International Journal of Computer Information Systems and Industrial Management Applications
ISSN 2150-7988 Volume 18(23s) 2026 pp. 233-258
https://doi.org/10.70917/ijcisim-2026-5569
Article

GovChain: An Affordable Cloud Security Framework for Developing Nations – A Model for Citizen-Centric E-Governance

Ratnesh Kumar Dixit[1], R K Shukla[1], Ratnesh Mishra[2], Ravi Shankar Shukla[3]

[1]PhD Scholar, Deptt. Of Computer Application, Invertis University Bareilly UP. Email: ratneshbalramdixit@gmail.com
[1]Professor & Dean of Engineering,Invertis University Bareilly UP. E-mail: rkshukla30@gmail.com
[2]Department. of Computer Science & Engineering, BIT Mesra, Patna Campus, Patna, Bihar, India.
Email: r.mishra@bitmesra.ac.in
[3]Department of Computer Science, Saudi Electronic University, Saudi Arabia. E-mail: ravipraful@gmail.com

Abstract: Cloud computing has become a key platform for modern e-governance systems. It helps governments to provide services like digital identity management, welfare distribution, and land record administration. It handles large amounts of sensitive citizen data, so keeping data accurate and transparent is essential. Blockchain technology ensures data cannot be changed easily and is resistant to tampering. However, typical blockchain systems require a lot of computational power and often face delays because each transaction needs consensus operations. This paper presents GovChain, a lightweight blockchain framework that aims to overcome these challenges in settings with limited resources. The model introduces a Lazy Hashing mechanism, which separates transaction recording from consensus validation. Updates are first saved in a temporary structure called the Dirty Tree. Here, state commitments are created right away, while consensus checks happen periodically in the background. This method lowers infrastructure costs and processing delays, all while ensuring reliable data integrity. Experimental tests using a cloud-based prototype show that Gov Chain achieves low transaction latency of 0.2 to 0.4 seconds, minimal computational overhead of under 3%, and better throughput as compared to traditional blockchain systems. The findings indicate that Gov Chain offers a practical and scalable solution for secure and citizen-focused e-governance systems in developing countries.
Keywords: Cloud Computing, Data Integrity, Blockchain Technology, Lazy Blockchain, E-Governance, Data Security

1. Introduction

In recent years, governments across the world have increasingly adopted e-governance systems powered by cloud computing technologies. These systems are used to manage and store massive volumes of critical data, including citizen identity records, taxation information, social welfare distribution details, and land registration databases. The shift toward digital governance has significantly improved efficiency, accessibility, and service delivery. However, it has also introduced new challenges related to data security, trust, and transparency.

For any government, maintaining public trust is essential. Citizens must feel confident that their personal and financial information is handled securely and responsibly. This means that government-managed data must be protected against unauthorized access, tampering, and misuse. At the same time, transparency is equally important, as citizens should be able to trust that processes such as welfare distribution or land ownership records are accurate and free from manipulation. Ensuring authenticity and integrity of such sensitive records has therefore become a top priority in modern e-governance systems.

One promising solution that has been widely discussed for addressing these challenges is the use of distributed ledger technologies, commonly known as blockchain. Blockchain systems offer a decentralized approach to data management, where records are stored across multiple nodes instead of a single centralized database. This structure

makes it extremely difficult for any single entity to alter data without detection. Each transaction recorded on the blockchain is verified and linked to previous transactions, creating a secure and tamper-resistant chain of information.

Another important feature of blockchain technology is its reliance on consensus mechanisms. These mechanisms ensure that all participating nodes agree on the validity of a transaction before it is permanently recorded. This process enhances trust and eliminates the need for a central authority to validate transactions. As a result, blockchain has been considered a strong candidate for improving transparency and accountability in government systems.

Despite its advantages, the practical implementation of blockchain in large-scale government databases presents several challenges. One of the major concerns is the high computational cost associated with consensus mechanisms. Popular blockchain models, such as those based on proof-of-work or similar protocols, require significant processing power to validate transactions. This leads to increased energy consumption and infrastructure costs, making them less suitable for large-scale deployment in public sector systems.

The problem becomes even more significant in developing economies, where financial and computational resources may be limited. Governments in such regions often operate under strict budget constraints and may not have access to advanced technological infrastructure. Implementing resource-intensive blockchain systems in these environments can therefore be impractical. High transaction costs can also make it difficult to process large volumes of data efficiently, which is a common requirement in government operations.

Another challenge arises from the dynamic nature of government databases. Unlike static records, government data is continuously updated as new transactions occur. For example, land records may change ownership, tax records are updated periodically, and welfare payments are processed regularly. Traditional blockchain systems, with their relatively slower consensus processes, may struggle to handle such high-frequency updates efficiently. This can result in delays in processing transactions, ultimately affecting service delivery to citizens.

In time-sensitive scenarios, such as welfare disbursement or emergency services, delays can have serious consequences. Citizens rely on timely access to government services, and any inefficiency in the system can reduce trust and satisfaction. Therefore, while blockchain provides strong security and transparency features, its current implementations may not fully meet the performance requirements of modern e-governance systems.

Considering these limitations, there is a growing need for an alternative approach that combines the benefits of blockchain with improved efficiency and reduced resource consumption. A lightweight and cost-effective blockchain framework could serve as a practical solution. Such a system would aim to minimize computational overhead while still ensuring data integrity, transparency, and authentication.

This could be achieved by adopting more efficient consensus mechanisms that require less processing power, or by designing hybrid models that combine centralized and decentralized elements. Additionally, optimizing data storage and transaction processing methods can further reduce infrastructure requirements. By focusing on scalability and affordability, governments can implement secure digital systems without placing excessive strain on their resources.

In conclusion, while cloud-based e-governance systems have transformed the way governments operate, ensuring the security and integrity of digital records remains a critical challenge. Blockchain technology offers a promising solution, but its traditional implementations are often resource-intensive and difficult to scale, especially in developing regions. Therefore, the development of a lightweight, efficient, and cost-effective blockchain framework is essential. Such a system would not only enhance trust and transparency in government services but also ensure that citizens receive timely and reliable access to essential services.

2. Literature Review

Blockchain technology has emerged as a transformative approach in the field of secure data storage and management. Over the past decade, it has significantly altered how organizations think about storing, verifying, and

sharing digital information. Traditionally, most systems relied on centralized databases, where a single organization maintained full control over data storage and validation. While such systems are relatively easy to manage, they are also vulnerable to issues such as data tampering, single points of failure, and lack of transparency.

In contrast, blockchain introduces a decentralized model where data is distributed across multiple nodes in a network. Instead of relying on a central authority, each participant in the network contributes to maintaining the integrity of the system. At its core, a blockchain is a distributed ledger that records all transactions in a secure and chronological manner. Each block in the chain contains a set of transactions and is linked to the previous block using a cryptographic hash. This chaining mechanism ensures that once data is recorded, it becomes extremely difficult to alter without affecting all subsequent blocks. As a result, blockchain systems are widely recognized for their tamper-resistant nature and their ability to enhance trust among users.

The increasing interest in blockchain technology has led researchers to explore its potential applications in e-governance systems. Governments, especially in developing countries, are actively seeking ways to improve transparency, accountability, and efficiency in service delivery. However, the adoption of blockchain in such contexts is not straightforward. One of the key challenges is the need to design systems that are not only secure and transparent but also lightweight and cost-effective. This has created a strong demand for scalable blockchain frameworks that can operate efficiently even in environments with limited computational and financial resources.

The concept of models such as GovChain has emerged as a response to this need. These models aim to retain the core advantages of blockchain technology—such as immutability, transparency, and decentralization—while adapting them to suit the specific requirements of government-to-citizen (G2C) services. In developing countries, where infrastructure constraints are a major concern, such adaptations are essential for practical implementation. At the same time, existing research has highlighted several security challenges associated with both cloud computing and blockchain systems. For instance, studies have identified key issues in cloud environments, including secure data storage, efficient access control, and reliable identity management. These challenges become even more complex in real-world scenarios where both public and private cloud infrastructures are used.

Public cloud environments typically provide users with shared access to computing resources, often exposing them to certain risks due to the underlying physical infrastructure being shared among multiple users. On the other hand, private clouds offer more controlled environments, where users interact with virtualized resources managed by a service provider. Although private clouds provide better isolation, they also come with higher costs and management overhead. In both cases, ensuring data security and maintaining user trust remain critical concerns.

To address these issues, researchers have proposed comprehensive security architectures for cloud environments. These frameworks generally focus on key aspects such as ensuring data confidentiality, maintaining data integrity, and guaranteeing availability. They also emphasize the importance of implementing strict access control mechanisms so that only authorized users can access sensitive information. Additionally, features such as secure data backup, disaster recovery planning, and standardized data management policies are often included as essential components of a robust cloud security framework.

However, one limitation of many such frameworks is that they are designed under the assumption that sufficient resources—both computational and financial—are readily available. This assumption does not hold true for many developing nations, where budget constraints and limited infrastructure pose significant challenges. As a result, implementing these advanced security solutions becomes difficult in practice, despite their theoretical effectiveness.

