

**A FRAMEWORK FOR ENTERPRISE BLOCKCHAIN FAULT
TOLERANCE WITH STATE MACHINE REPLICATION**

**IKOH, OBARO BENEDICT
(20PCG02180)**

DECEMBER, 2022

**A FRAMEWORK FOR ENTERPRISE BLOCKCHAIN FAULT
TOLERANCE WITH STATE MACHINE REPLICATION**

BY

**IKOH, OBARO BENEDICT
(20PCG02180)**

**BTech. Mathematics and Computer Science, Federal University of
Technology, Minna**

**A DISSERTATION SUBMITTED TO THE SCHOOL OF
POSTGRADUATE STUDIES IN PARTIAL FULFILMENT OF THE
REQUIREMENT FOR THE AWARD OF MASTER OF SCIENCE
(M.SC) DEGREE IN COMPUTER SCIENCE DEPARTMENT,
DEPARTMENT OF COMPUTER AND INFORMATION SCIENCES,
COLLEGE OF SCIENCE AND TECHNOLOGY, COVENANT
UNIVERSITY, OTA, OGUN STATE, NIGERIA**

DECEMBER, 2022

ACCEPTANCE

This is to attest that this dissertation is accepted in partial fulfilment of the requirements for the award of the degree of Master of Science in Computer Science in the Department of Computer and Information Systems, College of Science and Technology, Covenant University, Ota, Nigeria.

Miss. Adefunke F. Oyinloye
(Secretary, School of Postgraduate Studies)

Signature and Date

Prof. Akan B. Williams
(Dean, School of Postgraduate Studies)

Signature and Date

DECLARATION

I, **IKOH, OBARO BENEDICT (20PCG02180)** declare that this research was carried out by me under the supervision of Dr. Iheanetu U. Olamma of the Department of Computer and Information Sciences, College of Science and Technology, Covenant University, Ota, Ogun State, Nigeria. I attest that this dissertation has not been presented either wholly or partially for the award of any degree elsewhere. All sources of data, scholarly information used in this dissertation are duly acknowledged.

IKOH, OBARO BENEDICT

Signature and Date

CERTIFICATION

We certify that this dissertation titled “**A FRAMEWORK FOR ENTERPRISE BLOCKCHAIN FAULT TOLERANCE WITH STATE MACHINE REPLICATION**” is an original research carried out by **IKOH, OBARO BENEDICT (20PCG02180)** in the Department of Computer and Information Sciences, College of Science and Technology, Covenant University, Ota, Ogun State, Nigeria under the supervision of Dr. Iheanetu U. Olamma. We have examined and found this work acceptable as part of the requirements for the award of Master of Science (M.Sc.) in Computer Science.

Dr. Iheanetu U. Olamma
(Supervisor)

Signature and Date

Prof. Olufunke O. Oladipupo
(Head of Department)

Signature and Date

Prof. Olufunke R. Vincent
(External Examiner)

Signature and Date

Prof. Akan B. Williams
(Dean, School of Postgraduate Studies)

Signature and Date

DEDICATION

I dedicate this dissertation to the Almighty God unreservedly.

ACKNOWLEDGEMENTS

First, I want to acknowledge God Almighty for his mercy that enabled me to carry out this project. I like to specifically thank my wife for her relentless support and love and my parents for their continuous prayers.

My thanks also go to the Chancellor of Covenant University, Dr David O. Oyedepo, for his dedication to creating a generation of leaders with a difference. I applaud the entire management of Covenant University for the part they played during my programme. I would like to appreciate my supervisor, Dr. Iheanatu U. Olamma, for her support and for pushing me all the way. Furthermore, I want to appreciate the Faculty at Computer and Information Science Department for the learning experience, Dr Ezenwoke Azubike and Dr Ibukun Afolabi for their support while developing this dissertation, and the Head of Department, Professor Olufunke O. Oladipupo.

I also want to acknowledge my Post Graduate Colleagues; Seth Samuel and Otavie Okuoyo who stood by me and made sure I never gave up. To my colleagues Deborah Bassey Etukudo, Emmanuel Omonedo, Ogunsola Opeyemi, Chidera Eze, Christabel Uzor, Emmanuel Owoka, Excellent Ikeakanam, Gabriel Opeyemi, Isaac Martins, Samuel Olanipekun, Temitope Ogungbesan, Timilehin Owoseni, Victoria Robert, Olumide Adeosun who have contributed in many ways to my post graduate experience. I am grateful to you all, and I celebrate you all.

