Tightly Coupled DeFi in ETH2

pg

Agenda

1. What do we mean by tightly coupled Ethereum?
2. What are our options for tightly coupling ETH2?
3. How will DeFi look on scalable Ethereum?

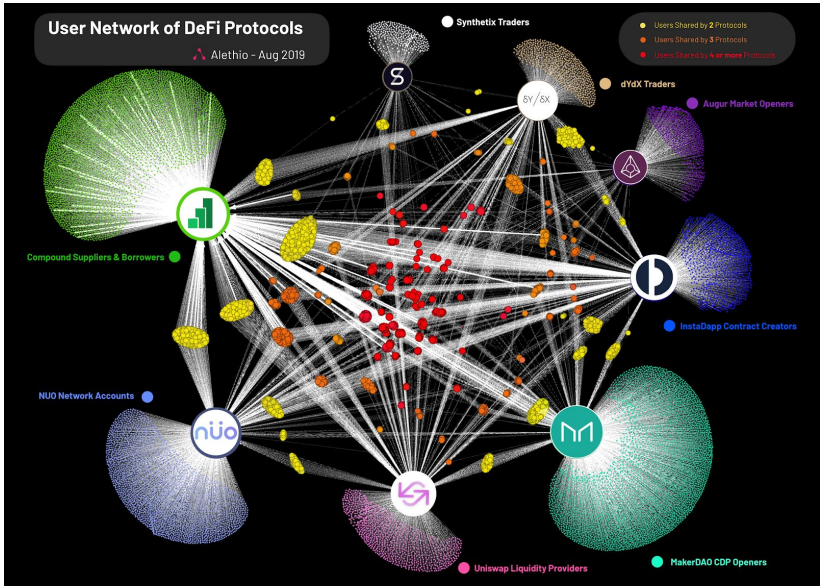
Agenda

1. What do we mean by tightly coupled Ethereum?
2. What are our options for tightly coupling ETH2?
3. How will DeFi look on scalable Ethereum?

What does it mean that DeFi is tightly coupled?

What does it mean that DeFi is tightly coupled?

- Single state space.
- Atomic transactions.

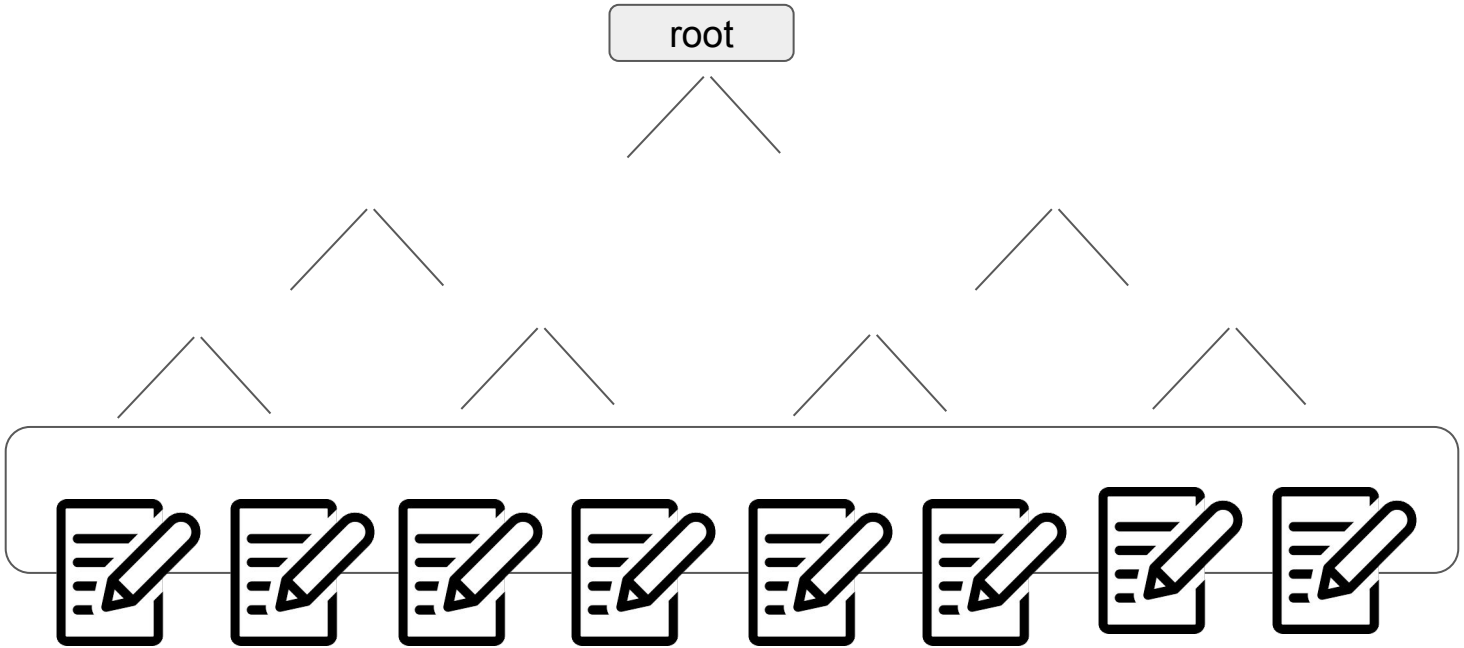


What does it mean that DeFi is tightly coupled?

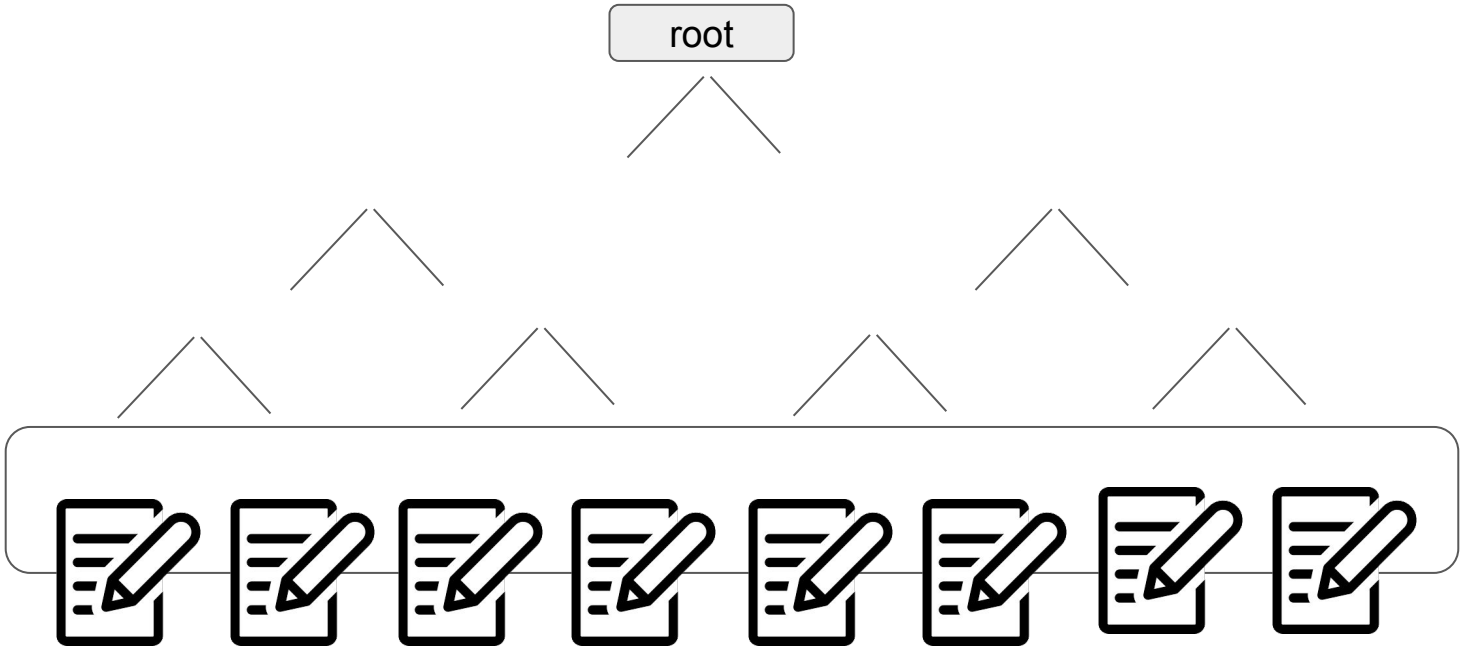
- Single state space.
- Atomic transactions.

