Dysing: Improving Blockchain Scalability

Empowering Web3 Ecosystems Through Transient Digital Contracts

Author: Mars Landin

marslandin@scifer.io

Abstract

Dysing, introduced by Scifer, aims to improve the scalability of blockchain systems by managing unused or obsolete contracts. These unnecessary records can bloat the blockchain, leading to higher costs and slower performance. Dysing cleans up these obsolete contracts, without losing the integrity of the blockchain, making it more optimized and cost efficient. Specifically, it achieves:

Optimization: Dysing frees up valuable data storage by removing unused or obsolete contracts. Dysing also enables down scaling. This provides an opportunity to further optimize resource utilization by reducing storage space allocation at the moment demand is low.

Cost efficiency: By cleaning up redundant data, the blockchain can validate the State and process transactions faster, reducing overall processing time. On this basis, overall costs are immediately reduced.

All in all, Dysing is a promising solution for blockchain applications in everyday use cases. It fits well with the need for scalability.

1. Introduction

The advent of blockchain technology, initiated by Satoshi Nakamoto [1] in 2008, marked a transformative moment in digital innovation and laid the foundation for a new era of decentralization, data integrity and transparency. As blockchain systems have evolved, they have become essential tools not only for digital cash such as Bitcoin, but also for a wide range of applications in Decentralized Finance (DeFi), healthcare, supply chain management and more. However, as these systems grow, they face a significant challenge: the accumulation of redundant or obsolete data, which can lead to bloated networks, higher costs and slower performance [2]. Dysing emerges as a breakthrough solution to this issue, poised to transform our understanding of blockchain as we know it today.

Dysing, developed by Scifer, addresses this crucial challenge through the management of unused or obsolete data on the blockchain. This is done in a decentralized, autonomous and random (DAR) manner. By enabling customers to randomly delete unnecessary contracts directly from the blockchain without the intervention of a centralized intermediary, Dysing significantly improves scalability, addressing a key part of the blockchain trilemma (the balance between scalability, decentralization, and security, as raised by Vitalik Buterin [3]). This innovation is particularly important for web3 [4], which is a more decentralized version of the web, a concept that evolved from Tim Berners-Lee's 1999 idea of a "Semantic Web" [5]. In 2013 web3 was developed as web3.0 [6] by Gavin Wood in the context of blockchain technology. Next to this it's also in support of Smart Contracts, which are self-executing online contracts, a concept first proposed by Nick Szabo [7] in 1996.

At its core, Dysing embodies the principles of Web3 by promoting decentralization and empowering users through trustless systems. It enables the removal of digital contracts on a blockchain – contracts that expire when their purpose is met or at the moment of deletion – allowing networks to remain efficient and responsive. This capability aligns perfectly with the vision of Web3.0, which seeks to reshape the internet infrastructure through technologies such as smart contracts, decentralized applications (DApps, made popular by blockchains Ethereum [8] and Cardano [9]) and tokenization.

In conclusion, Dysing is more than just a tool for blockchain networks – it represents a fundamental shift toward a decentralized, user-centric, and transparent internet. Dysing paves the way for wider adoption of blockchain technology and positions itself as a breakthrough innovation that can redefine the future of Web3.0.

This whitepaper explores these concepts and highlights Dysing's role in shaping the next generation of blockchain and Web3.0 systems.

2. Theory of Dysing

Dysing provides a framework for addressing redundant data on a blockchain by making it possible to remove outdated contracts while maintaining the integrity of the chain. To actually change an immutable blockchain, a new approach to understanding the logic behind the mechanism is needed.

Traditional blockchain architectures treat immutability as inviolable, but this rigidity comes at a price. Although immutability guarantees security, it also perpetuates redundancy.

A blockchain $\mathcal{B}$ can be thought of as a chain of digital blocks (or contracts), arranged in a fixed order:

$$\mathcal{B} = (x_n)_{n \in \mathbb{N}}$$

Here, each x_i represents a single contract (e.g., a transaction or smart contract). Over time, the chain grows as new contracts are added, but obsolete contracts (e.g., expired or unused) remain stored indefinitely, wasting space.

