Modular zk-rollup on-demand

Published in JNCA

Thomas Lavaur[1,2] Jonathan Detchart[1]

Jérôme Lacan[1] Caroline P. C. Chanel[1]

[1]ISAE-SUPAERO, Toulouse
[2]Université Toulouse III Paul Sabatier

JC2 2023

19/10/2023

1 Introduction

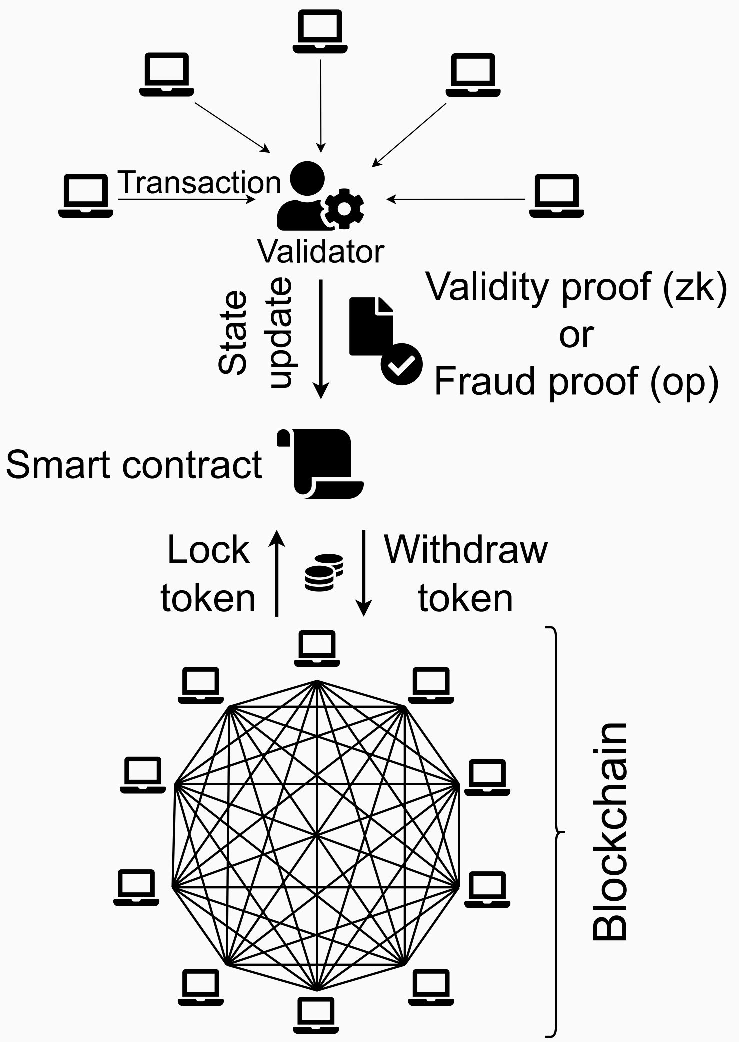
- zk-Rollups
- Motivations

2 Contribution

Rollups

Specifications[1]

- **A smart contract** stores the funds and accounts state of the rollup.
- Transaction **execution is centralized** by a validator.
- Needed data are stored on the blockchain.

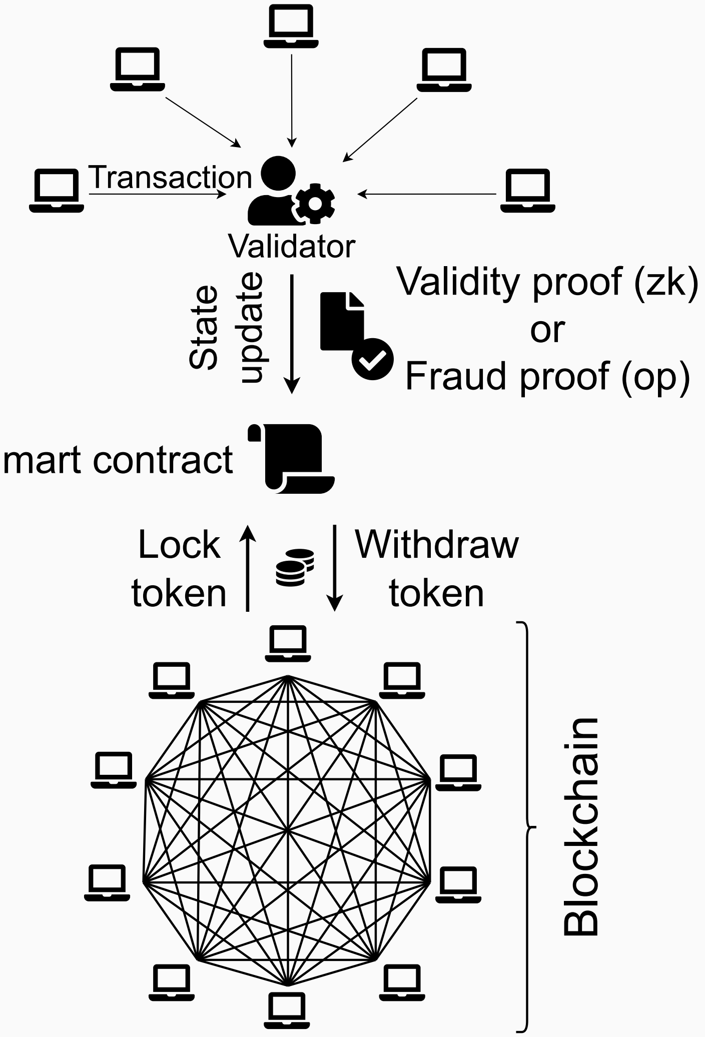


[1] Thibault, L. T., Sarry, T., Hafid, A. S. (2022). Blockchain scaling using rollups: A comprehensive survey. IEEE Access.

Rollups

Specifications[1]

- **A smart contract** stores the funds and accounts state of the rollup.
- Transaction **execution is centralized** by a validator.
- Needed data are stored on the blockchain.
- Correct transitions are **decentrally verified** by the smart contract via a zero-knowledge proof for zk-rollup or a fraud proof for optimistics.

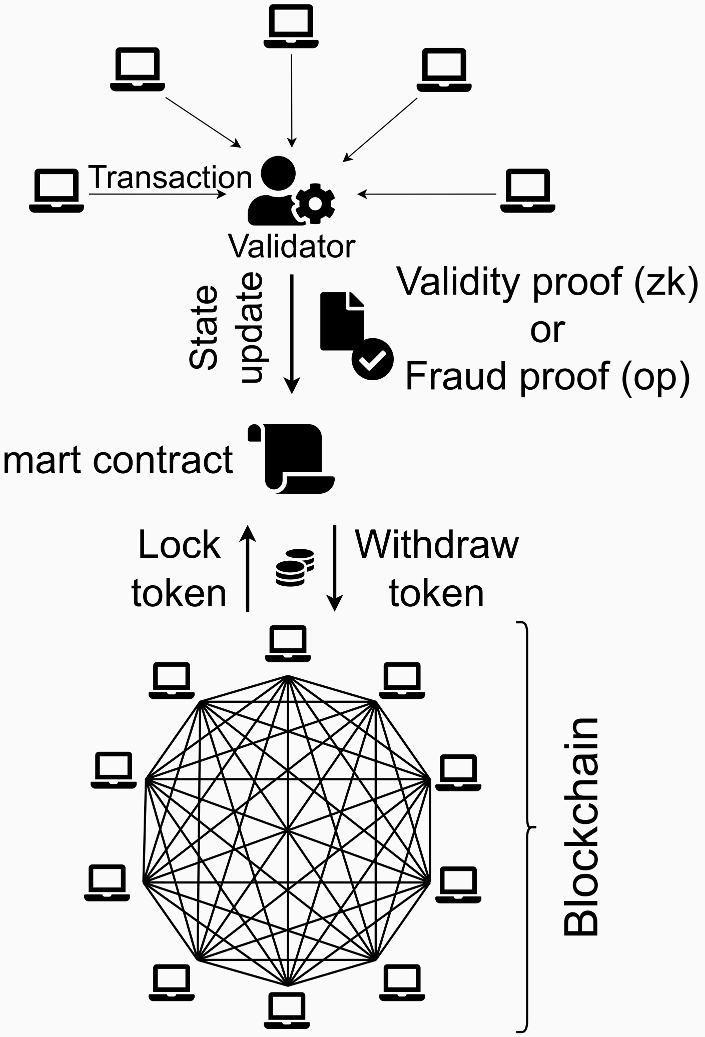


[1] Thibault, L. T., Sarry, T., Hafid, A. S. (2022). Blockchain scaling using rollups: A comprehensive survey. IEEE Access.

Rollups

Specifications[1]

- **A smart contract** stores the funds and accounts state of the rollup.
- Transaction **execution is centralized** by a validator.
- Needed data are stored on the blockchain.
- Correct transitions are **decentrally verified** by the smart contract via a zero-knowledge proof for zk-rollup or a fraud proof for optimistics.
- The **validator cannot** perform cryptographic **attacks but can censor** a transaction (only) on a zk-rollup.

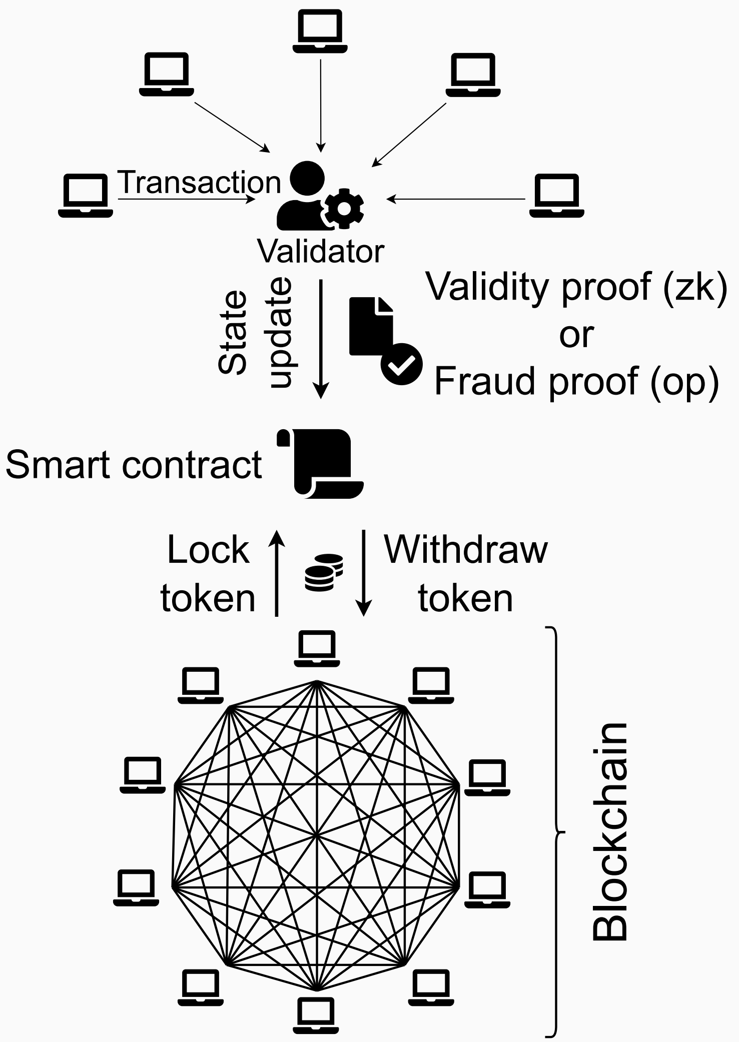


[1] Thibault, L. T., Sarry, T., Hafid, A. S. (2022). Blockchain scaling using rollups: A comprehensive survey. IEEE Access.

Rollups

Specifications[1]

- **A smart contract** stores the funds and accounts state of the rollup.
- Transaction **execution is centralized** by a validator.
- Needed data are stored on the blockchain.
- Correct transitions are **decentrally verified** by the smart contract via a zero-knowledge proof for zk-rollup or a fraud proof for optimistics.
- The **validator cannot** perform cryptographic **attacks but can censor** a transaction (only) on a zk-rollup.
- **Cheaper** transaction cost.