Single State Space

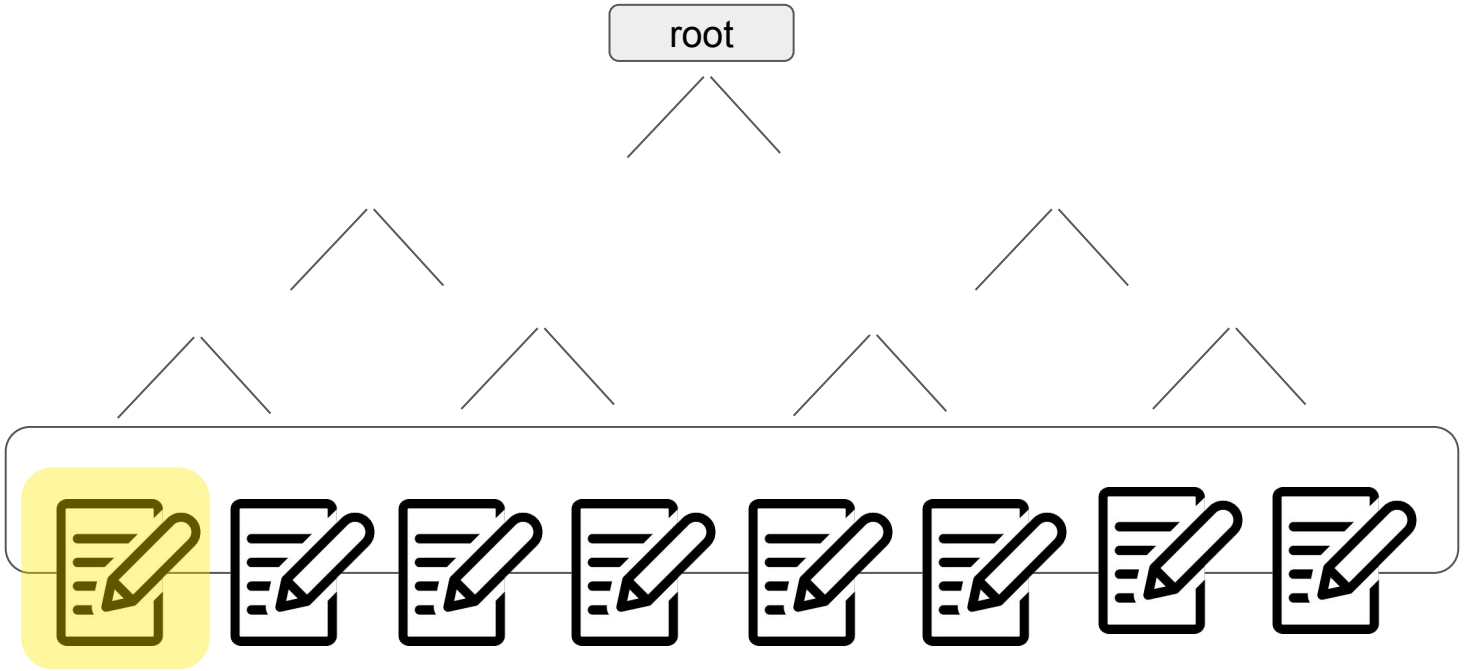
Single State Space



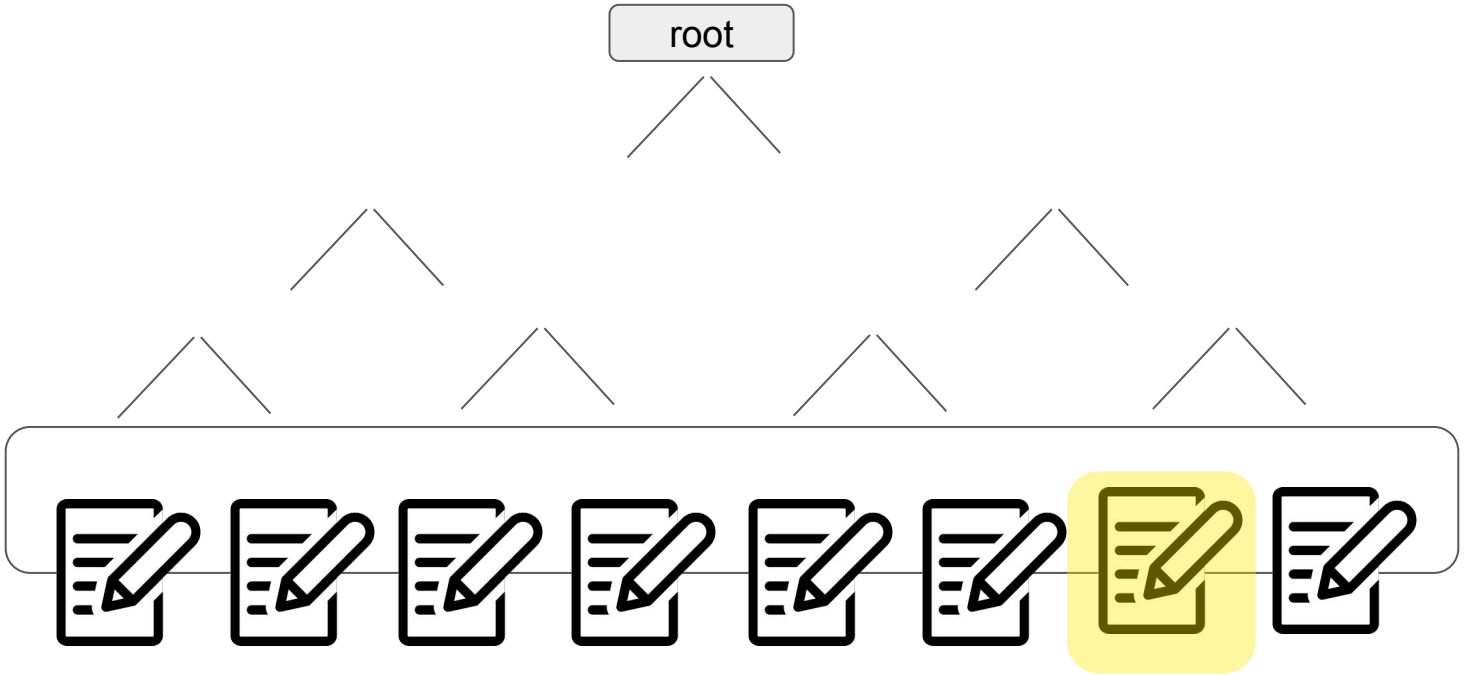
Single State Space



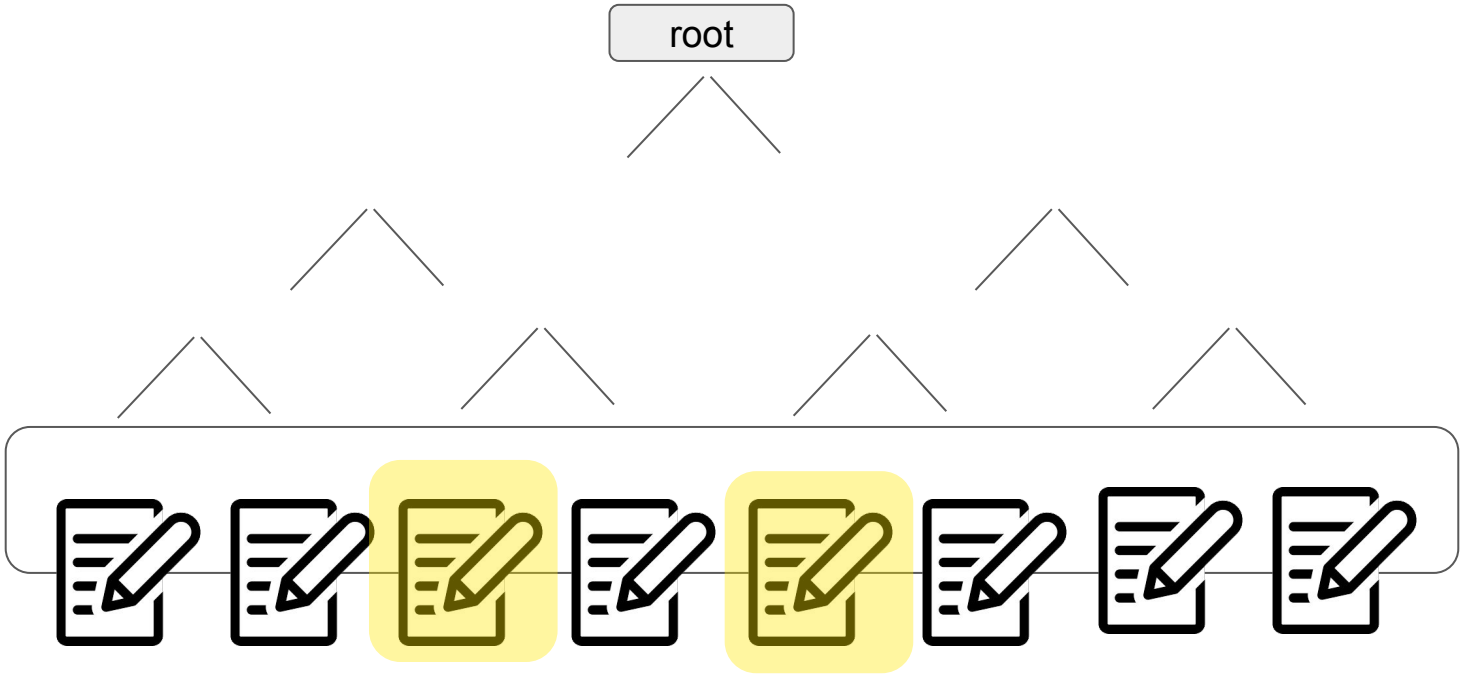
Single State Space



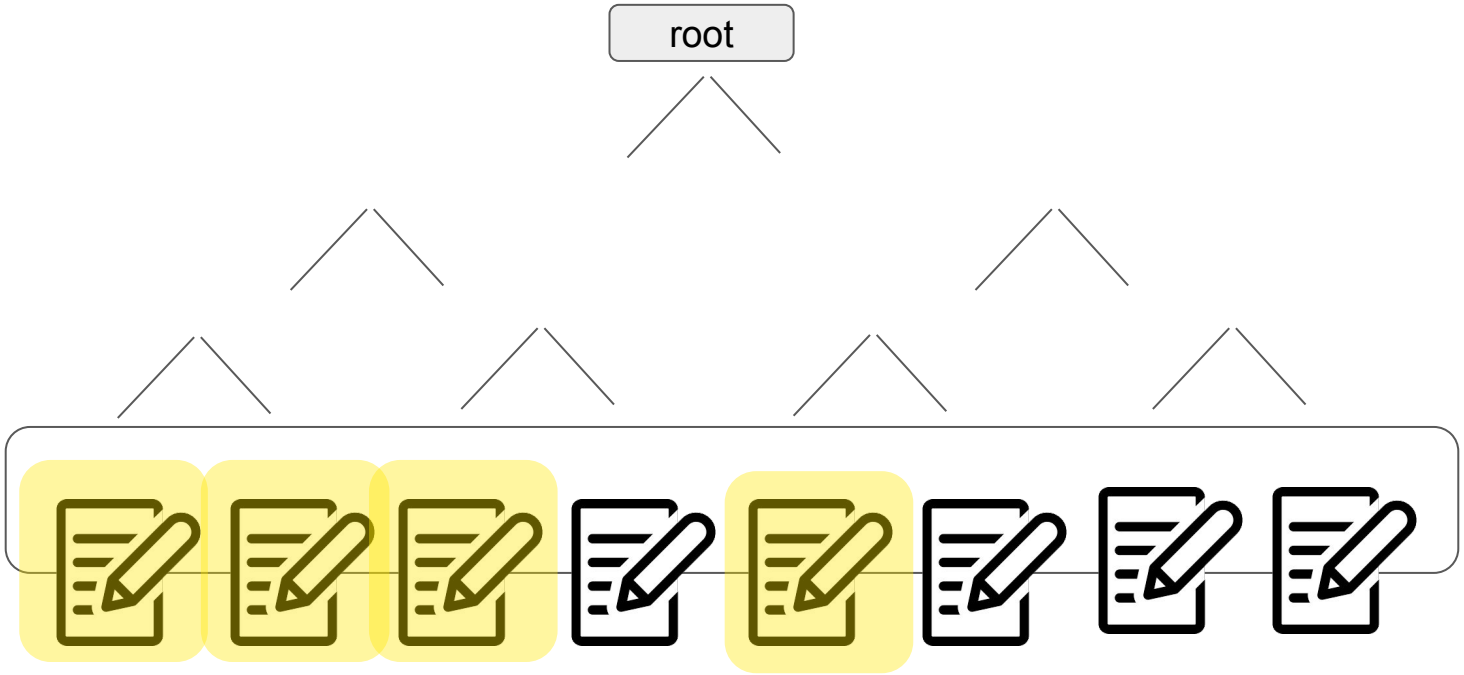