TABLE OF CONTENTS

CONTENTS	PAGES
COVER PAGE	i
TITLE PAGE	ii
ACCEPTANCE PAGE	iii
DECLARATION	iv
CERTIFICATION	v
DEDICATION	vi
ACKNOWLEDGEMENTS	vii
TABLE OF CONTENTS	viii
LIST OF FIGURES	xi
LIST OF TABLES	xiii
LIST OF ABBREVIATIONS	xiv
ABSTRACT	xvii
CHAPTER ONE: INTRODUCTION	1
1.1 Background to the Study	1
1.2 Statement of the Problem	5
1.3 Aim and Objectives of the Study	5
1.4 Study Methodology	6
1.5 Significance of the Study	6
1.6 Scope of the Study	7
1.7 Dissertation Organization	7
CHAPTER TWO: LITERATURE REVIEW	8
2.1 Introduction	8
2.2 Overview of Blockchain Technology	8
2.3 Blockchain Model	9
2.3.1 Public/Permissionless Blockchain	10
2.3.2 Private/Permissioned Blockchain	14
2.3.3 Hybrid Blockchain	19
2.3.4 Consortium Blockchain	19
2.4 Key Characteristics of Blockchain	20
2.4.1 Decentralization	20
2.4.2 Cryptography	21
2.4.3 Consensus Mechanism	21
2.4.4 Business Logic	21

2.5	Versions and Use Cases of Blockchain	21
2.5.1	Blockchain 1.0	23
2.5.2	Blockchain 2.0	24
2.5.3	Blockchain 3.0	25
2.5.4	Blockchain 4.0	27
2.6	Values of Blockchain Technology to Business	28
2.6.1	Immutability	28
2.6.2	Decentralisation	28
2.6.3	Transparency	28
2.6.4	Consensus	28
2.6.5	Integrity	29
2.6.6	Fault Tolerance	29
2.7	Challenges of Blockchain Technology	30
2.7.1	Throughput	30
2.7.2	Latency	30
2.7.3	Size and Bandwidth	31
2.7.4	Security	31
2.8	Concept of Enterprise Blockchain	31
2.8.1	Enterprise Blockchain Architecture	32
2.8.2	Enterprise Blockchain Consensus Algorithm	34
2.9	When to use Enterprise Blockchain	36
2.10	Use Cases of Enterprise Blockchain	37
2.11	Enterprise Blockchain Frameworks	39
2.12	Vulnerabilities of Enterprise Blockchain	40
2.12.1	Core Layer Vulnerability	40
2.12.2	Underlying Layer Vulnerability	41
2.12.3	Cross-Layer Function Vulnerability	41
2.12.4	User Layer and Service Layer Vulnerability	41
2.13	Overview of State Machine Replication (SMR)	42

2.14	Fault Tolerance in Blockchain Technology	43
2.14.1	Power Source	44
2.14.2	Computer Software	44
2.14.3	Computer Hardware	44
2.15	Need for Fault Tolerance for Enterprise Blockchain	44
2.16	Review of Related Works	45
2.17	Gaps in Literature	50
	CHAPTER THREE: METHODOLOGY	54
3.1	Introduction	54
3.2	Proposed Framework	55
3.2.1	Proposed SMR Fault Tolerance Framework for Enterprise Blockchain	56
3.2.2	Design of the Replicas Algorithms	58
3.2.3	Pseudocode of Replica Algorithm	59
3.3	Proof of Concept	61
	CHAPTER FOUR: RESULTS AND DISCUSSION	62
4.1	Introduction	62
4.2	Experimental Setup	62
4.3	Experimental Results	63
4.3.1	Evaluation of the Implemented Framework	63
4.3.2	Establishing the Network and Joining the Network	63
4.3.3	Results of the Network Ability and Reliability	64
4.4	Discussion	65
	CHAPTER FIVE: CONCLUSION AND RECOMMENDATION	66
5.1	Conclusion	66
5.2	Contribution to Knowledge	67
5.3	Recommendation	67
	REFERENCES	68
	APPENDIX	76