[1] Thibault, L. T., Sarry, T., Hafid, A. S. (2022). Blockchain scaling using rollups: A comprehensive survey. IEEE Access.

Motivations

Promizing

- Appear to be a **promising** way to **improve the scalability** of secure public blockchains while providing **possible privacy and cost savings**.
- Allow users to take **advantage of pre-established communities, pre-established cryptocurrencies** (and pre-audited security if they share the same smart contracts) while offering the **flexibility of private blockchains** designed for specific purposes.

Motivations

Promizing

- Appear to be a **promising** way to **improve the scalability** of secure public blockchains while providing **possible privacy and cost savings**.
- Allow users to take **advantage of pre-established communities, pre-established cryptocurrencies** (and pre-audited security if they share the same smart contracts) while offering the **flexibility of private blockchains** designed for specific purposes.

Issues

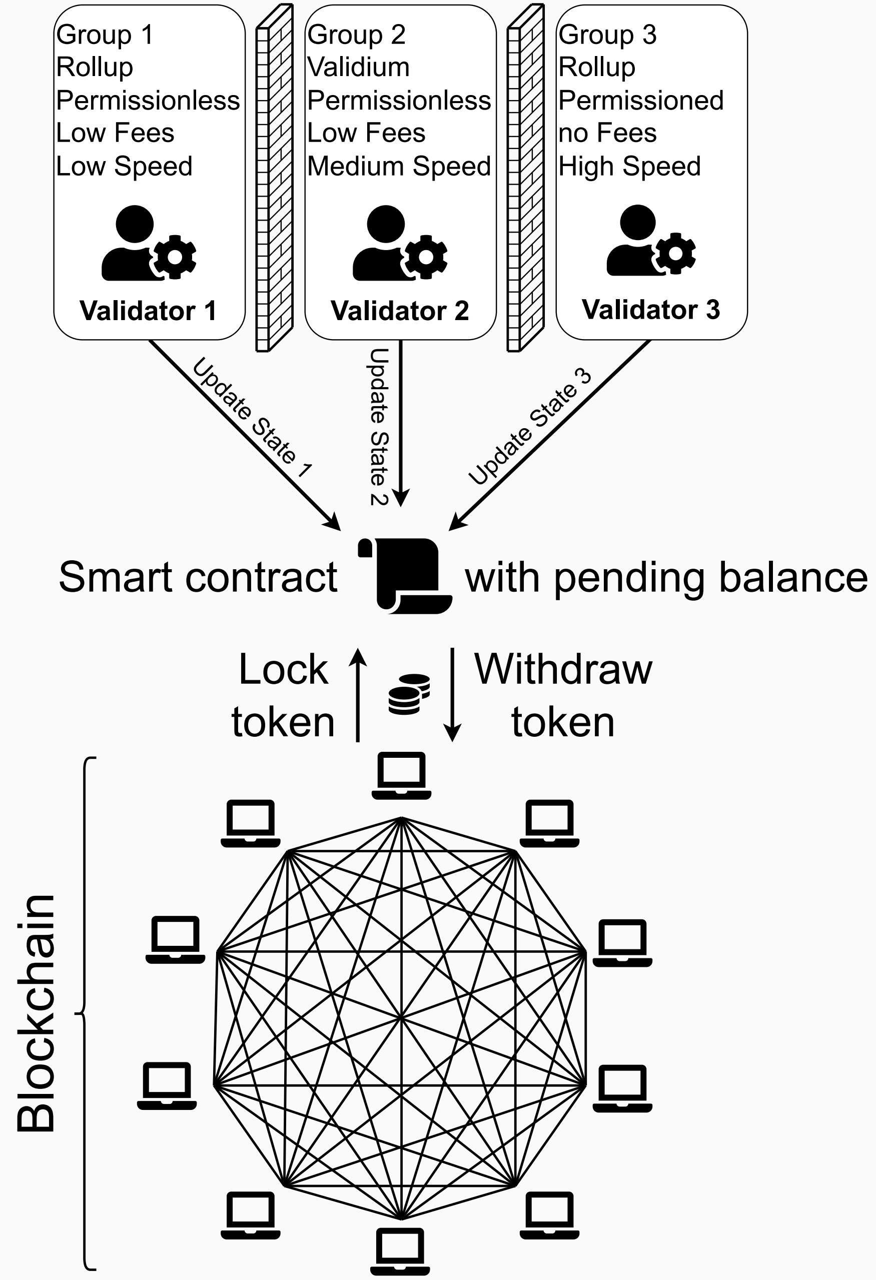
- One solution put forward by different companies is to extend these services providing **privacy and customization through layer 3s** built on top of their own rollup.
- **Sensitive data** have to be **publish** to a **centralized validator** that **can censorship transactions**.
- Even in a validium, **data privacy is concerning** if the validator is owned by an external entity.
- The **setup** of a zk-rollup can be **expensive** reducing the incentives for non-financial applications

1 Introduction

2 **Contribution**

- co-existing in smart contracts
- New transaction types
- Results

All rollups in the same contracts[2]

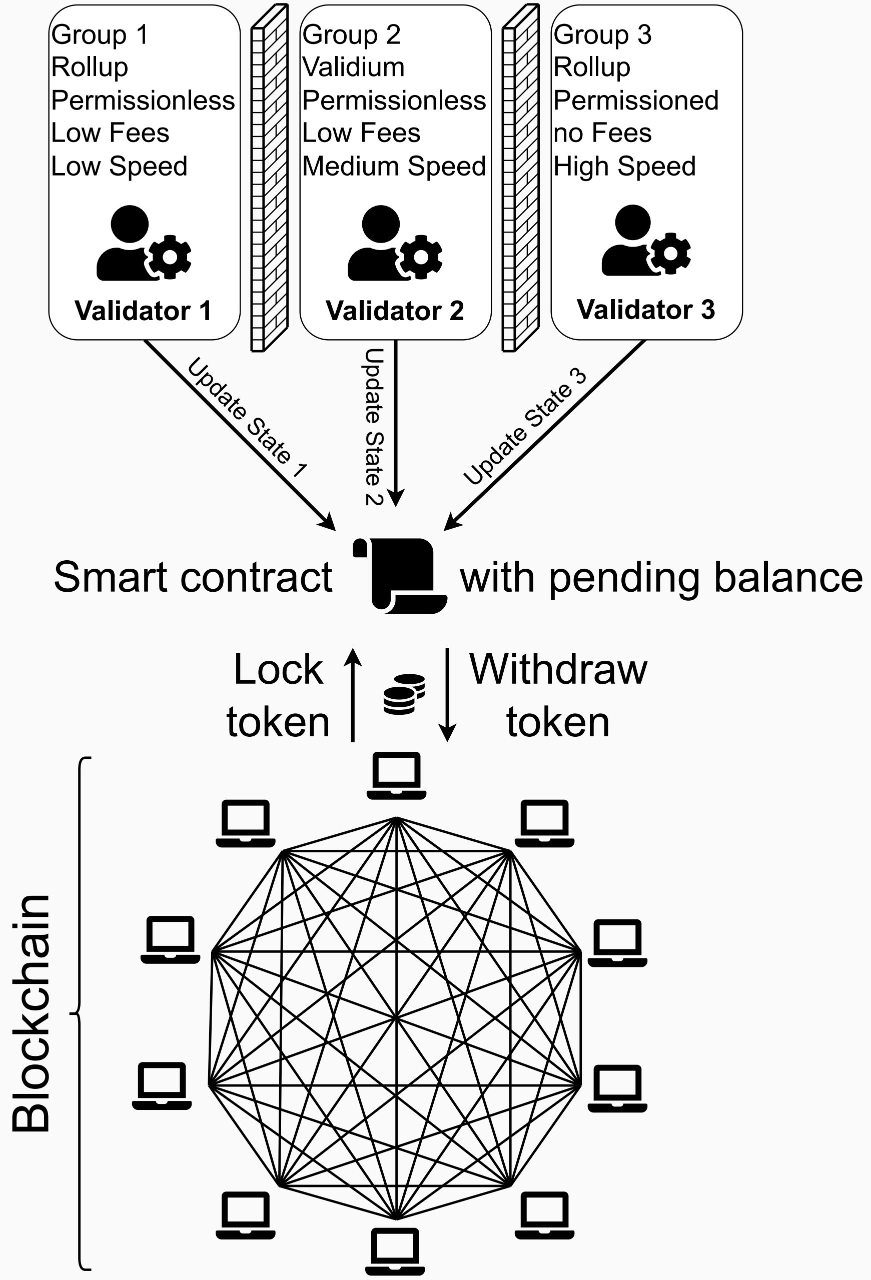


Proposition

- We propose allowing **several zk-rollups** to co-exist on the **same smart contract**, by including a group ID system into the smart contracts.

[2] Lavaur, T., Detchart, J., Lacan, J., Chanel, C. P. (2023). Modular zk-rollup on-demand. Journal of Network and Computer Applications, 103678.

All rollups in the same contracts[2]

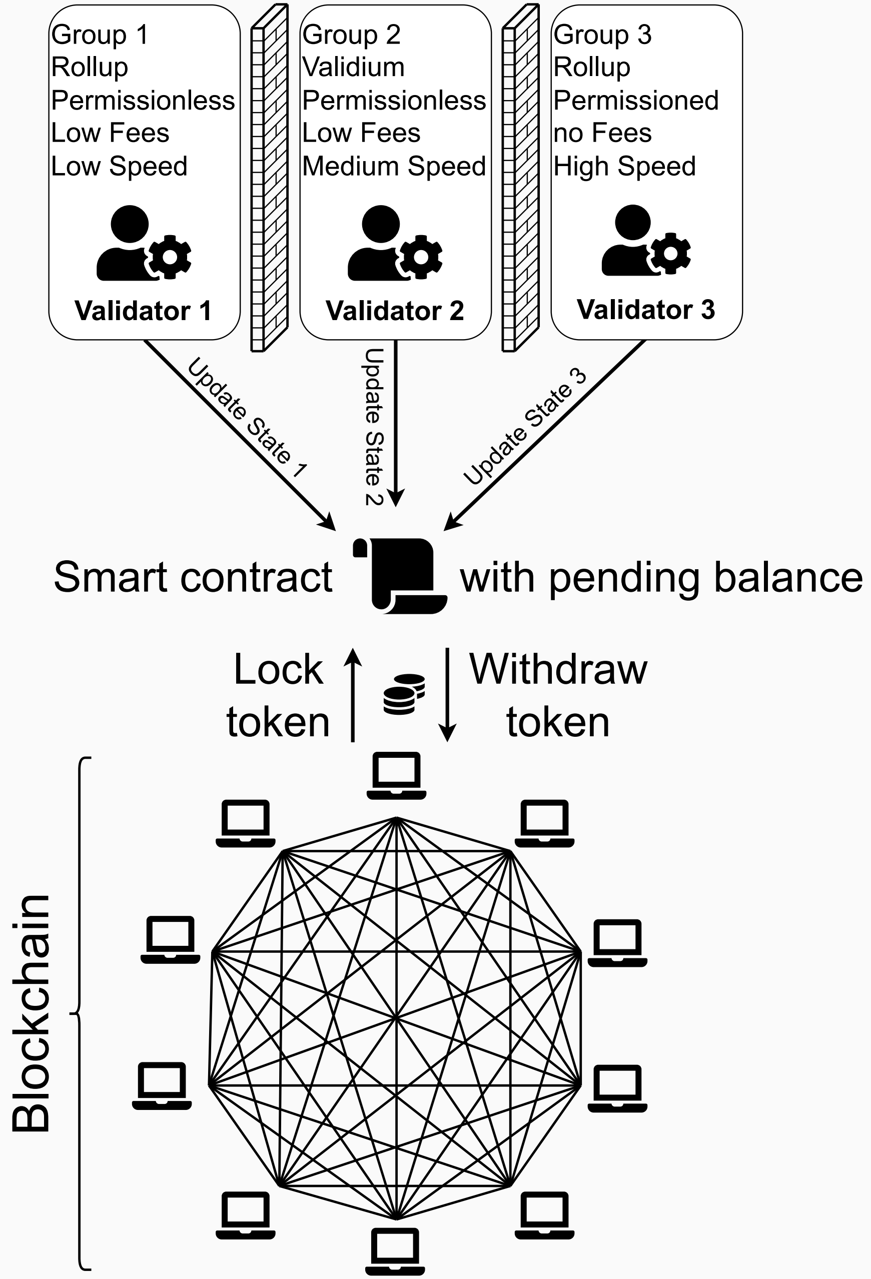


Proposition

- We propose allowing **several zk-rollups** to co-exist on the **same smart contract**, by including a group ID system into the smart contracts.
- The **functions** of the smart contracts are **shared** by the different groups, it is possible to choose a specific smart contract for proof checking in order to use different circuits or systems.