A blockchain can be modified in specific ways. In theory, the first nine blocks of a ten-block chain could be removed. However, this alters the 10th block into the genesis block, thereby changing the historical metadata of the blockchain. Such an approach carries the risk of invalidating historical evidence, which can ultimately lead to issues with the validation of the State [10]. Moreover, true decentralization requires more than centralized control over data and/or the linear erasure of historical data [11]. The owner of a contract must be able to autonomously decide whether this data is stored or erased.

2.1 Dysed Subsets

A dyse operation offers a solution for managing outdated digital contracts by generating a reduced version within a new contract, streamlining $\mathcal{B}$. This process removes all client-specific data from the old contract, preserving only essential metadata. The result is a compact digital record, termed a "dysed subset" (D).

A dysed subset *D* is defined as:

$$D \subseteq \mathcal{B}$$

If *D* contains a single contract $|D| = 1$, it is a stand-alone obsolete element. For larger subsets $|D| > 1$, Dysing reduces *D* to its minimal representation:

$$\text{Reduced}(D) = \{\min(D), \max(D)\} = \{x_k, x_{k+m}\}$$

Here, $min(D)$ and $max(D)$ are the first and last elements of *D*, respectively. Intermediate contracts $x_{k+1}, x_{k+2}, \ldots, x_{k+m-1}$ are permanently removed, but cryptographic bookmarks to prove the existence of the original sequence are still available on-chain.

Example: Consider a blockchain segment with contracts indexed from 1 to 10. If $D = \{x_2, x_3, x_4\}$, reducing brings this subset back to $\{x_2, x_4\}$, while a validator is able to cryptographically verify that x_3 existed between x_2 and x_4. This mechanism achieves two goals:

1. *Efficient storage:* Reduces redundant data while maintaining referential integrity
2. *Deterministic validation:* Ensures that nodes can reconstruct the history of the chain

2.2 Relational Conditions for Multiple Dysed Subsets

Multiple dysed subsets $D_1, D_2, \ldots, D_n$ can coexist on $\mathcal{B}$, provided they meet strict relational conditions. For two subsets D_1 and D_2:

$$\max(D_1) < x_n < \min(D_2)$$

These conditions ensure that the isolated subsets are disjoint (they do not overlap) and that they are separated by at least one live element x_n.

2.3 Maximum Dysed Proportions

A dyse operation can be initiated until the blockchain $\mathcal{B}$ consist out of only the genesis block ($\mathcal{G}$) - the first block ever created – which is inherently non-dysable, and a single condensed dysed subset. In this scenario, all subsequent contracts were considered obsolete and merged into a single dysed subset. For example, if the blockchain $\mathcal{B} = (\mathcal{G}, x_2, x_3, \ldots, x_n)$, maximum dysing reduces $\{x_2, \ldots, x_n\}$ to $\{min(D), max(D)\} = \{x_2, x_n\}$ resulting in:

$$\mathcal{B} = \langle G \rangle \oplus \langle x_2, x_n \rangle$$

Here, ⊕ denotes an ordered sequence, in which the integrity of the series is maintained. It is critical that adjacent dysed subsets D_1 and D_2 merge into a single subset if they are not separated by a non-dysed element.

$$D_1 = \{x_k, x_{k+m}\}, \quad D_2 = \{x_{k+m+1}, x_{k+p}\} \quad \Rightarrow \quad D = \{x_k, x_{k+p}\}$$

This ensures that the blockchain only retains essential metadata and eliminates redundancy. The genesis block $\mathcal{G}$ remains unchangeable, anchoring the historical validity of the chain even under maximum disruption, which ensures the validation of the Global State.

3. Research on Scalability: Optimization and Cost Efficiency

Dysing is designed to improve blockchain scalability by enabling the removal of obsolete digital contracts, reducing storage requirements, and increasing transaction speed. These improvements are predicted to translate directly into enhanced optimization and cost efficiency for blockchain systems. To validate this claim, Scifer conducted a controlled empirical study with the following hypothesis about using the Dysing process:

1. Scalability: There will be no significant reduction in the size of the client wallet proportional to the percentage of obsolete contracts removed.