Single State Space



Single State Space

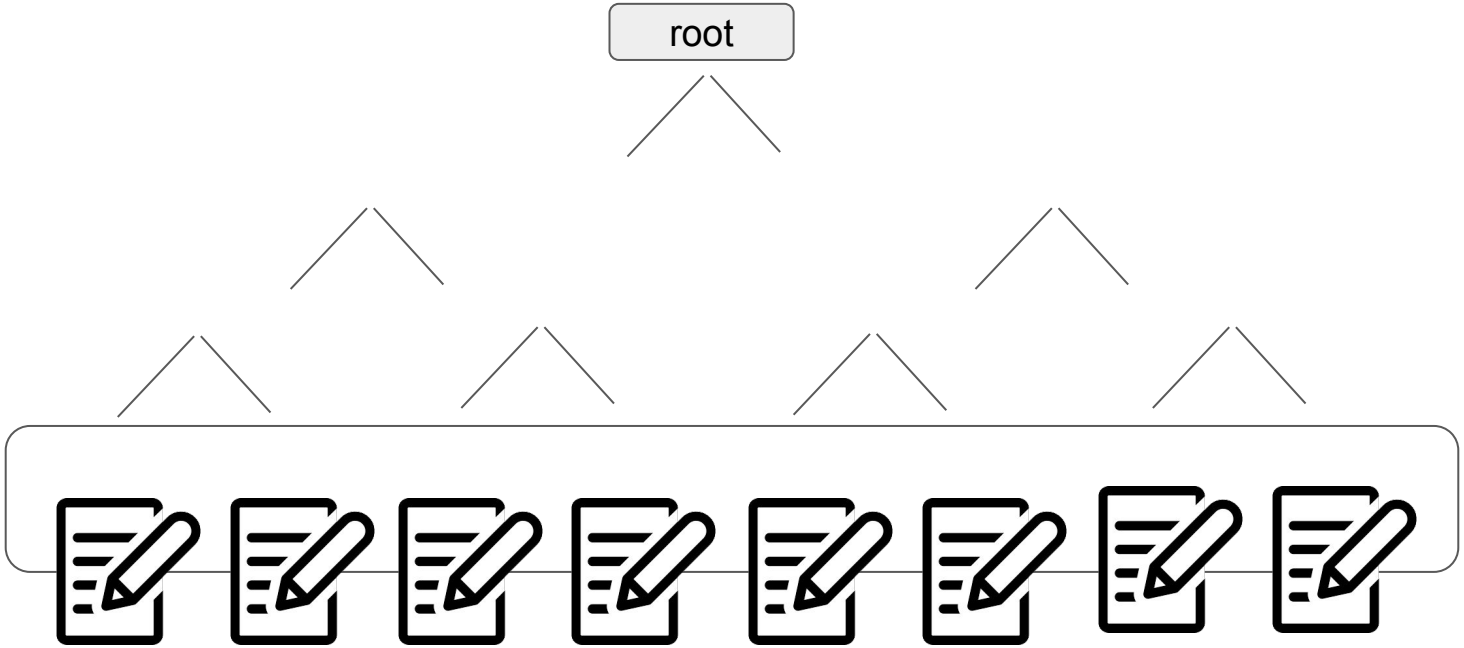


Single State Space



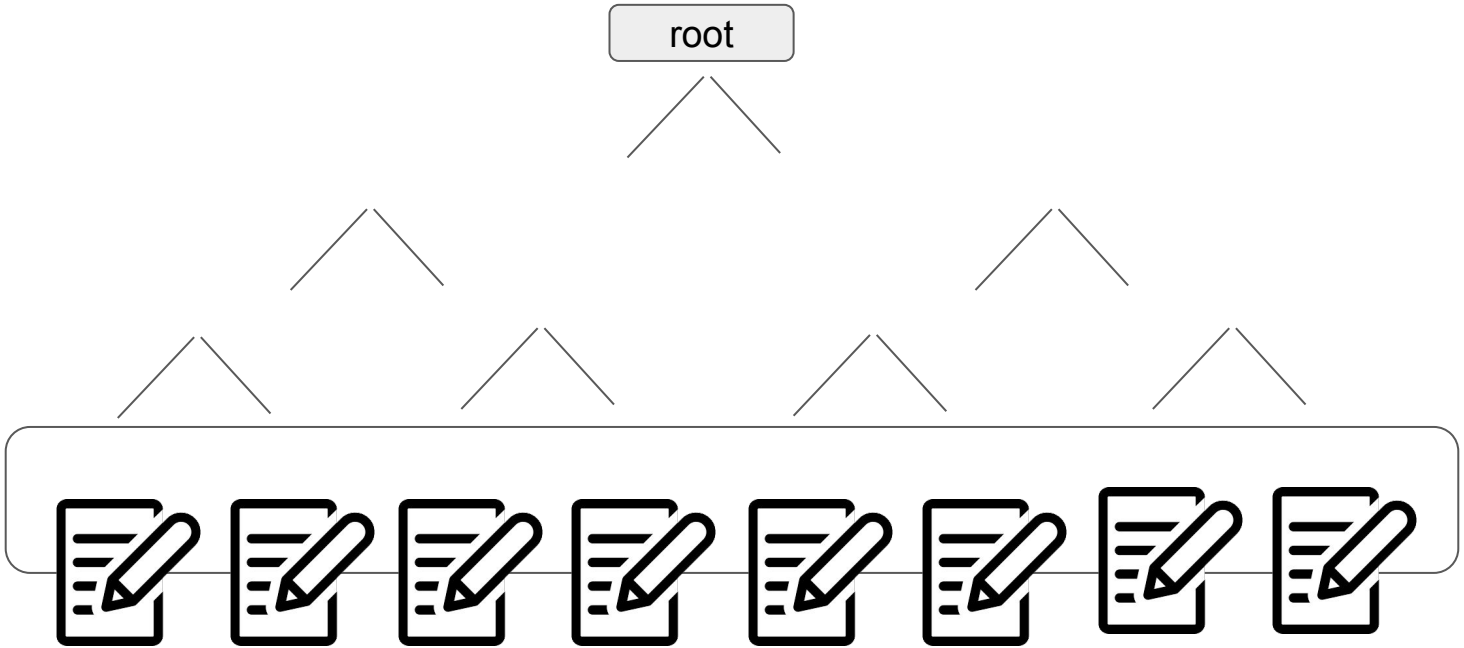
Single State Space

All contracts are accessible in every block.