[2] Lavaur, T., Detchart, J., Lacan, J., Chanel, C. P. (2023). Modular zk-rollup on-demand. Journal of Network and Computer Applications, 103678.

All rollups in the same contracts[2]

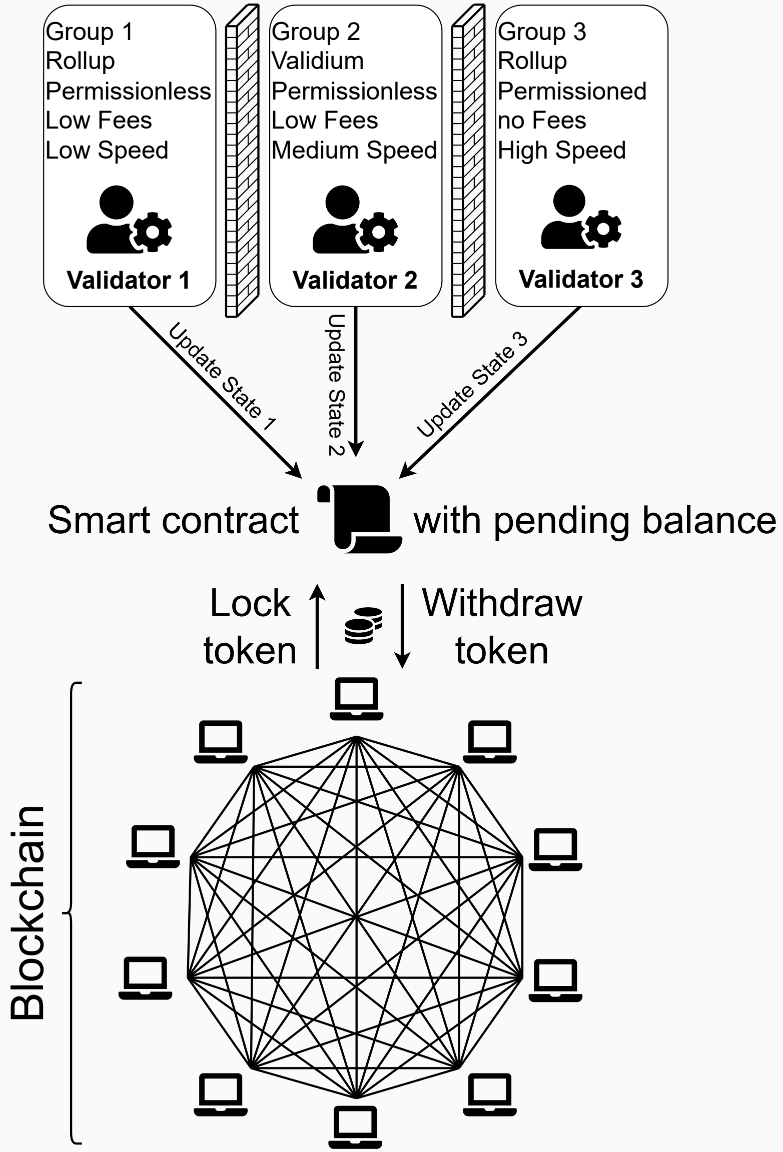


Proposition

- We propose allowing **several zk-rollups** to co-exist on the **same smart contract**, by including a group ID system into the smart contracts.
- The **functions** of the smart contracts are **shared** by the different groups, it is possible to choose a specific smart contract for proof checking in order to use different circuits or systems.
- Using **group-specific parameters**, the rollups would either be permissionless or permissioned, post data on-chain or off-chain and be optimistic or zk-rollup.

[2] Lavaur, T., Detchart, J., Lacan, J., Chanel, C. P. (2023). Modular zk-rollup on-demand. Journal of Network and Computer Applications, 103678.

All rollups in the same contracts[2]

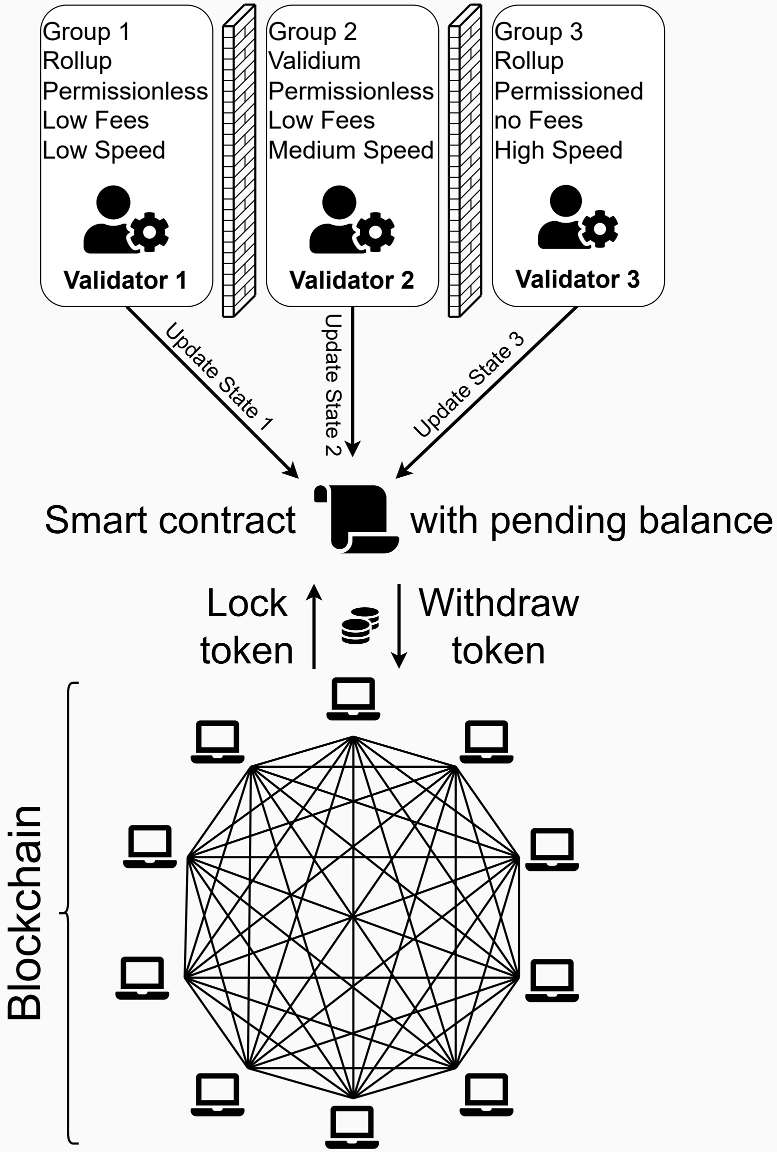


Benefits

- This drastically **reduces the cost** of subsequent **“deployments”** after an initial deployment.

[2] Lavaur, T., Detchart, J., Lacan, J., Chanel, C. P. (2023). Modular zk-rollup on-demand. Journal of Network and Computer Applications, 103678.

All rollups in the same contracts[2]

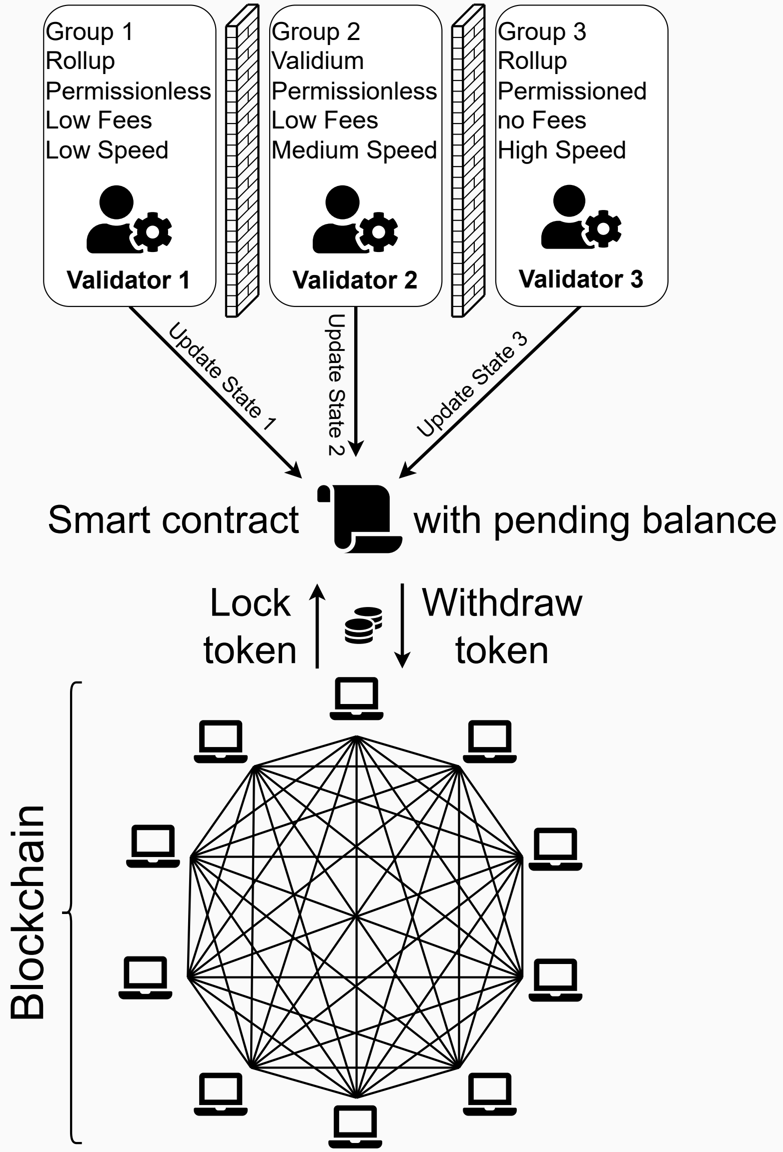


Benefits

- This drastically **reduces the cost** of subsequent **“deployments”** after an initial deployment.
- **Solves privacy issues** while democratizing **easy access to zk-rollups** for wider adoption.

[2] Lavaur, T., Detchart, J., Lacan, J., Chanel, C. P. (2023). Modular zk-rollup on-demand. Journal of Network and Computer Applications, 103678.