2. Cost-Efficiency – Single Contract Validation: There will be no significant improvement in the validation speed of a single remaining contract per percentage of obsolete contracts removed.

3. *Cost-Efficiency – State Validation:* There will be no significant improvement in the validation speed of the entire wallet state per percentage of obsolete contracts removed.

3.1 Experimental Setup

The study was conducted in an ideal configuration with a single client wallet containing 10 contracts of equal size (measured in MB). The Dysing process was applied incrementally:

1. **First dysed subset (33.3% of contracts)**: Contracts No. 2, 3 and 4 were removed.
2. **Second dysed subset (66.6% of contracts)**: Contracts No. 6, 7 and 8 were removed.
3. **Third dysed subset (70% of all contracts)**: Contract No. 5 was removed.

The final step, the removal of all content from the wallet, was omitted because it would theoretically reduce the size of the wallet to zero, making further analysis irrelevant.

3.2 Optimization Findings

The primary measure of enhanced optimization was the reduction in wallet size after each Dysing step. Removing digital contracts from the wallet should result in a proportional reduction in wallet size that closely correlates with the percentage of contracts dysed.

3.2.1 Measure enhancement in optimization

1. Wallet total size in MB →				After 33.3% Dysed (Contracts No 2, 3, 4)		After 66.6% Dysed (Contracts No 6, 7, 8)		After 70% Dysed (Contracts No 5)	
Test No	Contract size in MB	Amount	Wallet	Wallet Size in MB	Decrease %	Wallet Size in MB	Decrease %	Wallet Size in MB	Decrease %
1	0.091961	10	0.945807	0.669921	29.17%	0.394035	58.34%	0.302073	68.06%
2	0.126733	10	1.293537	0.913332	29.39%	0.533127	58.78%	0.406392	68.58%
3	0.534902	10	5.375227	3.770515	29.85%	2.165803	59.70%	1.630899	69.66%
4	1.008517	10	10.111395	7.085832	29.92%	4.060269	59.85%	3.051748	69.82%
5	5.321272	10	53.238927	37.275105	29.98%	21.311283	59.97%	15.990009	69.96%
6	11.187318	10	111.899387	78.337427	29.99%	44.775467	59.99%	33.588147	69.98%
7	17.292263	10	172.948837	121.072042	29.99%	69.195247	60.00%	51.902982	69.99%
8	21.636643	10	216.392637	151.482702	29.99%	86.572767	60.00%	64.936122	69.99%

Mean: Approximately 69.34% Standard Deviation: Approximately 0.85%

- *Average reduction:* About 69.34% after 70% of contracts were dysed.
- *Standard deviation:* About 0.85%, signals a high consistency across different contract sizes.
- *Conclusion:* The data reveals a consistent decrease in storage utilization across all contract sizes, confirming that Dysing effectively and immediately enhances optimization of storage allocation.

3.3 Cost-efficiency Findings

To evaluate cost efficiency by means of Dysing, two critical performance tests measuring validation speeds where conducted. The first is Single Contract Validation, assessing the processing time for individual contract validation, a key metric for transaction throughput. The second test is State Validation, measuring the system's ability to verify the complete current state of a client's wallet, reflecting real-world operational load.

3.3.1 Validation Speed of a Single Contract

2. Validation speed of a single contract in seconds →				After 33.3% Dysed (Contracts No 2, 3, 4)		After 66.6% Dysed (Contracts No 6, 7, 8)		After 70% Dysed (Contracts No 5)	
Test No	Contract size in MB	Amount	Wallet	Validation Speed Single Contract	Decrease %	Validation Speed Single Contract	Decrease %	Validation Speed Single Contract	Decrease %
1	0.091961	10	0.029305	0.031994	-9.18%	0.028551	2.57%	0.021003	28.33%
2	0.126733	10	0.236667	0.211529	10.62%	0.216694	9.45%	0.168242	28.91%
3	0.534902	10	0.224701	0.178544	20.54%	0.190951	15.79%	0.144224	35.82%
4	1.008517	10	0.206476	0.219886	-6.49%	0.185275	10.05%	0.153305	25.75%
5	5.321272	10	0.256184	0.246023	3.97%	0.231361	9.79%	0.181895	29.00%
6	11.187318	10	0.336845	0.300139	10.90%	0.166237	50.35%	0.247043	26.66%
7	17.292263	10	0.426594	0.384468	9.87%	0.370361	13.17%	0.328035	23.10%
8	21.636643	10	0.456452	0.449559	1.51%	0.407725	9.70%	0.368874	19.19%