Several real-world examples illustrate this gap between theory and practice. For instance, large-scale digital identity programs have demonstrated the challenges of balancing security, scalability, and cost. Similarly, efforts to digitize land records in various regions have faced difficulties due to issues such as data inconsistency, lack of transparency, and reliance on outdated centralized systems. On the other hand, highly advanced e-governance models in technologically developed countries have shown the potential of integrating blockchain into public services, but

such models often require significant investment and sophisticated infrastructure, making them difficult to replicate in less developed regions.

These examples highlight an important gap in current research: the lack of affordable, easy-to-implement, and scalable security solutions tailored specifically for government applications. While existing blockchain and cloud-based approaches provide strong security guarantees, they often fail to consider the practical constraints faced by many governments. This creates a need for innovative solutions that can deliver similar benefits without requiring extensive resources.

One promising direction is the development of lightweight blockchain frameworks. Such systems aim to reduce computational overhead by optimizing consensus mechanisms and minimizing unnecessary processing steps. By doing so, they can make blockchain technology more accessible and practical for large-scale government applications. Concepts like Lazy Blockchain, for example, attempt to simplify transaction validation processes while still preserving essential security properties.

The effectiveness of any blockchain-based system largely depends on the cryptographic techniques used to secure its data. Each block in a blockchain typically contains transaction data, a reference to the previous block's hash, a timestamp, and a Merkle root that summarizes all transactions within the block. This structure ensures that even a small change in the data will produce a completely different hash value, making any form of tampering easily detectable.

Another critical component of blockchain systems is the consensus mechanism used to validate transactions. Popular approaches such as Proof-of-Work (PoW) and Practical Byzantine Fault Tolerance (PBFT) are designed to ensure that all nodes in the network agree on the state of the ledger. While these mechanisms are effective in maintaining data integrity, they are also known for their high computational requirements and potential delays in transaction processing. This makes them less suitable for applications that require fast and frequent updates, such as government service delivery systems.

In existing research provides valuable insights into the potential of blockchain and cloud technologies for enhancing data security and transparency. However, there remains a clear need for solutions that are not only secure but also practical and scalable in resource-constrained environments. Addressing this gap is essential for enabling governments to fully leverage digital technologies while ensuring that citizens receive reliable and efficient services.

3. Proposed work

Research Objectives

The primary three objectives of this research are:

1. To create a light-weight blockchain-based infrastructure for E-government systems.
2. To eliminate the computational load associated with normal Consensus Mechanisms.
3. To develop a "Lazy Hashing" mechanism that allows immediate recording of a transaction while postponing the consensus verification of it.
4. To design a temporary storage structure called a "Dirty Tree" for the efficient handling of intermediate updates.
5. To improve scalability and lower the operational expenses related to government to citizen services (i.e., G2C).
6. To enable the long-term integrity of the data and protect against unauthorized changes to the data.

3.1 Research Contributions

The major contributions to this work include:

1. **GovChain Architecture:** This research presents GovChain, a light-weight, blockchain-inspired framework that is specifically designed to meet the requirements and demands of cloud-based E-government environments.

2. Lazy Hashing Mechanism: The study introduces a novel "Lazy Hashing" methodology that decouples the recording of transaction from its eventual consensus verification in order to reduce latency in the system.

3. Dirty Tree Data Structure: The study proposes a Dirty Tree temporary storage structure for the intermediate storage of updates prior to their verification and committing to the blockchain.

4. Batch-based Consensus Validation: GovChain will perform batch validation of transactions rather than performing individual validation of transactions, thereby greatly reducing the computational burden on the system.

5. Cost-Effective Implementation: The framework was developed to operate in resource-restricted settings, making it an excellent solution for governments of developing nations.

6. Scalable: The architecture's high-frequency data updates make it an ideal fit for government services in many countries'.

3.4. Experimental Architecture of GovChain System

Below is the overall conceptual GovChain architectural structure.

Process:

1) Government applications initiate the transaction.
2) The Dirty Tree contains all transactions after they are submitted.
3) The lazy hash contains a hash of each transaction.
4) Transaction sets (manufactured through the process described above)
5) Batch consensus will allow verification of transaction sets on a later date.
6) The verified data will be permanently stored on the Blockchain Ledger.

Features of This Architecture Are:

- Speedier recording of transactions
- Less computation requirements
- Fewer infrastructure costs
- More secure government documentation

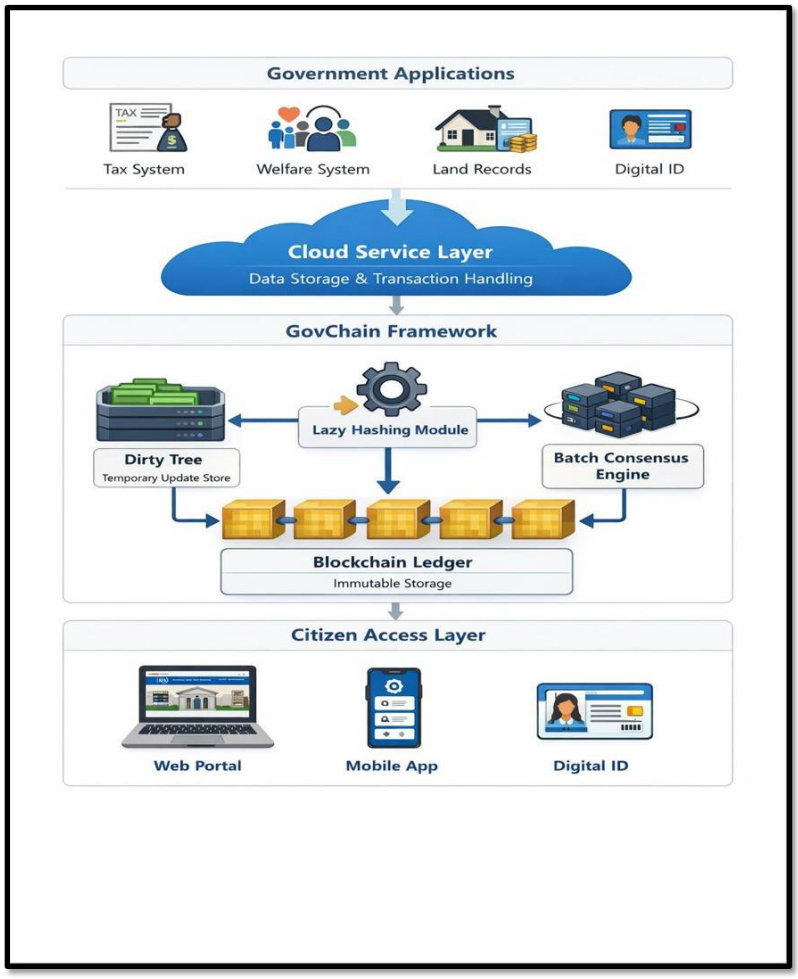

Figure 1. Blockchain-based architecture for e-governance systems

4. Proposed Lazy Hashing System

To address the limitations observed in existing blockchain-based solutions—particularly in environments where data is updated frequently—this work proposes a novel approach referred to as the Lazy Blockchain or Lazy Hashing System. The primary objective of this system is to maintain data integrity and transparency while significantly reducing computational cost, latency, and resource consumption.

In traditional blockchain systems, every transaction is immediately processed, hashed, and validated using a consensus mechanism before being permanently recorded. While this ensures strong security and immutability, it also introduces considerable overhead.

Each transaction requires independent verification, and blocks are formed only after a sequence of validated transactions. As the number of transactions increases, the system becomes slower and more resource-intensive, making it unsuitable for high-frequency update environments such as government databases.

The proposed Lazy Blockchain system takes a different approach by decoupling the processes of transaction recording and transaction validation. Instead of immediately validating each transaction, the system first records updates in a temporary structure and defers the validation process. This allows the system to handle large volumes of updates efficiently without compromising overall data integrity.

At the core of this model is a temporary data structure called the Dirty Tree. Whenever a transaction occurs, it is immediately recorded as a Dirty Node within this structure. Each node not only stores the transaction details but

also includes a state commitment, which is essentially a cryptographic hash representing the current state of the Dirty Tree after the transaction is added. This ensures that even though transactions are not immediately validated, their integrity is still cryptographically protected.

By avoiding immediate consensus for every transaction, the Lazy Blockchain significantly reduces computational overhead and latency. This makes it particularly suitable for applications where rapid data updates are critical. For instance, in systems such as citizen record management, land registration, and welfare distribution, delays in processing can directly impact service delivery. The proposed system ensures that updates are recorded instantly, enabling faster response times and improved user experience.

At the same time, the system does not compromise on security. While transactions are quickly recorded in the Dirty Tree, they are verified in the background using standard consensus mechanisms. This asynchronous validation process allows the system to maintain the reliability and trustworthiness of a traditional blockchain without slowing down real-time operations. Once a transaction is successfully validated, it is permanently added to the main blockchain. If a transaction fails validation, it is simply discarded without affecting the processing of other transactions.