All rollups in the same contracts[2]



Benefits

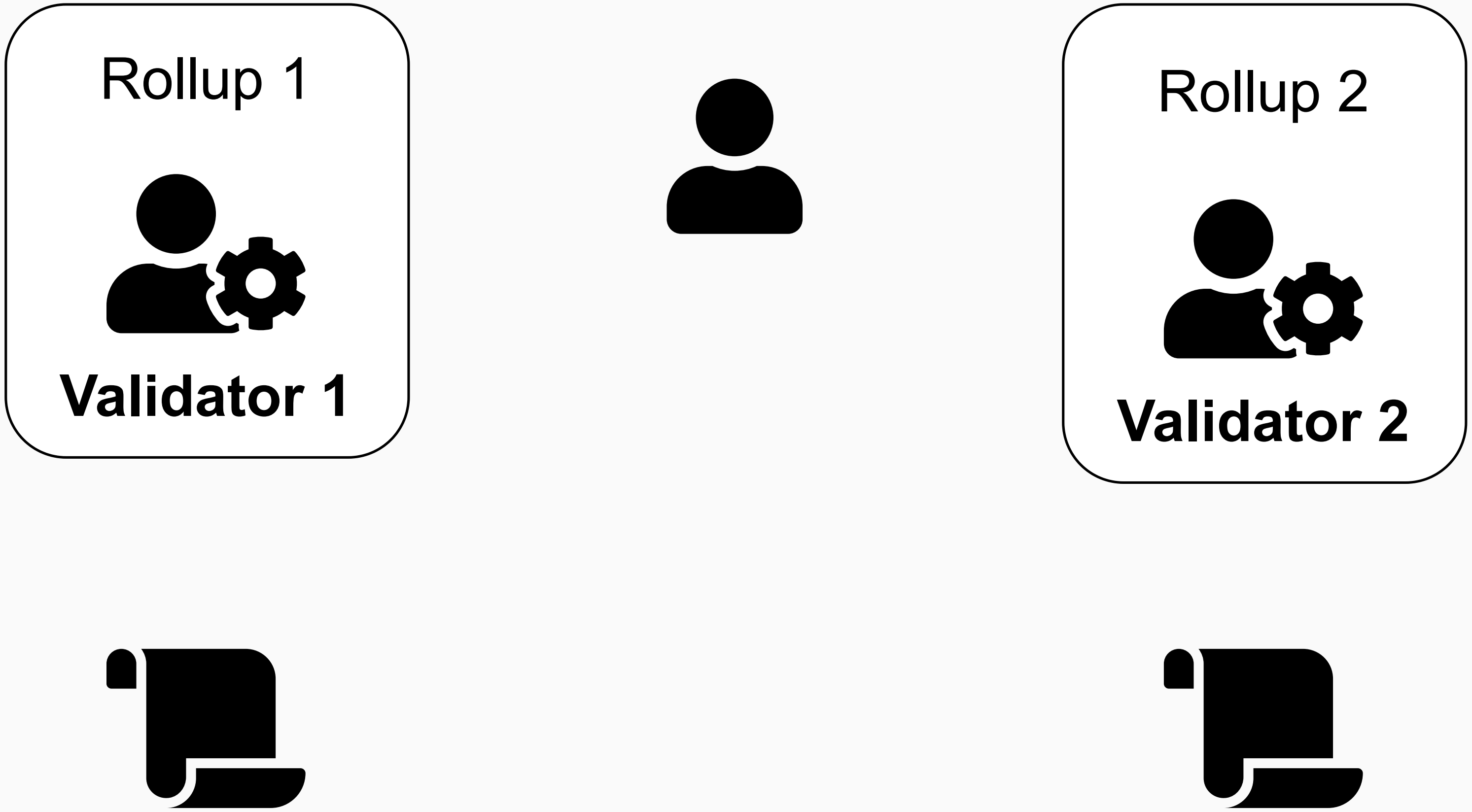
- This drastically **reduces the cost** of subsequent **"deployments"** after an initial deployment.
- **Solves privacy issues** while democratizing **easy access to zk-rollups** for wider adoption.
- Can be very interesting even if they are all public and permissionless, bringing different prices, finalities, systems and applications.

[2] Lavaur, T., Detchart, J., Lacan, J., Chanel, C. P. (2023). Modular zk-rollup on-demand. Journal of Network and Computer Applications, 103678.

New transaction types

Communications

Communications between rollups are actually **bulky, laborious and costly**. They have to return to user's address before being deposited elsewhere.



Rollup 1

Validator 1

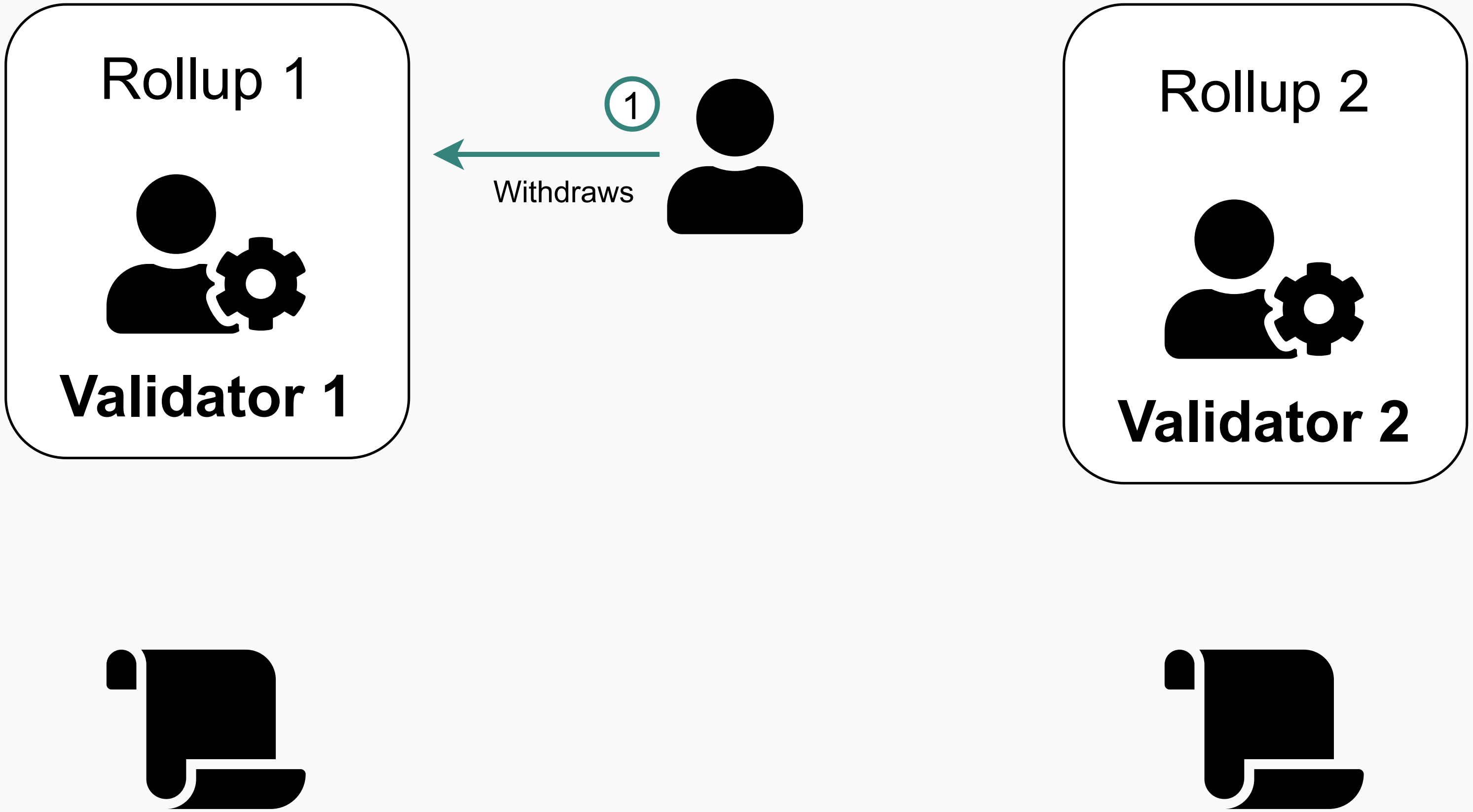
Rollup 2

Validator 2

New transaction types

Communications

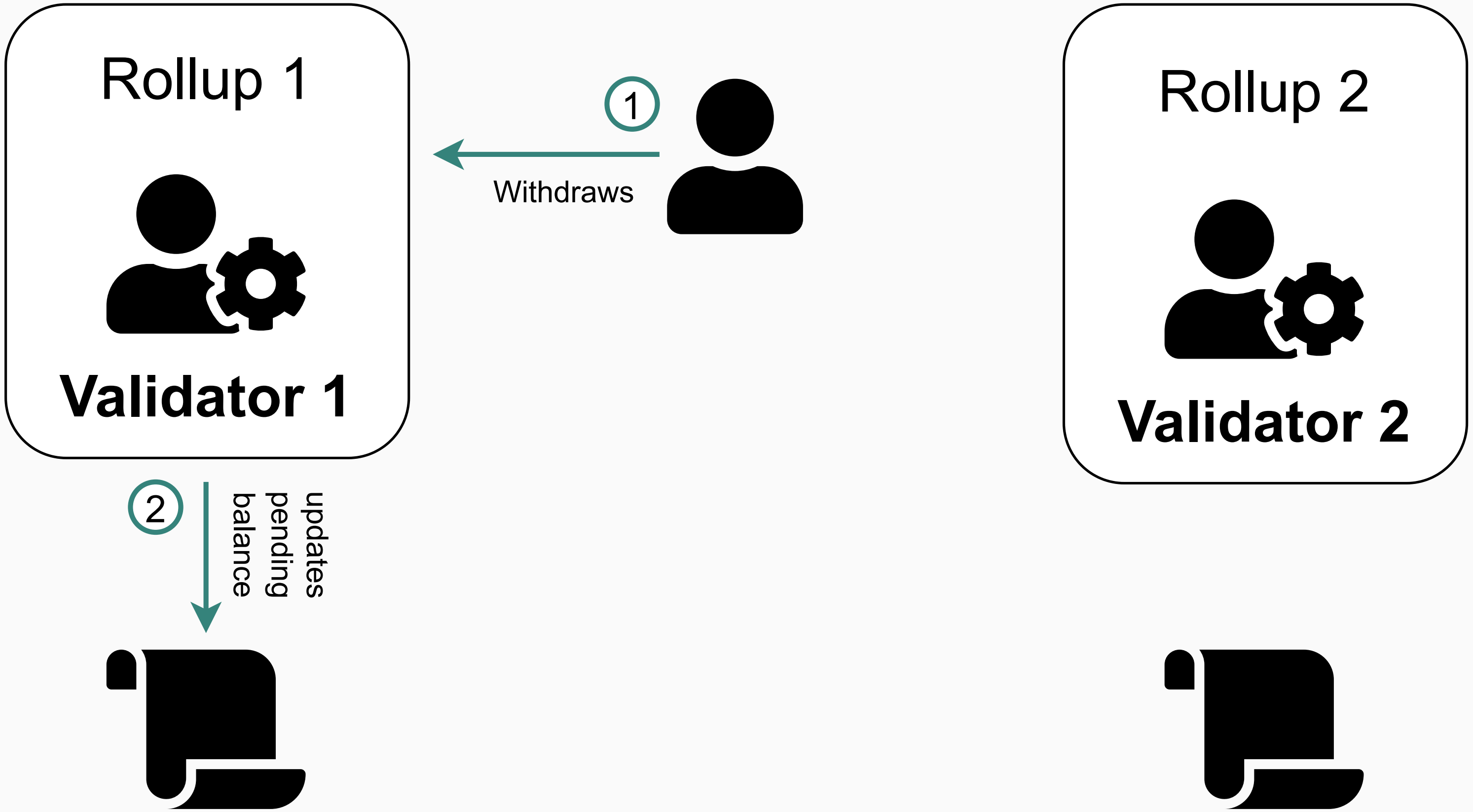
Communications between rollups are actually **bulky, laborious and costly**. They have to return to user's address before being deposited elsewhere.