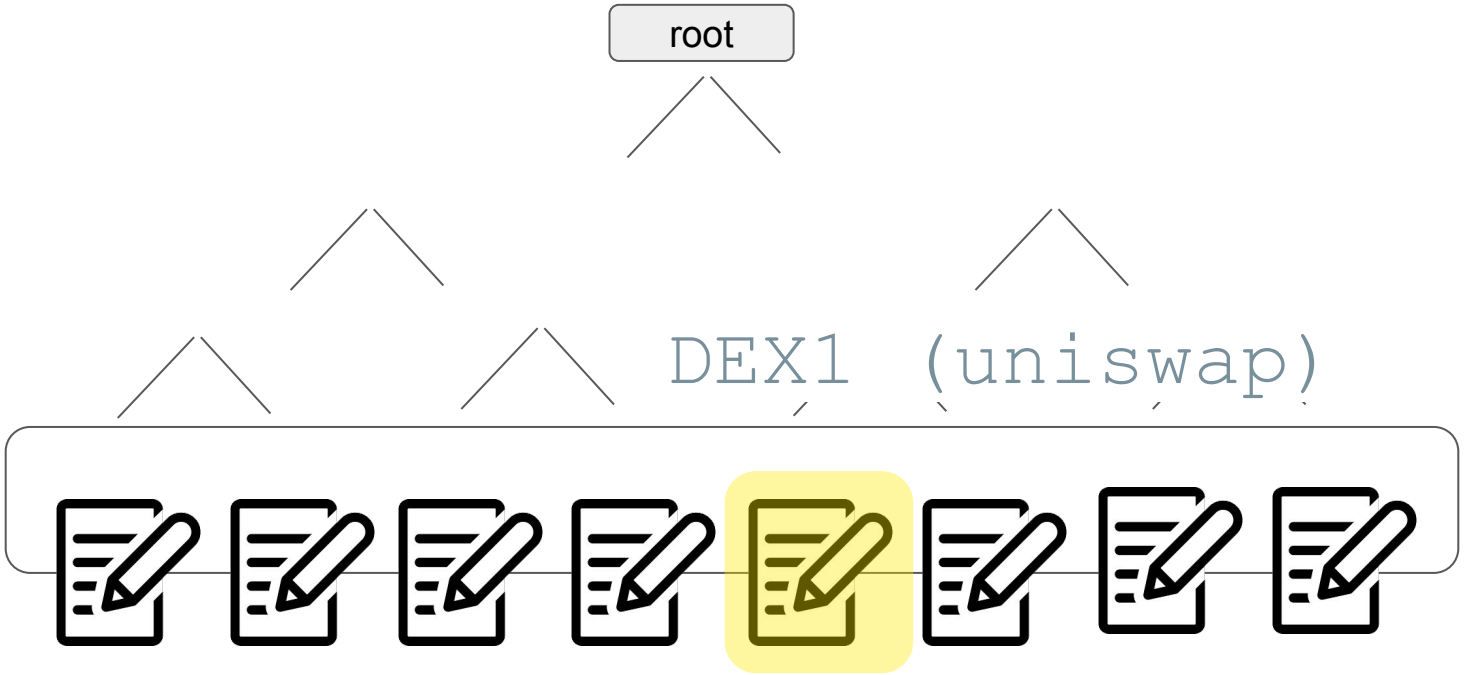
Atomic Transactions

Atomic Transactions



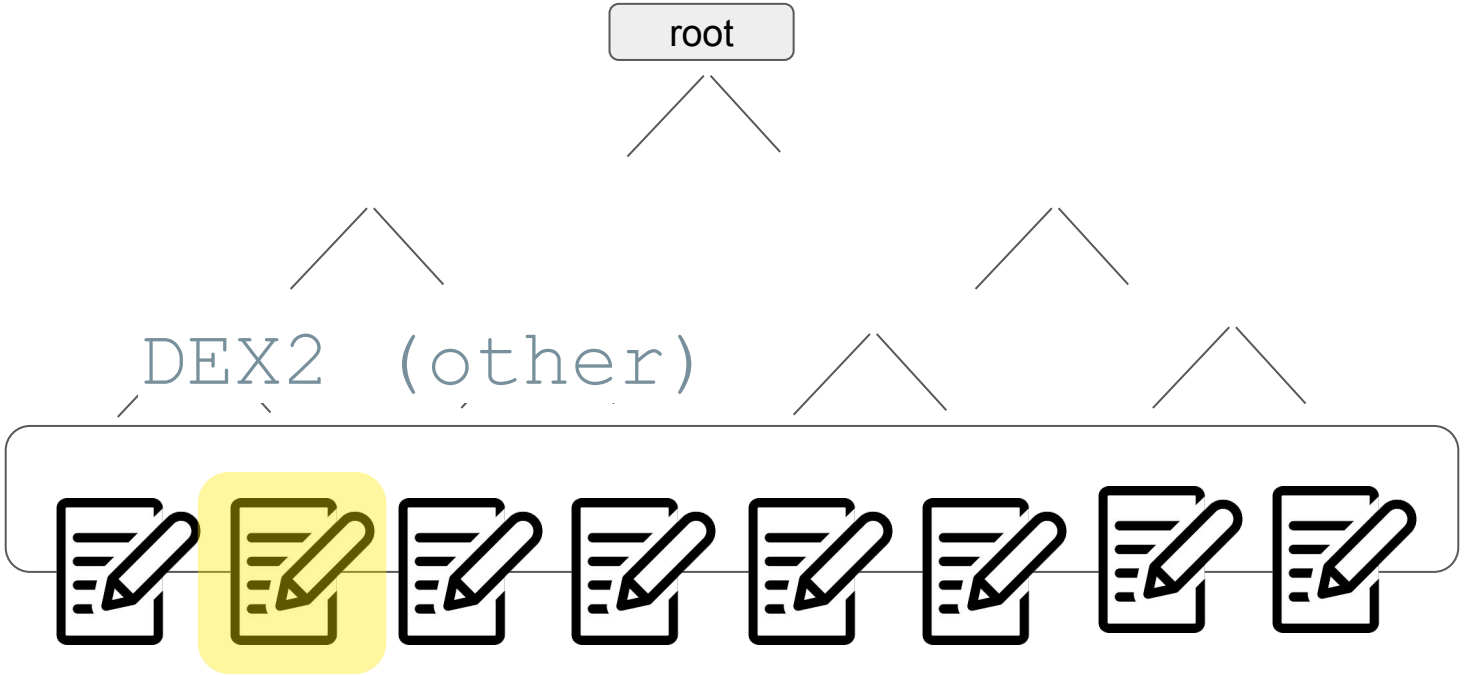
Atomic Transactions

Selling token for $10

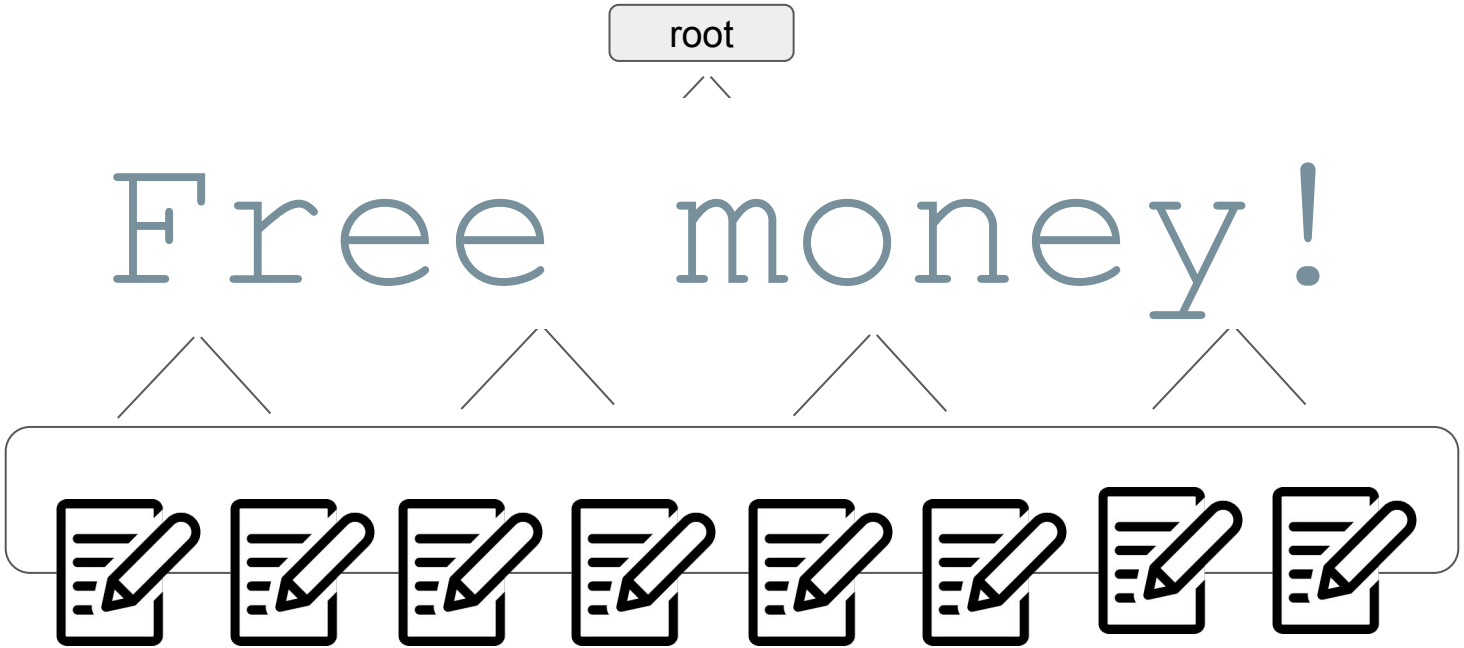