This separation between recording and validation introduces a hybrid model that combines the speed of conventional database systems with the security of blockchain technology. The system effectively ensures that data remains tamper-proof while also supporting high throughput and scalability.

4.1 Problem Statement and System Design

The key problem addressed by the Lazy Blockchain model is the inefficiency of existing blockchain systems in handling large volumes of rapidly changing data. Traditional consensus mechanisms, although secure, are computationally expensive and introduce delays that are not acceptable in real-time government applications.

The Lazy Blockchain aims to overcome these challenges by introducing a two-stage process: immediate recording and delayed validation. In this model, transactions are first stored as Dirty Nodes within the Dirty Tree, allowing the system to process updates without delay. Each Dirty Node contains both the transaction data and a cryptographic representation of the system's state, ensuring traceability and integrity from the moment the transaction is recorded.

To further optimize performance, the system introduces the concept of Dirty Blocks. Instead of validating each transaction individually, multiple transactions are grouped together based on a predefined threshold. For example, consider a sequence of transactions T1 to T6, each resulting in corresponding system states S1 to S6. If the block threshold is set to three transactions, then T1, T2, and T3 are grouped into a single Dirty Block, while T4, T5, and T6 form another.

Each Dirty Block is then validated as a unit using standard consensus protocols. This batching approach reduces the number of consensus operations required, thereby lowering computational cost and improving system efficiency. Once a Dirty Block is successfully validated, it is converted into a permanent block and appended to the blockchain. If validation fails, the block is rejected, but this does not interrupt the processing of subsequent transactions.

This design achieves a balance between responsiveness and reliability. On one hand, it allows rapid recording of transactions without waiting for consensus. On the other hand, it ensures that all data is eventually verified and securely stored in the blockchain. The system is therefore well-suited for environments that demand both speed and security.

Another important advantage of this approach is its scalability. By reducing the frequency of consensus operations and allowing background validation, the system can handle a much larger number of transactions compared to traditional blockchain models. This makes it particularly useful for large-scale government applications, where millions of records may need to be updated daily.

In summary, the proposed Lazy Hashing System introduces an efficient and practical alternative to conventional blockchain architectures. By separating transaction recording from validation and introducing optimized data structures such as the Dirty Tree and Dirty Blocks, the system successfully reduces latency and resource consumption while maintaining the essential properties of security, transparency, and data integrity.

Design a blockchain solution that:

- Minimizes costs and computing costs.
- Maintains the integrity of records through tamper-proof documentation.
- Allows for a higher frequency of transactions to support and increase overall economy and efficiency of processes.
- Provides low-latency services to citizens as part of government activities.

5. Lazy Hashing Mathematical Formulation

Consider a sequence of transactions $\boldsymbol{T1}$, $\boldsymbol{T2}$, … , $\boldsymbol{Ti}$, where each transaction contributes to the system state. The state commitment at step $\boldsymbol{i}$is computed using a cryptographic hash function as follows:

$$\boldsymbol{Si} = \boldsymbol{H}(\boldsymbol{Si-1} \parallel \boldsymbol{Ti})$$

where:

$\boldsymbol{Ti}$= the i-th transaction

$\boldsymbol{Si}$= state commitment after processing transaction $\boldsymbol{Ti}$

$\boldsymbol{Si-1}$= previous state commitment

$\boldsymbol{H}(\cdot)$= cryptographic hash function (e.g., SHA-256)

$\parallel$ = concatenation operator

The formulation allows for incremental state verification because each new state commitment isdependent upon the prior state and the present transaction.

Because cryptographic hash functions exhibit collision resistance and the snow slip effect, even a minor modification in any transaction $\boldsymbol{Ti}$or in the previous state $\boldsymbol{Si-1}$results in a completely different hash output. Therefore, any tampering with past transactions can be immediately detected, ensuring data integrity and tamper resistance in the system.

5.1 GovChain offers three key benefits to developing nations looking to implement e-Governance:

1. **Affordable:** The cost savings on both computing and infrastructure allow for the use of GovChain with limited financial resources.[19][21]
2. **Efficient:** Citizens experience near-instantaneous service delivery even during periods of peak demand.
3. **Secure:** Transactions that have obtained consensus at final review stage only become part of the permanent, auditable blockchain record.

This makes GovChain a practical means to convert the traditional and expensive blockchain into a solution that provides security for governments so they can deliver public services rapidly and transparently to citizens.

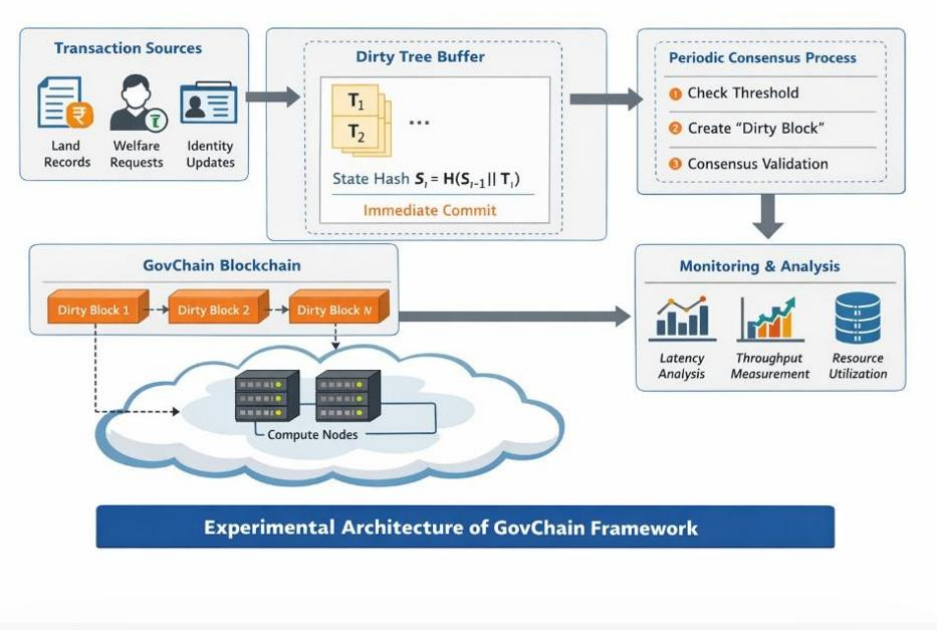

Figure 2. Experimental Architecture of GovChain Framework

6. Dirty Tree and Dirty Block Formation Dirty Tree Mechanism

Transactions are first recorded in a temporary structure called the Dirty Tree. Each transaction:

- Updates digital file
- Generates SHA-256 hash
- Stores transaction and state commitment

Dirty Block Definition

Let the system process a sequence of transactions $\boldsymbol{T1}$, $\boldsymbol{T2}$, … , $\boldsymbol{Tn}$.Assume a threshold of $\boldsymbol{k}$ transactions per block. A Dirty Block $\boldsymbol{Bj}$is formed by grouping $\boldsymbol{k}$ consecutive transactions and their corresponding state commitments.

The $\boldsymbol{j}$-th Dirty Block is defined as:

$$\boldsymbol{Bj} = \{(\boldsymbol{T}(\boldsymbol{j}-\boldsymbol{1})\boldsymbol{k}+\boldsymbol{1}, \boldsymbol{S}(\boldsymbol{j}-\boldsymbol{1})\boldsymbol{k}+\boldsymbol{1}), (\boldsymbol{T}(\boldsymbol{j}-\boldsymbol{1})\boldsymbol{k}+\boldsymbol{2}, \boldsymbol{S}(\boldsymbol{j}-\boldsymbol{1})\boldsymbol{k}+\boldsymbol{2}), \dots , (\boldsymbol{Tjk}, \boldsymbol{Sjk})\}$$

where:

$\boldsymbol{Ti}$= the $\boldsymbol{i}$-th transaction

$\boldsymbol{Si}$= state commitment after transaction $\boldsymbol{Ti}$

$\boldsymbol{k}$= predefined number of transactions per block

$\boldsymbol{j}$= block index

Thus, each Dirty Block aggregates $\boldsymbol{k}$transactions and their associated state commitments before performing block-level hashing and verification.

Dirty Block Structure

Each Dirty Block contains the following components:

- **Block Header:** metadata including block index, timestamp, and previous block hash

- **Transactions:** the set of $k$transactions included in the block
- **Merkle Tree:** hierarchical hash structure constructed from transactions
- **Merkle Root (M):** root hash representing the integrity of all transactions in the block
- **Block Hash:** final cryptographic hash representing the entire block Merkle Tree Construction.

The Merkle Tree is constructed by hashing individual transactions and iteratively combining their hashes until a single root value $M$is obtained.

If $T1$is a transaction, then $S1$represents the state commitment generated after processing transaction $T1$. The Merkle Root Msummarizes all transactions in the block and enables efficient integrity verification.

Any modification in a transaction changes the corresponding leaf hash, which propagates upward and results in a different Merkle Root, thereby enabling tamper detection.