New transaction types

Communications

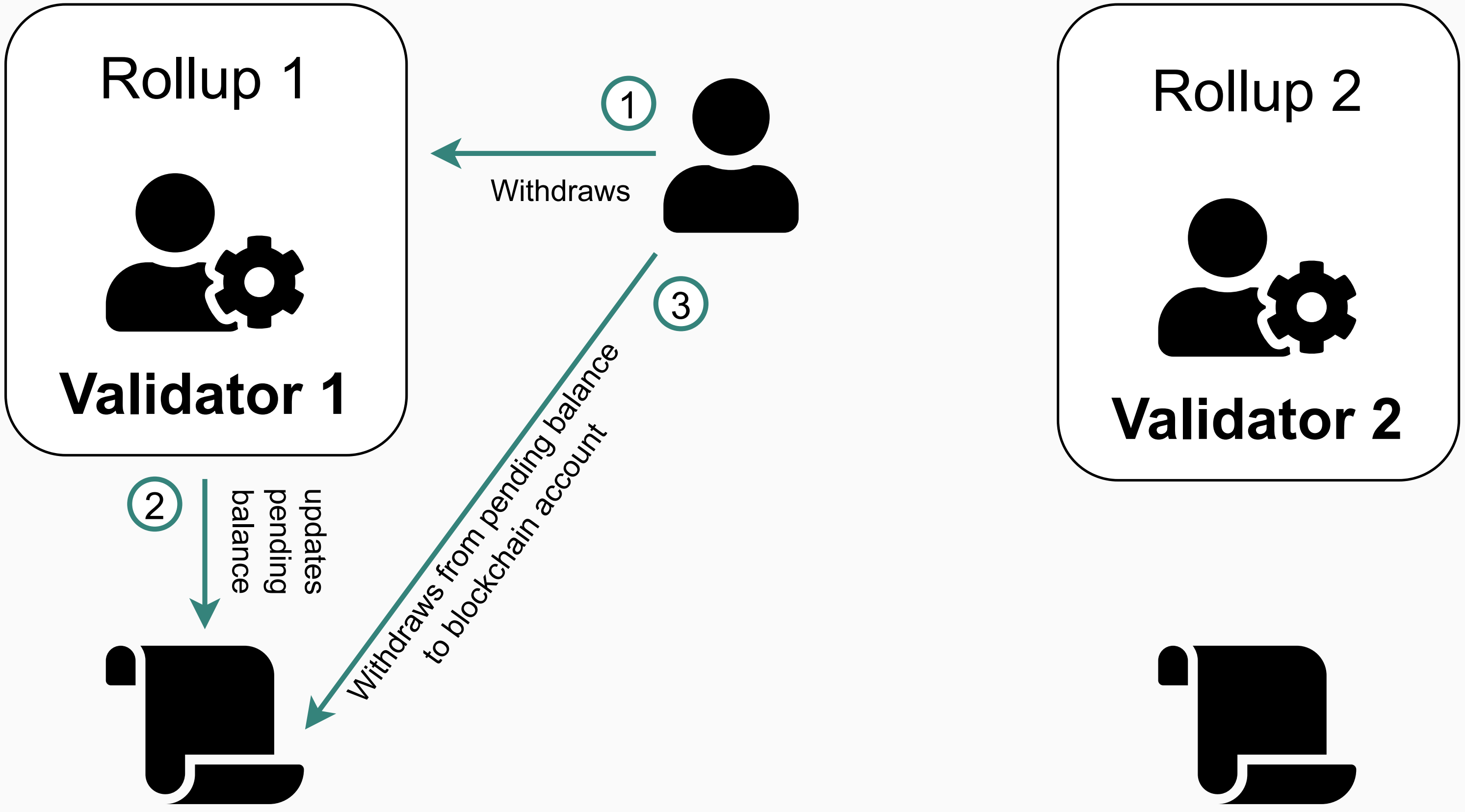
Communications between rollups are actually **bulky, laborious and costly**. They have to return to user's address before being deposited elsewhere.



New transaction types

Communications

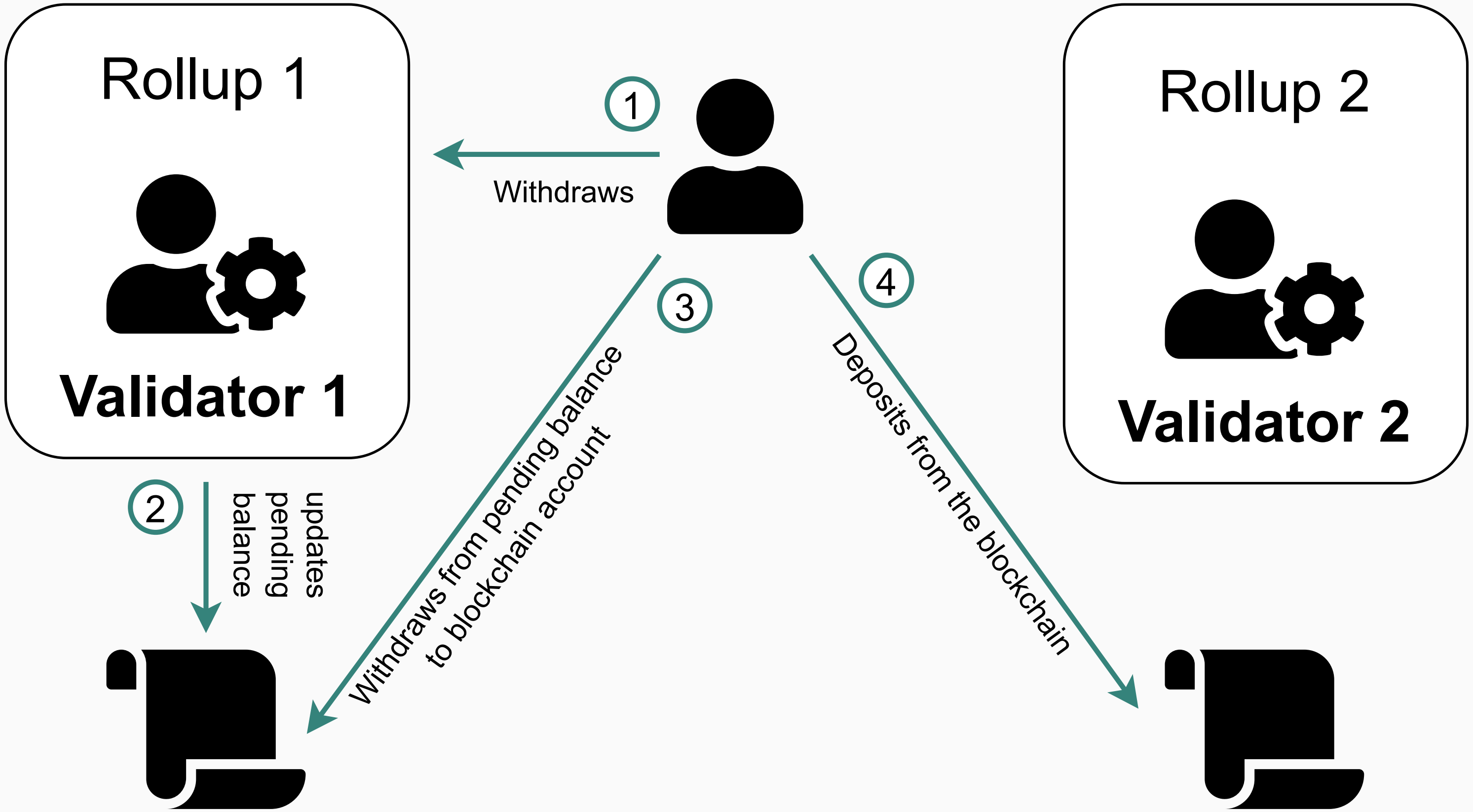
Communications between rollups are actually **bulky, laborious and costly**. They have to return to user’s address before being deposited elsewhere.



New transaction types

Communications

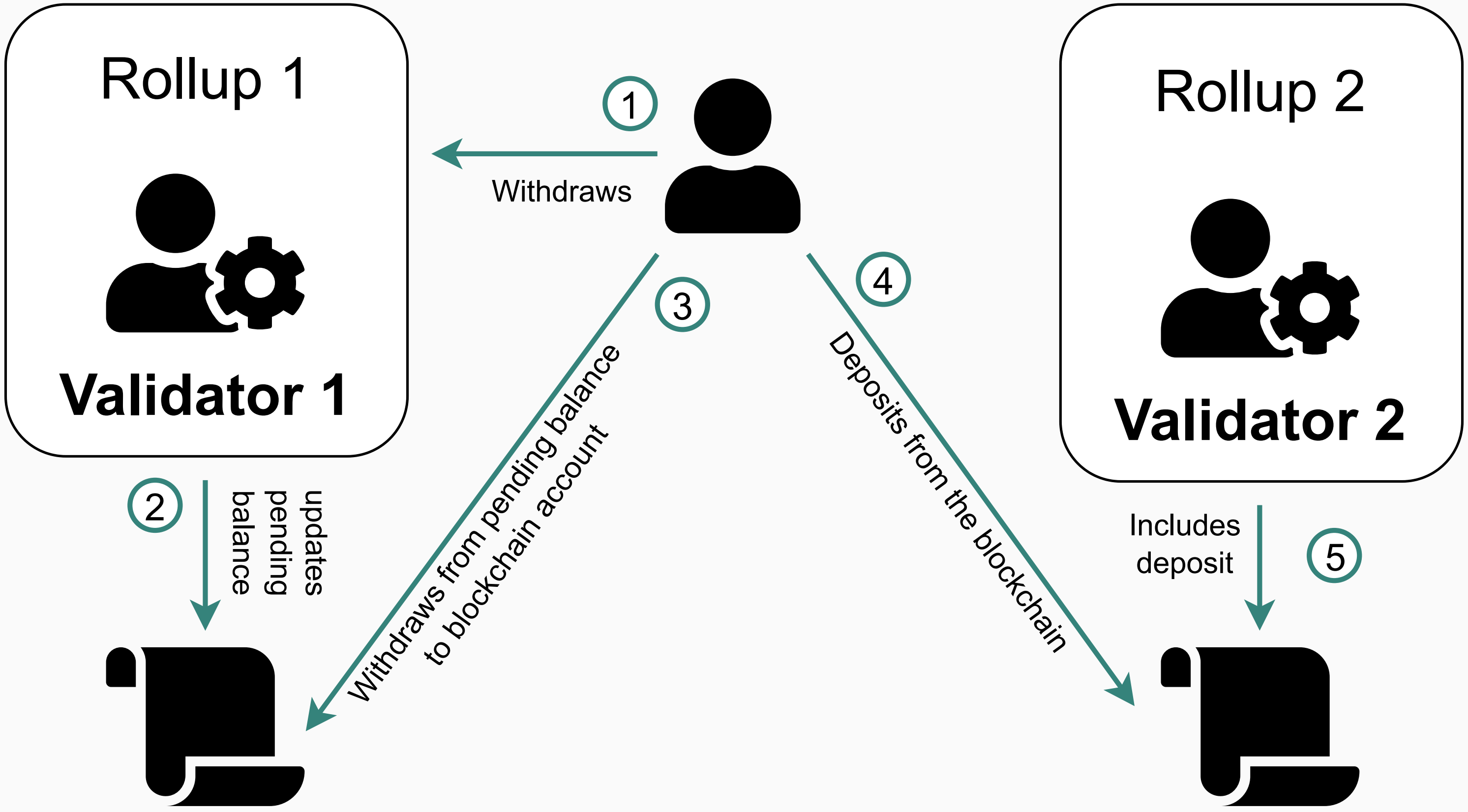
Communications between rollups are actually **bulky, laborious and costly**. They have to return to user's address before being deposited elsewhere.


