Daggers Over Chains

Sebastian Tobar Quintero
sebastiantq@protonmail.com

BASE COMPUTING S.A.S.
social@basecomputing.com.co

Index

1. Executive Summary

This whitepaper presents an innovative and detailed vision regarding the evolution and next steps for the future of distributed ledger technologies (DLTs). It proposes an advanced architecture called Micro-DLTs Chain, which combines the efficiency of Directed Acyclic Graphs (DAGs) with functional specialization through interconnected Micro-DLTs. This approach seeks to resolve scalability and efficiency limitations present in current distributed ledger technologies such as Blockchain and DAGs.

The proposal includes creating a DAG of DAGs structure, where each node in the main DAG is a block, which in turn, is a DAG. This allows for multi-level scalability. The concept of Micro-DLTs is then introduced, specialized units for different functions within the network, enabling modular scalability and operational efficiency.

The technical foundations of this architecture are explored in detail, including its benefits in terms of scalability and security and the protocols necessary for implementation. Practical applications and case studies are also presented, demonstrating the potential of this architecture in various scenarios.

Finally, technical challenges and proposed solutions are discussed, as well as areas for future research and development. This whitepaper aims to offer a comprehensive and technical vision to guide researchers, developers, and enthusiasts in building the future of distributed ledger infrastructure.

2. Introduction

Blockchain technology has revolutionized the way we perceive security, transparency, and decentralization across various fields, from finance to the food industry. Its ability to provide an immutable and distributed ledger has enabled the emergence of innovative applications and has transformed how we manage trust and transactions.

However, as the adoption of Blockchain has grown, its limitations have also become evident, especially in terms of scalability and efficiency. Traditional Blockchains, such as Bitcoin and Ethereum, are designed with a linear chain of blocks structure that, while secure and decentralized, faces significant challenges when it comes to processing a high volume of transactions quickly and efficiently. This linear design creates bottlenecks and high transaction fees, limiting its ability to effectively scale for real-world applications.

To address these limitations, researchers and developers have explored various alternatives and improvements to Blockchain architecture. One of these alternatives is the use of Directed Acyclic Graphs (DAGs), which offer a more flexible and scalable structure for managing transactions. DAGs allow for the parallel validation of multiple transactions, significantly improving processing speed and efficiency.

However, while DAGs represent a significant advancement, they also face their own challenges, such as ensuring consistency, security, and interoperability. To overcome these limitations, this whitepaper proposes an advanced architecture called Micro-DLTs Chain.

The Micro-DLTs Chain combines the flexible and scalable structure of DAGs with functional specialization through Micro-DLTs. This approach not only enhances horizontal scalability but also introduces a hierarchy of hash dependencies that reinforces network security and consistency, making the hash of one chain recursively dependent on others.

The objective of this proposal is to create a distributed ledger infrastructure that can scale effectively and efficiently, adapting to the changing needs of various industrial applications. The Micro-DLTs Chain enables modularity that facilitates functional specialization, resulting in a more robust and adaptable network.

3. Fundamentals of DAGs and Their Evolution

3.1. Explanation of Directed Acyclic Graphs (DAGs)

A Directed Acyclic Graph (DAG) is a data structure consisting of nodes and directed edges, where each edge has a direction, and there are no cycles. This means that no directed path returns to the starting node. DAGs are useful for representing processes with unidirectional dependencies, such as workflows, task graphs, and in this case, transaction networks and transaction network hierarchies, similar to blocks in Blockchain.

3.1.1. Structure of a DAG

A DAG is composed of:

- Nodes: Representing data or states within the network.
- **Edges**: Indicating the relationships between nodes, reflecting dependencies or information flow.

Diagram of a basic DAG:

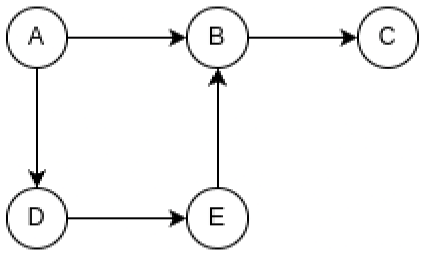


Figure 1

In this example, node A has directed edges to B and D, D to E, E to B, and B to C. The acyclic nature is maintained since no path allows returning to A from any other node.

3.1.2. Comparison with Traditional Blockchain

In a traditional Blockchain:

- Data is stored in blocks.
- Each block contains a set of transactions.
- Blocks are connected in a linear chain, where each block points to the previous one.

