind kumar	A grouping on several kinds of faults that takes place and methods for fault tolerance in real time shared system.	Reliability , scalability and feasibility are achieved .
4	Optimizing Fault tolerance in real time cloud computing in Iaas environment	Bashir Mohammed 1, Mariam Kiran2 ,Irfan-Ullah Awan3 and Kabiru M. Maiyama	Fault environment in rendered real time in cloud store by using virtualization and checkpoints-replay scheme.	Reliability and high availability.
5	A VFT: A Virtualization and Fault Tolerance Approach for Cloud Computing	Pranesh Das, Dr.Pabitra Mohan Khilar	A Cloud Manager (CM) module and a Decision Maker (DM) are utilized in our system to carry off the virtualization, load balancing and to dealing the faults	diminish service time and high availability .
6	Providing proactive fault tolerance	Jing lui	PFTaas is rendered in convenient way in which cloud service reboot is done when breakdown occurs and whole cloud service re-execution is evaded.	High availability with feasibility and cost effectiveness .
7	Fault tolerance management in Iaas cloud computing.	Ravi Jhawar and Vincenzo Piuri	The paper configured a framework on the principles of FTM, where virtualzation method is used to render high availability and reliability by copying duplicate vms and moving task to Vms	High availability and reliability.
8	An Improved Digital Signature Scheme with Fault Tolerance in RSA	Iuon-Chang Lin and Hsing-Lei Wang	the strategy is capable to detecting and accurating the error happening in the calculating processes or data transferring rocedure.	Confidentiality and safety

9.	Fault Tolerance-Challenges, Techniques and Implementation in Cloud Computing	Anju Bala and Inderveer Chana	The Architecture used in the system founded on HAProxy software. Autonomic fault tolerance handles several software faults.	Reliability
10	BFTCloud: A Byzantine Fault Tolerance Framework for Voluntary-Resource Cloud Computing	Yilei Zhang, Zibin Zheng and Michael R. Lyu	This fault tolerance structure for constructing powerful systems in voluntary-resource cloud surroundings.	High reliability and good performance and robust behaviour.

Fig.comparative analysis of parameters achieved

various architecture developed for fault tolerance :

Low latency fault tolerance (LLFT) : LLFT model render fault tolerance by making duplicate copies of processes of application through a replication method with powerful stability . It can deal with crash, timing faults but not Byzantine fault. LLFT render unreliable transmission service to reliable and inferior delay. The virtualization framework ensures that backup replicas can perform identical as primary copies. The procedure that takes place in LLFT includes retrieval from primary failure, backup failure and discharge of copies. The LLFT middleware render degrade overhead, transparent application which is best for applications in cloud.[18]

Adaptive fault tolerance AFTRC :AFTRC is model configured for providing fault tolerance in real time computing. It assess the reliability of reach virtual node and dependability of each virtual node is altered in nature which modifies after a very computing cycle. Checkpointing are also stored at last when the results of all nodes are produced.[19]

Byzantine Fault tolerance (BFTcloud) model can tolerate various type of faults including handling byzantine faults. It ensures good performance and improved reliability as compared to various approaches. It guarantees vigorous of system by using cloning method on voluntary resources cloud environment. When the node found to be faulty cloud will update the BFT to assure the system reliability.

Proactive fault tolerance as service (PFTaas) is an architecture which render high availability for cloud application with cost effectiveness . This framework use PFTaas in situation like live migration where faulty components are migrated for mitigating the caused degradation and software rejuvenation which diminish the aging problem of hypervisor which can cause several failures.[16]

FT Models /architecture	Technique used	Reliability	Availability	Resource awareness
FTM	Reactive	YES	YES	YES
LLFT	Reactive	YES	YES	NO
VFT	Reactive	NO	YES	YES
AFTRC	Reactive/ Proactive	YES	NO	NO
BFTcloud	Reactive	NO	YES	NO
PFTaas	proactive	NO	YES	NO

Virtualized fault tolerance(VFT) : This given model works on the strategy of reactive fault tolerance that tolerate the faults by measuring success rate of each virtual node's physical server. A client submits task to Cloud Service Provider(CSP) and then submits further to Cloud Manger(CM) which performs virtualization and maintain records of which node belonging to which virtual machine and then shrewd decision maker (DM) makes judgment on the basis of success rates. The given model focus on high availability by using virtualization with fault tolerance scheme. [15]

FAULT TOLERANCE MACHINE(FTM) : The FTM model offers on demand service with fault tolerance to users by replicating user applications. It detects faults using heartbeat message exchange procedure and invoke the fault masking and recovery service which increases the lifetime of system. It allows users to use replication technique of fault tolerance without knowing its detailed implementation.[4]

V. CONCLUSION

Fault tolerance is quite essential to be implied in this modern era for providing on demand services to cloud users. Fault tolerance helps the users to accomplish their task even in the occurrence of breakdownand errors. This paper have mentioned various types of faults and failures. IV section have compared various models on the basis of their parameters achieved. various techniques and architecture used by research experts are focused to get better understanding of working of fault tolerance. Reliability and high availability are the major services to be attained in fault tolerance.