Mean: Approximately 27.19% Standard Deviation: Approximately 4.61%

- *Average improvement:* About 27.19% after 70% of contracts were dysed.
- *Standard deviation:* About 4.61%, indicating relatively stable performance across different contract sizes.
- *Conclusion:* As the percentage of dysed contracts increases, validation speed for remaining contracts improves significantly.

3.3.2 Validation Speed of the State (All contracts inside a single wallet)

3. Validation speed of all Transactions in seconds →				After 33.3% Dysed (Contracts No 2, 3, 4)		After 66.6% Dysed (Contracts No 6, 7, 8)		After 70% Dysed (Contracts No 5)	
Test No	**Contract size in MB**	**Amount**	**Wallet**	**Validation Speed All Transactions**	**Decrease %**	**Validation Speed All Transactions**	**Decrease %**	**Validation Speed All Transactions**	**Decrease %**
1	0.091961	10	0.101486	0.087485	13.80%	0.079068	22.09%	0.049073	51.64%
2	0.126733	10	0.574348	0.491501	14.43%	0.383311	33.26%	0.355649	38.08%
3	0.534902	10	0.699475	0.690092	1.34%	0.501648	28.34%	0.344368	50.77%
4	1.008517	10	0.671371	0.550955	17.93%	0.480459	28.49%	0.418705	37.63%
5	5.321272	10	0.531994	0.563459	-5.91%	0.548021	-3.17%	0.253121	52.42%
6	11.187318	10	0.595466	0.525944	11.68%	0.578477	3.63%	0.410281	31.10%
7	17.292263	10	0.796004	0.729722	8.33%	0.514638	35.62%	0.407178	48.84%
8	21.636643	10	0.795925	0.408696	48.65%	0.549907	31.33%	0.426267	46.44%

Mean: Approximately 44.62%   Standard Deviation: Approximately 7.43%

- *Mean improvement:* About 44.62% after 70% of the contracts were dysed.
- *Standard deviation:* About 7.43%, indicating variability likely due to external factors such as hardware or software limitations.
- *Conclusion:* Despite its higher variability, Dysing significantly speeds up State validation, improving overall system performance.

3.4 Discussion

The results confirm Dysing's scalability benefits:
Optimization: Near-linear storage reduction (69.34% vs. 70% removal) rejects the hypothesis for scalability.
Cost-Efficiency: 27.19% faster single-contract validation. 44.62% faster full-wallet validation.
Limitations: Minor deviations suggest external factors (e.g., hardware) may slightly influence validation speeds. Testing used idealized conditions; real-world blockchains may show more variability.

3.5 Conclusion

The results confirm Dysing's core demonstrate: improved scalability. The data consistently shows that as obsolete contracts are dysed, storage utilization significantly decreases, directly addressing the issue of optimization. Moreover, notable improvements in both the validation speed of individual contracts and the overall State of the wallet indicate enhanced system performance reducing transaction costs. By providing a mechanism for secure and systematic removal of unnecessary data, Dysing emerges as a practical solution for making blockchain technology more scalable, optimizing resource utilization and cost-efficiency, and ultimately making it more suitable for widespread, everyday applications.

4. Security measurements

Dysing's theory for improving blockchain scalability requires unique security measures. While the design preserves chain integrity and enables dynamic data management, robust security measures are essential to prevent unauthorized changes, ensure cryptographic validity and maintain system-wide consistency. This section describes the Dysing framework, which addresses critical complications such as owner-controlled dysing and dysed subset validation. The implementation of these security protocols ensures the trustworthiness of the entire ecosystem while maintaining the decentralized principles that underpin blockchain technology. By establishing comprehensive safeguards against potential vulnerabilities, Dysing creates a resilient framework that can withstand various attack vectors while still delivering on its promise of improved efficiency.