Diagram of a basic Blockchain:

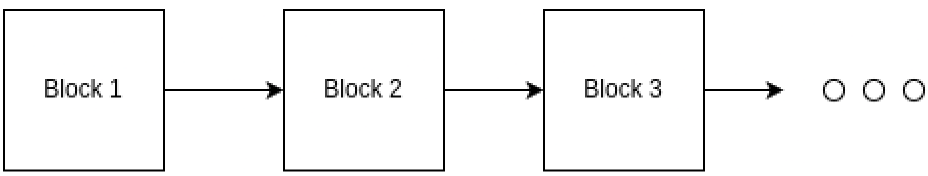


Figure 2

The limitations of this structure include:

- **Limited scalability:** Only one block can be added at a time, slowing down transaction processing.
- **Low efficiency:** Validation and addition of blocks require global consensus, which can be costly and slow.

3.1.3. Advantages of DAGs

DAGs offer several significant advantages over traditional Blockchains:

- **Improved scalability**: DAGs enable the parallel addition of multiple nodes, not constrained to a linear sequence as in traditional blockchains. This facilitates the simultaneous processing of a larger transaction volume, crucial for high-performance applications like real-time payment systems.

- **Energy efficiency and cost reduction**: DAGs eliminate the need for extensive computational and energy resources by not using mining to validate transactions. Instead, each new node (transaction) verifies n previous nodes, making the network more eco-friendly and cost-effective regarding mining and equipment maintenance costs.

- **Faster confirmation times**: Thanks to their parallel structure, DAGs can confirm transactions more quickly than linear blockchains. In a DAG, each newly added node helps verify and confirm previous transactions, distributing the validation workload across the network and speeding up the overall process.

3.2. Evolution of DAGs

The evolution of DAGs in the context of distributed ledgers has led to the creation of various innovative solutions. Notable examples include:

3.2.1. IOTA

IOTA uses a DAG structure known as Tangle to manage transactions in the Internet of Things (IoT). In the Tangle, each new transaction must approve two previous transactions, eliminating the need for miners and enabling greater scalability.

3.2.2. Nano

Nano employs a DAG structure called block-lattice, where each account has its own Blockchain. This allows for fast, fee-less transactions as each account can be updated independently.

These examples demonstrate how DAGs can provide more efficient and scalable solutions compared to traditional Blockchains. However, challenges regarding security and consensus remain to be addressed.

3.3. Modularity and Interoperability Across Chains

In addition to DAG-based solutions like IOTA and Nano, other innovative architectures have explored scalability and interoperability in distributed systems through different approaches:

3.3.1. Polkadot

Polkadot introduces a structure of **parachains**, specialized chains that operate in parallel, connected to a central chain known as the **Relay Chain**. This design enables interoperability among parachains and shared security, where the Relay Chain coordinates and validates transactions across the system. Each parachain can be customized according to its specific needs, optimizing its performance for particular tasks. However, Polkadot uses traditional blockchains instead of DAGs, limiting total parallelization to the level of individual chains.

3.3.2. Cosmos

Cosmos proposes a **zone-based** approach, independent blockchains interconnected through the **IBC (Inter-Blockchain Communication)** protocol. This model allows each zone to operate autonomously while maintaining interoperability with other zones within the Cosmos Hub network.

Although similar in concept to Polkadot, Cosmos stands out for its emphasis on decentralizing the chains, enabling each zone to implement its own consensus mechanism. Nevertheless, like Polkadot, Cosmos's structure relies on linear blockchains.

3.3.3. Interledger Protocol (ILP)

ILP is designed to facilitate payments between different ledgers, whether blockchains or traditional financial systems. It uses **connectors** to transfer value between systems, preserving the independence of each ledger. While efficient for cross-system transactions, ILP does not define a hierarchical structure or provide a native framework for modular interoperability.

3.4. Conclusion of the Fundamentals

The structure of DAGs and their evolution represent a significant advancement in distributed ledger technology. As we continue to explore their applications, it is essential to understand both their advantages and challenges to fully harness their potential in creating more scalable and efficient infrastructures.

Approaches to modularity and interoperability in distributed ledger technologies have made significant progress. However, none address the hierarchical design and DAG-level parallelization that characterizes the architecture proposed in this document.