Atomic Transactions

Buying token for $11

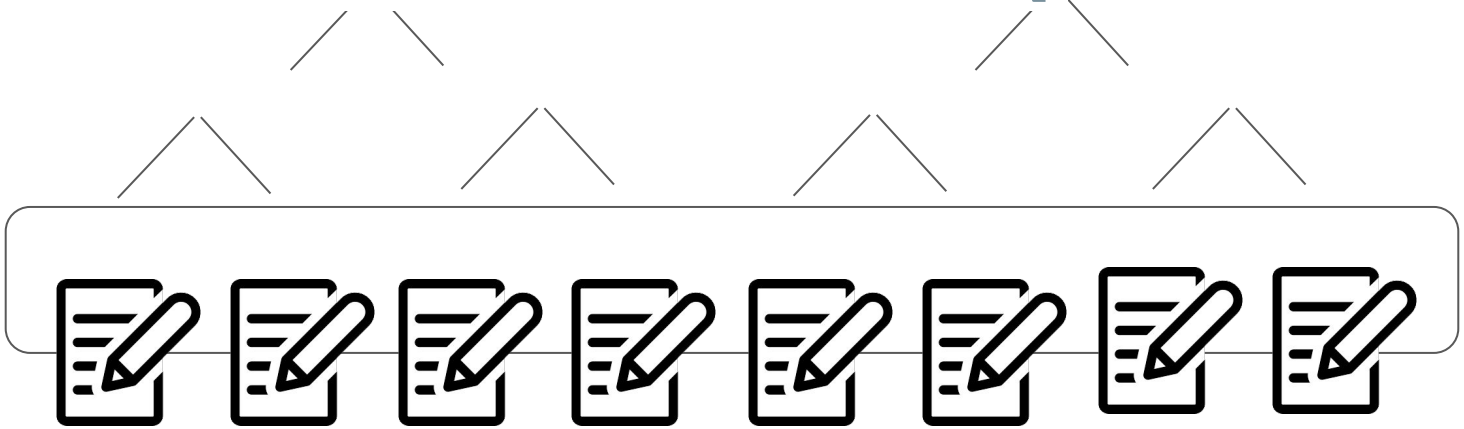


Atomic Transactions



Atomic Transactions

Atomic transaction:
[buy token on Uniswap,
 sell token on OtherDex]

