OKIMS

Scaling blockchains zkRollup

Jordi Baylina

Technical Lead

iden3

OKIMS

Rollup limits (In ETH 1.0)

Current Eth:	32 Tx/s
Before Istanbul:	682 Tx/s
After Istanbul:	**2.048 Tx/s**

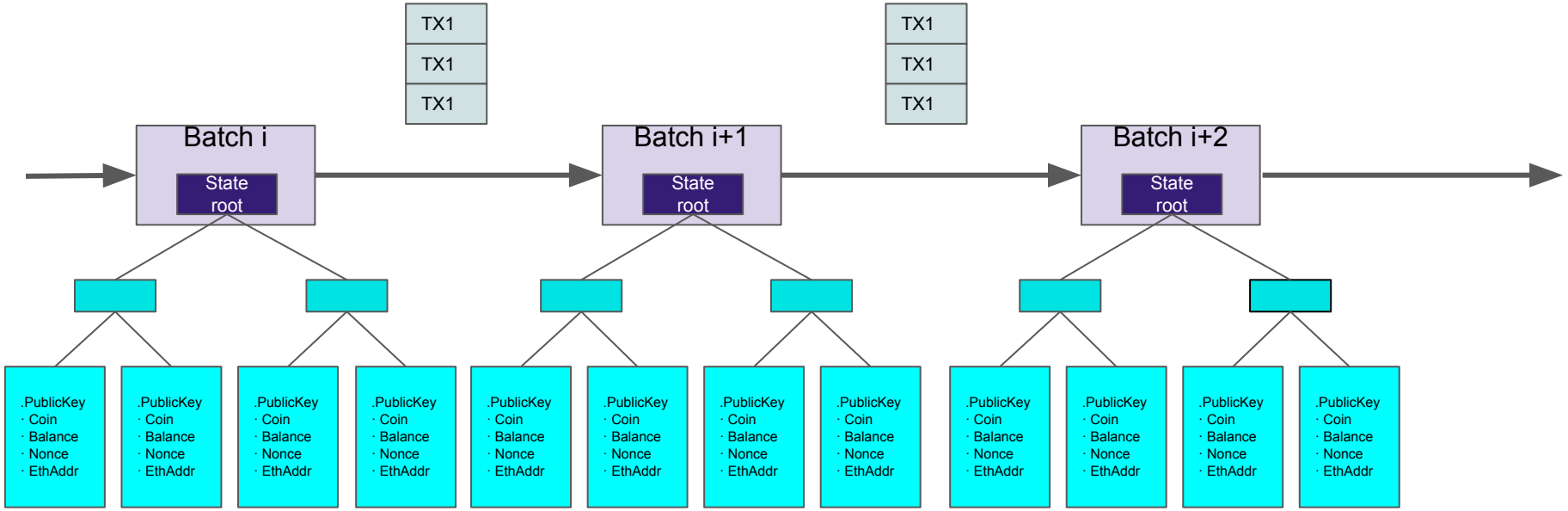
VISA network estimates

Average use:	2.000 Tx/s
Daily Peak:	4.000 Tx/s
Max. Capacity:	54.000 Tx/s

iden3

OKIMS

Rollup scalability general idea



iden3

OKIMS

Succinct Verification

- Constant verification time
- Small size
- Fast proving time
- Secure
- Quantum resistant
- No trusted setup

zk-Snarks

iden3

OKIMS

Data availability

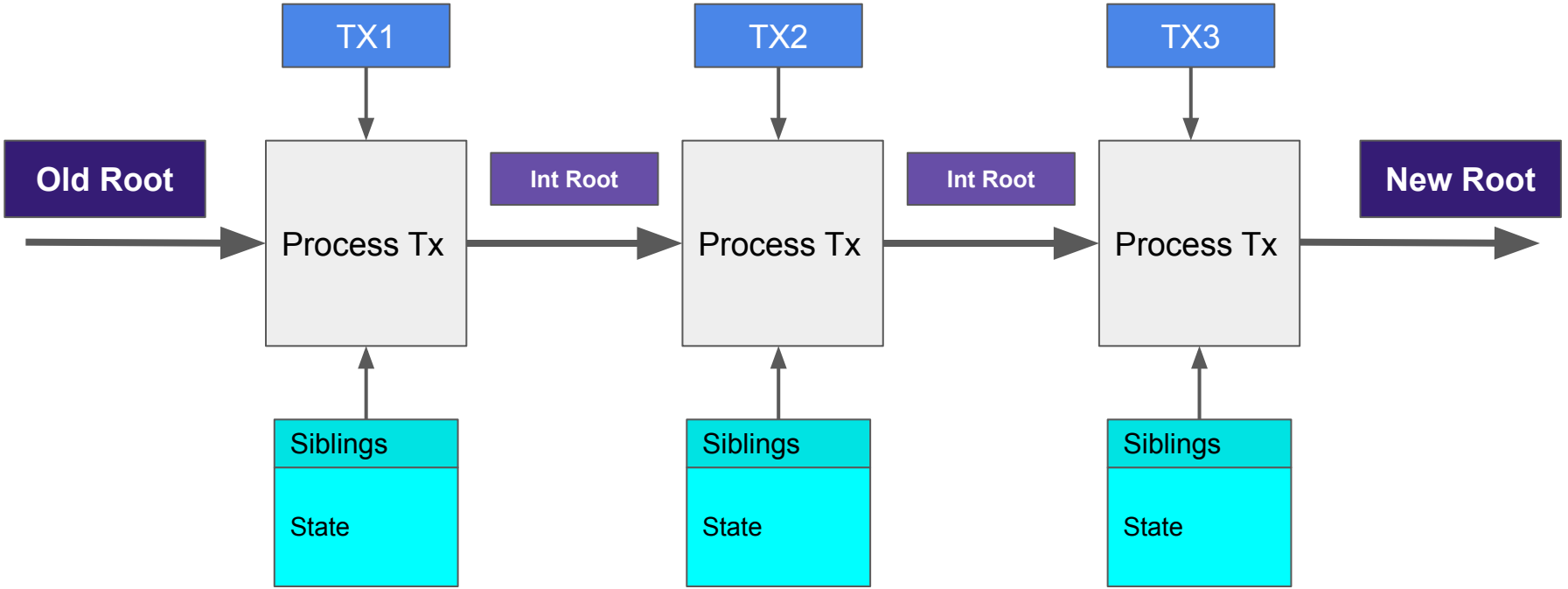
- On Chain <- ROLLUP
- Other chains (BCH ?)
- Ethereum 2.0
- Filecoin
- A federation of servers.
- ...