4. Proposal for DAG of DAGs

4.1. Concept and Structure

In a DAG of DAGs:

- **Main DAG**: The nodes of the main DAG are what would traditionally be known as blocks in Blockchain.
- **Secondary DAGs**: Within each node of the main DAG are secondary DAGs, where each node represents the transactions of the block.

Diagram of a DAG of DAGs:

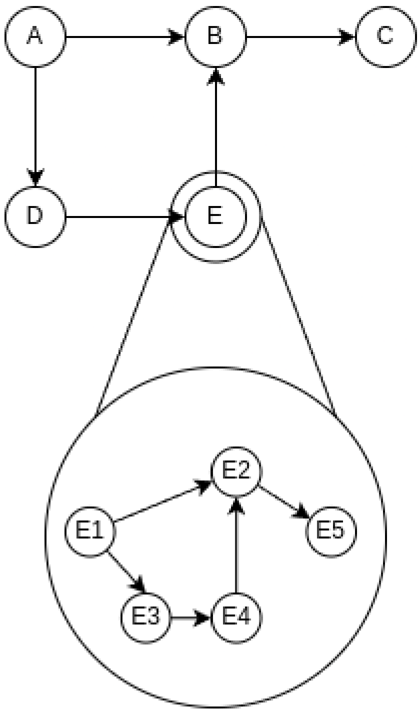


Figure 3

4.2. Benefits in Terms of Scalability and Efficiency

1. **Multi-Level Scalability**: By parallelizing the processing of transactions and blocks, the network gains scalability at various levels of operation compared to current solutions.
2. **Reduction of Bottlenecks**: The architecture eliminates bottlenecks as all growth operations of the network, both at the transaction and block levels, are completely parallelized.

5. Introduction of Micro-DLTs

5.1. Definition and Purpose of Micro-DLTs

Although the previous architecture already scales well, it is possible to go further, which is why Micro-DLTs are introduced. These are DAGs of DAGs focused on specific functions within the global network ecosystem. Each Micro-DLT can handle a particular type of task, allowing greater efficiency and scalability by distributing workloads modularly.

One of the main advantages of Micro-DLTs is the ability to allocate resources individually and arbitrarily based on the specific needs of each task. This means that:

- Dedicated Resources: Each Micro-DLT can receive dedicated resources (such as computing power, storage, and bandwidth) depending on its importance and workload.
- Dynamic Optimization: Resources can be dynamically adjusted to respond to changes in demand or operational priorities, ensuring optimal performance at all times.
- Scaling Flexibility: Micro-DLTs with higher demand can scale horizontally by adding more resources, while those with less demand can operate with minimal resources, optimizing infrastructure usage.
- Task Isolation: Allocating resources individually minimizes the impact of problems or failures in one Micro-DLT on the performance of others.

This precise and adaptable resource management improves operational efficiency and enables better planning and use of available resources, adapting to the ecosystem's changing needs.

Key features of Micro-DLTs:

1. **Functional Specialization**: Each Micro-DLT is dedicated to a specific function, such as asset exchange, smart contract execution, or community sovereignty over the network.
2. **Modular Scalability**: Enables adding and updating Micro-DLTs independently, facilitating the system's expansion and improvement.
3. **Interoperability**: Micro-DLTs are designed to communicate and collaborate, ensuring the coherence and efficiency of the global system, as all instances deployed on a node operate under the same gateway, allowing intelligent traffic redirection as if they were a single chain.
4. **Flexibility and Adaptability**: Facilitates implementing a wide range of applications and use cases, ensuring acceptance across various industries (finance, healthcare, IoT, supply chains, etc.), customization of functionalities, and continuous evolution.

5.2. Use Cases

5.2.1. Example 1: Value Exchange

A Micro-DLT can manage the network's cryptocurrency exchange, solely handling value transactions between users.

5.2.2. Example 2: Execution of Smart Contracts

Another Micro-DLT can specialize in executing smart contracts, managing their deployment and execution within the network.

6. Hierarchy of Micro-DLTs

6.1. Hierarchical Structure and Hash Dependencies

Micro-DLTs do not constitute an ecosystem by themselves, but their interconnection does. This section introduces a hierarchical structure where each Micro-DLT depends on a parent DLT to compute the hashes of its blocks. This creates a chain of dependencies ensuring network coherence and integrity.

