



**Shendi University**

**Faculty of Post Graduate Studies**



# **A Novel Model of Transaction's Consistency in NoSQL Databases**

A Thesis Submitted in fulfillment of the requirements for the degree of Doctor  
of Philosophy in Computer Science

**BY: Mohamed Siddig Hassan Abdelgadir**

**B.Sc. of Statistics & computer Science (2007)**

**M.Sc. of Computer Science (2015)**

**Shendi University, Sudan**

**Supervisor:**

**Dr. Mohamed Bakri Bashir**

**Co-Supervisor:**

**Prof. Ahmed Salah-Eldeen Abdullah**

**December 2023**

بِسْمِ اللّٰهِ الرَّحْمٰنِ الرَّحِیْمِ

{ فَتَعَالَى اللَّهُ الْمَلِكُ الْحَقُّ وَلَا تَعْجَلْ بِالْقُرْآنِ مِنْ قَبْلِ أَنْ

يُقْضَىٰ إِلَيْكَ وَحْيُهُ وَقُلْ رَبِّ زِدْنِي عِلْمًا }

صدق الله العظيم

سورة طه الاية ١١٤

## **Dedication**

To my beloved family and cherished friends, who have consistently demonstrated unwavering belief in my abilities and provided unwavering support throughout my journey, even during moments when I harbored doubts about my own capabilities, I express my deepest gratitude.

To my esteemed and revered parents, whose unwavering dedication to fostering a deep-rooted love for acquiring knowledge and an insatiable thirst for wisdom within me has left an indelible mark on my being, I extend my heartfelt appreciation. It is with utmost gratitude that I acknowledge your relentless efforts in propelling me towards becoming the best version of myself, and for that, I am eternally indebted.

To my beloved siblings, who have served as both a pillar of strength and an unwavering source of unwavering support throughout the various ebbs and flows of my life, I sincerely thank you. Your boundless love, unwavering support, and infectious laughter have been integral in shaping my journey, and for that, I am eternally grateful.

To my cherished friends, who have consistently provided a safe haven for me to express my thoughts, serve as my enthusiastic supporters, and entrusted me with their deepest secrets, I am forever indebted to you. Your presence has not only enriched my journey but has also made it immensely enjoyable, and for that, I extend my heartfelt appreciation.

It is with the utmost sincerity that I express my profound gratitude for the profound love and unwavering support that each and every one of you have bestowed upon me. Without your unwavering presence, I would not have been able to surmount the various challenges that accompanied this significant milestone in my life.

## **Acknowledgment**

I would like to express my deepest gratitude to my PhD supervisors, Dr. Mohamed Bakri Bashir, for their unwavering support and guidance throughout my study. Your expertise, insights, and encouragement have been invaluable. I am truly grateful to have had you as my supervisors.

I am grateful to my colleagues in the Shendi University for their support and collaboration. I have learned so much from working with you all. I would also like to thank the staff at the Shendi University and the faculty of computer science for their assistance.

I am also indebted to my friends and family for their love and support during my PhD journey. You have been there for me every step of the way. Thank you for believing in me.

This study would not have been possible without the support of all of the people mentioned above. I am truly grateful for your help and guidance.

## **Abstract**

NoSQL databases have recently experienced a surge in popularity, primarily driven by their capacity to effectively manage and process substantial amounts of data and their exceptional scalability capabilities. Nonetheless, it is important to acknowledge that these databases also bring about a set of trade-offs, particularly in relation to transaction consistency and overall performance. Consequently, it becomes imperative to have a comprehensive understanding of these trade-offs, as they play a pivotal role in making well-informed decisions when selecting a NoSQL database that aligns with the specific requirements of a given application. One of the primary challenges of NoSQL databases in supporting transaction consistency is the lack of full ACID support. While some NoSQL databases provide ACID guarantees for single operations, they may not offer the same level of consistency for multi-operation transactions. This can lead to inconsistencies when multiple operations are performed concurrently, impacting data integrity.