We don't need consensus. We just need the data to be available.

iden3

OKIMS

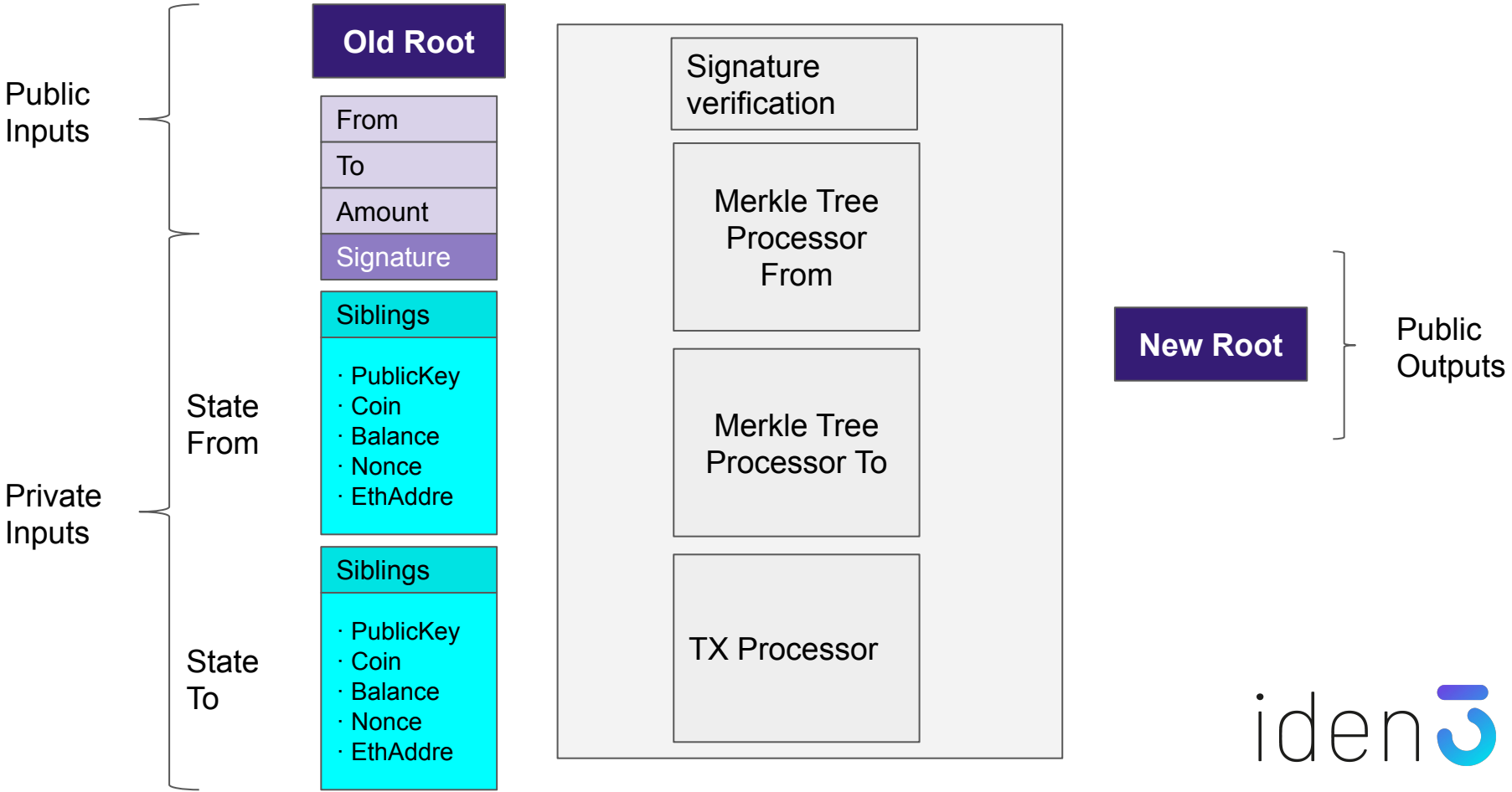
Rollup main circuit



iden3

Circuit that process a TX.

