Efficient BFT Technique render Reliable Coordination problem in WS-AT specification

Mrs.V.Kavitha[1], P.Jamuna[2]

[1]Assistant professor,
Department of Computer Science and Engineering,
Christ College of Engineering and Technology,
[2]M.Tech-II Year,
Department of Computer science and Engineering,
Christ College of Engineering and Technology,

ABSTRACT

In this paper,we present Byzantine Fault Tolerance (BFT) Technique in order to overcome the reliable coordinator problem in distributed transaction processing over Internet.In addition,we proposed Efficient Enhanced Three Phase Commit protocolEE3PC) which reduces the usage of Byzantine Agreement which is prohibitively expensive during the registration of each participant. The EE3PC performs the work of View change Algorithm (i.e. Elect new primary replica when the existing replica is suspected to be Byzantine faulty) and it overcome the problem of resource blocking.In WS_AT specification,the coordinator provides some core services to participant and to initiator,which makes the transaction reliable even in untrusted environment over the Internet.Our resulting BFT technique will achieves higher availability of data due to server replication,security,trust and it doesn’t have much communication overhead.

Keywords: Byzantine Fault, Web Service, Atomic Transaction, Byzantine Agreement, Distributed Transaction Processing.

1. INTRODUCTION

The Web Service Atomic Transaction standard makes many such business transaction processing are being deployed in Internet using Web Service technology. Always distributed transaction processing involves three attributes are Initiator, Coordinator and Participant. Initiator is responsible for starting and ending up the transaction. Coordinator is responsible for maintaining different states in transaction and provide interoperability among multiple web service and Participant are the end users. Reliable Coordinator in business transaction processing is crucial. The coordinator should tolerate the byzantine Fault [3] and should perform well in untrusted environment in Internet.

The WS_AT specification will provide interoperability among distributed transactional Web Services .In WS_AT standard specifies , the reliable coordinator provide some core services to Initiator and to Participant namely Activation service, Registration service , Coordination service, completion service[7]. The Byzantine Fault Tolerance Technique which makes system to avoid Byzantine Fault. The Byzantine Fault refers system may get failed in arbitrary ways such as hardware failure, software failure etc. In order to have high availability of data,server is replicated and all replicas should reach an agreement to avoid Byzantine Fault such an agreement called Byzantine Agreement. Among various BFT algorithm [2], [4], [5], here we chooses the Practical Byzantine Fault Tolerance for Byzantine Agreement [1].

In our work, we proposed Efficient Enhanced Three Phase Commit protocol (EE3PC) which reduces the usage of Byzantine Agreement during the registration of each participant. If non faulty Participant gets registered with coordinator they can enter into EE3PC protocol[6]. This EE3PC will overcome some of the disadvantages faced by 2PC, 3PC, Quorum based protocol etc. The EE3PC protocol performs the work of view change algorithm [8]. According to WS_AT standard specifies two protocols are EE3PC protocol and Achievement protocol. In which, Achievement protocol is allowed to run between Initiator and Coordinator and EE3PC is allowed to runbetween Coordinator and Participant.Achievement protocol is a simple protocol that notify about the outcome of the transaction if it is committed successfully and EE3PC protocol is explained briefly in section 2.3.

2.RELATED WORK

In this section we provide here a Synopsis about various protocols which tolerate the Byzantine Fault such as 2PC, 3PC**.**

2.1 Two Phase Commit protocol (2PC)

Two Phase Commit protocol or blocking protocol is a standard distributed commit protocol which involves 2 phase are Prepare phase and Commit phase.

First phase (Prepare phase)

1. Coordinator will broadcast request message to all participants.

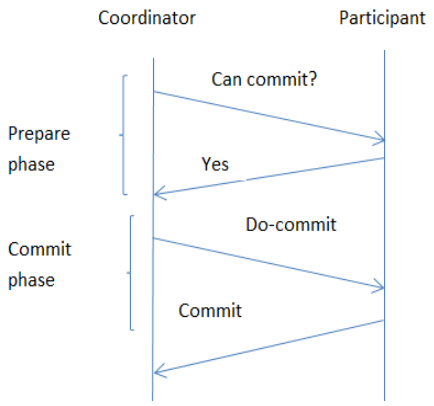
2. If the participant want to commit the transaction means it respond with yes otherwise no.

