

**AN EFFICIENT TECHNIQUE FOR OPTIMIZING TIMESTAMP ORDERING
SCHEDULER IN SYNCHRONIZATION**

BY

ABRAHAM IBOROMA

**Bsc Comp Sci. (UPH)
G2014/M.Sc/COMP/FT/500**

**DEPARTMENT OF COMPUTER SCIENCE
FACULTY OF SCIENCE
UNIVERSITY OF PORT HARCOURT, NIGERIA**

APRIL, 2018

**AN EFFICIENT TECHNIQUE FOR OPTIMIZING TIMESTAMP ORDERING
SCHEDULER IN SYNCHRONIZATION**

BY

ABRAHAM IBOROMA

**Bsc Comp Sci. (UPH)
G2014/M.Sc/COMP/FT/500**

A Dissertation submitted to the School of Graduate Studies in partial fulfilment of the requirements for the award of degree of Master of Science (M.Sc) in the Department of Computer Science, Faculty of Science, University of Port Harcourt.

APRIL, 2018

DECLARATION

I, ABRAHAM IBOROMA with Registration Number G2014/MS/COMP/FT/500 declare that this work on AN EFFICIENT TECHNIQUE FOR OPTIMIZING TIMESTAMP ORDERING SCHEDULER IN SYNCHRONIZATION was carried out by me, that it is my original work and it has not been submitted wholly or in part for the award of any degree in any institution.

Name of student: Abraham Iboroma

Signature/Date

Confirmation by Supervisors

Name of supervisor: Prof. E. O. Nwachukwu

Signature/Date.....

Name of supervisor: Dr. E.E Ogheneovo

Signature/Date.....

CERTIFICATION

**UNIVERSITY OF PORT HARCOURT
SCHOOL OF GRADUATE STUDIES**

**AN EFFICIENT TECHNIQUE FOR OPTIMIZING TIMESTAMP ORDERING
SCHEDULER IN SYNCHRONIZATION**

BY

ABRAHAM IBOROMA

**Bsc Comp Sci. (UPH)
G2014/MSC/COMP/FT/500**

The Board of Examiners certifies that this Dissertation is accepted in partial fulfilment of the requirement for the award of the Degree of Master of Science (M.Sc.) in Computer Science.

DESIGNATION	NAME	SIGNATURE	DATE
Supervisor (I)	Prof. E. O. Nwachukwu	-----	-----
Supervisor (II)	Dr. E.E. Ogheneovo	-----	-----
Head of Department	Dr. Laeticia N. Onyejebu	-----	-----
Chairman, Departmental Graduate Studies Committee	Prof. E.O. Nwachukwu	-----	-----
Dean, Faculty of Science	Prof. G.C. Obute	-----	-----
External Examiner	Prof. F.O. Ikpotokin	-----	-----
Dean, School of Graduate Studies Chairman, Board of Examiners	Prof. Anthonia. A. Okerengwo	-----	-----

DEDICATION

This work is dedicated to God Almighty for his loving kindness and protection over my life during the period of this work.

ACKNOWLEDGEMENTS

In completing this work, i have been fortunate to have help, support and encouragement from lots of people. I wish to express my sincere gratitude to them for their cooperation.

First, I would like to thank my supervisors Prof. E. O. Nwachukwu and Dr E.E Ogheneovo who guided me throughout the project tenure, provided me each and every details, references, and technical helps without which it was impossible to complete this project.

My appreciation could be incomplete if I fail to express my gratitude to my lectures Prof. Asagba Prince O, Dr. C. Ugwu, Dr Laeticia .N. Onyejegbu, Dr Friday Onuodu, Dr Eke Barthlomey O, Dr. C. I. Ejiofor For spending their time to impart knowledge into us, you're God's gift and blessing to students of this generation,

I am indebted to my well-wishers and friends Mr Zaara Kara, Mrs Nyingi Allison, Mrs Chikoadi Iboroma, Comrade Livinus K. Eli, Comrade Sunday Egbuta, Comrade Kailayo Lawrence and many other, who encouraged me in completion of the project.

Finally. I also wish to thank all guest faculties and non-teaching staffs for supporting me during this work. May the blessings of God Almighty be with you and your families.

ABSTRACT

This dissertation presents an efficient techniques for optimizing timestamp ordering scheduler in synchronization. We analyze the problems of concurrency control systems and developed an Optimistic Thomas Algorithm (OPTH) which is a hybrid of Optimistic Concurrency Control Algorithm and Thomas' Write Rule Timestamp Algorithm for optimizing the serialization of Read-Write/Write-Write (RW/WW) synchronization in distributed database management system (DDBMS). The proposed system consist of three phases: Read phase, Validation phase and Write phase. The proposed system fixed the problem of lost update, uncommitted data and long executing transactions. Object Oriented Analysis and Design methodology were adopted to analyse and design the system. The programming languages and tools used for the development of the system are C#, JavaScript, ASP.Net, Css, Visual Studio and Ms SQL Server. The results shows that the system is effective and provide an execution that has the same outcome as a serial execution. It reduces the rate transactions restart and improve transaction throughput in a distributed database management system. The integration of other pessimistic concurrent control algorithms and optimistic concurrent control algorithms should be used to address the anomalies of conflicting transactions in a centralized, distributed and mobile database system.