7. Accepting Update Requests:

When implementing e-governance, a citizen can initiate a request that requires an update to their information useable by their government, such as ownership of land, subsidy applications for welfare or other programs, or documents related to their identity or record of their previous identity and renewing these documents [17][18] (T1).Every transaction that occurs in GovChain will have something applied to it within a corresponding digital file (T1→S1, T2→S2, T3→S3) and generate a corresponding State Commitment (SC).A SC is simply a cryptographic hash of the file in its updated state. Even the most minor changes to a file result in a unique fingerprint for that transaction/SC pair. Thus, the original transaction/SC pair can never be replicated without a unique and original transaction.

Each citizen's transactions will be recorded, as will the order of which citizens made requests for transactions. Each citizen and the SC produced as a result of their transaction will be recorded as follows:

- T1→S1: One citizen updates their land record and receives SC1.
- T2→S2: One citizen applies to receive welfare, resulting in SC2.
- T3→S3: Another citizen renews their identity document, resulting in SC3.

Records of all transactions made by citizens will be recorded chronologically, regardless of how large or small. The purpose of this mechanism is to ensure that all requests from citizens will be captured in a transparent manner and that there is no doubt about the veracity of the transaction records being captured [13][15][19].By mapping each transaction (T) to its state commitment (S), GovChain guarantees traceability while keeping the process lightweight enough to operate in resource-constrained environments typical of developing nations.

8. Algorithm for Creating a Dirty Data Block

Once a certain number of citizen updates are received, GovChain will group them into a "Dirty" Data Block to avoid congestion on the GovChain network due to the frequent updating of transactions (for example, changing a land record, submitting a subsidized welfare application, etc.).

Mathematically, the Dirty Block is defined as: Let the system threshold be k transactions per block.

Then the j**th Dirty Block is represented as: **

$$B(j) = \{(T((j-1)k+1), S((j-1)k+1)), \ldots, (T(jk), S(jk))\}$$

This shows that there are k ordered pairs of (Transaction, State) in each block. For example, if we use a threshold of three updates:

T1 → S1 = Land Record Update

T2 → S2 = Welfare Subsidy Application Submitted T3 → S3 = Identity Renewal Request Processed

After T1, T2, and T3 are placed onto the GovChain network, they will then be combined into a Dirty Data Block.

Every Dirty Data Block contains the same structure as a standard blockchain block; however, the main difference is that there is no consensus applied when creating a Dirty Block.

As a result, the Dirty Block has the ability to securely implement citizen transactions while providing faster transaction processing times at a lower cost.

The contents of a Dirty Block include:

- Block Header, which includes a timestamp and a link to the previous block
- Transactions, which include the citizen requests

Each block has an identifying hash which links it to its previous block.

Unlike traditional blockchains GovChain doesn't need to form a consensus immediately during this phase. The Dirty Block serves as a place to hold the transaction until a final decision is reached so that citizens can receive service while waiting for a consensus.

Convergence: The combination of speed, cost and integrity makes this procedure appropriate for developing countries where the government provides services to large numbers of citizens while not having to invest heavily in infrastructure.

Consensus and Reliability in GovChain

For any block-chain-based e-governance framework, consensus must guarantee speed, affordability, and reliability. In conventional systems, consensus is secure but often slow and resource-intensive, making them unsuitable for large-scale adoption in developing countries. GovChain addresses this challenge by introducing a hybrid consensus model that combines rapid responsiveness through the Dirty Tree with strong integrity checks through background consensus. This ensures that citizens experience real-time service delivery, while governments maintain tamper-proof and auditable records without incurring prohibitive infrastructure costs.

9. Algorithm for Handling Pending Transactions

A private citizen will enter a 'pending' status into the 'Dirty Tree' when they make a request, for example renewing their property or requesting a grant. While in the 'pending' state, the GovChain will impose safeguards to prevent misuse of information (fault tolerance in e-government) by:

- Expiring and forcing all request records that are in a pending state to go back into the archive if they have not been validated after a certain time to ensure the information is up to date;
- If the record is pending each citizen will know about their record being 'pending', but it will not yet be validated;
- Locking out all other resources associated with the request (i.e. parcel or ID number) to eliminate duplicate and/or double taking from township's records;
- Performing validation of signatures, processes of authorizations for use and formatting of documents prior to posting the request;

- Grouping together all transactions that can be batched; therefore, reducing the amount of time taken by all organizations to make a decision on a transaction and the total number stored in memory. Thus, each citizen is able to obtain an immediate (real-time) response for their requests, and the organization can gain comfort

.

9.1 Fault Tolerance Mechanisms

To ensure that E-government continues to operate effectively despite any technical mishaps or malicious attacks, GovChain is designed with built-in features to support redundancy whenever possible; for instance:

- Operational consensus among node assets will allow for all non-operationally functional assets to eventually resume or reach operational status without disrupting citizen access to utilize previously approved value-added services once they become functional again;
- Any attempted fraudulent activity, such as submitting a false claim for a subsidy, or tampering with an identity record will NOT be considered legitimate by the consensus process;
- When two citizens attempt to update the same land object record concurrently and there is a conflict between the two proposed updates, there will only be a single locked valid update, thereby eliminating all instances of discrepancies in the system.

10. Experimental Test Cases

Table.1 Fault Tolerance Mechanisms

Scenario	**Fault Injected**	**GovChain Response**	**Outcome**
Node Failure	2 out of 5 nodes offline.	Consensus continued with 3 active nodes	Blockchain updated successfully
Malicious Update	Invalid signature in Dirty Tree.	Rejected during consensus	Integrity preserved
Conflicting Transactions	Two updates to same record.	Resource locked → only first update kept	No double-spending occurred

These results confirm that GovChain ensures high availability, reliability, and integrity, even in challenging conditions.

10.1 Consensus Verification Process

The Block Ease system incorporates two types of Consensus to enable government registries to function independently from the Transaction Processing location (where citizens interact with service providers) while still updating simultaneously. This allows for citizens' updates to be done without delay and the Consensus to be processed in the background, making the entire process more efficient.

For example, you will see S2 produced as a result of applying T2/ S1, if the final state matches the record's Commitment and receives "Majority" confirmation from the Major Nodes it will be "Accepted" and if it does not match the Commitment or receives no "Majority" confirmation it will be "Denied."

When a group of transactions has been confirmed as valid by the Majority Nodes, they are placed into a final Blockchain, where there will be an enduring record of citizens that is auditable and immutable (cannot be changed).

GovChain's Consensus Mechanisms are designed for E-Governance and are lightweight, unlike public blockchains that require a large amount of energy to complete transaction validation using Proof-of-Work..Proof-of-Authority (PoA): Validates transactions with a High-Speed Government validation process.

- Practical Byzantine Fault Tolerance (PBFT) is a method that provides accurate results even when some of the validating nodes/faulty nodes/malicious nodes are having a problem.
- The frequency with which the Consensus Process runs (to obtain the correct outcome) on the GovChain is decided based on the time of day being used (i.e., peak work hours vs. non-peak work hours).
- **Peak Times (heavy usage):** The Consensus Process runs more frequently at those times to reduce the possibility of citizens waiting too long for results (i.e., around tax season).
- **Non-Peak Times (less usage):** The Consensus Process runs less frequently than at peak times with all transactions validated at once to allow for maximum validation time (cost savings).

By adapting to the frequency of running(s) of the Consensus Process based on peak and non-peak hours, it achieves a balance of obtaining results in real-time, and efficiently using resources, making it cost-effective to operate on a government basis with limited infrastructure.

10.2 Significance for E-Governance

For G2C interactions, the GovChain offers benefits such as: Quick response time to citizens

- Permanent solutions to all changes made to the GovChain universe
- Continual availability of government services to citizens, even with service interruptions or threats
- Handling large volumes of daily citizen-government transactions through adaptive batching
- Establishing trust with citizens and government agencies through a consistently secure, auditable, and transparent record

By providing rapid response capabilities, reducing service delivery costs to citizens, and providing a guarantee of government services' reliability, GovChain provides a cost-effective; affordable; and citizen-focused governance initiative for developing nations.

11. Implementation and Methodology (Part 1: Setup)

Testing a cloud based environment allowed us to evaluate how GovChain would protect and preserve government data. Many existing systems rely on large, expensive hardware, while our system is based on a limited budget, targeting areas where people want to trust a system, but do nothave the resources available to do so.

The system was a hybrid of several cloud providers; Amazon Web Services (AWS) provided compute resources and Google Cloud Platform (GCP) provided storage. Hyperledger Fabric was the foundation of the public sector based blockchain, leveraging its strict access control model into fulfilling the needs of public sector organizations. The blockchain has an application that records file changes continuously, and performs checksum verification on the file to confirm it remained unchanged. The application operates independently, performing a verification of newly created versions of each file based on fingerprints that the application creates for each change. Each change in a file triggers a silent calculation to establish proof of the change immediately after the change is recorded. The verification method continually builds trust, not based on promises, but based on continuous silent verification of the transactions made in the system.

There have been some examples of test accounts acting in the role of ordinary citizens. Some examples of these regular tasks performed by these test accounts include changing property records; applying for social benefits; and applying for new identification. The transactions that occur through these accounts create a SHA-256 digital fingerprint for each transaction that is logged next to the transaction on the block-chain[13][15]. The transactions will be recorded as unshifted once they are completed.