The main objective of this study is to propose a novel model to provide strong consistency guarantees and offer the best performance for NoSQL databases. Therefore, this study proposed a transaction consistency model (PMC) for achieving the objectives. The proposed model uses existing techniques such as Two-Phase locking for isolation of single and multiple transactions and snapshot isolation for comparing versioning data on the replication node.

The proposed model for transaction consistency (PMC) exhibited superior performance, scalability, and consistency when compared to alternative models. In terms of performance, the PMC model excelled by achieving higher throughput and lower latency than its counterparts. Moreover, the PMC model demonstrated

exceptional scalability, being capable of accommodating a larger number of nodes than the other model. This ability to scale effectively is of utmost importance in modern distributed systems, where handling a larger number of nodes is crucial. Lastly, the PMC model distinguished itself by successfully upholding a high level of consistency, even under conditions of high load, which further solidifies its merit. The achievement of such a consistently high level of consistency can be attributed to the well-thought-out design and implementation of the PMC model. It is undeniable that the PMC model's superior performance, scalability, and consistency make it ready to apply in the realm of transaction processing systems.

***Keywords: NoSQL, Transactions, Consistency, NoSQL Transaction Consistency***

## المستخلص

شهدت قواعد البيانات الغير علائقية مؤخرًا ارتفاعًا في شعبيتها، ويرجع ذلك في المقام الأول إلى قدرتها على إدارة ومعالجة كميات كبيرة من البيانات بشكل فعال وإمكاناتها الاستثنائية في قابلية التوسع. ومع ذلك، من المهم الاعتراف بأن قواعد البيانات هذه تؤدي أيضًا إلى مجموعة من المقايضات، خاصة فيما يتعلق باتساق المعاملات والأداء العام. وبالتالي، يصبح من الضروري أن يكون لديك فهم شامل لهذه المقايضات، لأنها تلعب دورًا محوريًا في اتخاذ قرارات مستنيرة عند اختيار قاعدة بيانات الغير علائقية التي تتوافق مع المتطلبات المحددة لتطبيق معين. أحد التحديات الأساسية لقواعد البيانات الغير علائقية في دعم اتساق المعاملات هو الافتقار إلى دعم الاتساق الكامل. في حين أن بعض قواعد البيانات الغير علائقية توفر ضمانات اتساق للعمليات الفردية، إلا أنها قد لا تقدم نفس المستوى من الاتساق للمعاملات متعددة العمليات. يمكن أن يؤدي هذا إلى حالات عدم اتساق عند تنفيذ عمليات متعددة بشكل متزامن، مما يؤثر على سلامة البيانات. الهدف الرئيسي من هذه الدراسة هو اقتراح نموذج جديد لتوفير ضمانات اتساق قوية وتقديم أداء لقواعد بيانات الغير علائقية ولذلك اقترحت هذه الدراسة نموذج اتساق المعاملات (PMC) لتحقيق الأهداف. النموذج المقترح المستخدم قد يحتوي على تقنية موجودة مثل القفل على مرحلتين لعزل المعاملات الفردية والمتعددة وعزل اللقطة لمقارنة بيانات الإصدار على عقدة النسخ المتماثل.

أظهر النموذج المقترح لاتساق المعاملات أداءً فائقًا وقابلية للتوسعة والاتساق عند مقارنته بالنماذج البديلة. من حيث الأداء، تفوق نموذج من خلال تحقيق إنتاجية أعلى وزمن وصول أقل من نظرائه. علاوة على ذلك، أظهر نموذج المقترح قابلية توسع استثنائية، حيث كان قادرًا على استيعاب عدد أكبر من العقد مقارنة بالنموذج الآخر. تعد هذه القدرة على التوسع بفعالية ذات أهمية قصوى في الأنظمة الموزعة الحديثة حيث يعد التعامل مع عدد أكبر من العقد أمرًا بالغ الأهمية. وأخيرًا، من المميزات للنموذج المقترح الحفاظ بنجاح على مستوى عالٍ من الاتساق، حتى في ظل ظروف الأحمال العالية، مما يعزز من تحقيق أهدافه.