This approach differs from technologies like the Interledger Protocol (ILP), which connects different ledgers using connectors but does not establish strict cryptographic dependencies. The key innovation of this proposal lies in the cryptographic hash hierarchy, ensuring each Micro-DLT is intrinsically linked to its parent DLT, strengthening both the coherence and security of the ecosystem. Unlike systems such as Polkadot or Cosmos, which implement interoperability through autonomous modules, this hierarchical architecture enables robust synchronization and modular interconnection based on DAGs.

Key features of the hierarchical structure:

1. **Hash Dependencies**: Each block in a Micro-DLT must include the hash of the last block of its parent Micro-DLT, creating a chain of dependencies.
2. **Security**: Hash dependencies reinforce security by ensuring that any change in a block affects all descendant blocks.
3. **Coherence**: This structure ensures that all Micro-DLTs are synchronized and that any change in one Micro-DLT is reflected across the entire network.

Micro-DLTs Hierarchy Diagram:

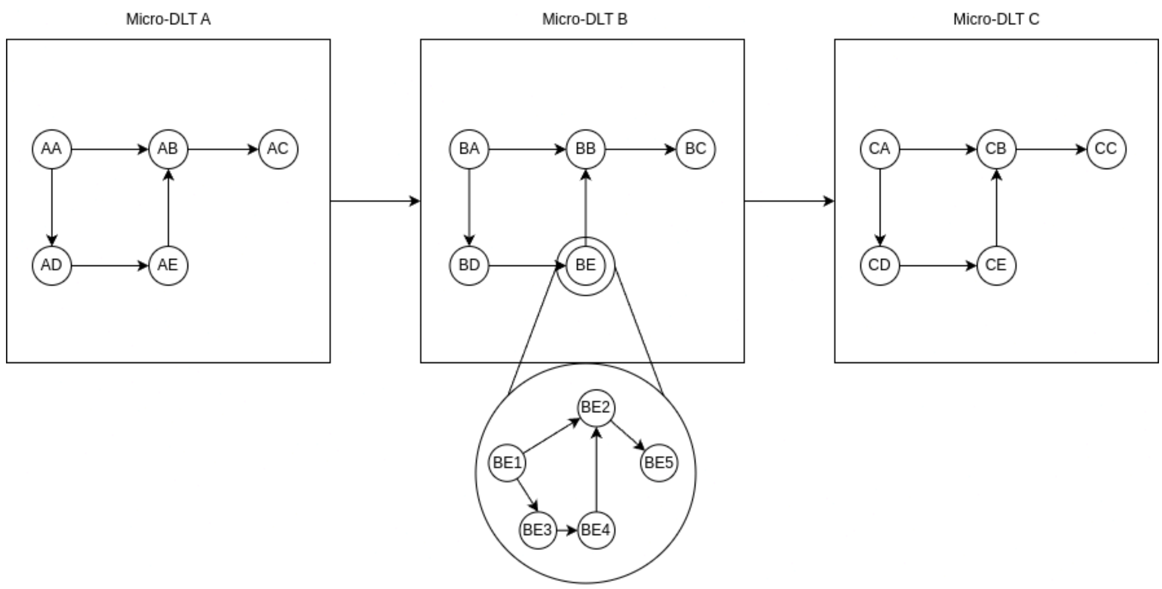


Figure 4

6.2. Benefits of the Hierarchical Structure

1. **Enhanced Security**: The hash dependency structure reinforces security, making any manipulation attempt extremely difficult.
2. **Data Integrity**: Coherence of data is maintained across the network, ensuring all transactions and states are consistently validated.

7. Implementation and Protocols

7.1. Technical Details on Implementation

1. **Data Structure**: The data structure itself is a nesting of a list of DAGs of DAGs. This spatially may not scale well for storing these data structures, so implementing encoding and decoding techniques for interacting with memory is recommended. This will be particularly useful for handling public keys, implementing a Micro-DLT for a DNS service within the ecosystem.
2. **Transaction Validation**: Each Micro-DLT and secondary DAGs must implement local validation mechanisms to enhance network efficiency without compromising security. These validations occur at three levels: when a new transaction enters the network, it is directed to a Micro-DLT, which assigns it to a block. Once in the block, it must verify the authenticity of n previous transactions. When a transaction reaches m validations, it will be processed. Finally, once a block is completed, it is signed and validated by a participating node in the network validation process (similar to PoS). This system allows transactions to be validated, processed, and confirmed in parallel at three levels, differing significantly from traditional blockchains.
3. **Hash Computation**: For new projects, a post-quantum world is more appropriate, so the public-private key pair generation algorithm will be quantum-resistant, as will the transaction signing algorithm. This highlights the importance of the DNS service, as the length of these keys can become unmanageable.