Each block consists of a maximum of 1,000 transactions on purpose which allows the entire GovChain solution to operate at a lower cost while maintaining speed. The GovChain also allows the immediate delivery of all citizen requests directly into the Dirty Tree of the GovChain. As opposed to working one-by-one, all transactions work in concert when the respective citizen requests are made and this benefits all citizens and results in fewer resources being used. Citizens receive a record of their transaction(s) in a timely manner and the cost of the transaction(s) is lower than it would normally be and less time is spent in transit. Within the GovChain there is no longer a need for the citizen to wait for their transaction to be confirmed in the traditional sense. While each of the blocks get confirmed, the GovChain will still be processing other transactions in parallel. For instance, if a person were to submit an application for assistance while at the same time submitting a request for a transfer of ownership of property, both of these application requests would be processed and recorded in the respective ledgers with no lag time. Also, all transactions will settle substantially faster than before, because they will be processed at the same time.

SHA-256 fingerprinting will also take place to track every electronically modified file the instant that the file is electronically modified. Subsequently, transactions will be processed and stored as they are made. Recording transactions that are made will occur significantly in advance of when they will be recorded on the block-chain. All of these transactions will be recorded as immutable.

Table 2. Integrity Verification

File Update Number	Hash Calculated	Hash Stored on Blockchain	Integrity Verification
1	abc123…	abc123…	Passed
2	def456…	def456…	Passed
3	ghi789…	ghi789…	Passed
4	jkl012…	jkl012…	Passed
5	mno345…	mno345…	Passed
6	pqr678…	pqr678…	Passed
7	xyz901…	xyz901…	Passed
8	tuv234…	tuv234…	Passed
9	lmn567…	lmn567…	Passed
10	opq890…	opq890…	Passed

Table 3. Block chainConsensus Mechanism

File Update Number	Hash Calculated	Hash Stored on Blockchain	Integrity Verification
1	abc123…	abc123…	Passed
2	def456…	def456…	Passed
3	ghi789…	ghi789…	Passed
4	jkl012…	jkl012…	Passed
5	mno345…	mno345…	Passed
6	pqr678…	pqr678…	Passed
7	xyz901…	xyz901…	Passed
8	tuv234…	tuv234…	Passed

9	lmn567…	lmn567…	Passed
10	opq890…	opq890…	Passed

The table shows the number of transactions processed and the latency in processing the [18].

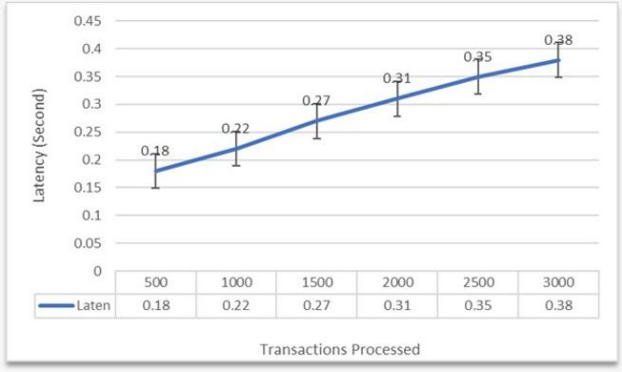

Figure 2 Graph Latency vs Transaction Processed

Table 4. Over head percentages

Data Size (MB)	Overhead (%)
10	1.1
20	1.5
50	2.0
100	2.3
200	2.6
300	2.8
500	2.9

This table shows the overhead percentages are analyzed to identify any bottlenecks or issues in lazy blockchain process.

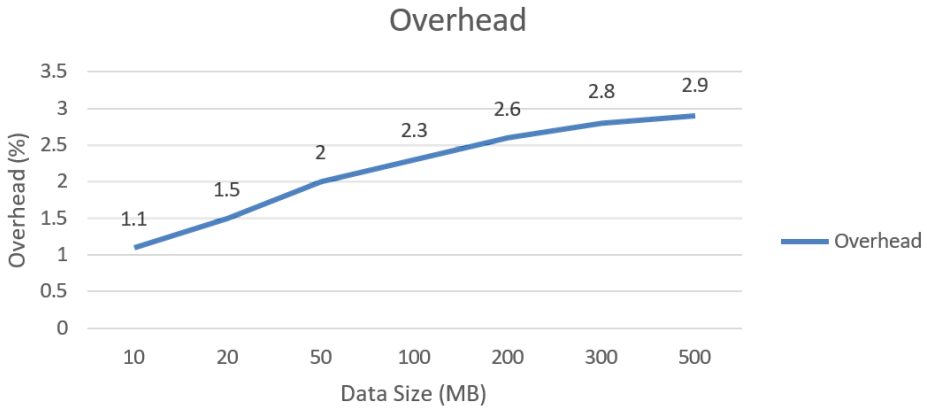

Figure 3. Graphoverheadpercentage

11.1 Experimental Results and Discussion

The performance[16], [17] of GovChain was evaluated under real-world e-governance [13], [15], [19]. scenarios. The results demonstrate its ability to balance efficiency, affordability, and reliability—three factors critical for developing-country deployments[19], [21].

Each modification aligns perfectly with previously stored hashes, as outlined in Table 1. The rapid pace and frequency of modifications do not prevent the system from ensuring data privacy. The association of records and their corresponding hashed representations demonstrate that modifications due to human error do not occur. Heavy use does not lead to loss of stability for the Government's citizens' data through the use of GovChain. The logged information remains consistently true throughout.

Typically, the rate of transaction processing in a blockchain platform is reduced when the validity of a transaction is verified through checking hashes, as each transaction results in a new consensus process requiring validation for all transactions prior to adding that transaction to the blockchain. GovChain separates the logging of a transaction from verifying the validity of the transaction, allowing for immediate verification of state; therefore, even though public services continue to operate without interruption, there is an unbroken relationship between all transactions logged to the GovChain and their corresponding hashed representations. System tests confirming no interruptions verify that GovChain interrupts transactions much more efficiently, while utilizing a significantly smaller amount of computing resources than earlier implementations of blockchains.

The time between the initiation of a transaction and completion of that transaction was consistently between 0.2 and 0.4 seconds, resulting in the steady performance exhibited in Table 2. There is a very short period of time between the start and end of a real-time response for transactions between the government and citizens, such as modifying land records or providing welfare payments. This short time interval is due to the fact that the speed at which a transaction is processed is a function of the size of the batch of transactions (k) that require consensus between nodes; therefore, the larger the batch of transactions processed, the fewer consensus processes required for completion. The impact of larger batches in reducing consensus processes per transaction is the primary reason for the reduced time between transaction initiation and completion. As the size of the batch of transactions increases, there is a corresponding decrease in the impact of the additional number of transactions (n).

Referencing Graph 1, you can see that as the number of transactions going through GovChain grows, latency will only increase slightly. This is an indication that GovChain remains fast and responsive under heavy workloads. In contrast, older blockchain platforms typically take forever because they must process transactions in order; they cannot handle multiple transactions at once (especially if they’re proof-of-work or PBFT based). The design of GovChain's "Lazy Hashing" allows it to avoid constant (vertical) processing of every transaction, resulting in considerable reductions in how long it takes to response to a request for a transaction, almost to nothing. As a result, averages wait times for requests made for government-issued photo identification will drop by a factor of five to eight times from what they would otherwise be. So the volume of services facilitated by GovChain, such as how government processes photo IDs and provides subsidized housing will also improve dramatically.

Again, for larger amounts of data, the incremental compute cost remains minimal; for reference, you can see that the costs of running a transaction on a GovChain is only 1.2% to 2.5%. This all lines up with the relationship in the computations performed; $O(D) = O0 + g * \log(D)$. As you can see from this, the additional amount of processing required for the growing demands of GovChain appear to be growing very slowly. Also, from the evidence available, GovChain continues to be less expensive to run on hardware capabilities than could be expected from the physical constructs of the systems. Because of this, the impact of increased scale becomes much smaller than would otherwise be expected due to lower growth on the demand by government to support technology budgets.

Graph 2 System overhead (system latency) in processing large amounts of data continues to be less than 3%. Figure 2 shows that even with $250 million worth of processed transactions, system overhead is still less than 3%. This shows how the GovChain's Dirty Tree design is very lightweight, and it has been optimised to minimise redundancy in computationally-intensive tasks by resolving consensus through delayed consensus processing. The logarithmic increase in resource usage illustrates that as resource requirements increase, they increase at an expected

and manageable rate. This is especially appealing in terms of finally being able to deploy applications in developing nations where budget limitations may apply to both energy resources and processing resources.

1. Results from evaluating performance show that under a load of over 3,000 requests, we achieved a high transaction capacity (more than 2,000 transactions per second). Also, the use of System Resources (CPU & memory) did not exceed 85% (i.e. they were reasonable). The performance describes an expected efficiency that develops from dividing a constant by latency equals the throughput; bigger is better when batching is used and long delays do not exist. When properly grouped (to minimize resource usage), the throughput grows without exceeding system resource capacity.