4.1 Owner-Controlled Dysing

One of the fundamental security principle is that only the owner of a contract can initiate the dyse. This ensures that unauthorized parties cannot tamper with active or old contracts. Ownership claims are validated by on-chain evidence, such as nonce-based transaction histories/digital contract bonds, which ensure that no entity can revoke legitimate ownership.

4.2 Validation of Dysed Subsets

Dysed subsets must be verifiable to prevent manipulation or fraudulent modifications. Dysing uses the main cryptography protocol of the blockchain and Historical Data verification.

4.2.1 PoW anchored integrity

Dysing uses a cryptographic evidence protocol to validate dysed subsets. Each subset is captured in a digital contract. This method ensures that Dysing operations cannot be changed by copy-paste actions without detectable inconsistency.

4.2.2 Historical Data verification

Using canonical blockchain history, the metadata of the dysed contracts is accessible to verify that the dysed subsets match the immutable ledger record. This prevents attacks where malicious actors attempt to insert illegal contracts.

Dysing's security architecture balances decentralization with strict security. The integration of canonical history, cryptographic evidence and ownership controls ensure that Dysing improves blockchain efficiency without compromising the immutability of the contract or trust. These measures are in line with the decentralized ethos of Web3.0 and respond to the Semantic Web's emphasis on structured, verifiable data, positioning Dysing as a secure solution for next-generation blockchain ecosystems.

5. Dysing's Innovative Approach to Scalability

Through its unique approach to improve blockchain scalability, Dysing's proposal directly addresses the technology's "cost efficiency" and "optimization" pain points. By managing storage and increasing validation handling, Dysing aligns layer 1 blockchains with the efficiency requirements of Web3.0, where digital contracts and DApps require leaner, faster and more adaptable infrastructures.

5.1 Cost Efficiency: Reducing Financial Overheads

The decentralized nature of blockchain requires that ledger data be stored in perpetuity, ultimately creating network bloating. Dysing addresses this inefficiency. The removal of outdated contracts from the chain simplifies State validation processes, as nodes can process transactions faster and require less computing power to reach consensus. This efficiency lowers transaction costs.

In addition to State validation, Dysing enhances transaction validation. By optimizing the process of verifying and adding new transactions to the blockchain, Dysing ensures that transactions are handled with reduced computational effort. This leads to faster processing times and further cost savings, benefiting both enterprises and individual users.

5.2 Optimization: Enabling Dynamic Management of Resources

Dysing offers an innovative approach to fundamental data storage limitation by allowing users the ability to down scale their wallets. By using Dysing's DAR principle, customers themselves have the ability to remove unnecessary contracts directly from the blockchain. This proactive removal of redundant data leads to a tangible reduction in the ledger footprint, reducing computer load and decreasing energy consumption during the State validation process. These results meet the dynamic demands of real-world applications without sacrificing the core idea of decentralization. This feature of dynamic scalability ensures blockchain flexibility, allowing users to consistently preserve optimal performance levels, even as their operational requirements change over time. This crucial optimization is paramount to align blockchain technology with the ambitious vision of Web3.0, within which a seamlessly interconnected internet infrastructure can support a new generation of decentralized applications and services.

6. Application

Dysing aligns with both the decentralized, trustless concept of Web3.0 and the structured, machine-readable data frameworks of the Semantic Web through the following key points:

6.1 Strengthen Web3.0 with Scalability

Dysing directly address the critical challenge of blockchain data storage, offering a revolutionary approach to Web3.0 scalability. By enabling random and autonomous blockchain data management, Dysing drastically reduces Web3.0 storage requirements while maintaining the integrity of the blockchain. This streamlined approach accelerates both State and transaction validation processes, leading to significantly improved throughput. By solving the fundamental storage efficiency problem, Dysing eliminates a major bottleneck in Web3.0 adoption, creating leaner, faster DApps capable of supporting truly global-scale applications while promoting customer sovereignty over their digital assets and interactions.

6.2 Semantic Interoperability and Transient Digital Contracts

The framework for transient digital contracts enabled by Dysing can leverage semantic Web ontologies (e.g., RDF[12]/OWL[13]) to automate life-cycle management, ensure regulatory compliance and transform static ledgers into dynamic systems. New blockchain billing models such as wallet-based billing align with semantic Web agents and incorporate semantic triggers to manage resources autonomously and reflect context-aware workflows. This synergy improves trust and automation in decentralized systems.