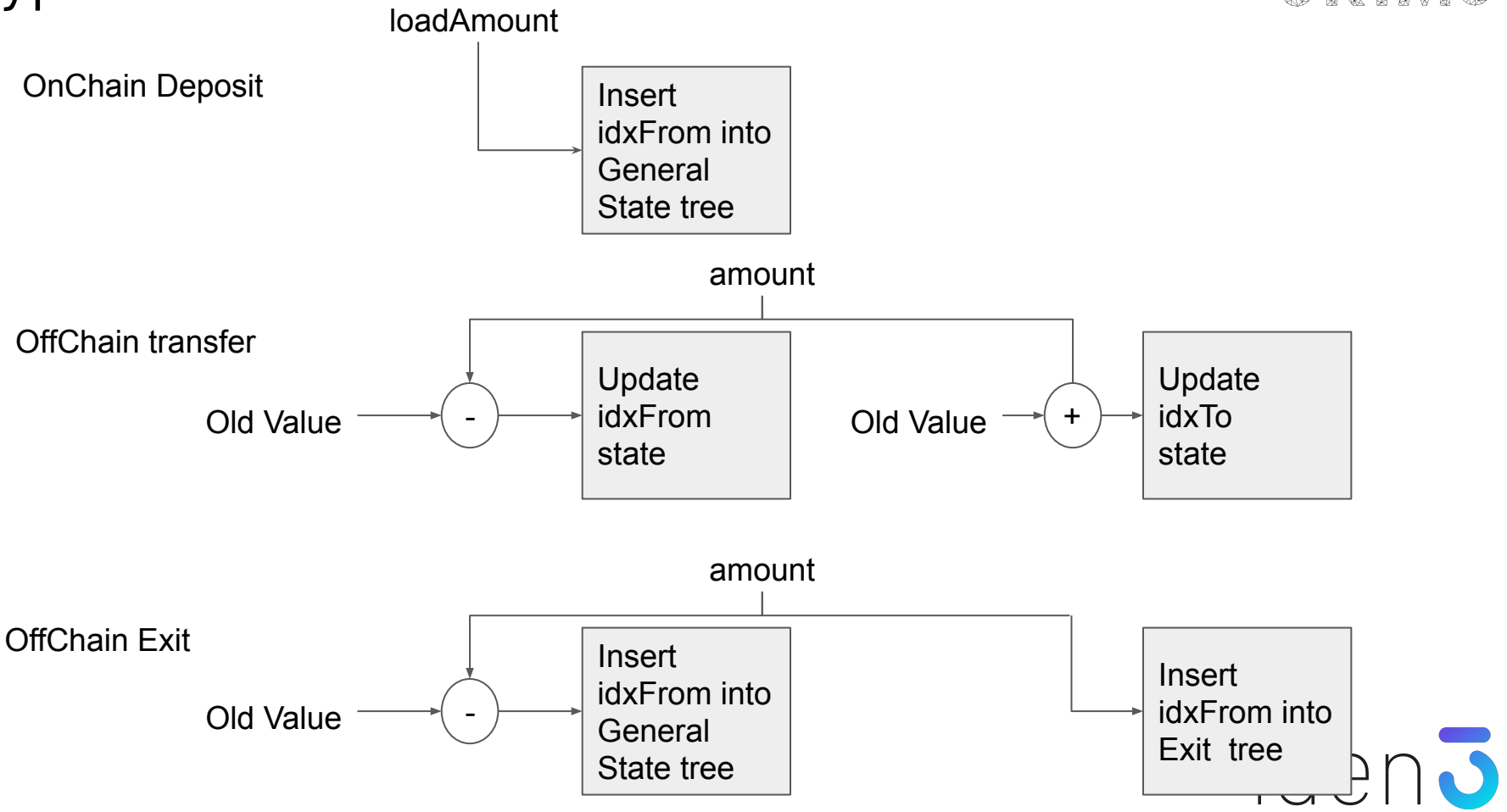
OKIMS

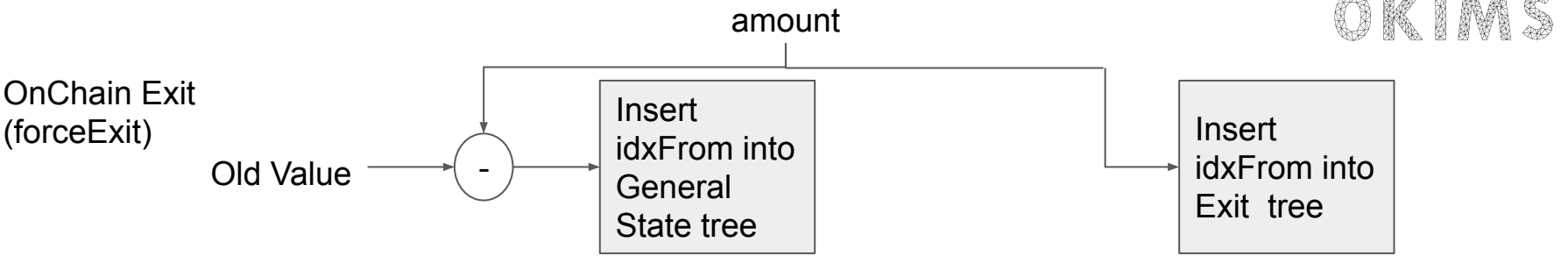


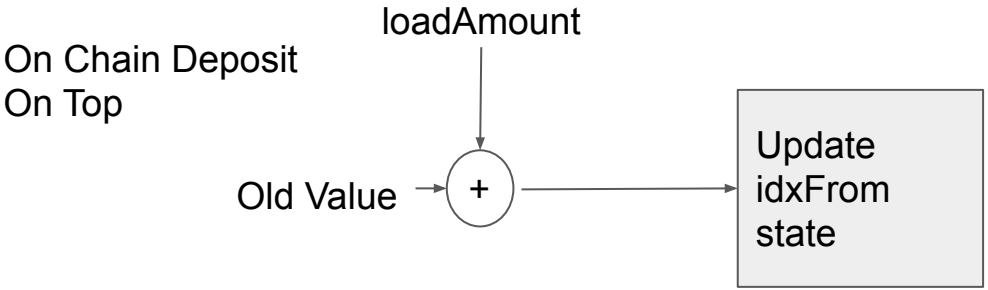
iden3

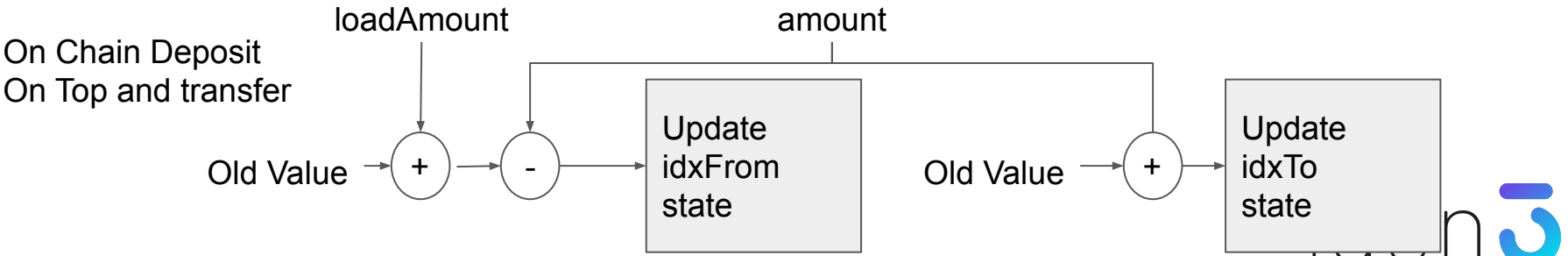
Types of TXs

OKIMS




amount
OnChain Exit
(forceExit)
Old Value
-
Insert
idxFrom into
General
State tree
Insert
idxFrom into
Exit tree


loadAmount
On Chain Deposit
On Top
Old Value
+
Update
idxFrom
state


loadAmount
amount
On Chain Deposit
On Top and transfer
Old Value
+
-