7.2. Communication Protocols Between Micro-DLTs

1. **Synchronization Protocol**: The network's complexity increases the likelihood of forks, making proper synchronization among network nodes critically important.
2. **Consensus Protocol**: While block addition consensus occurs locally within each Micro-DLT, an efficient and secure mechanism is needed for synchronizing consensus among network nodes.
3. **Interoperability Protocol**: Ensuring all Micro-DLTs function as a unified whole requires mechanisms that allow seamless interoperation, creating a true ecosystem.

7.3. Interoperability Management

1. **Interface Standardization**: Define standardized interfaces for communication between Micro-DLTs and an interface to interact with ecosystems outside the Micro-DLT chain.

8. Practical Applications and Use Cases

8.1. Concrete Examples of Applications

8.1.1. Ecosystem Voting System

The Micro-DLT chain allows for adding more Micro-DLTs depending on system needs. If ecosystem consensus decides to add a new Micro-DLT to dedicate specific resources to a project, it can be done. This is exactly what would happen with the ecosystem voting system, which would be implemented at the infrastructure's layer 0, and its information would be managed exclusively by a dedicated Micro-DLT.

8.1.2. Smart Contract Voting System

The Smart Contracts Micro-DLT will enable applications developed on the infrastructure to execute, exclusively managing the deployment and execution of smart contracts. Among them, a voting system could be deployed. Unlike the previous example, this system would not have a dedicated Micro-DLT but would be managed by the Smart Contracts Micro-DLT, sharing system resources with other smart contracts.

9. Challenges and Solutions

9.1. Identification of Possible Technical Challenges

Despite the advantages and potential of the Micro-DLTs chain architecture, several technical challenges must be addressed to ensure its successful implementation and operation.

Main Challenges:

1. **Implementation Complexity**: The proposed architecture is inherently complex, which may complicate its implementation and maintenance. It requires meticulous planning and rigorous management of dependencies between nodes and Micro-DLTs.
2. **Security**: Ensuring network security against attacks, such as double-spending and Sybil attacks, is crucial. The interdependence between Micro-DLTs may introduce new vulnerabilities that need mitigation.
3. **Consensus and Synchronization**: Maintaining consensus and synchronization among the network's multiple nodes is a significant challenge. Efficient mechanisms must be developed to ensure that all Micro-DLTs are updated and coherent.
4. **Interoperability**: Communication and collaboration between Micro-DLTs require robust and efficient protocols. Ensuring interoperability without compromising security and efficiency is a key challenge.
5. **Scalability and Performance**: Although the architecture is designed to enhance scalability, ensuring optimal performance in real-world scenarios is necessary. Resource management and process optimization are essential to maintain high performance.

9.2. Proposed Solutions and Mitigations

To address these challenges, several solutions and strategies are proposed:

1. **Complexity Management**: Develop specific tools and frameworks for implementing and managing the Micro-DLTs chain architecture. Encourage the use of best development practices and modularity to simplify implementation.
2. **Enhanced Security**: Implement robust consensus algorithms to secure the network. Develop attack detection and mitigation mechanisms, including real-time monitoring and response systems.
3. **Consensus and Synchronization Mechanisms**: Utilize efficient consensus algorithms that can handle validation and synchronization among multiple network nodes. Implement synchronization protocols to ensure that all Micro-DLTs are updated and coherent.
4. **Interoperability Protocols**: Develop standards and communication protocols that ensure secure and efficient interoperability between Micro-DLTs. Use standardized interfaces to facilitate integration and collaboration among different Micro-DLTs.
5. **Scalability and Performance Optimization**: Conduct continuous testing and optimizations to ensure that the architecture can handle real-world workloads efficiently. Implement resource management strategies that optimize the use of available infrastructure.