6.3 Compliance and Data Integrity

Both visions prioritize accountability and compliance. Dysing ensures blockchain integrity while allowing users to autonomously remove data in adherence to semantic retention policies, such as semantic retention policies (e.g., GDPR [14]-compliant data expiration dates). This dual focus on decentralized trust and structured governance positions Dysing as a compliance layer for Web3.0 ecosystems, ensuring that data practices meet legal and semantic standards.

Dysing bridges the gap between the decentralized autonomy of Web3.0 and the machine-readable intelligence of the Semantic Web by embedding transient contracts and DAR principles in data life-cycle management. It optimizes blockchain's scalability while enabling semantic interoperability, reducing redundancy and mitigating compliance risks. Dysing unlocks through transient digital contracts cost-effective and trustless ecosystems.

7. Sources

1. Bitcoin: A Peer-to-Peer Electronic Cash System – Satoshi Nakamoto – 2008 – PDF

   https://bitcoin.org/bitcoin.pdf

2. Data redundancy of blockchain systems in construction projects – Weisheng Lu, Liupengfei Wu, Chen Chen – 2024 – PDF

   https://www.frontiersin.org/journals/built-environment/articles/10.3389/fbuil.2024.1355498/full

3. The Limits to Blockchain Scalability – Vitalik Buterin – Ethereum – 2021 – HTML

   https://vitalik.eth.limo/general/2021/05/23/scaling.html

4. Web 3.0: The Future of Internet – Wensheng Gan, Zhenqiang Ye, Shicheng Wan, Philip S. Yu – 2023 – PDF

   https://arxiv.org/pdf/2304.06032

5. The Semantic Web – Tim Berners-Lee, James Hendler, Ora Lassila – 2001 – PDF

   https://www-sop.inria.fr/acacia/cours/essi2006/Scientific%20American_%20Feature%20Article_%20The%20Semantic%20Web_%20May%202001.pdf

6. ĐApps: What Web 3.0 Looks Like – Gavin Wood – 2014 – HTML

   https://gavwood.com/dappsweb3.html

7. Smart Contracts building block for Digital markets – Nick Szabo – 1996 – PDF

   https://www.truevaluemetrics.org/DBpdfs/BlockChain/Nick-Szabo-Smart-Contracts-Building-Blocks-for-Digital-Markets-1996-14591.pdf

8. Ethereum: A Next-Generation Smart Contract and Decentralized Application Platform – Vitalik Buterin – 2014 – HTML

   https://ethereum.org/content/whitepaper/whitepaper-pdf/Ethereum_Whitepaper_-_Buterin_2014.pdf

9. Why we are building Cardano – Charles Hoskinson – IOHK – 2017 – PDF

   https://cryptopapers.info/assets/pdf/cardano.pdf

10. Robust and Fast Blockchain State Synchronization – Enrique Fynn, Ethan Buchman, Zarko Milosevic, Robert Soulé, Fernando Pedone – Yale – 2022 – PDF

https://www.cs.yale.edu/homes/soule/pubs/opodis2022.pdf

11. Blockchain for decentralization of internet – Javad Zarrin, Hao Wen Phang, Lakshmi Babu Saheer, Bahram Zarrin – NIH – 2020 – PDF

https://pmc.ncbi.nlm.nih.gov/articles/PMC8122205/pdf/10586_2021_Article_3301.pdf

12. RDF 1.1 XML Syntax – Fabien Gandon, Guus Schreiber and Dave Beckett – W3C – 2004 – HTML

https://www.w3.org/TR/rdf-syntax-grammar/

13. OWL Web Ontology Language – Deborah L. McGuinness en Frank van Harmelen – W3C – 2004 – HTML

https://www.w3.org/TR/2004/REC-owl-features-20040210/#s1.2

14. REGULATION (EU) 2016/679 – The European Parlement and the Council – 2016 – PDF

https://eur-lex.europa.eu/legal-content/EN/TXT/PDF/?uri=CELEX:32016R0679