Update
idxFrom
state
Old Value
+
Update
idxTo
state

OKIMS

Exit Mechanism

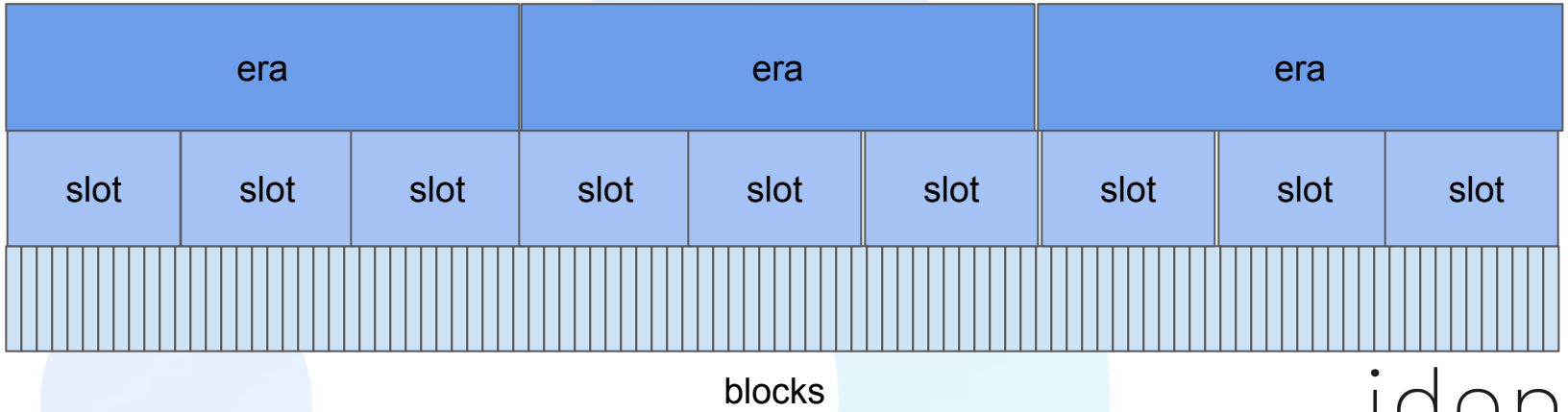
- Sends to 0 TX which are considered exits.
- In each batch there is an Exit tree.
- To withdraw, users have to prove they are in the exit tree.
- You can force an exit by sending an onchain transaction.

iden3

OKIMS

Decentralized operator

- Fixed # blocks per slot. Fixed # slots per era.
- Registration / deregistration happens in era+2