VI. REFERENCES

1. P.Guo and Z.Xue Real-Time "Fault-Tolerant Scheduling Algorithm with Rearrangement in Cloud Systems" IEEE journal(2017)

2. Q.zhang, L.Cheng,R.Boutaba "Cloud computing:state-of-the-art and research challenges" ,J Internet serve appl(2010).

3.A.D.Meshram, A.S.Sambare S.D.Zade"Fault Tolerance Model for Reliable Cloud Computing"International Journal on Recent and Innovation Trends in Computing and Communication(JULY 2013).

4. R.Jhawar, V.Piuri, M.Santamborgio "Comprehensive conceptual system level approach to fault tolerance in cloud computing"(2011).

5. B.Mohammed1, M.Kiran ,I.Awan and Kabiru.M.Maiyama "Optimising Fault Tolerance in Real-time Cloud Computing IaaS Environment" 2016 IEEE 4th International Conference on Future Internet of Things and Cloud.

6. A.Kumar, R.Yadav, Ranvijay, A.Jain "Fault Tolerance in Real Time Distributed System"(2011)International Journal on Computer Science and Engineering.

7. H.Aggarwal, A.Sharma "A Comprehensive Survey of Fault Tolerance Techniques in Cloud Computing"2015 Intl. Conference on Computing and Network Communications (CoCoNet'15).

8. Salma M. A. Ataallah, Prof. Salwa M. Nassar "Fault Tolerance in Cloud Computing - Survey".

9. S.Prathiba, S. Sowvarnica "Survey of breakdownand Fault Tolerance in Cloud"2017 Second International Conference On Computing and Communications Technologies(ICCCT'17)

10. J.Liu, J.Zhou , R.Buyya "Software Rejuvenation based Fault Tolerance Scheme for Cloud Applications", 2015 IEEE 8th International Conference on Cloud Computing.

11. S.Talwani, I.Chana "Fault Tolerance Techniques for Scientific Applications In Cloud"2017 2nd International Conference on Telecommunication and Networks (TEL-NET 2017)

12. T.Chang and H.Meling " Byzantine Fault-Tolerant Publish/Subscribe: A Cloud Computing Infrastructure" 2012 31st International Symposium on Reliable Distributed Systems.

13. AbdElfattah, M.Elkawkagy and A.El-Sisi "A Reactive Fault Tolerance Approach For Cloud Computing Eman"2017 IEEE journal.

14. AbdElfattah, M.Elkawkagy, A.El-Sisi "Towards a Taxonomy of Cloud Recovery Strategies" 2014 44th Annual IEEE/IFIP International Conference on Dependable Systems and Networks.

15. P.Das, Dr.P.M.Khilar "VFT: A Virtualization and Fault Tolerance Approach for Cloud Computing" Proceedings of 2013 IEEE Conference on Information and Communication Technologies (ICT 2013).

16. J.Liu, J.Zhao "Providing Proactive Fault Tolerance as a Service for Cloud Applications "2016 IEEE World Congress on Services Computing.

17. A.Tchana, L.Broto, D.Hagimont "Approaches to Cloud Computing Fault Tolerance"2012 IEEE .

18. W.Zhao, P.M.Melliar-Smith and L.E.Moser "Fault Tolerance Middleware for Cloud Computing" 2010 IEEE 3rd International Conference on Cloud Computing.

19. S.Malik, F.Huet "Adaptive Fault Tolerance in Real Time Cloud Computing" 2011 IEEE World Congress on Services.

20. Y.Zhang, Z.Zheng and M.R.Lyu "BFTCloud: A Byzantine Fault Tolerance Framework for Voluntary-Resource Cloud Computing" 2011 IEEE 4th International Conference on Cloud Computing.

21. R.Jhawar, V.Piuri "Fault Tolerance Management in Cloud Computing: A System-Level Perspective"(2012 IEEE)