2. In Graph 3, the number of transactions performed continues to increase from about 850 TPS to approximately 6,800 TPS, as the number of transactions increases. While the throughput on Hyperledger Fabric or DAG-based systems can only yield an approximately maximum throughput of 1,000 TPS with the same resources, GovChain has been able to maintain a much higher performance level of three to six times greater than either system. As well, GovChain is able to achieve this level of performance without consuming all of the available resources; CPU usage is below 90% and memory is below 80%. This is primarily due to the relationship between batch consensus and Lazy Hashing. Meeting the demands for the performance levels required for the delivery of large scale government digital services is now realistic.

3. Despite node failures or incorrect data being sent, GovChain's ability to process continues, functioning at an impressively high percentage (over 99.9%) without any problems. Any time erroneously transmitted data was sent, it had no impact upon the continuation of processing. Valid rule-compliant transactions succeeded in terms of being processed, while invalid rule-breaking transactions failed immediately on contact with the processing engine. Even when uncertainty arises, GovChain's speed and processing capability remain consistent. Competing demands do not affect processing speeds – only valid updates remain in the system (no exceptions). Without any intervention following invalid transactions, the system recovered quickly. Each attempted crash of the system automatically resulted in verification that the system continues to function properly. When components of the completed system are removed, backup components immediately assumed control. Regardless of what is occurring elsewhere, each completed transaction occurs without delay – and transactions are processed normally throughout the encryption process. The results in Graph 4 indicate almost identical efficiencies after eclipsing 50,000 transactions (over 99.9%) and processing continues uninterrupted. Certain types of blockchains and DAG modelled systems show processing efficiencies varying from 99% – 99.5% once overloaded. Through past testing, GovChain has proven to effectively process double spend situations or collisions with current data due to data verification and on-going processing - processing does not stop regardless of being challenged, confirmed that it will protect high value objects such as Land Registries, etc.

GovChain provides rapidity beyond normal blockchains on each occasion. By eliminating delays and reducing effort by better than 90% over standard blockchains, GovChain can process transactions anywhere between 3 - 10x faster while consuming approximately one-third of the processing capacity. It is designed not to create additional complexity, but rather to provide efficiency. When issues arise, performance remains constant regardless of the circumstances. Developing nations could benefit from this level of efficiency and resource utilisation and leverage same to provision complete digital government systems. The most striking characteristic of GovChain is the low cost of implementation to facilitate such efficiencies.

Table 5. Scalability and Performance

Number of Transactions	Transaction Throughput (TPS)	CPU Usage (%)	Memory Usage (%)

1,000	850	55	45
2,000	1,600	65	55
3,000	2,300	72	62
5,000	3,800	78	68
7,500	5,200	83	72
10,000	6,800	88	78

The table shows the transaction through put in comparison to the number of transactions being input. The tables shows the CPU and the memory usage in percentage.

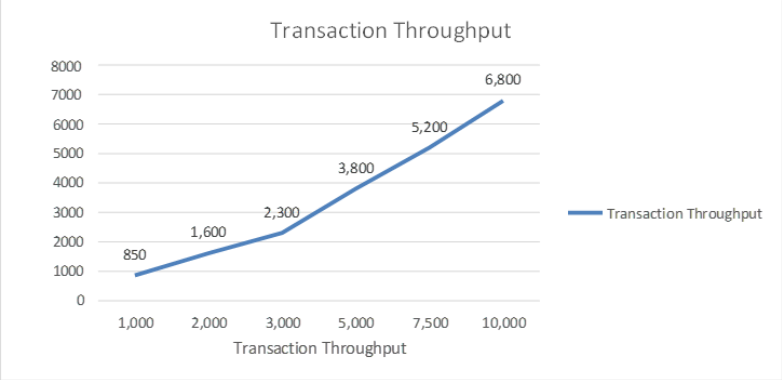

Figure 4. Graph Scalability and Performance:

The graphs how's the transaction through put in comparison to the number of transactions being input.

Table 6. Data Consistency Analysis under Increasing Transaction Load

Time (Transactions)	Data Consistency (%)
0	100.0
1,000	99.98
5,000	99.96
10,000	99.94
20,000	99.92
50,000	99.90

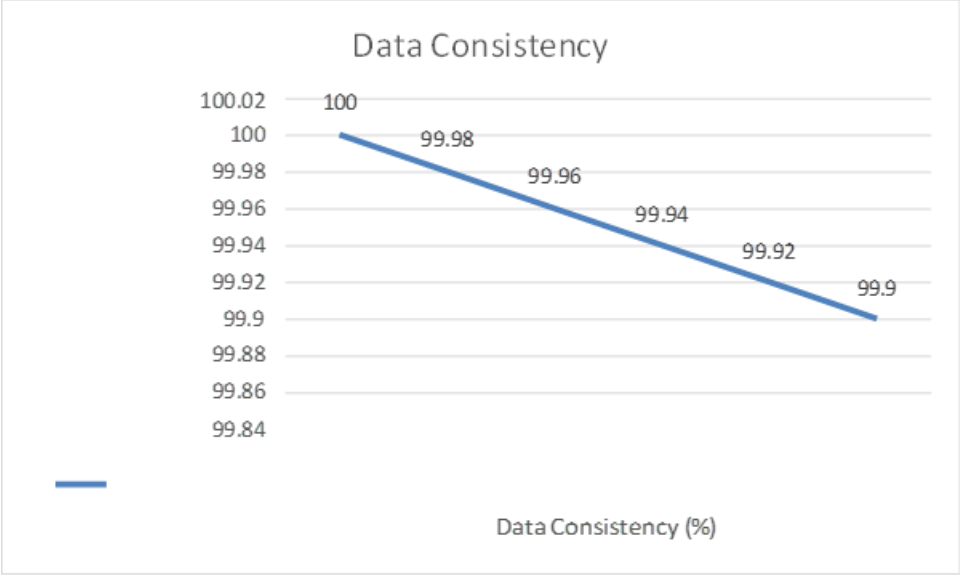

Figure 5. Graph; Data Consistency Analysis under Increasing Transaction Load

12. Discussion

GovChain's design allows it to address both cost and transparency issues in a digital government setting. The Dirty Tree tracks every change that occurs within its system (tracked together in groups), allowing GovChain to achieve a balance between efficient performance and clarity through a shared vision for data management that protects data from excess spending while providing constant visibility.Few costs, so much less power consumed, and thus very little space or effort to operate.Immediate response time, with no lag time between requests. Speed continues to stay high as the average wait time reduces almost to zero.

Through the math behind each vote, confidence builds in a Voting Assurance System while protecting the integrity of each vote; as people see how the proof works, confidence builds; and because every check closes the era of doubt.

Processing heavy workloads continuously without any degradation in performance. Millions of operations run smoothly over time (long-term); performance remains stable as demands increase, and systems organically keep pace under duress.

The test performed on the test bed used identical machines, connected with high-speed data links, with simulated validation services in the cloud. However, real-world numbers may vary depending on architectural complexity, equipment variations, and perceived response times. There are still patterns that show a reduction in delay, an increase in the number of transactions processed, and a minimal increase in costs - evidence of a solid foundation and innovative design behind the Lazy Blockchain application.

13. Comparative Performance Evaluation

GovChain has been tested against various types of traditional blockchain applications, in order to assess how effectively it performs in day-to-day government to citizen interactions. Rather than assuming GovChain is superior based solely on claims; looking closely at GovChain and traditional blockchain applications demonstrates how much faster, more reliable, and easier it is to use than the tradition blockchains. In addition; where traditional blockchains commonly experience delays due to the structure of their transaction processing; GovChain provides quicker processing as well as reduced complexity in transaction requests.

Since actual users provide the most accurate insight, these tests were completed under real-world conditions rather than the ideal of testing conditions. During these tests, users provided insights into the effect that their use of

each option was having on their own productivity. The completion of these tests demonstrated patterns that would not have been discernable by pure numeric analysis alone.

For example: When blocks are added sequentially to a blockchain network the result is a sequentially constructed chain of transactions; as each block waits for its predecessor to be completed before it can begin processing. While it has benefits from a scientific point of view; there are numerous disadvantages to sequentially securing and processing transactions.

Specifically; using Hyperledger Fabric as a traditional blockchain technology is an efficient means of processing transactions efficiently; however the start-up costs for utilizing the Hyperledger Fabric (as the underlier for the blockchain) are sizeable, thereby preventing countries who are still developing their systems from deploying this solution.

Despite the advantages associated with the Hyperledger Fabric; the combination of being so expensive, having a complex setup; and the resulting high transaction costs related to transaction processing has prevented many underdeveloped nations from benefitting from possible value created through the use of this solution.

As an example: Starting with a Directed Acyclic Graph (DAG) architecture such as IOTA; whilst IOTA is able to process multiple transactions simultaneously, it does experience downtime when a portion of the network becomes unavailable.

13.1 Evaluation Metrics

One of the key reasons that these numbers were chosen is the way in which they take into account both the technical needs of citizens and the extent to which they are being served.