## List of Contents

Contents	Page
<b>Dedication</b>	III
<b>Acknowledgment</b>	IV
<b>Abstract</b>	V
المستخلص	VII
<b>List of Contents</b>	VIII
<b>List of Tables</b>	XII
<b>List of Figures</b>	XIII
<b>List of Abbreviation</b>	XIV
<b>Chapter One</b>	1
<b>Introduction</b>	1
1.1 Background	1
1.2 Research Contributions	2
1.3 Research Objectives	4
1.4 Research Motivation	4
1.5 Research problems	6
1.6 Research Questions	6
1.6 Research outlines	9
<b>Chapter Two</b>	11
<b>Literature Review</b>	11
2.1 Background	11
2.2 NoSQL Models	11
2.2.1 Key-Value Data Model	12
2.2.2 Column-Family Data Model	14
2.2.3 Document Data Model	16
2.2.4 Graph Data Model	20
2.3 Advantages of NoSQL	22
2.4 NoSQL Data Partitioning	24
2.4.1 Scaling	26
2.4.2 Replication	27
2.4.3 Load Balancing	28
2.4.4 Garbage collection	28
2.4.5 Recovery and Identification of Failures	29
2.4.6 Transaction Management	29



<b>Contents</b>	<b>Page</b>
2.5 CAP Vs. BASE	30
2.6 The Data Consistency in NoSQL Databases	36
2.7 NoSQL Transactions	46
2.7.1 Understanding NoSQL Transactions	47
2.7.2 Challenges in NoSQL Transactions Consistency	47
2.7.3 Solutions to NoSQL Transactions Consistency	48
2.7.4 ACID Characteristics	49
2.7.5 BASE Transactions	52
2.7.6 ACID, CAP vs. BASE in NoSQL Databases	52
2.7.7 NoSQL Transactional Databases	54
2.7.8 Transaction as a services for NoSQL databases	54
2.7.9 Concurrency Control Techniques in NoSQL	55
2.7.9.1 Locking	55
2.7.9.2 Two Phase Locking Protocol	56
2.7.9.3 Protocol for Timestamp Ordering (TSO)	59
2.7.9.4 Optimistic Concurrency Control (OCC)	59
2.7.9.5 Multi-Version Concurrency Control	60
2.7.10 Transaction Isolation Levels	62
2.7.11 Conflicts Read and Write Transactions	65
2.7.12 Snapshot Isolation Protocol	68
2.7.13 Serializability	71
2.7.14 PACLEC	72
2.7.15 PAXOS	72
2.8 Previous Studies	74
<b>Chapter Three</b>	
<b>Methodology and Conceptual Design</b>	
3.1 Research Methodology	85
3.2 NoSQL Models Experiment	88
3.2.1 Pre-Experiment Tools	90
3.2.2 Pre-Experiment Overview	93
3.2.3 Pre-Experimental Steps	93
3.2.4 Pre-Experiment Benchmark	95
3.2.5 Pre-Experiment Workloads	96
3.2.6 Pre-Experimental Metrics	97

<b>Contents</b>	<b>Page</b>
3.2.7 Pre-Experimental setup	97
3.2.8 Pre-Experiment Results	98
3.3 Proposed Model of NoSQL Transactions' Consistency (PMC)	103
3.3.1 Model Design	103
3.3.1.1 Model Design Principles	103
3.3.1.2 Key Components	104
3.3.1.3: Consistency Levels	104
3.3.1.4 Conflict Resolution Mechanisms	105
3.3.1.5 Resource Management	105
3.3.1.6 Monitoring and Debugging	106
3.3.2 Transaction Protocol	107
3.3.3 Model Variables and Concepts	108
3.3.4 Mathematical Model	108
3.3.5 Model Architecture	110
<b>Chapter Four</b>	<b>116</b>
<b>Implementation</b>	<b>116</b>
4.1 Model Implementation	116
4.2 Experiment Design	126
4.2.1 Experiment Objective	126
4.2.2 Consistency models selection	126
4.2.3 NoSQL database selection	126
4.2.4 Experiment Dataset	129
4.2.5 Experiment Environment	129
4.2.6 Benchmarking	131
4.2.7 Performance measurement	132
<b>Chapter Five</b>	<b>133</b>
<b>Experiment Results</b>	<b>133</b>
Experiment results	133
<b>Chapter Six</b>	<b>143</b>
<b>Discussion</b>	<b>143</b>
6.1 Consistency requirements	148
6.2 NoSQL workload	150
<b>Chapter Seven</b>	<b>151</b>
<b>Conclusion</b>	<b>151</b>