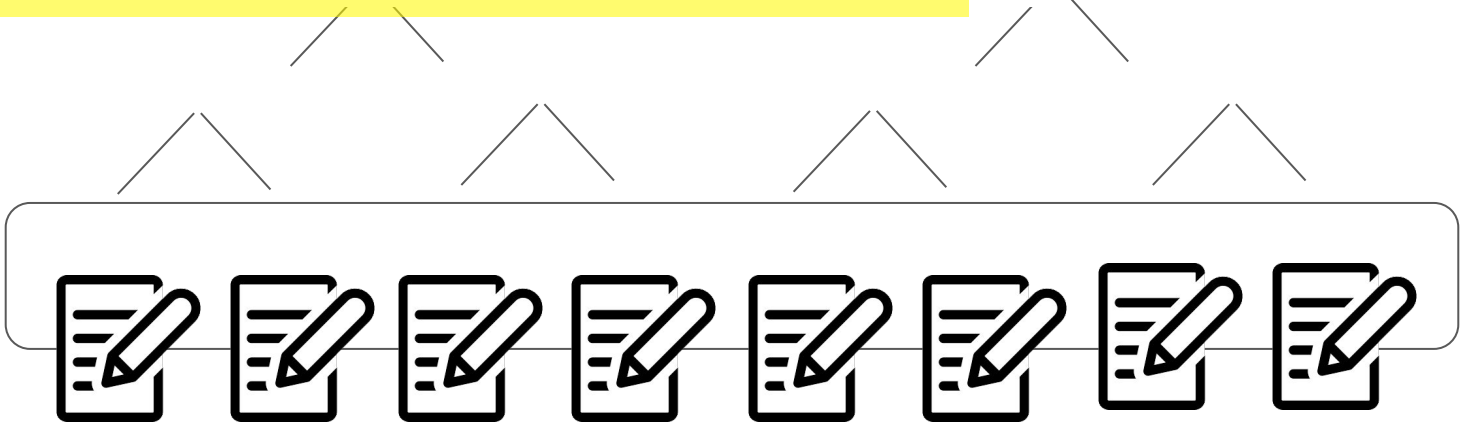


Atomic Transactions

All or nothing!

Atomic transaction:

[buy token on Uniswap,
sell token on OtherDex]



Atomic Transactions

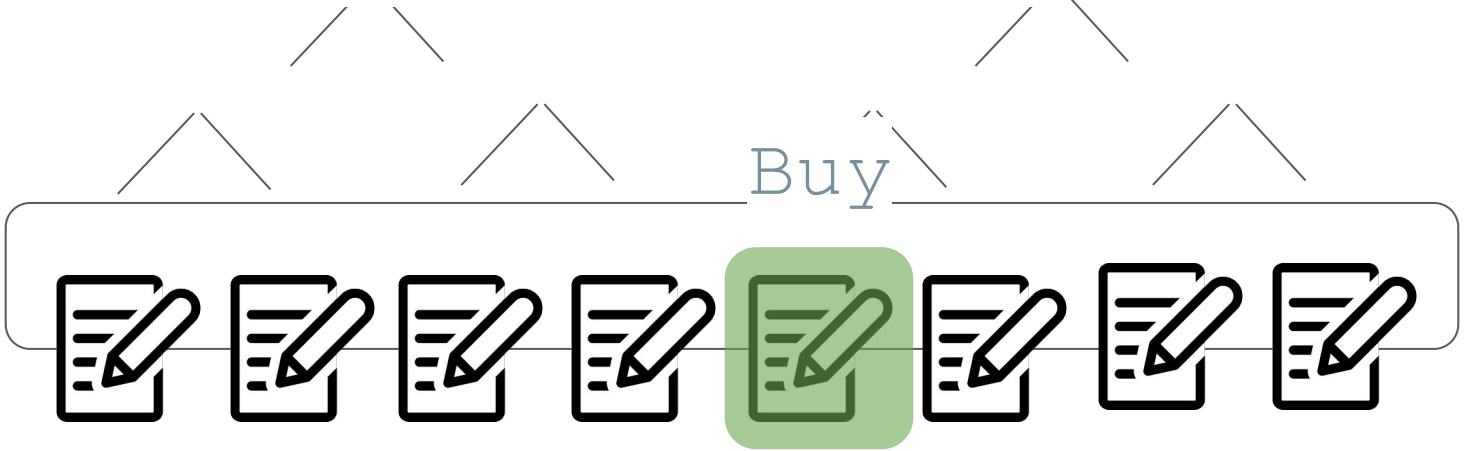
All or nothing!

Atomic transaction:
[buy token on Uniswap,
sell token on OtherDex]

Atomic Transactions

All or nothing!

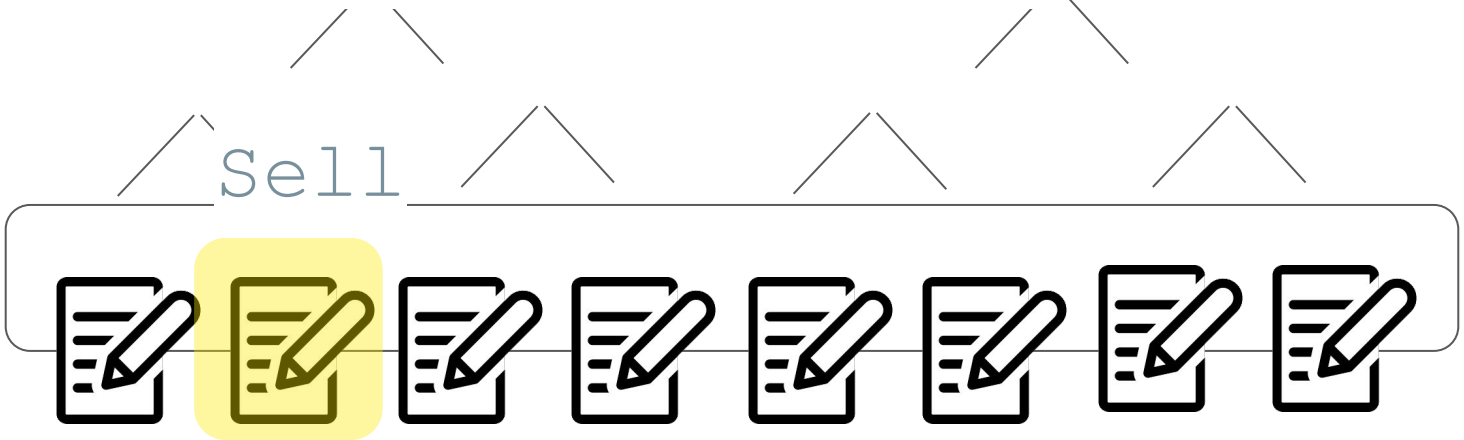
Atomic transaction:
[buy token on Uniswap,
sell token on OtherDex]



Atomic Transactions

All or nothing!

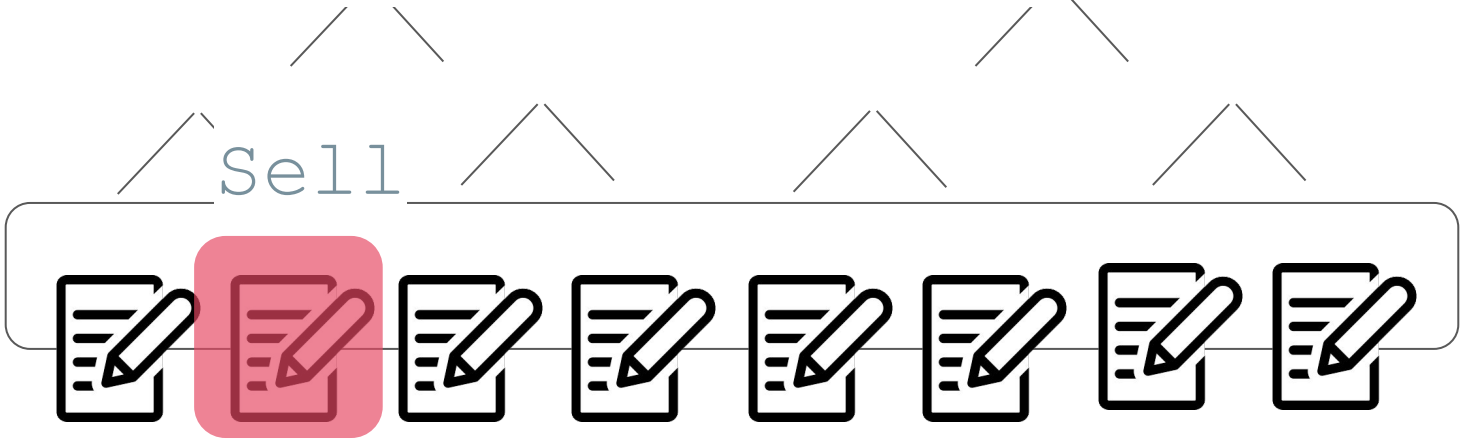
Atomic transaction:
[buy token on Uniswap,
 sell token on OtherDex]



Atomic Transactions

All or nothing!