TABLE OF CONTENTS

Title page	ii
Declaration	iii
Certification	iv
Dedication	v
Acknowledgement	vi
Abstract	vii
Table of content	viii
List of figure	xi
List of table	xii

CHAPTER ONE: INTRODUCTION

1.1 Background to the study	1
1.2 Statement of the problem	4
1.3 Aim and Objectives of the Study	5
1.4 Significance of the Study	5
1.5 Scope of the Study	6
1.6 Limitations of the study	6

CHAPTER TWO: LITERATURE REVIEW

2.1 Overview of Timestamps and their Origin	7
2.2 Distributed Database Management System (DDBMS)	9
2.3 The Concept of Transactions and Schedules	11
2.3.1 Conflicting Operations	14
2.3.2 Schedule	15
2.3.3 Serial Schedule	18
2.3.4 Serializable Schedule	21

2.3.5 Problems in Concurrency Control	23
2.4 Concurrency Control in DBMS	27
2.4.1 Taxonomy of Concurrency Control Algorithm (CCA)	28
2.4.2 Pessimistic View	28
2.4.3 Optimistic View	29
2.5 Implementation of Concurrency Control Techniques	31
2.5.1 Two Phase Locking Algorithm (2PL)	31
2.5.2 Timestamp Ordering (TO)	35
2.5.3 Basic Timestamp Ordering Algorithm (BTSO)	36
2.5.4 Thomas' Write Rule for Timestamp Ordering Algorithm	37
2.5.5 Optimistic Concurrency Control Algorithm (OCCA)	38
2.5.6 Apply Optimistic Concurrency Control	40
2.13 Related works	41

CHAPTER THREE: MATERIALS AND METHODS

3.1 Analysis of existing system	45
3.1.1 Disadvantages of the Existing System	46
3.2 Analysis of the Proposed System	48
3.2.1 Advantages of the Proposed System	51
3.3 Methodology of the Proposed System	52
3.3.1 The High Level Model Diagram of the Proposed System	54
3.3.2 The Use Case Diagram of the Proposed System	56
3.3.3 The Activity Diagram of the Proposed System	59
3.3.4 Transaction Execution Model of Propose Algorithm	62
3.3.5 The Optimistic Thomas Algorithm (OPTH)	65
3.4 Output/Input Specification	70
3.4.1 Output/Input Design	70
3.5 Database Design	70

CHAPTER FOUR: RESULTS AND DISCUSSION

4.1 System Requirements	73
4.1.1 Hardware Requirements	73
4.1.2 Software Requirements	74
4.2 System Testing	74
4.2.1 Unit Testing	74
4.3 Choice of Programming Language Used	76
4.4 Presentation of Data	77
4.6 Result and Discussion	84

CHAPTER FIVE: SUMMARY, CONCLUSION AND RECOMMENDATIONS

5.1 Summary	86
5.2 Conclusions	87
5.3 Recommendation	87
5.4 Contribution to Knowledge	88
REFERENCES	89
Appendix A	93
Appendix B	116

LIST OF FIGURES	PAGE
2.1 DDBMS Architecture	10
2.2 Execution state of Transaction	13
2.3 A schedule of Transactions T1 and T2	17
2.4 The Sequence of Execution T1 and T2	17
2.5 A Schedule Involving two Transactions T1 and T2()	19
2.6 Lost Update Problem	24
2.7 Inconsistent retrievals problem	26
2.8 Basic Taxonomy of Concurrency Control	30

2.9 Two-phase Locking	33
2.10 Strict two-phase Locking	33
2.11 Phases of Optimistic Concurrent Control	39
3.1: Architecture of the Existing System Source	47
3.2: transactions at Various Stages	49
3.3: Architecture of the Proposed System	53
3.4: High Level Model Diagram	55
3.5: Use Case Diagram	58
3.6: Activity Diagram of the Proposed System	61
3.7: Illustration of the Propose System in an Online Banking System	68
3.8: Bank Read and Write Operations	69
4.1a: Customer Login Page	78
4.1b: Admin Login Page	78
4.2 (a): Teller initiate transaction	79
4.2 (b): Customer initiate transaction	79
4.3 (a): Teller Withdrawing From Customers Account	81
4.3 (b): Customer Transferring From Savings Account	81
4.4: Teller Complete Transaction	82
4.5: Validating Transaction	82
4.6: Conflict is Detected Transaction Terminate	83

LIST OF TABLES	PAGE
2.1: Compatibility Matrix for T_i and T_j	16
2.2: A serial execution of transactions T_1 and T_2	20
2.3: A serializable execution of transactions T_1 and T_2	22
3.1: Admin Table	71
3.2: Transaction Table	71
3.5: Customer Table	72