10. Conclusion and Future Research

10.1. Summary of the Proposal’s Benefits and Potential

The Micro-DLTs chain architecture represents a significant advancement in distributed ledger technology. By combining the scalability of DAGs with the modularity of Micro-DLTs, this architecture offers numerous advantages:

1. **Enhanced Scalability**: Enables parallel transaction processing at multiple levels. By using a nested DAG structure, bottlenecks common to traditional Blockchain architectures are eliminated.
2. **Operational Efficiency**: Improves efficiency through functional specialization in Micro-DLTs. Distributing workloads across different Micro-DLTs, each with a specific purpose, optimizes resource utilization and reduces processing time.
3. **Security and Coherence**: Ensures network integrity and coherence through a hierarchy of hash dependencies. This structure guarantees that any change in a node securely propagates throughout the network, maintaining data consistency.
4. **Flexibility and Adaptability**: Facilitates continuous adaptation and improvement through the addition and updating of Micro-DLTs. The architecture's modularity allows for integrating new functionalities and enhancing existing ones without disrupting network operations. This is particularly useful in dynamic environments where technological needs can change rapidly, enabling constant evolution to meet new demands and opportunities.

10.2. Areas for Future Research and Development

1. **Protocol Optimization**: Enhance communication and consensus protocols to increase efficiency and security. Develop new consensus algorithms capable of handling high transaction volumes with lower latency and greater resistance to attacks.
2. **Practical Applications**: Develop and test practical applications of the architecture in various scenarios, including Micro-DLT implementations in sectors like healthcare, supply chains, and energy.
3. **Security and Resilience**: Strengthen security mechanisms and resilience against attacks. Research advanced techniques for detecting and mitigating threats, including Sybil and double-spending attacks.
4. **Standardization and Regulation**: Collaborate with international organizations to establish standards for interoperability and security of Micro-DLTs. Ensure compliance with legal and regulatory frameworks in different jurisdictions.
5. **Advanced Cryptography and Key Management**: Explore new cryptographic techniques and key management systems, particularly those resistant to quantum computing, and develop efficient algorithms for integration with the Micro-DLTs chain.
6. **Interoperability Across Heterogeneous Networks**: Investigate mechanisms to enable interoperability between Micro-DLTs and other distributed ledger systems, including traditional Blockchain and DAGs.
7. **Artificial Intelligence and Machine Learning**: Integrate AI and machine learning techniques to optimize network management and operations, including anomaly detection

and dynamic resource optimization.
8. **Applications in the Internet of Things (IoT)**: Develop IoT-specific applications using the Micro-DLTs chain architecture for secure device management and efficient data processing.
9. **Cryptocurrency Economics and Incentive Models**: Explore economic models and incentive structures compatible with the Micro-DLTs chain, including native cryptocurrencies and decentralized financing mechanisms.
10. **Decentralized Governance**: Research governance models that enable collaborative and transparent decision-making within the Micro-DLTs ecosystem.
11. **Scalability and Performance in Global Networks**: Conduct studies to ensure the architecture scales effectively globally, managing latency, load balancing, and data replication across regions.
12. **Integration with Legacy Systems**: Investigate methods for integrating Micro-DLTs chain architecture with existing legacy systems in various industries.
13. **Environmental Impact and Sustainability**: Explore ways to minimize the environmental impact of Micro-DLTs chain operations, including renewable energy use and hardware optimization.

11. References and Annexes

11.1. References

Nakamoto, Satoshi. "Bitcoin: A Peer-to-Peer Electronic Cash System." 2008. Available at: bitcoin.org/bitcoin.pdf.

Popov, Serguei. "The Tangle." 2017. Available at: iota.org/research-papers.

LeMahieu, Colin. "Nano: A Feeless Distributed Cryptocurrency Network." 2015. Available at: nano.org/en/whitepaper.

Wood, Gavin. "Polkadot: Vision for a Heterogeneous Multi-Chain Framework." 2016. Available at: https://github.com/polkadot-io/polkadot-white-paper/blob/master/PolkaDotPaper.pdf.

Kwon, Jae; Buchman, Ethan. "Cosmos: A Network of Distributed Ledgers." 2016. Available at: https://github.com/cosmos/cosmos/blob/master/WHITEPAPER.md.