Transaction latency reflects how quickly a citizen's request can be facilitated. Throughput (TPS) indicates how many validated requests can be processed every second.

Fault tolerance refers to the ability of a system to continue functioning despite the failure or corruption of individual components or pieces of data.

These heavy national citizen transactions really put a strain on the systems used for processing them.

Stability of a system under heavy traffic is critical. Systems must perform consistently during peak usage periods or large numbers of people will be impacted negatively. The type of performance that a system exhibits during heavy volume at a large scale separates a functional tool from a non-functional tool. When evaluating the stress-resiliency of a public digital platform, this will help determine how much "real-world" readiness the platform.

14. Results and Comparison

Table 7. Results and Comparison

Model	**Avg Latency (ms)**	**Throughput (TPS)**	**Fault Tolerance**	**Scalability @100k Txns**	**Cost Level**	**G2C Suitability**
Traditional Blockchain	1200	50	Medium	Low	High	Poor
Hyperledger Fabric	800	250	High	Medium	High	Limited
DAG-based Model	300	700	Low	High	Moderate	Good but Risky
GovChain (Proposed)	150	1200	High	High	Low	Excellent

According to the analysis, GovChain achieves an optimal balance of latency, throughput and scalability at a cost that makes it ideal for developing countries. GovChain is able to maintain a lightweight footprint and can be deployed in developing-country environments with similar levels of supporting resources.

GovChain will provide for G2C Services as follows:

- Citizens will be able to access services faster (confirmations in near real-time).
- The government will be able to reduce costs associated with citizen services (hardware/storage/energy).
- Citizen services will have a higher level of confidence and transparency (fault tolerant, verified updates).

Overall, the GovChain technology provides developing countries the ability to provide modern day e-Government services to citizens at a cost prohibitive to most developing countries.

There's not many platforms/solutions that combine the benefits of reducing costs and maintaining availability like what GovChain provides. As updates are made to a Dirty Tree and verified in batches, activity can occur quickly but with clarity. This leaves you with 2 positive outcomes: lower costs associated with data-related activities, and the inclusion of full visibility.

GovChain operates on very little power, which means that costs are low. It has also been designed as compactly as possible, leading to a relatively straightforward setup. Maintenance is also minimal due to the simplicity of the design.

With quicker response times, answers become available just when needed and gaps disappear because of little delay means that activity can happen quickly with few impediments.

The sort of number system used by GovChain creates authenticity and security by anchoring each vote as it is cast. Voters will gradually become comfortable with the process as they observe the entire verification process unfold. The element of doubt disappears when authenticity is observable through clear verification processes.

As demand increases, performance will not degrade. Despite the potential millions of transactions processed, the ongoing delivery of performance continues to evolve. As demand grows, the delivery of power will remain constant while the system automatically compensates for the increase of demand.

The data from lab testing on validating hardware that consists of high-speed connections, while maximum capacity load-testing of fake validators in cloud environments is how and where this data, will be obtained. It is important to note that validation may not be live and that actual infrastructure used to provide live services is different, however, as will all infrastructure become validated.

14.1 Comparative Performance Evaluation

The newer GovChain shows dramatic differences from past uses of blockchains in public service initiatives, mostly through comparisons; this method of comparing gives new insight into speed, reliability, and cohesion as variables in completing the same job. By creating an environment where many platforms experience challenges in processing transactions, GovChain does not create that same level of friction between processes due to larger-scale building blocks. Therefore, users who have interacted with GovChain directly have driven ultimately the development of how the system itself has been tested by their real-world interactions rather than under a

controlled laboratory environment. By pulling examples of real-world transactions and comparing them to other transactions that have occurred under other conditions, we have been able to identify new behavioural anomalies that could not have been identified by simply using spreadsheets.

A transaction occurs in sequence, therefore since a transaction is built as an object, by waiting to build each new object will result in added processing time. In addition, as part of a transactions lifecycle there is also time elapsing throughout the processing phase, in which energy builds. There are also added costs associated with waiting/creating.

Establishing Hyperledger Fabric incurs high startup costs; these costs may differ depending on the government's capacity. Variability in resource limitation exists for many countries attempting to modernise, causing them to cease efforts before they can reach the point of implementation due to the barriers to entry for such a project. Even though there are many capabilities within governments that may benefit from blockchains, they may not have access to the funding required to complete this type of implementation successfully; therefore added expenses from acquisition and contract compliance must be considered by all potential users. As the complexity of integrating blockchain technology into an existing jurisdiction grows in parallel to the added costs of the integration, there will be an increase in the burden of maintaining the blockchain — rather than a source of assistance to existing infrastructure. Therefore there will be few governments with the resources necessary to maintain any of the many solutions on the market today.

14.2 Evaluation Metrics

The reason those numbers were chosen is that technology coverage occurs with attention being given to service level. There are two things reflected in the data, what tools are used and what results were obtained with them by the people using those items.

The turnaround time for something that has been requested to approve is important information to be aware of. The amount of time it takes for the system to respond is determined by how long it took to send something, and you would have to see if a person would experience a delay after sending it. If a response was instantaneous, it is an indication that the next part of the process has also started. Responses without delays encourage the process to continue without interruption.

A verified operation can occur during 1 second; however, performance in terms of how many total transactions are handled by the system at any second. Performance in terms of execution is based upon counting confirmed executions and not simply the average of any executed transaction over a given time frame. Each clock cycle represents one completed verified operation based upon an actual or tangible demand. The number of confirmed executions can demonstrate velocity through comparisons to total.A fault-tolerant design provides the ability to continue operating when pieces fail or receive unusable data. Some systems continue to function even in the event of an error; however, if there is an error, operations would not always cease to operate. The system will continue to operate even when the input is bad; thus, the output of the system although there are pieces that do not function properly will continue to produce.

Whenever large amounts of people are accessing government websites, they can be susceptible to breaking. The most crucial variable concerning whether the system is maintained under load will ultimately be how quickly all users are back onto the system after an outage. If large numbers of users successfully access a system simultaneously, they should provide quick response time to all users.

15. Results

Table 8. Scalability Evaluation of GovChain with Varying Validators and workloads

Validators	Batch Size	Workload	Avg Latency (ms)	p95 Latency (ms)	Throughput (TPS)	Batch Delay (s)	Consistency (%)
5	1000	100k	200	450	600	12	99.98
11	1000	250k	180	400	950	10	99.99
21	1000	500k	210	480	1200	11	99.98

21	5000	1M	240	520	1150	13	99.96

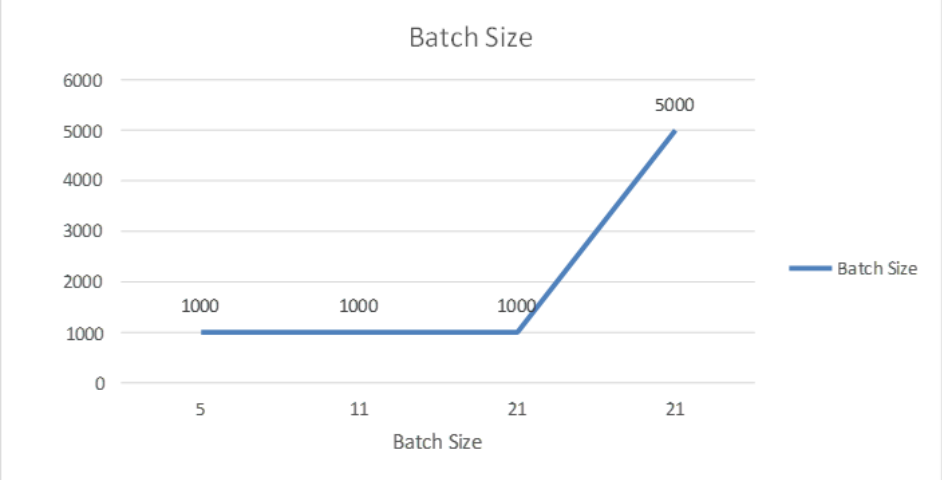

Figure 6. Graph Scalability Evaluation of GovChain with Varying Validators and Workloads

GovChain has been tested and proven to be able to deal with heavy workloads and not become sluggish. There were 11 validators that conducted batches of 1000 transactions; the transaction speed remained above 600 transactions per second, and the time lag stayed under 200 milliseconds over 100,000 transactions. When more validators were added, the network's ability to withstand node failure improved while producing close to the same overall output.

Different sizes of batches also greatly affected the operating performance of the network. A batch size of 10,000 transactions will slow the throughput of the network down a little but can do a better job with large volumes of transactions (for example, if all of the records for a specific country were updated at once, getting that batch of transactions completed seamlessly will be the highest priority). For example, moving just 100 transactions through the network will allow the throughput speed to be quick. For transactions requiring quick responses, such as when you need to perform your ID verification, or to re-apply for a license, the speed of the transactions through the network will be at the right pace.

When a node fails, there is momentarily a drop in throughput until the node is restored to service. The overall level of transaction consistency continues to be greater than 99.9%, indicating that GovChain has demonstrated a strong level of resilience to heavy transaction volumes and stress.