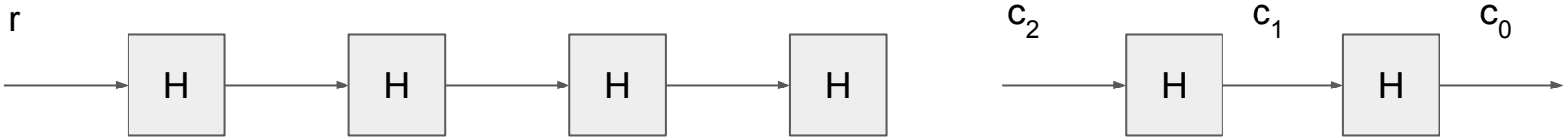


iden3

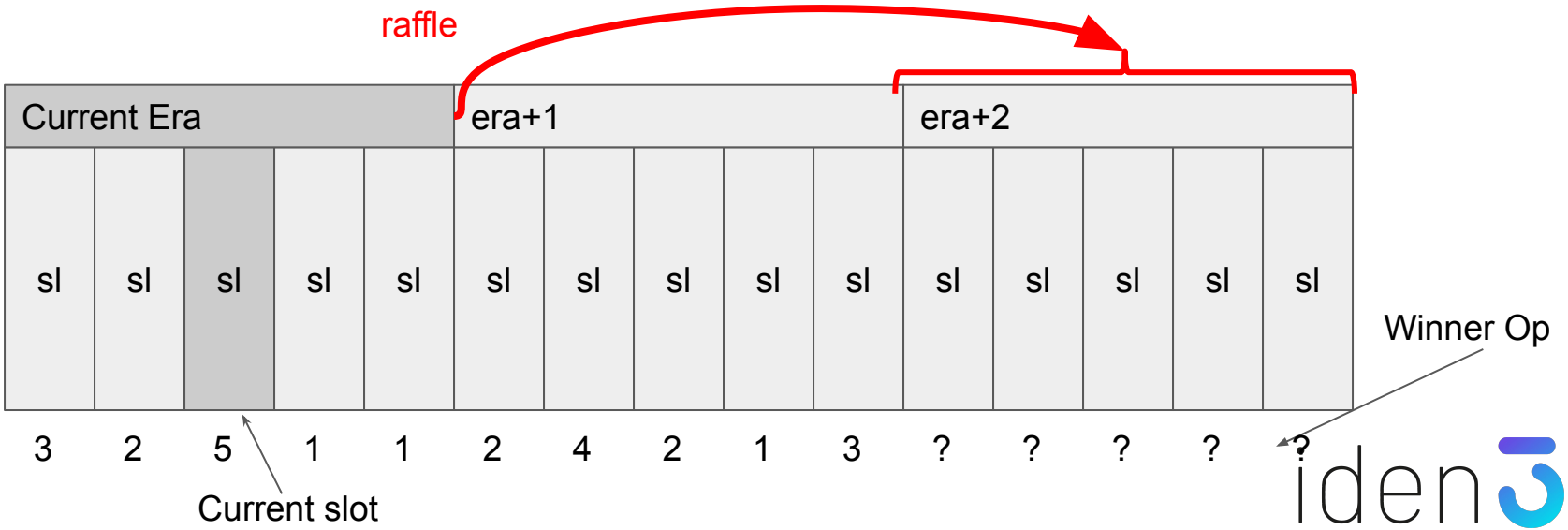
The raffle

OKIMS

The operators reveals a precommited random number in the first batch of each slot.



The hash of all revealed randoms, determine the seed for the raffle of the era+2



iden3

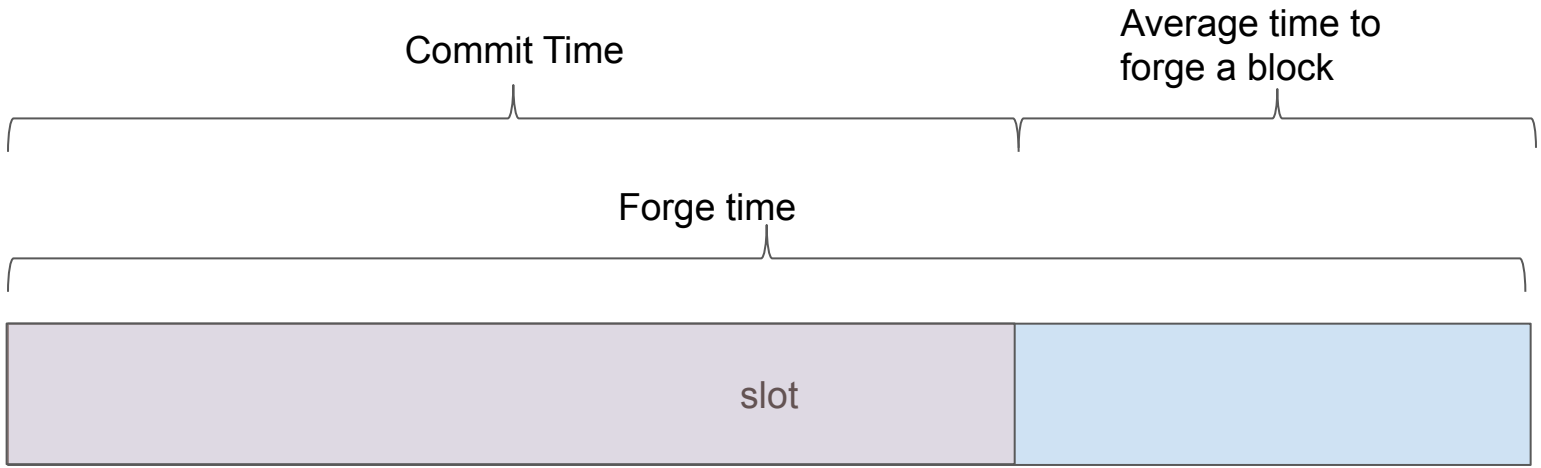
OKIMS

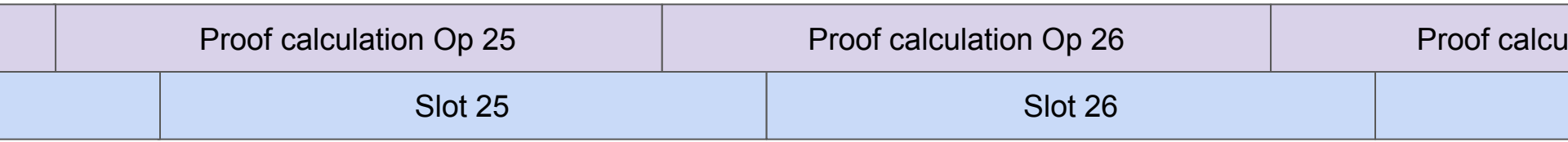
Effective Stake:
Compensation for risk concentration

$$
\begin{aligned}
f(2x) &= (1+p)2f(x) \\
f(x) &= x^{1+log_2(p)} \\
10\% &\Rightarrow x^{1.13} \\
20\% &\Rightarrow x^{1.26} \\
50\% &\Rightarrow x^{1.58} \\
100\% &\Rightarrow x^{2}
\end{aligned}
$$