3. When the coordinator receives the reply from participant it starts 2nd phase.

Second phase (Commit phase)

1. Coordinator will notify participant about the outcome of the transaction

However 2PC protocol is easy to implement it has some major drawbacks such as blocking problem.The Blocking problem is described with a given scenario is , If the participant locks some required resources while it is waiting for the reply message of failure coordinator at the mean time other process competing for the same resources will have to wait for the locks to be released, there exist Blocking problem.



2.2 Three Phase Commit protocol (3PC)

Three Phase Commit protocol (3PC) or non- blocking protocol which avoids blocking problem faced by 2PC protocol as the participant can hold some resources for some mean time and it will release the locks after timeout and it doesn't wait for the reply message from the fault coordinator.

Three Phase Commit protocol involves three phases are Pre-prepare phase,Prepare phase, Commit/Abort phase. Among these Pre-Commit phase which avoids blocking problem.

First phase (Prepare)

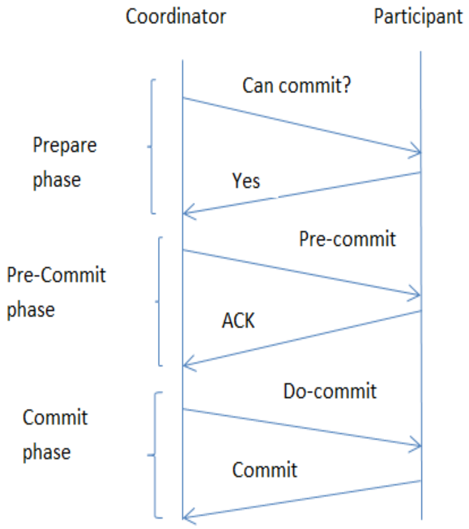
1. Coordinator broadcast can commit?Request to all participant.
2. If the participant is able to commit the transaction it responds with yes otherwise no.
3. When the coordinator receives reply from participant it starts the 2nd phase.

Second phase (Pre-Commit or Buffering)

1. If the coordinator receives yes message within time it broadcast Pre-Commit message to all participant. (In this phase preliminary decision can be made by the coordinator)
2. If the participant accept the message it will acknowledge the coordinator
3. When the Coordinator receive ACK it starts 3rd phase

Third phase (Commit/Abort phase)

1. Notify the participants about the transaction outcomes.
2. Coordinator decides to commit or Abort



However 3PC have some drawback such as difficult to implement, having communication overhead, inconsistency in network partitioning. In order to overcome above problems we proposed Efficient Enhanced 3PC commit protocol makes the distributed transaction as trustworthy.

3.PROPOSED WORK

3.1 Efficient Enhanced Three Phase Commit Protocol (EE3PC)

Our proposed EE3PC is quorum based 3PC will overcome some of the drawback faced by 2PC,3PC such as blocking problem, inconsistency in network partitioning, communication overhead.EE3PC recovery procedure includes, at each invocation the site try to elect new coordinator if the existing site is suspect to be byzantine fault. Thus EE3PC reduces the overhead of View change Algorithm usage. Here we described the synopsis of view change algorithm with the following scenario when a faulty primary broadcast conflicting message to different replica view change algorithm is used and the backup replica will initiate this view change algorithm. This algorithm is used to select a new primary when the existing primary is suspected to be Byzantine faulty.

EE3PC achieve higher availability than 3PC by maintaining 2 additional counters

Pre_Elected:- Declare its Initial value is 0, It takes count the no. of election took part in site and the variable value is updated when new coordinator is elected.

Pre_Attempt:- Declare its initial value is 0. It denotes Election number in last election.

Operations involved in EE3PC

1. Elect a new coordinator(r),in case of failure, r hears the value of Pre_Elected and Pre_Attempt from all sites. r Determine Tot_Elected and Tot_Attemptand sets Pre_ Elected→Tot_Elected+1 and notify value Tot_Elected to all participants(P).