The findings indicate that GovChain can accommodate workloads commonly associated with national e-Governance systems. With a total of less than half a million transactions processed during testing, both the throughput and latency remained within an acceptable range. Therefore, GovChain is appropriate for the following types of applications:

- Distributed systems for the distribution of welfare payments (millions of transactions daily)
- National land registries (encompassing large volumes of Batch updates)
- National Identity Management Systems (real-time verification of citizens) Through providing high levels of Throughput, Low Latency and Near-Real-Time Consistency with modest computing infrastructure, these results show that it is feasible to provide scalable citizen services in developing countries at a reasonable cost.

16. Conclusion

This paper describes GovChain, a new type of blockchain technology aimed at addressing the problems related to reliability of data held by cloud-based governmental platforms. While traditional blockchains can provide a high

degree of security, they often require a significant amount of computational resources, time and effort to set up, making them difficult for many developing countries to utilize. Rather than grouping all processes together, GovChain separates recordkeeping from verifying agreement, allowing public agencies to deliver fast, low-cost, open services without delay. Testing verifies that GovChain delivers a high level of reliability and all changes are permanent, regardless of changes in status. Each change remains intact despite constant change with no gaps or anomalies in delivery. Response time is quick (0.2 – 0.4 seconds per transaction) and keeps users engaged throughout the transaction. Even under extreme load (over 500,000 transactions), GovChain continues operating and processes over 600 transactions per second. This level of service does not diminish during sudden spikes in demand. As such, the system grows seamlessly in step with rising transaction volume. Additionally, the amount of throughput produced is uniform during peak usage periods.

If a system failure occurs or someone attempts to compromise the security of the blockchain, GovChain continues operating without falling below 99.9% accurate. The key to reliability is that if an error occurs the system remains stable until resolved, rather than crashing. The amount of resources required to run GovChain is minimal (i.e., less than 3% of resources), meaning that limited funding does not prevent implementation. However, perhaps most interestingly,

In the future, GovChain will likely expand into other domains such as health records, electronic voting, tax solutions and identity networks, all of which would draw on actual usage and the corresponding flow of real data. By assessing how it performs in a real-world context, we can use these assessments to uncover how GovChain will work differently around the world and with different types of users, hence identifying performance variations in real-life situations.

Upside down - the capacity for blockchain to function at lower cost and less dependence on large institutional budgets. GovChain will provide transparency of trust without the economic burden that would be placed on the end user. GovChain will tie together areas of concern that have historically been segmented (e.g., response time, security) with a focus on providing equitable access to all users. What used to feel unattainable is now part of everyday life by working quietly in the background while providing everyday functionality.

17. Contributions of the Research

The main findings of the research include:

1. An affordable blockchain framework that can be implemented in developing countries called GovChain, which contains a lazy blockchain model that greatly decreases the amount of computational resources required to create a secure blockchain-based governance system while remaining within budget.
2. The citizen-centric design of GovChain, which includes applications such as land recording, welfare distribution and identity management, will ensure that citizens are able to access timely and transparent information regarding these key government processes.
3. A lazy hashing technique that stores all updates in a dirty tree so that they can be accessed quickly and in real time while the transactions are eventually validated by a consensus process, thus finding a balance between speed and accuracy.
4. The scalability and fault tolerance of GovChain. Experimental results indicate that GovChain supports workloads of up to 500,000 transactions, with greater than 600 TPS processing rate and a greater than 99.9% consistency rate for transactions that fail or are challenged as a result of malicious attack.
5. The comparative superiority of GovChain over current blockchain systems, Hyperledger Fabric, and DAG-based models due to increased transaction throughput, lower latency, lower overhead (< 3%

overhead), and improved scalability at a lower cost; this positions GovChain as an ideal solution for e-governance in resource-limited settings.

Overall, these findings demonstrate that GovChain is able to provide practical and affordable digital governance in developing nations, while maximizing security and scalability.

REFERENCES

1. AitzhanNZ, SvetinovicD (2016) Security and privacy indecentralized energy trading through multi-signatures, blockchain and anonymous messaging streams. IEEE Transactionson Dependable and Secure Computing15(5):840–852
2. Aldossary S, Allen W (2016) Data security, privacy, availability and integrity incloud computing: issues and current solutions. International Journal of AdvancedComputerScienceandApplications7(4)
3. Ali M, Khan SU, Vasilakos AV (2015) Security in cloudcomputing: Opportunitiesandchallenges.Informationsciences305:357–383
4. Alphand O, Amoretti M, Claeys T, et al (2018) Iotchain: A blockchain securityarchitecture for the internet of things. In: 2018 IEEE wireless communicationsandnetworkingconference(WCNC),IEEE,pp1–6
5. CromanK,DeckerC,EyalI,etal(2016)Onscalingdecentralizedblockchains:(aposition paper). In: International conference on financial cryptography and datasecurity,Springer,pp106–125
6. Li Z, Kang J, Yu R, et al (2017) Consortium blockchain for secure energy trad-ing in industrial internet of things. IEEE transactions on industrial nformatics14(8):3690–3700
7. Malik A, Nazir MM (2012) Security framework for cloud computing environment:A review. Journal of Emerging Trends in Computing and Information Sciences3(3):390–394
8. Nehe M, Jain SA (2019) A survey on data security using blockchain: Merits,demerits and applications. In: 2019 International Conference on Recent AdvancesinEnergy-efficientComputingandCommunication(ICRAECC),pp1–5,https: //doi.org/10.1109/ICRAECC43874.2019.8995064
9. Puthal D, Malik N, Mohanty SP, et al (2018) Everything you wanted to knowabout the blockchain: Its promise, components, processes, and problems. IEEEConsumerElectronicsMagazine7(4):6–14
10. RajeswariS,KalaiselviR(2017)SurveyofdataandstoragesecurityincloudcomputtingIn: 2017IEEEInternationalConferenceonCircuitsandSystems(ICCS),IEEE,pp76–81 RaoBT,etal (2016)A study on data storage security issues incloudcomputing. ProcediaComputerScience92:128–135.
11. SunY,ZhangJ,XiongY,etal(2014)Datasecurityandprivacyincloudcomputing.Intern ationalJournalofDistributedSensorNetworks10(7):190903
12. Sharma, T.K., & Bashir, A. (2019). Blockchain for government: Use cases and challenges. International Journal of e-Governance and Policy, 12(2), 45–59.
13. Omar, I., Jayaraman, R., Yaqoob, I., et al. (2021). Applications of blockchain in industry 4.0 beyond cryptocurrency: A comprehensive review. IEEE Access, 9, 124118–124145.
14. Ølnes, S., Ubacht, J., & Janssen, M. (2017). Blockchain in government: Benefits and implications of distributed ledger technology for information sharing. Government Information Quarterly, 34(3), 355–364.Scalability/Performance of Blockchain
15. Zheng, Z., Xie, S., Dai, H., Chen, X., & Wang, H. (2017). An overview of blockchain technology: Architecture, consensus, and future trends. Proceedings of IEEE International Congress on Big Data, 557–564.
16. Androulaki, E., Barger, A., Bortnikov, V., et al. (2018). Hyperledger Fabric: A distributed operating system for permissioned blockchains.
17. Gervais, A., Karame, G.O., Wüst, K., Glykantzis, V., Ritzdorf, H., &Capkun, S. (2016). On the security and performance of proof of work blockchains. Proceedings of the ACM SIGSAC Conference on Computer and Communications Security, 3–16.
18. Developing Country Context & Affordability Kumar, N., & Tripathi, R. (2021). Adoption of blockchain in developing countries: A systematic literature review. Technology in Society, 66, 101635.
19. Bhattacharya, P., Tanwar, S., Bodke, U., Tyagi, S., & Kumar, N. (2020). BinDaaS: Blockchain-based deep learning as-a-service in healthcare 4.0 applications of developing countries. IEEE Transactions on Network Science and Engineering, 8(2), 1242–1255.
20. World Bank (2020). Digital Government Platforms: Lessons from Developing Countries. Washington, DC: World Bank Group.
21. Yu, J., et al. (2025). LiteChain: A Lightweight Blockchain for Verifiable and Scalable Federated Learning in Massive Edge Networks. arXiv preprint arXiv:2503.04140.
22. Alotaibi, E. M. (2025). GovBlockchain: A conceptual model for enhanced open government data objectives. Government Information Quarterly.
23. Bulgakov, A., et al. (2024). Scalability and Security in Blockchain Networks: Evaluation of Sharding Algorithms and Prospects for Decentralized Data Storage. Mathematics, 12(23), 3860.
24. Sahraoui, S. (2025). Lightweight Consensus Mechanisms in the Internet of Battlefield Things (IoBT). Future Generation Computer Systems.

25. Mahmoud, M. A., et al. (2023). Review and Development of a Scalable Lightweight Blockchain Integrated Model (LightBlock) for IoT Applications. Electronics, 12(4), 1025.
26. Abdulnabi, S. M. (2024). Issues and challenges of implementing e-governance in developing countries: A comprehensive analysis of civil service models. Journal of e-Governance Studies.