iden3

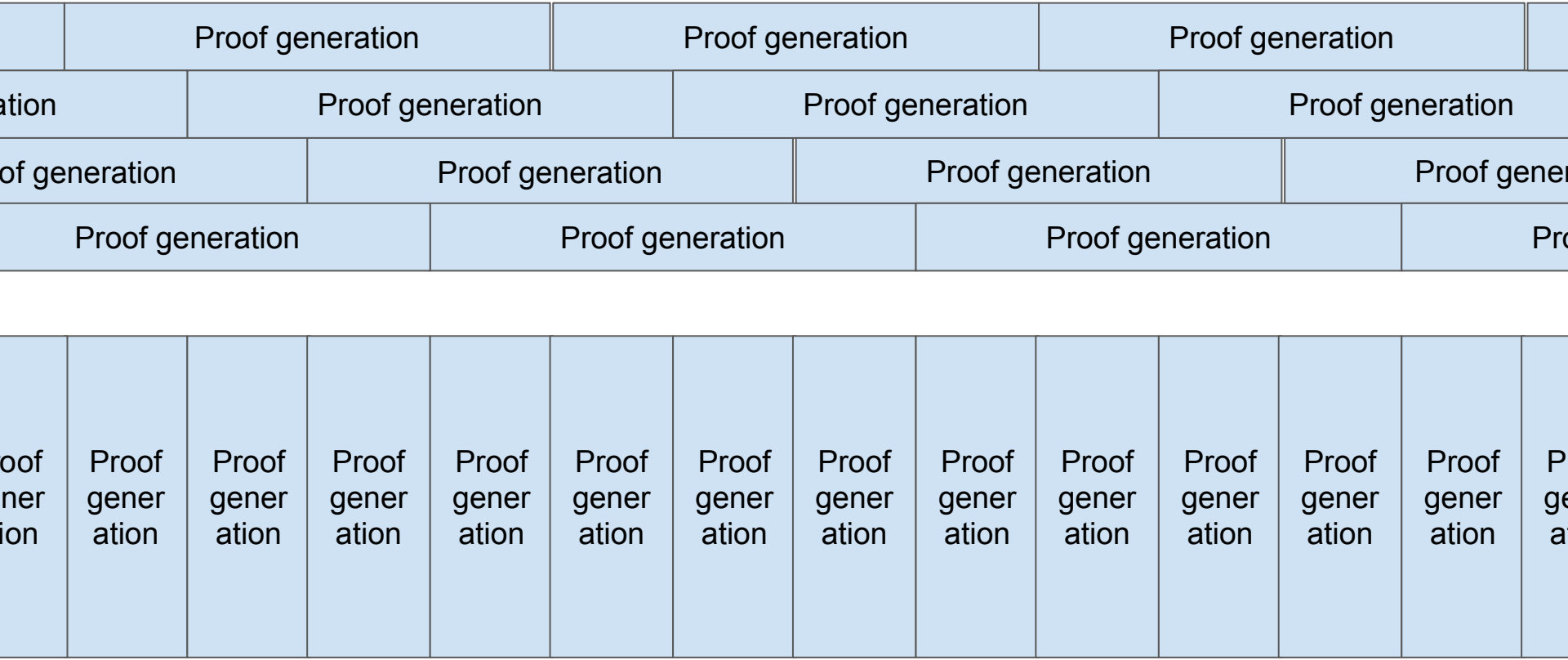
Pipeline batch forging





Parallelize Batch calculation

OKIMS



iden3

OKIMS

Slashing

Operators are slashed with the full stake if:

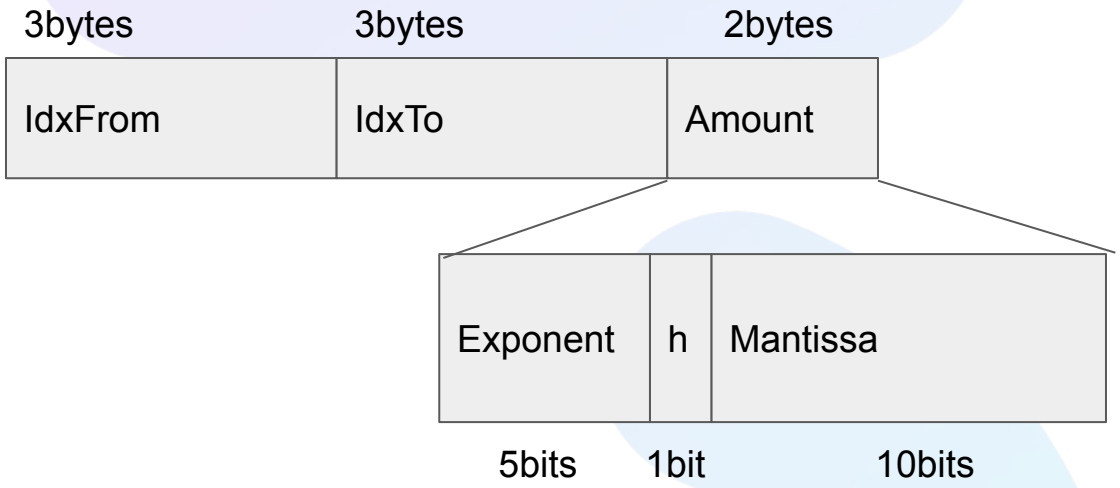
- They don't forge a single block in the assigned slot.
- They don't forge a committed block.

Instead of being burnt, the slashed stake is sent to a pot. This pot is used to reward developers who are able to break cryptographic primitives.