Atomic transaction:
[buy token on Uniswap,
 sell token on OtherDex]



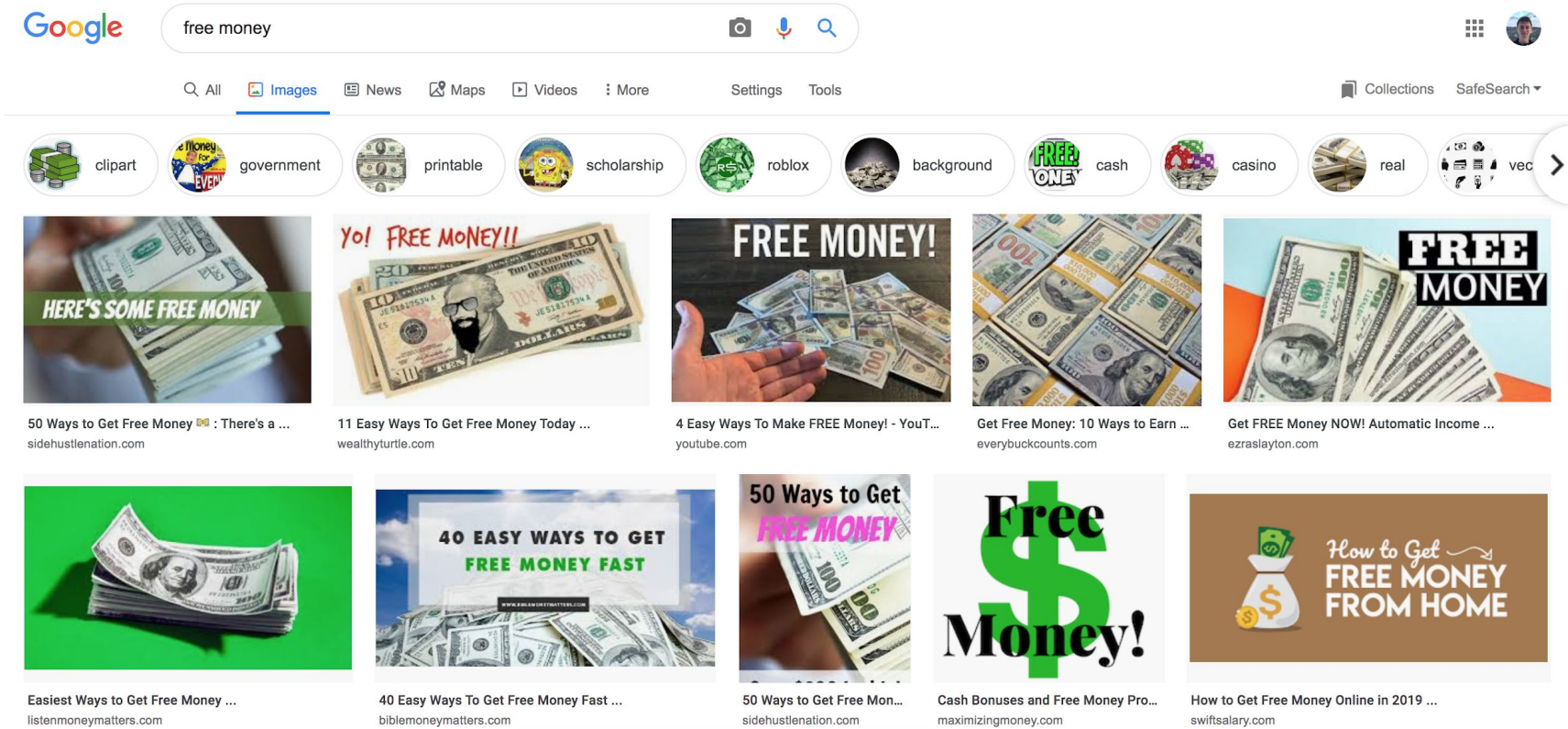
Atomic Transactions

All or nothing!

Atomic transaction
[buy token on
sell toke

REVERT

Atomic Transactions



Agenda

1. What do we mean by tightly coupled Ethereum?
2. What are our options for tightly coupling ETH2?
3. How will DeFi look on scalable Ethereum?

Agenda

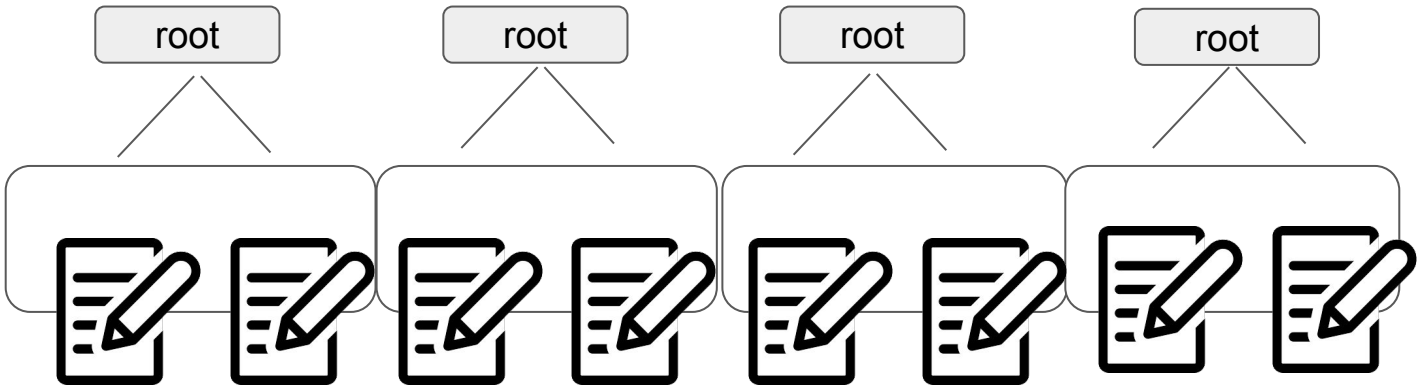
1. What do we mean by tightly coupled Ethereum?
2. What are our options for tightly coupling ETH2?
3. How will DeFi look on scalable Ethereum?

DeFi in ETH2: Why are we worried?

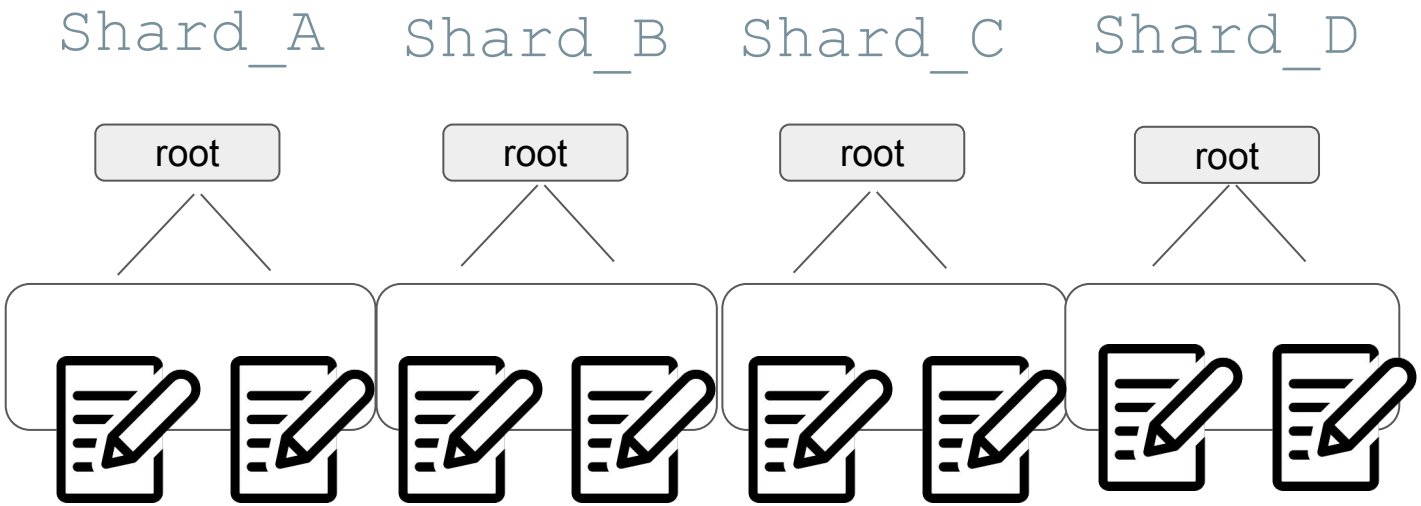
DeFi in ETH2: Why are we worried?