Rollup 1
Validator 1
Rollup 2
Validator 2
1
Withdraws
2
updates pending balance
3
Withdraws from pending balance to blockchain account
4
Deposits from the blockchain

New transaction types

Communications

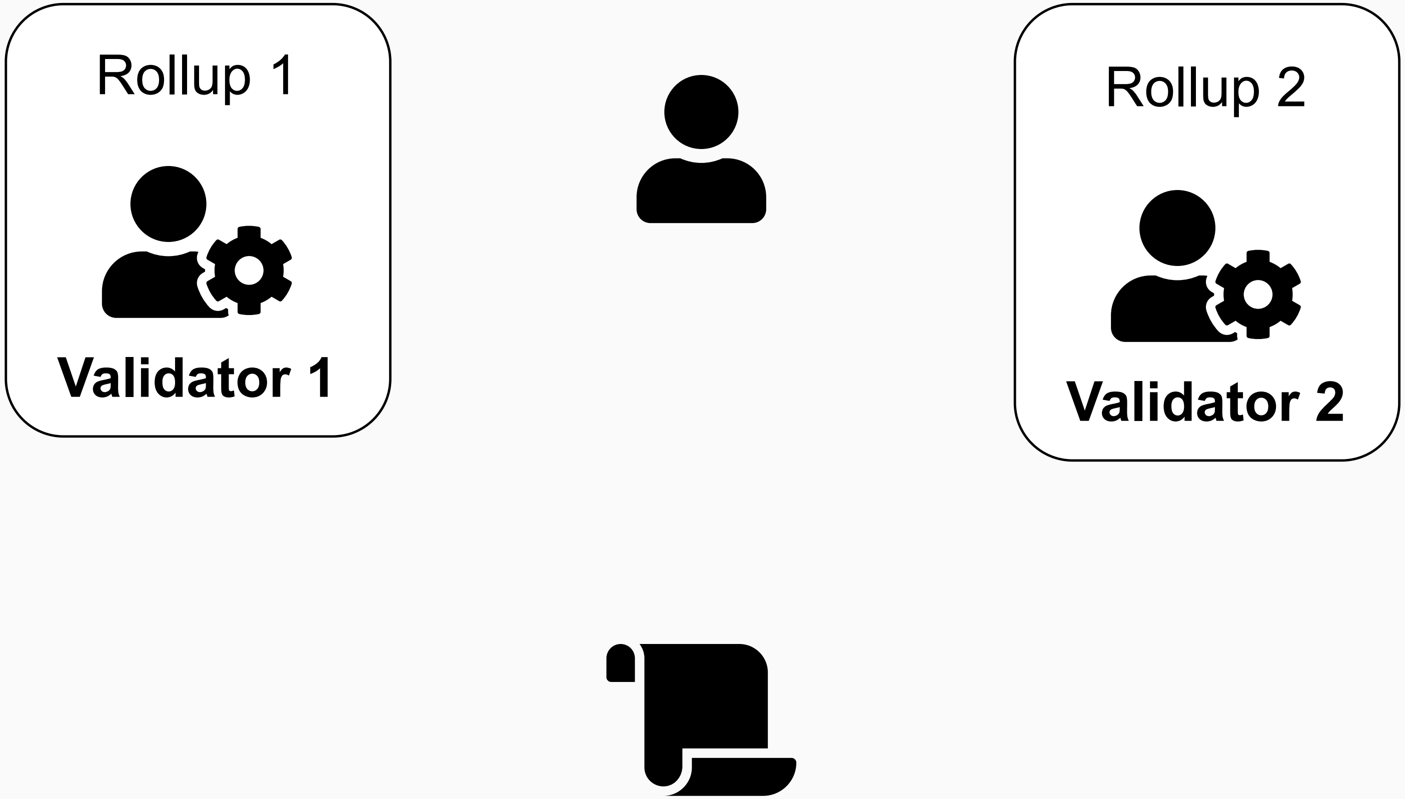
Communications between rollups are actually **bulky, laborious and costly**. They have to return to user's address before being deposited elsewhere.



New transaction types

New transaction types

We propose **adding new transaction types** that can be interpreted by smart contracts and act as a bridge between two rollups. The main idea is to easily allow users to send/receive information or funds from one group to another **without having to return** them **to the user's address**. This acts as a proof of burn to the underlying layer, automatically triggering a deposit request to the targeted rollup.



Rollup 1

Validator 1

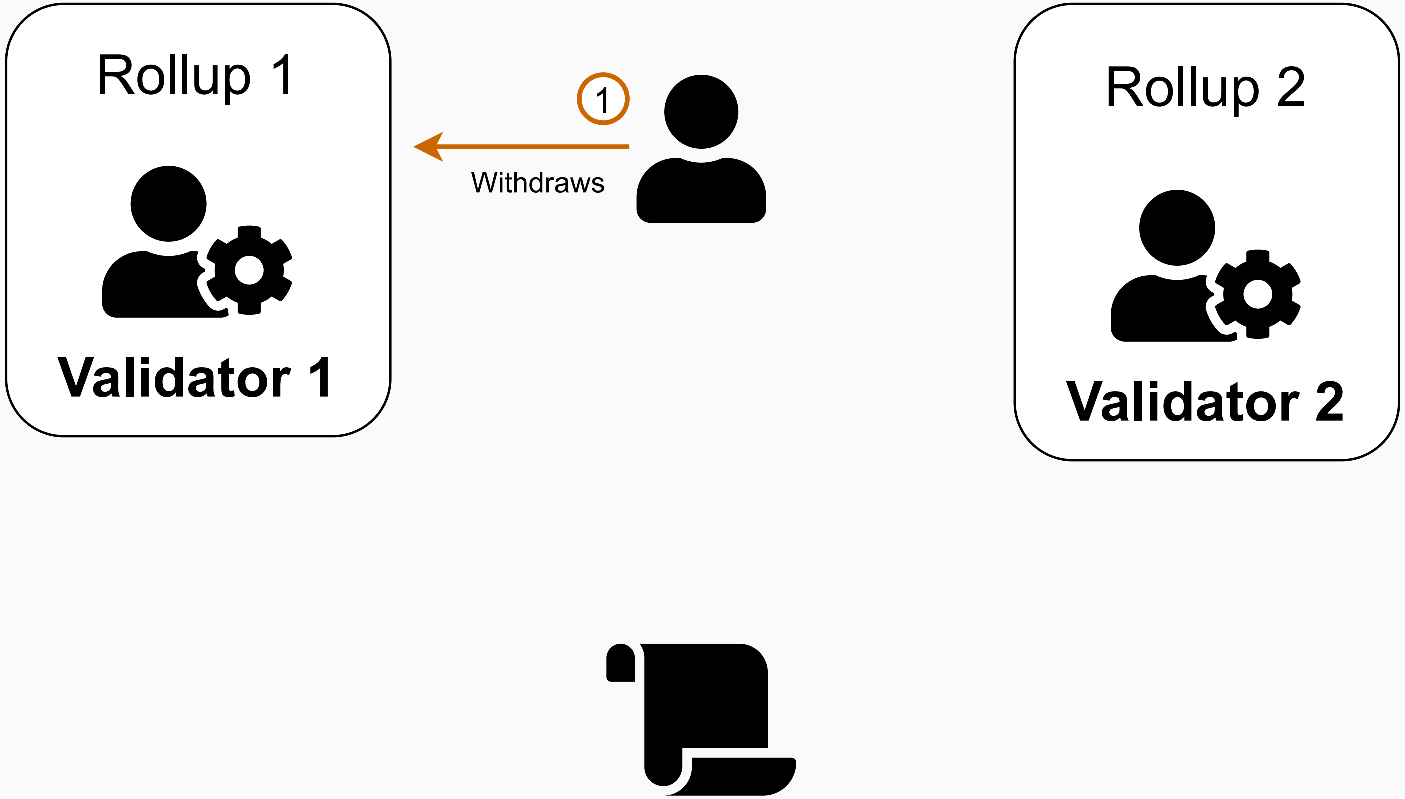
Rollup 2

Validator 2

New transaction types

New transaction types

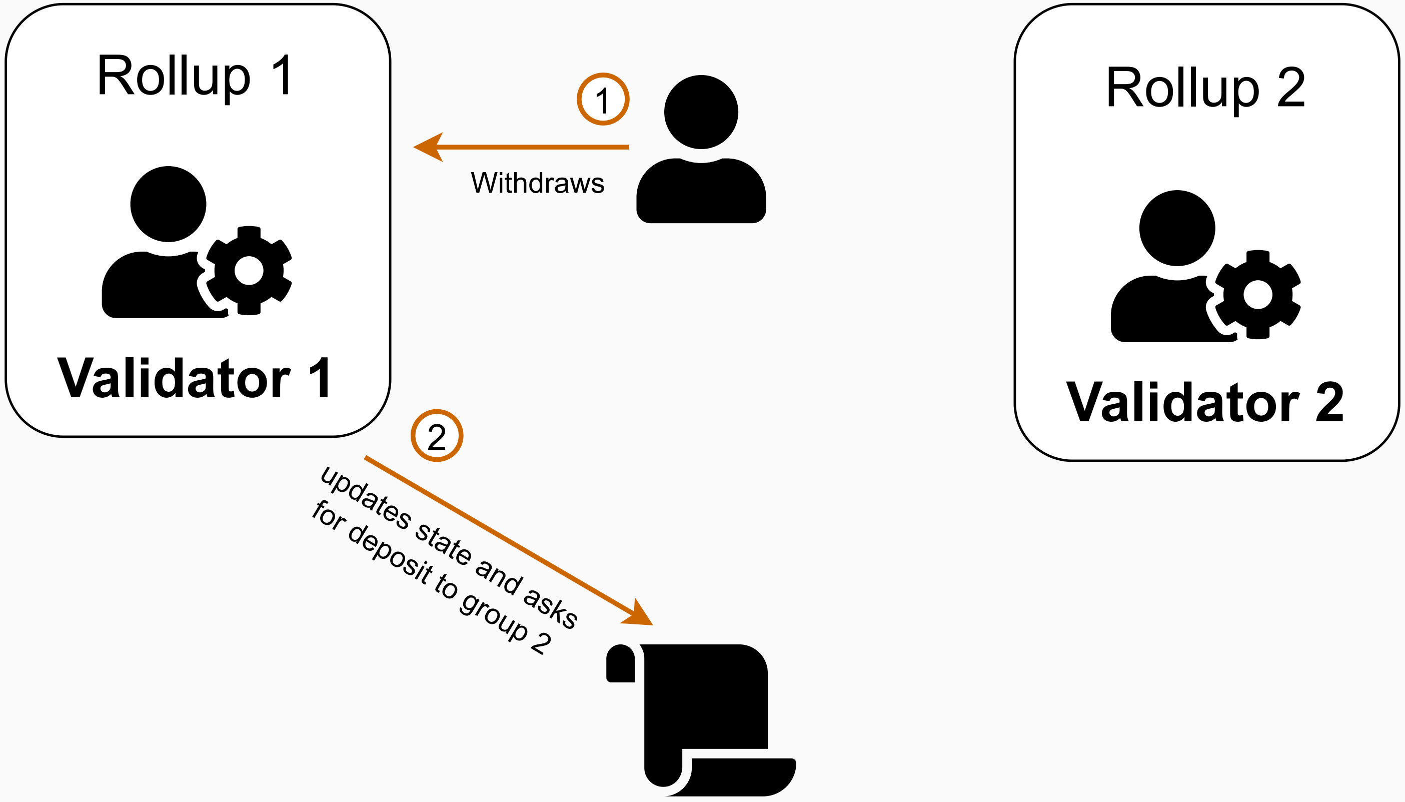
We propose **adding new transaction types** that can be interpreted by smart contracts and act as a bridge between two rollups. The main idea is to easily allow users to send/receive information or funds from one group to another **without having to return** them **to the user's address**. This acts as a proof of burn to the underlying layer, automatically triggering a deposit request to the targeted rollup.