2. P sets Pre_ Elected→Tot_Elected+1 and report to coordinator r.

3. Coordinator collects states from all sites P and make the decision. Upon decision r sets Pre_Attempt→Pre_Elected

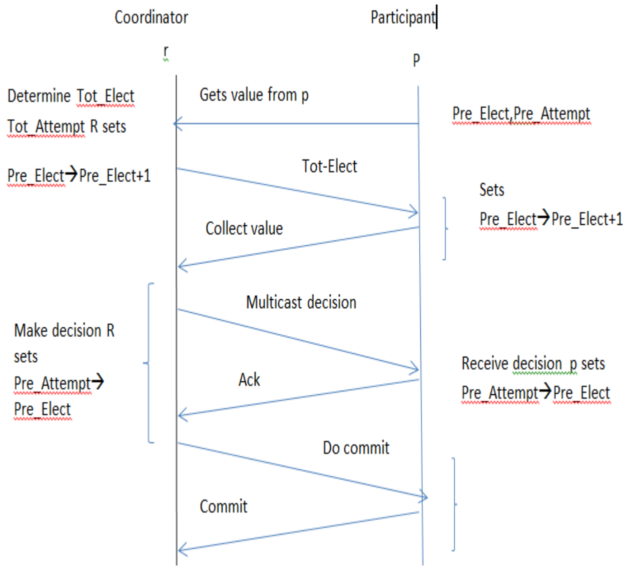
Collected States	Decision
ЭABORTED	ABORT
ЭCOMMITTED	COMMIT
Tot_Attempt_Committable (true)	PRE-COMMIT
Tot_Attempt_Committable (False)	PRE-ABORT
Otherwise	BLOCK

Where Tot_Attempt_commitable is true iff all the members in non-final state and Pre_Attempt is equal to Tot_Attempt.

4. Uponreceiving pre-commit or Pre-abort from r, P sets Pre_Attempt→Pre_Elected and send ACK to r.

5. Upon receiving ACK, r multicast its decision

6. Upon receiving message from r,process the transaction

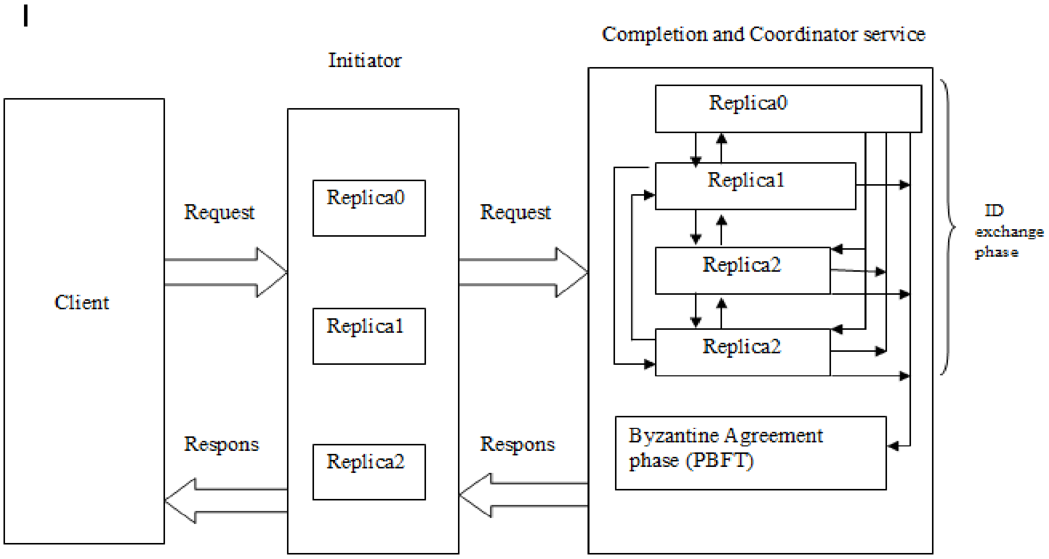


4. BFT CORE SERVICES

According to Web Service Atomic Specification,the coordinator offers a set of services to Participants and to the Initiator are Registration service, Activation service, Completion service and coordination service. By making core service as harden, the coordinator can be trustworthy even in the untrusted environment of the Internet.