	<b>Contents</b>	<b>Page</b>
7.1	Conclusion	151
7.2	Recommendations	152
	<b>References</b>	<b>154</b>
	<b>Appendies</b>	<b>168</b>

## List of Tables

<b>Table No.</b>	<b>Page</b>
2.1 Graph Model Structure	26
2.2 NoSQL Consistency Models	48
2.3 Consistency Models in NoSQL Databases	49
3.1 YCSB Workloads	91
3.2 Performance level of Eventual Consistency vs. Quorum	102
5.1 The transaction accuracy of consistency for each model	132
5.2 The results of the experiment	133
5.3 The results of read/write transactions of each model	134
5.4 The characteristic tests	138
5.5 The replication factors vs. consistency percentage	139
5.6 Failure Scenario	139
5.7 CPU Consumption	140
5.8 Memory Consumption	140
5.9 Network Bandwidth Consumption	141

## List of Figures

Figure No.	Page
2.1 shows the NoSQL Data-Models	20
2.2 shows the case studies of NoSQL Data Models	23
2.3 shows the types of graph model	27
2.4 shows the NoSQL Reasons	28
2.5 shows the CAP Characteristics' of distributed databases	38
2.6 Shows the Technology timeline for primary databases vs. accompanying systems	39
2.7 Shows the ACID characteristics	54
2.8 Shows The Transactional Life Cycle	56
2.9 Shows the Two-Phase Commit Locking Algorithms	67
3.1 shows the application of the research technique's process model.	85
3.2 YCSB Architecture	92
3.3 Terminal output of experimental loading phase	97
3.4 workload A – Throughputs	98
3.5 workload B – Throughputs	99
3.6 workload D – Throughputs	100
3.7: Proposed NoSQL Transaction Model	115
4.1: Single Transaction Algorithm	121
4.2: Multi-version Transaction Algorithm	122
4.3: Sequence diagram of a successful Transaction T1 read X and Transaction T2 update Y	122
5.1 shows The throughput of PMC vs. MVCC	135
5.2 shows The latency of PMC vs. MVCC	136
5.3 shows a comparing between PMC vs. MVCC in throughput and latency	137

## List of Abbreviations

Abbreviation	
<b>NoSQL</b>	Not Only SQL
<b>SQL</b>	Structured Query Language
<b>CAP</b>	Consistency, Availability, Partition Tolerance
<b>BASE</b>	Basically Available, Soft State, Eventual Consistency
<b>ACID</b>	Atomic, Consistency, Isolation, Durability
<b>PMC</b>	Proposed Model of Transaction's Consistency
<b>MVCC</b>	Multi-Version Concurrency Control
<b>YCSB</b>	Yahoo Cloud Services Benchmark
<b>2PCL</b>	Two-Phase Commit Locking Technique
<b>CC</b>	Concurrency Control
<b>SR</b>	Serializability
<b>PCC</b>	pessimistic concurrency control
<b>OCC</b>	optimistic concurrency control
<b>SI</b>	Snapshot Isolation
<b>RC</b>	Read Commit
<b>SPI</b>	Single Partition Interior
<b>MPI</b>	Multi Partition Initiator
<b>RWTX</b>	Read/Write Transactions