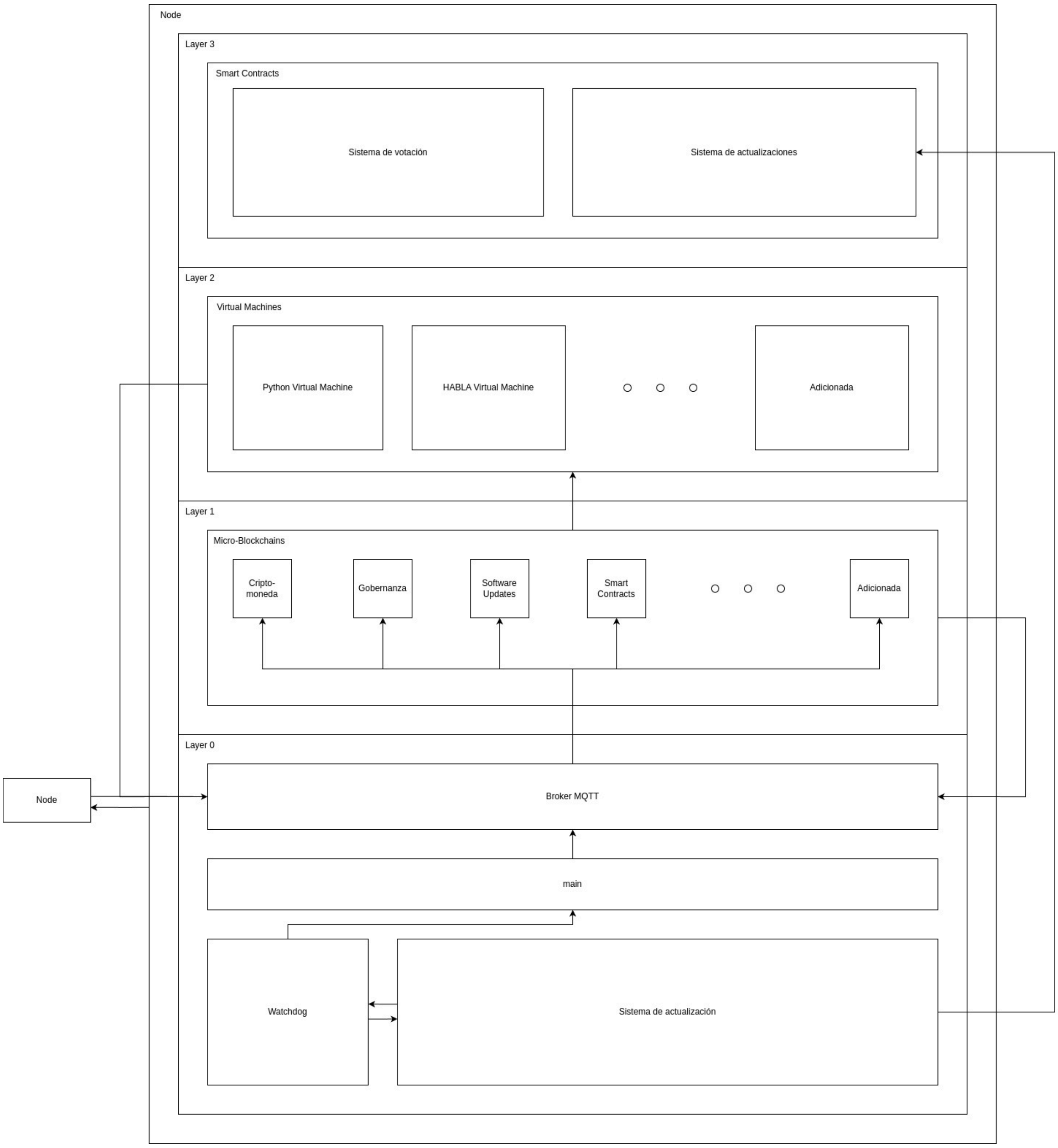
Thomas, Stefan; Schwartz, Evan. "A Protocol for Interledger Payments." 2015. Available at: https://interledger.org/interledger.pdf.

11.2. Annexes

Detailed Diagram of the Micro-DLTs Chain Network Node Architecture

Description: This annex includes a detailed diagram illustrating the architecture and interconnection of nodes within the Micro-DLTs Chain network. It provides a clear view of how the nodes are structured and communicate within the network.

Image:



Example Code for Implementing a Micro-DLT

Description: This annex provides an example code snippet illustrating how to implement a Micro-DLT within the network. It includes code fragments and detailed explanations to facilitate understanding and reproduction of the process by developers.

URL: https://bitbucket.org/base-computing/micro_dlts_chain_mvp/src/main/ (Still closed source)

These annexes are designed to provide additional visual and practical support to the concepts discussed in the whitepaper, facilitating the understanding and application of the proposed architecture.