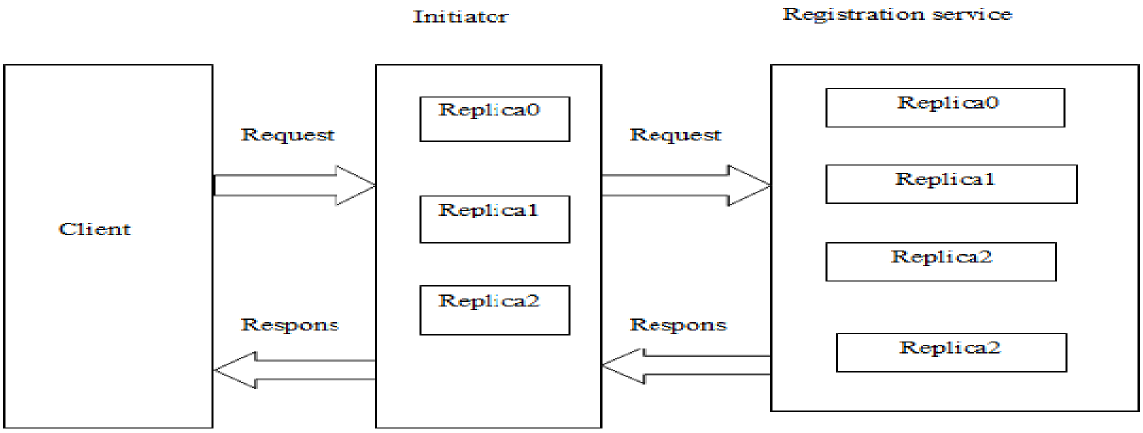
4.1 Activation Service

Activation Service which creates coordination object and transaction id where Transaction id is a part of coordination object.In order to maintain security transactionid can be chosen from random number so that it cannot be predicted by other replicas. The Activation service which allows inter replica communication it allows to generate the unique transaction id and initiate the Byzantine Agreement phase. In this paper, we choose the Practical Byzantine Fault Tolerance (PBFT) Algorithm for the Byzantine Agreement in the implementation [].For example,when we requesting for any transaction operation at the time the activation service only activate the particular operation.



4.2 Registration service

The Registration service allows the Participants and the Initiator to register their endpoint references and it has no inter replica communication. For example If the new user wants to create the new account means at the movement bank service send the request to the registration service ,the registration service verify all information and response to bank service. Bank service response to customer.