iden3

OKIMS

Tx Format



m*10^exp + h*(10^exp)/2

3.5 Decimal Digits

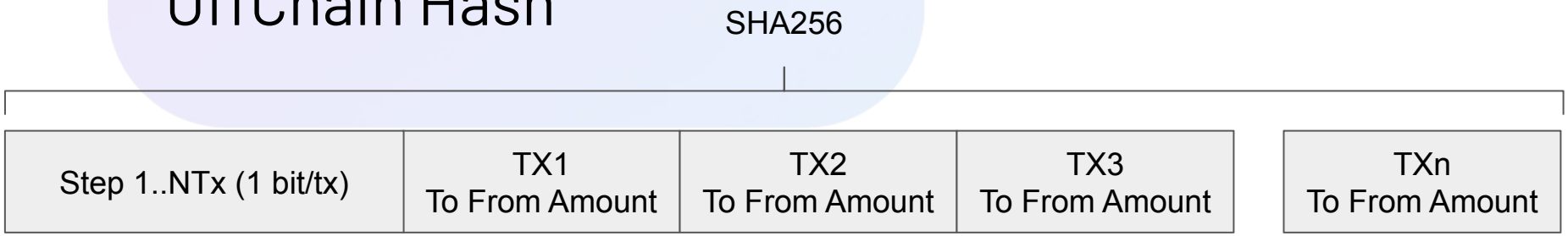
Examples

8.34
15.35
464.50
1830
1835

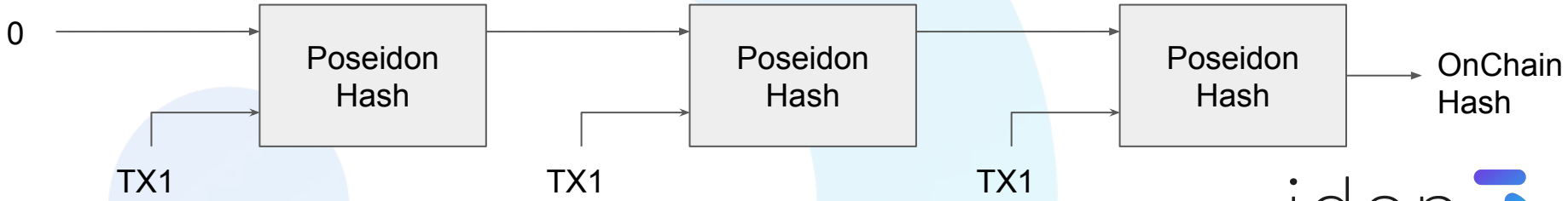
iden3

OKIMS

OffChain Hash



OnChain Hash



iden3

OKIMS

Fees

Fee Plan

CoinIdx1	Fee1
CoinIdx2	Fee2
...	
CoinIdx15	Fee15

CoinIdx15	.....	CoinIdx2	CoinIdx1

Fee15	.....	Fee2	Fee1

FeePlanCoin

FeePlanFee

Public Inputs

iden3

OKIMS

Signature verification

- EDDsa Using Baby Hub
- Poseidon hash function
- Batching ?? -> Requires Operations Module the order of the BabyJub Curve.

iden3

OKIMS

On chain transactions

- Force Exit
- Deposit New account / On top
- Force transfer (Works with smart contracts).

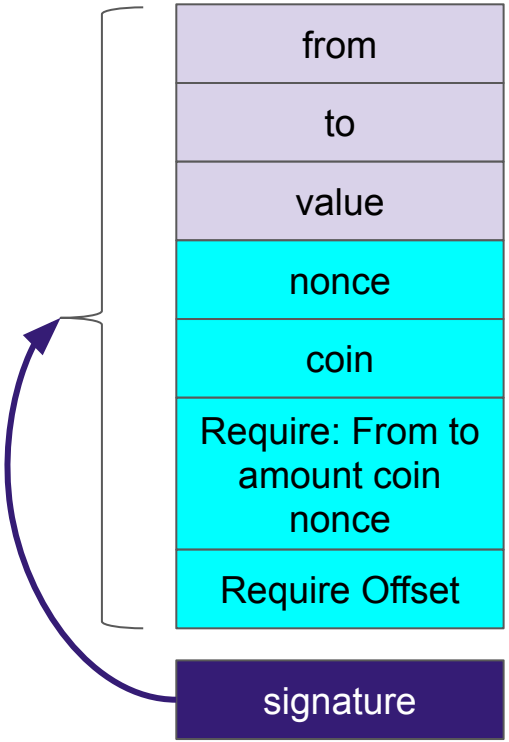
Operators are forced to include onchain TX in order for the batch to be valid.

iden3

Atomic Swaps

OKIMS

TX i-3	TX i-2	TX i-1	TX i	TX i+1	TX i+2	TX i+3



iden3

OKIMS

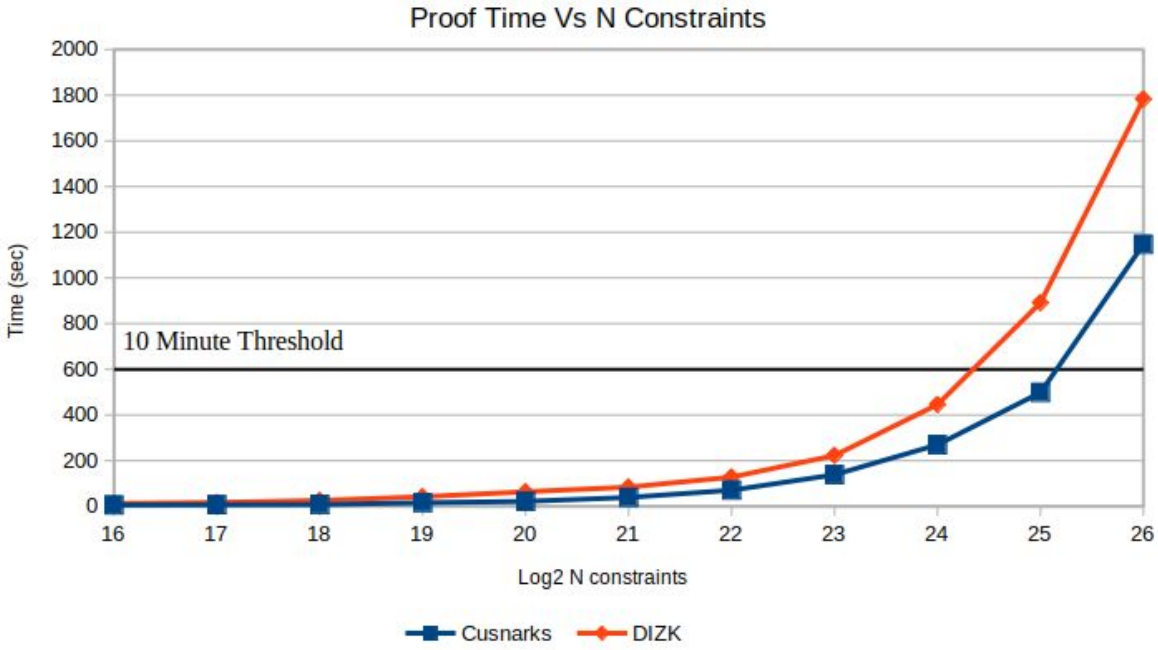
CuSnarks development

- Parallel implementation of Groth16 prover based on snarkjs over curve BN128.
- Written in CUDA and wrapped in Python.
- Works together with circom and snarkjs.
- Objective is speed.