LIST OF FIGURES

FIGURES:	TITLE OF FIGURES	PAGES
2.1	Classes of Permission and Permissionless Blockchain Consensus Mechanisms	9
2.2	Evolution of Blockchain Technology	23
2.3	A Flowchart of When to use Blockchain	37
3.1	Research Methodology Workflow	54
3.2	Identification of Threat Models	55
3.3	The Proposed Framework	56
3.4	Phases of the Proposed Replicated Algorithm	58
3.5	Phases of Replicas	60
4.1	Graphs Illustrating the Average Time Taken for Nodes to Joining the Network in Seconds	64

LIST OF TABLES

TABLES	TITLE OF TABLES	PAGES
1.1	Mapped out Methodology	6
2.1	Summary of the Different Versions Blockchains	27
2.2	Use Cases of Enterprise Blockchain	38
2.3	Enterprise Blockchain Technology and Framework	39
2.4	Identified Research Gaps in Literature	50
4.1	Experimental Setup	62
4.2	Nodes Joining the Network in Seconds	64
4.3	Execution Results	65

LIST OF ABBREVIATIONS

AI	Artificial Intelligence
BFT	Byzantine Fault Tolerance
CFT	Crash Fault-Tolerant
CPU	Central Processing Unit
DAC	Decentralized Autonomous Corporation
DAG	Directed Acyclic Graph
DAO	Decentralized Autonomous Organization
DApp	Decentralized Application
DBFT	Delegated Byzantine Fault Tolerance
DDoS	Distributed Denial of Service
DLT	Distributed Ledger Technology
DoS	Denial-of-Service
DR	Disaster Recovery
EBDF	Enterprise Blockchain Design Framework
FBA	Federated Byzantine Agreement
FBFT	Federated Byzantine Fault Tolerance
FSM	Finite State Machine
GPU	Graphics Processing Unit
HPE	Hewlett Packard Enterprise
hOCBS	hybrid Off-Chain Blockchain Systems
IBM	International Business Machines
IT	Information Technology
IoT	Internet of Things
NEM	New Economic Movement
NONCE	Number used ONCE
O & M	Operation and Maintenance
P2P	Peer-to-Peer
PBFT	Practical Byzantine Fault Tolerance
PoET	Proof-of-Elapsed-Time

PPCA	Point-to-Point Channel Authentication
PoI	Proof-of-Importance
PoL	Proof-of-Location
PoS	Proof-of-Stake
PoW	Proof-of-Work
PPCA	Point-to-Point Channel Authentication
RAM	Random Access Memory
RDMA	Remote Direct Memory Access
RSM	Replicated State Machine
SAP	Systems Applications and Products
SC	Smart Contract
SCADA	Supervisory Control and Data Acquisition
SMR	State Machine Replication
SSD	Solid State Drive
TPS	Transactions Per Second
UCLA	University of California, Los Angeles
VM	Virtual Machine

ABSTRACT

Enterprise blockchain is a decentralized system which has become critical to maintaining continuous operation and maximising value for private blockchain networks across industries. This study aims to design a distributed state replicated machine with multiple replicas for fault tolerance in enterprise blockchain. The proposed framework will be based on a threat and node model, with various replicas and an authenticated point-to-point channel that ensures the network's resilience if some of its nodes crash or fail and the ability to recover from this failure. It also presents fundamental concepts of blockchain and fault tolerance algorithms and a critical review of existing approaches to implementing and evaluating fault tolerance for enterprise blockchain technology. The study provides validation of the implemented framework by testing the network's ability and reliability to detect faults, restore crash nodes and re-adding them to the network.

The findings show that it takes around 0.114 seconds for the first node to join the network, while the worst-case scenario for any node to join the network is around 0.119 seconds. The result of the implemented framework also shows that it requires 2 seconds to detect an attempted crash, and after crashing or wiping out all the Nodes in the network, it takes an additional 13 seconds to reestablish a new node. After that, the restored Nodes are ready in 50 seconds. Based on these findings, it can be concluded that no matter how many crash nodes there are, the network cannot go down for longer than 50 seconds and all network defects can be detected in 2 seconds or less.

Keywords: Consensus Algorithm, Enterprise Blockchain, Fault Tolerance, Hyperledger, Replica Algorithm, State Machine Replication