New transaction types

New transaction types

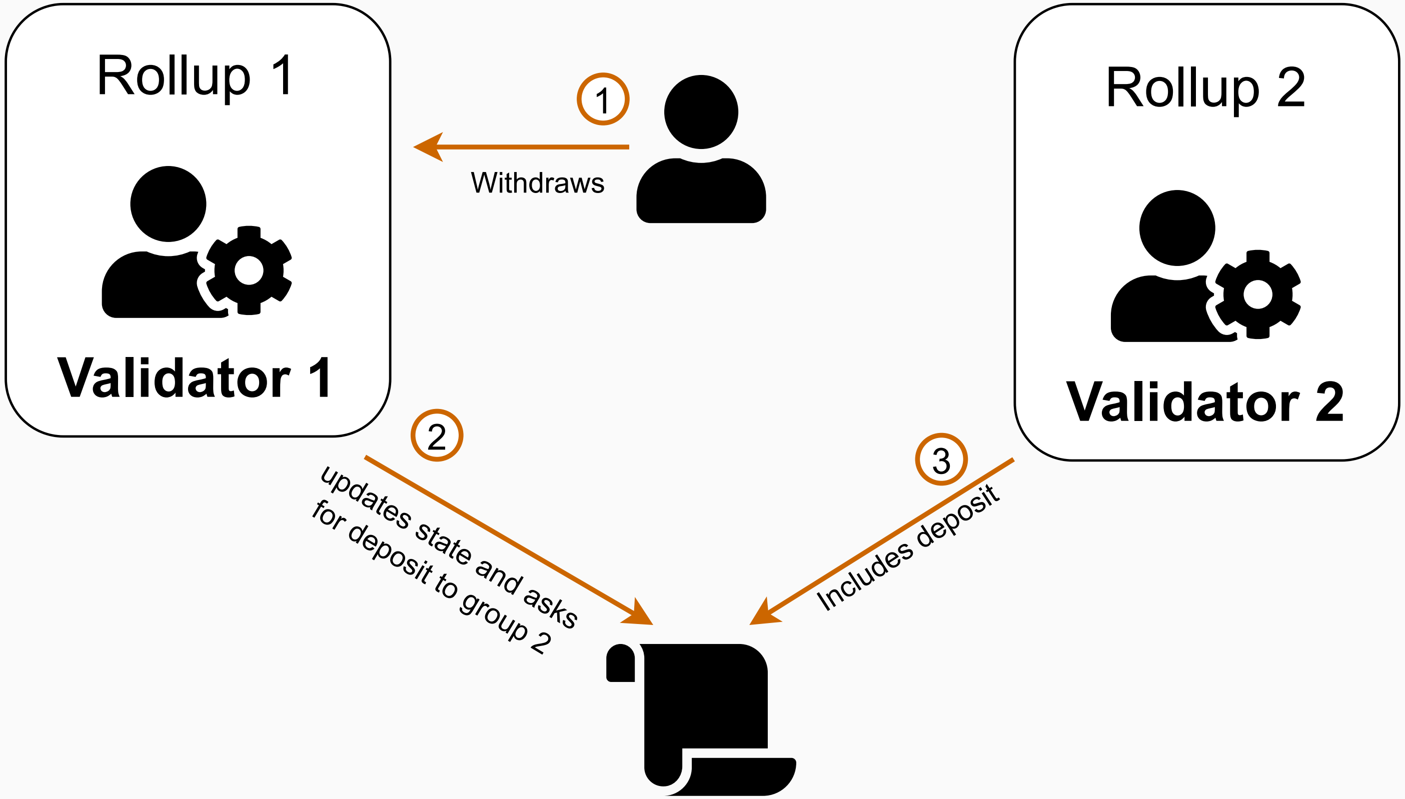
We propose **adding new transaction types** that can be interpreted by smart contracts and act as a bridge between two rollups. The main idea is to easily allow users to send/receive information or funds from one group to another **without having to return** them **to the user's address**. This acts as a proof of burn to the underlying layer, automatically triggering a deposit request to the targeted rollup.



New transaction types

New transaction types

We propose **adding new transaction types** that can be interpreted by smart contracts and act as a bridge between two rollups. The main idea is to easily allow users to send/receive information or funds from one group to another **without having to return** them **to the user's address**. This acts as a proof of burn to the underlying layer, automatically triggering a deposit request to the targeted rollup.



Results

Material

To compute the **proofs**, we used a computer with an Intel Xeon Platinum 8164 CPU and 400GB of RAM.

Overhead

The **addition** of the two new operation types, the inclusion of the group in the transactions and the modification of the public input create **almost no overhead** for the prover. The size of the first circuit only **increases** from **0.18%** for the smallest blocks **to 0.32%** for the largest blocks, and the difference in **proof time is not significant**.

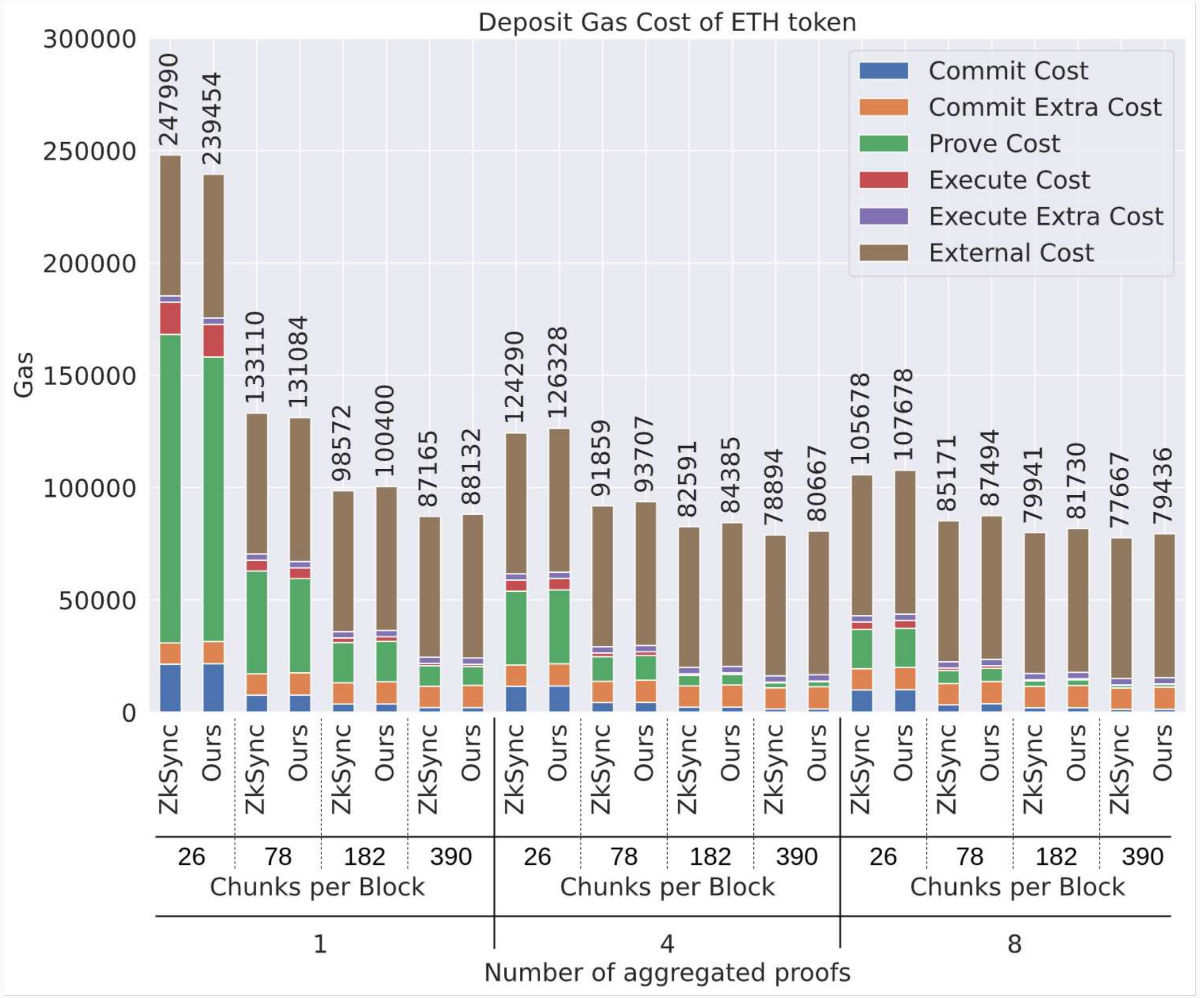
Block Chunk Size	zkSync		Our Proposition	
26	8,526,701c	71s	8,542,124c	71s
78	16,908,690c	142s	16,952,713c	144s
182	33,672,019c	289s	33,773,242c	289s
390	67,185,536c	588s	67,401,159c	588s

Table: First circuit comparison (c mean constraints, s seconds).

Results

Impact on existing transaction types

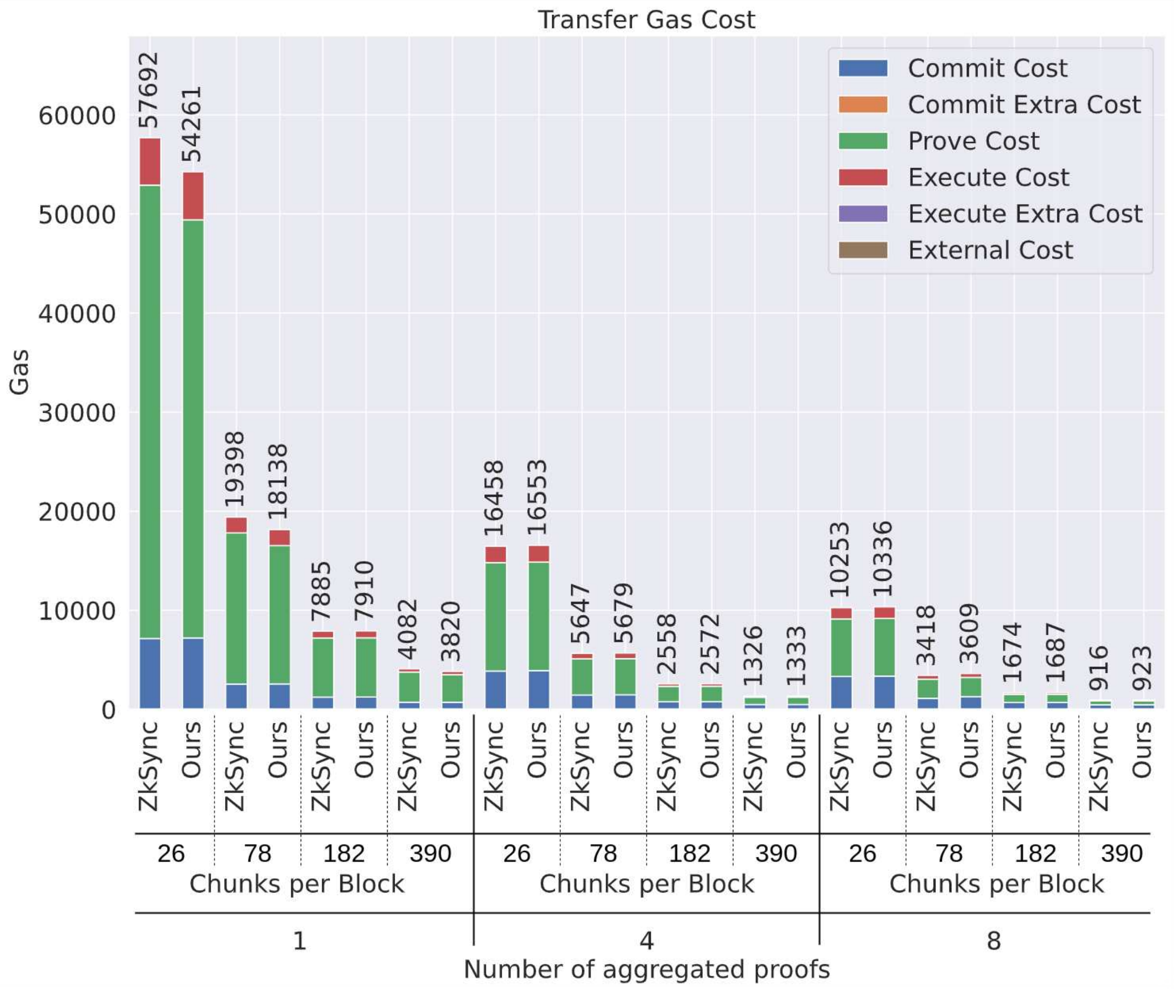
When block size is the largest and the number of aggregated proofs is the highest, the cost of a **deposit** is only **increased by 3%** for ERC20 and 2% for ETH, while **the rest** of the transactions only see their costs increase **by less than 1%**.