iden3

CuSnarks benchmarking

- With the current system, and 2^{26} constraints, the proof takes 19 minutes
- Dell T640 server - 2xIntel Xeon Silver 4110@2.1GHz (32 cores total) /128GB memory / 2xGPU GTX 1080 (2560 cores@1607MHz/ 8GB GDDRX5) - **$10K**



DIZK : results taken from H.Wu, W. Zheng, A. Chiesa, R.A. Popa, and I. Stoica. DIZK: A distributed zero knowledge proof system. Results obtained with a different circuit but same number of constraints. Results provided as a reference.

OKIMS

Rollup limits (In ETH 1.0)

Current Eth: 10M / 21K / 15 = **32 Tx/s**

Before Istanbul:

Data cost/batch: 1 block: 2048tx * 8.125bytes/tx * 68gas/byte = 1,131,520

TX proof/batch = 800,000 (aprox)

Tx/s = 5 Batches/Block -> **682tx/s**

After Istanbul:

Data: 1 block: 2048tx * 8ebytes/tx * 16gas/byte = 266,240

TX proof = 350,000 (aprox)

Tx/s = 15 batches/Block -> **2,048tx/s**

iden3

OKIMS

Costs estimations

An investment of 20,000$ Can mine 2048tx / 10min

Amortization cost: 20,000$ / (2048 * 6 * 24 * 365 * 5) = 0.00003$/tx

Electricity cost: (4Kw * 0,15 $/KwH) / (2048 * 6) = 0,00006$/tx

Other operation costs: 10,000/year = 0.00009 $/tx

Gas TX Cost: 350,000*10Gwei = 0.0003 $/tx

So the cost of the TX = 0.00048$ = **0.048¢/tx**

Current Ethereum TX cost = **4¢/tx**

iden3

OKIMS

zkSnarks Tools

- Circom
  - Building specs for version 1.0
  - https://github.com/iden3/circom
- Circomlib
  - Auditing
- Snarkjs
  - Should migrate to websnark
  - Will be keeped for academic uses.
  - https://github.com/iden3/snarkjs
- Websnark
  - MNT Verification support. Completing other curves.
  - Possible development of PLONK
  - https://github.com/iden3/websnark
- CuSnarks
  - Target is to forge a proof of 2^{27} constraints in less than 10 min
- Trusted setup with trusted hardware.
  - Working in a SGX compatible version.

iden3

OKIMS

Let’s go scale!

Current limit at **2.048 tx/s** and increasing

Atomic transactions

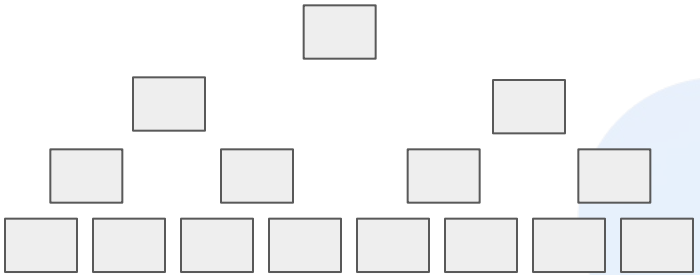
Lower transaction cost

Current state: Preparing and auditing for test network

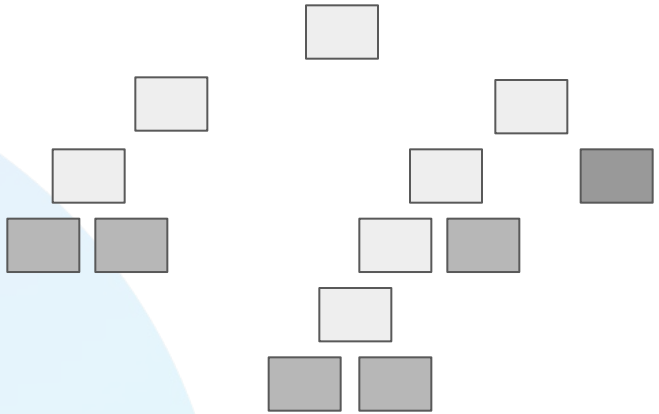
@jbaylina
@identhree

iden3

Normal Tree - Sparse Merkle tree - Other Accumulators



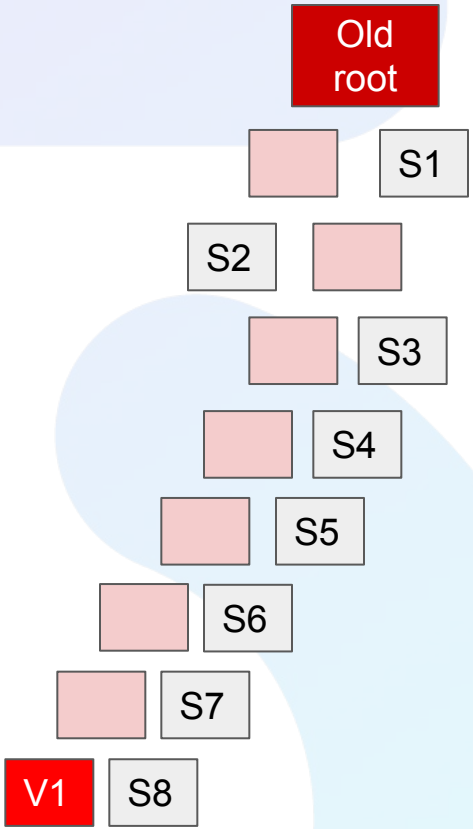
Normal tree



Sparse merkle

OKIMS

Verifying state



iden3

Updating State