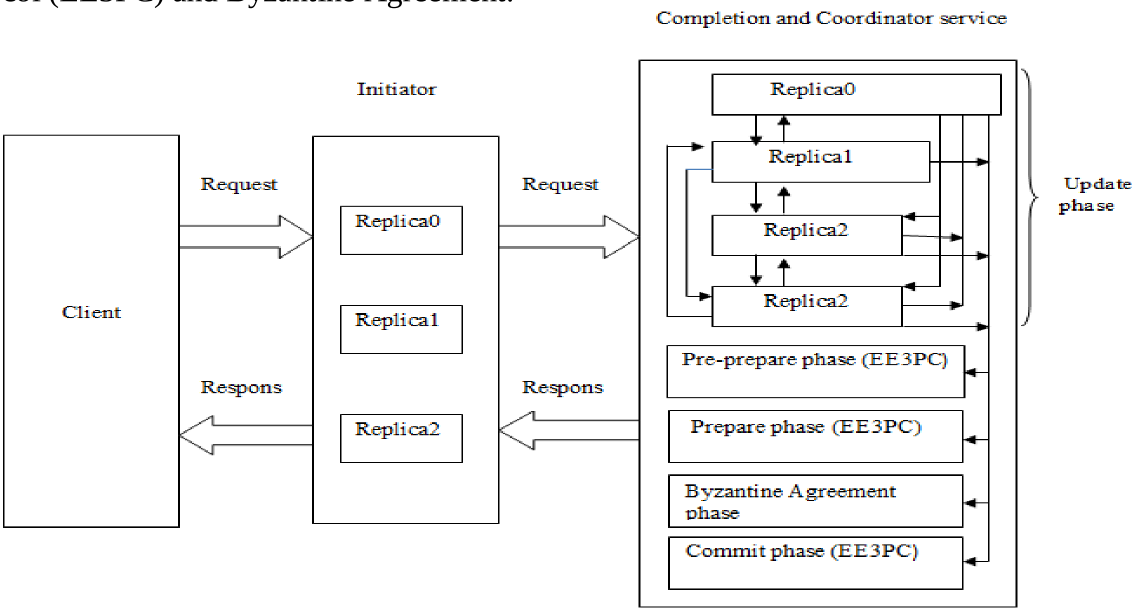


4.3 Completion Service

Completion service is allowed to activate a successful commits operation only. For example If user want to transfer amount from one account to another account at the time it send the request to the coordinator service, the coordinator analyze the account details and fund transaction and reply to the completion service then it send to the response bank service and bank response to the customer transaction.

4.4 Coordinator service

It provides the interoperability among all service. Certainly any service is fault means it redirect toother replica service. It contains all information about the transaction. The coordinator service which allows the inter replica communication, it allows to update the registration record (i.e. it checks whether the entirenon-faulty participant are registered successfully or not.).Coordinator service which runs the Efficient Enhanced Three Phase Commit protocol (EE3PC) and Byzantine Agreement.



5. CONCLUSION

Thus our Byzantine Fault Tolerance (BFT) Technique with the Efficient Enhanced Three Phase Commit protocol (EE3PC) had overcome the reliable coordinator problem in distributed transaction processing over Internet and our proposed work had reduced the usage of Byzantine Agreement. In addition, our EE3PC performs the work of View change Algorithm so our work shows only moderate runtime overhead. By hardening the core services, the coordinator can be a reliable entity even in untrusted environment over the internet. Thus resulting BFT technique will achieves higher availability of data due to server replication,security,trust and it doesn't have much communication overhead.

REFERENCE

[1] M.Castro and B. Liskov, "Practical Byzantine Fault Tolerance and Proactive Recovery,"*ACM Trans. Computer Systems,vol.20, no. 4,pp. 398-461,Nov.2002.*

[2] Y, Amir, B.A. Coan, J.Kirsch, and J. Lane," Byzantine Replication under Attack," *Proc. IEEE Int'l Conf. Dependable Systems and Networks, pp. 105-144, June 2008.*

[3] L. Lamport, R. Shostak, and M. Pease, "The Byzantine Generals Problem," *ACM Trans. Programming languages and Systems, vol. 4, no. 3,pp. 382-401, July 1982.*

[4] J. Cowling, D. Myers, B. Liskov, R. Rodrigues and L. Shri," HQ Replication: A Hybrid Quorum Protocol for Byzantine Fault Tolerance," *Proc. Seventh Symp. Operating Systems and Implementation, pp. 177-190, Nov. 2006.*

[5] R. Kotla, L. Alvisi, M. Dahlin, A.Clement, and E. Wong,"Zyzzyva: Speculative Byzantine Fault Tolerance," *Proc. 21st ACM Symp. Operating Systems Principles,pp. 45-58,Oct.2007.*

[6] IditKeidar, Danny Dolev," Increasing the Resilience of Distributed and Replicated Database Systems*", J. Comut. Systems Science, December 1998*.

[7]Wenbing Zhao,"Byzantine Fault Tolerant Distributed Commit Protocol," *Dependable, Autonomic and Secure Computing, 2007. DASC 2007.*

[8]Honglei Zhang, Hua Chai, Wenbing Zhao, P.M.Melliar-Smith, L.E. Moser, "Trustworthy Coordination of Web Services Atomic Transaction", IEEE Transactions on Volume: 23 , Issue: 8