Results

Impact on existing transaction types

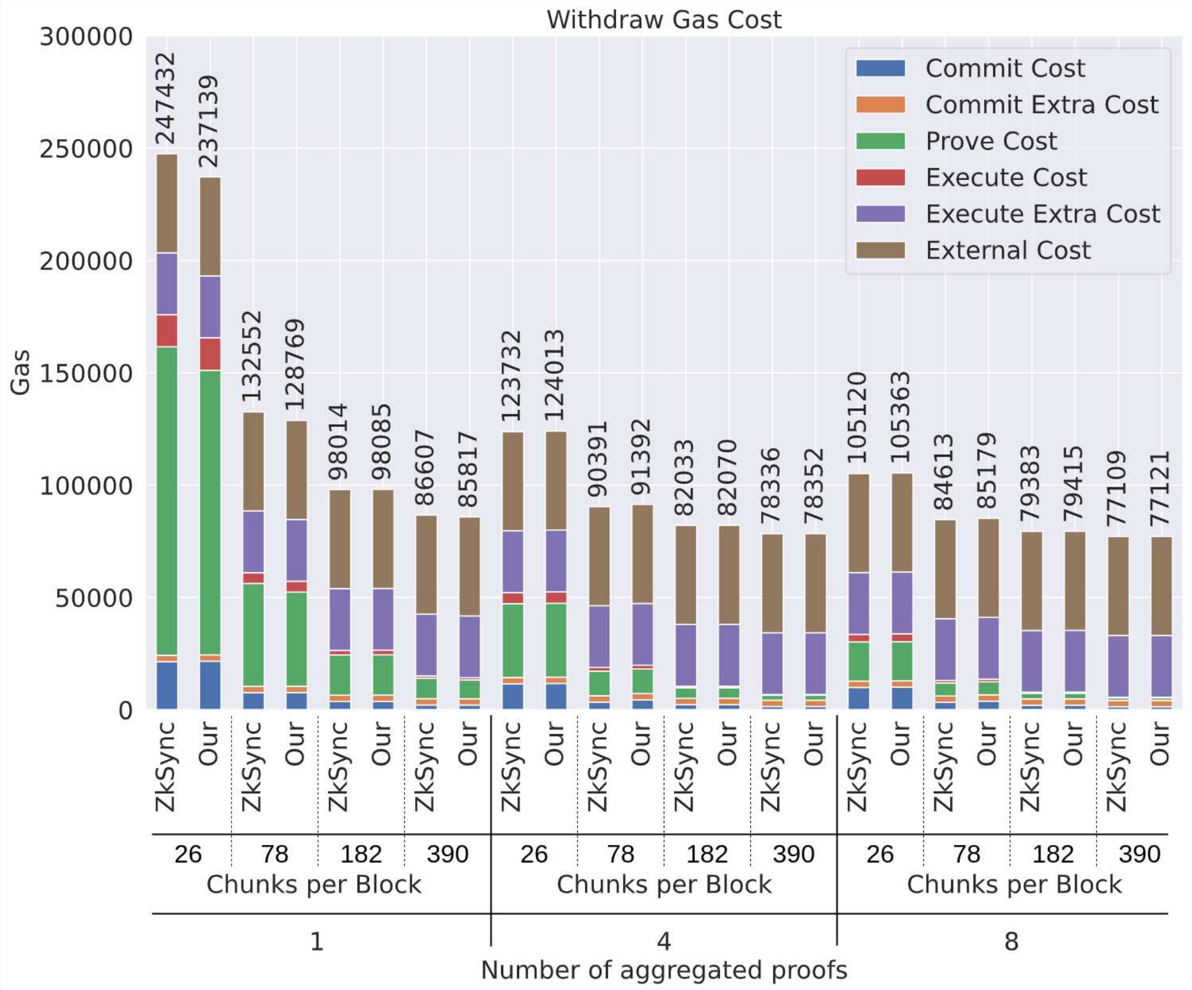
When block size is the largest and the number of aggregated proofs is the highest, the cost of a **deposit** is only **increased by 3%** for ERC20 and 2% for ETH, while **the rest** of the transactions only see their costs increase **by less than 1%**.



Results

Impact on existing transaction types

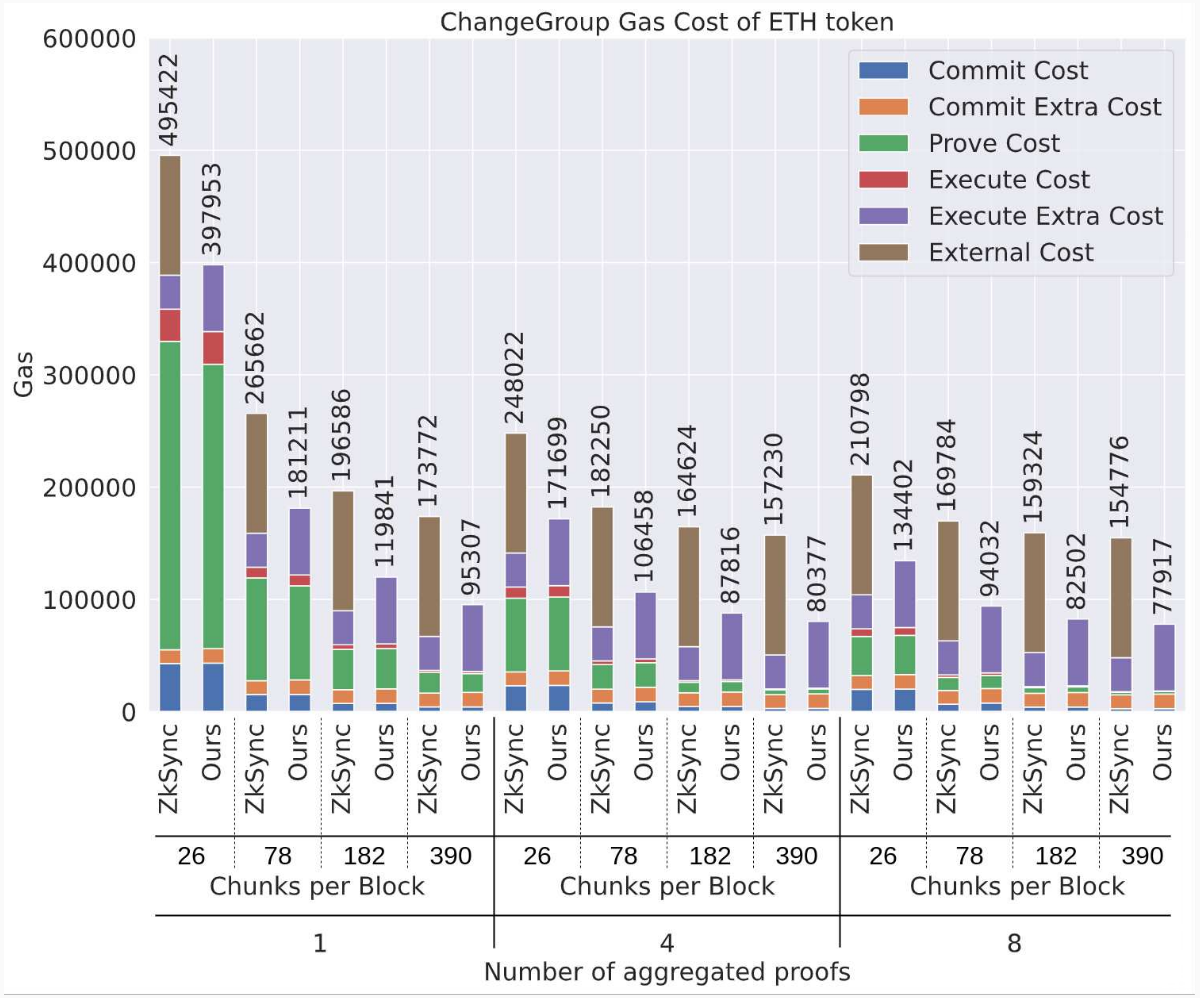
When block size is the largest and the number of aggregated proofs is the highest, the cost of a **deposit** is only **increased by 3%** for ERC20 and 2% for ETH, while **the rest** of the transactions only see their costs increase **by less than 1%**.



Results

Impact of new transaction types

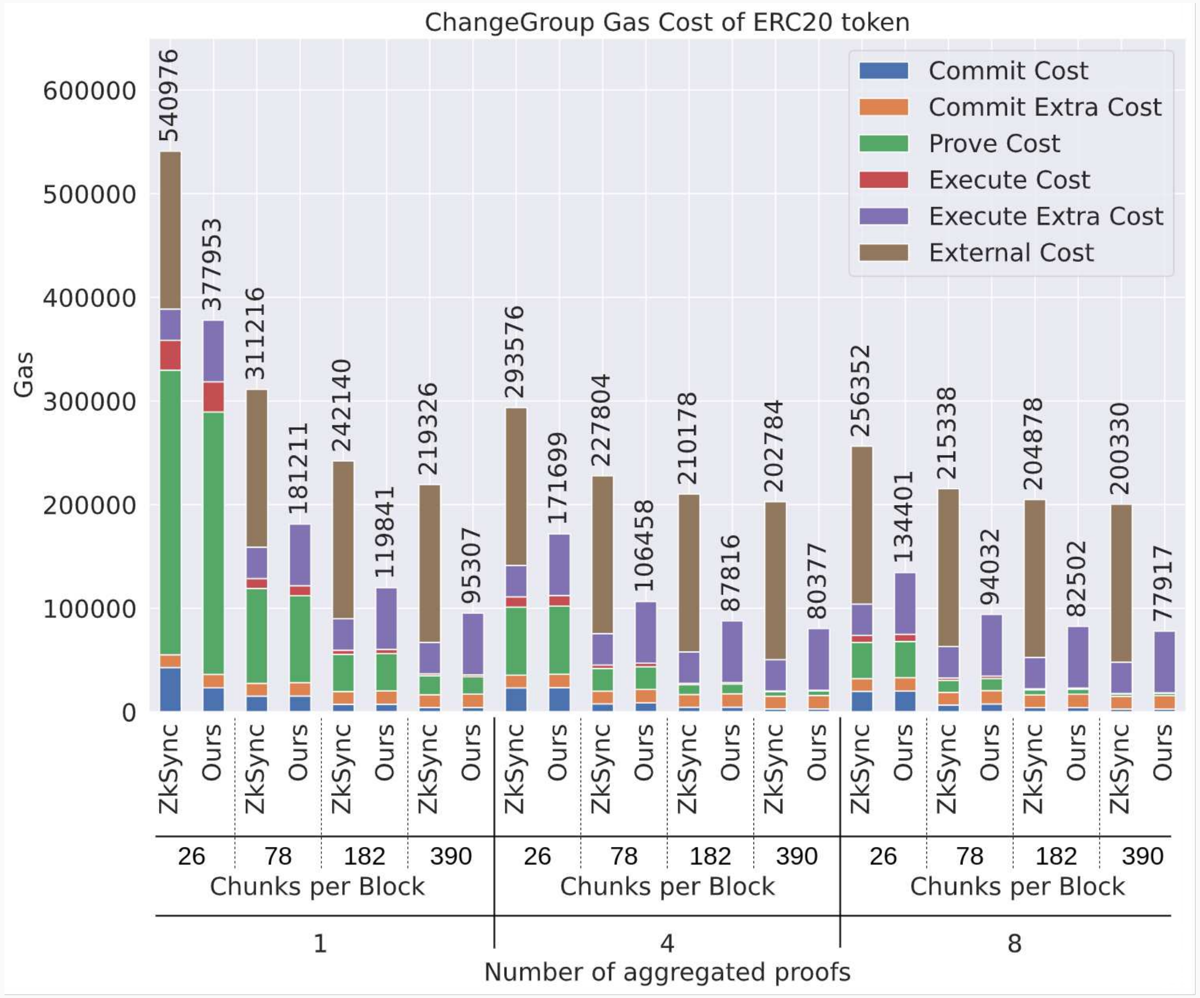
The **ChangeGroup** operation **reduces** gas consumption **by more than 49%** for ETH and **by more than 61%** for ERC20.



Results

Impact of new transaction types

The **ChangeGroup** operation **reduces** gas consumption **by more than 49%** for ETH and **by more than 61%** for ERC20.



Results

Deployment

During the **first deployment of** the smart contracts, our proposal leads to an **additional cost** of about **4%**, going from 22,106,772 gas to 22,904,219 gas.
However, when we compare the cost of **redeploying** zkSync Lite with the cost of creating a group with our proposal, **costs** are **reduced by more than 99%** from 22,106,772 gas (zkSync Lite) to 184,258 gas.

Results

Deployment

During the **first deployment of** the smart contracts, our proposal leads to an **additional cost** of about **4%**, going from 22,106,772 gas to 22,904,219 gas.
However, when we compare the cost of **redeploying** zkSync Lite with the cost of creating a group with our proposal, **costs** are **reduced by more than 99%** from 22,106,772 gas (zkSync Lite) to 184,258 gas.

Data availability

All the **graphics** and the **code** of our implementation are available **on github**:

Results

Deployment

During the **first deployment of** the smart contracts, our proposal leads to an **additional cost** of about **4%**, going from 22,106,772 gas to 22,904,219 gas.
However, when we compare the cost of **redeploying** zkSync Lite with the cost of creating a group with our proposal, **costs** are **reduced by more than 99%** from 22,106,772 gas (zkSync Lite) to 184,258 gas.

Data availability

All the **graphics** and the **code** of our implementation are available **on github**:

Thanks for your attention.