- **Multiple state spaces.**
  - Shard_a, shard_b, shard_c, etc.
  - What do I deploy to?

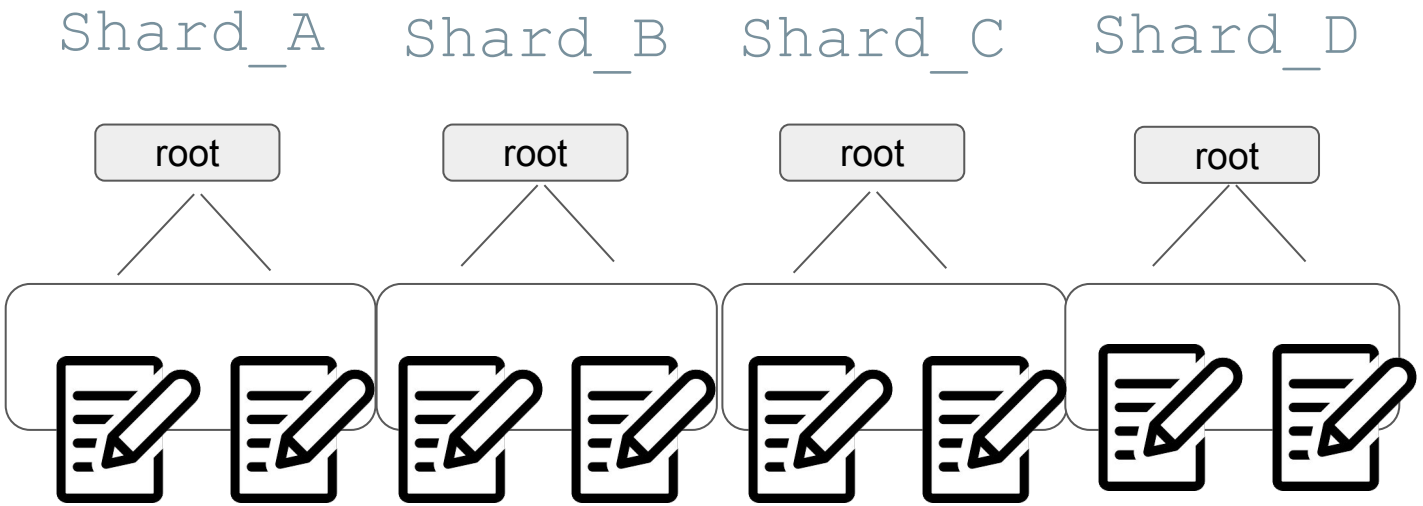
Multiple State Spaces



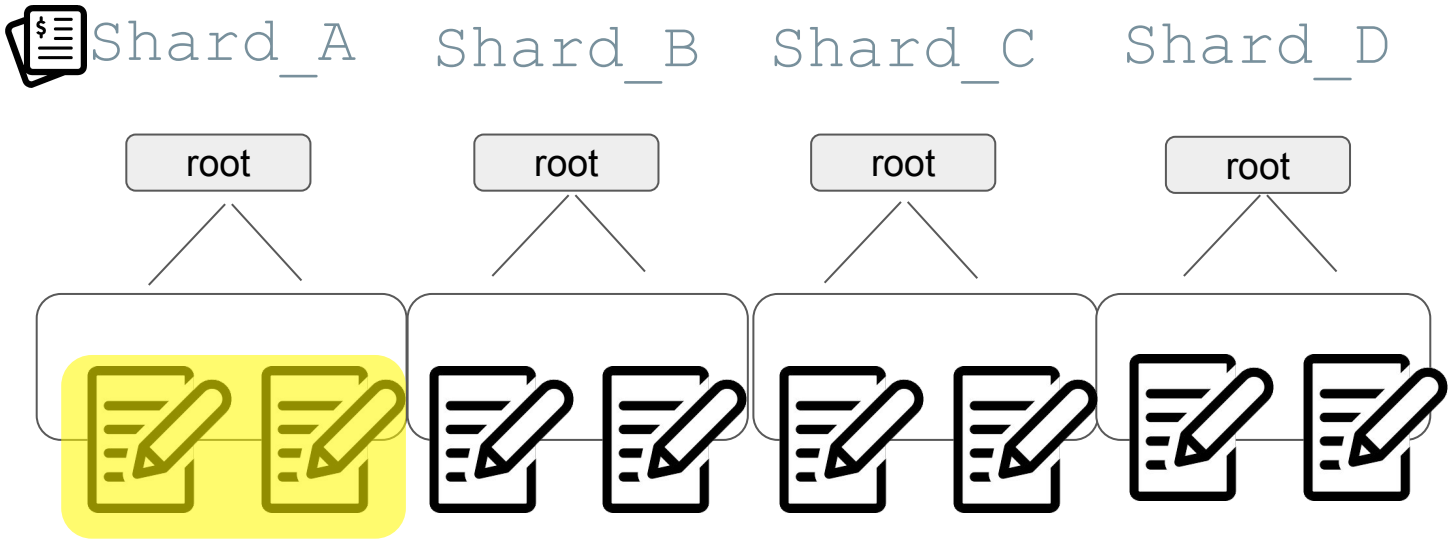
Multiple State Spaces



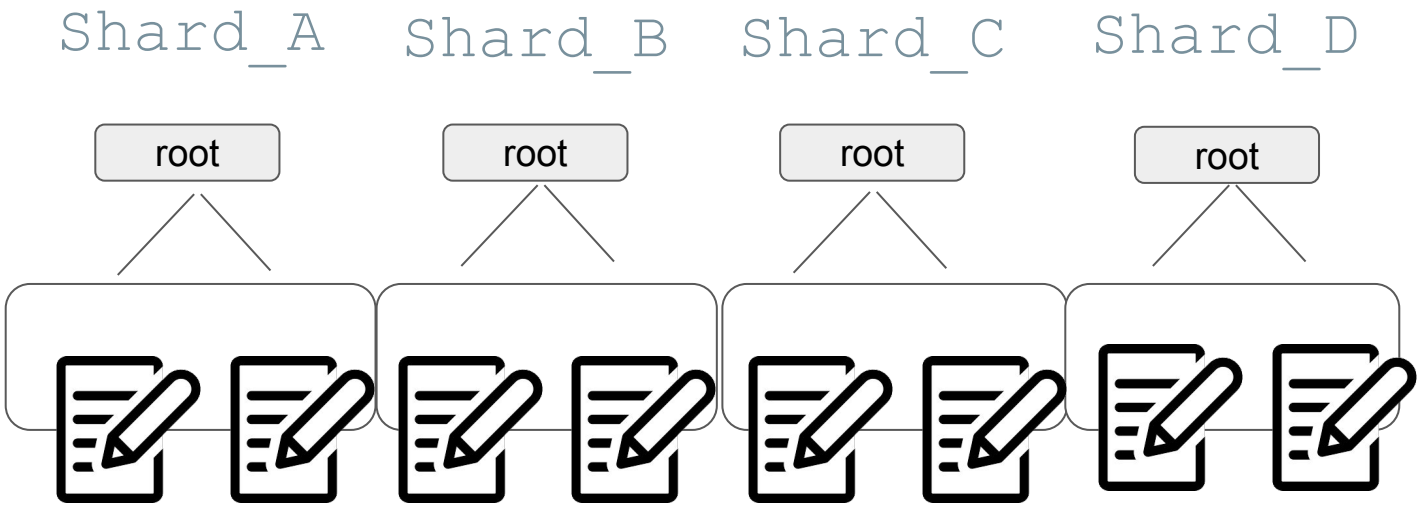
Multiple State Spaces



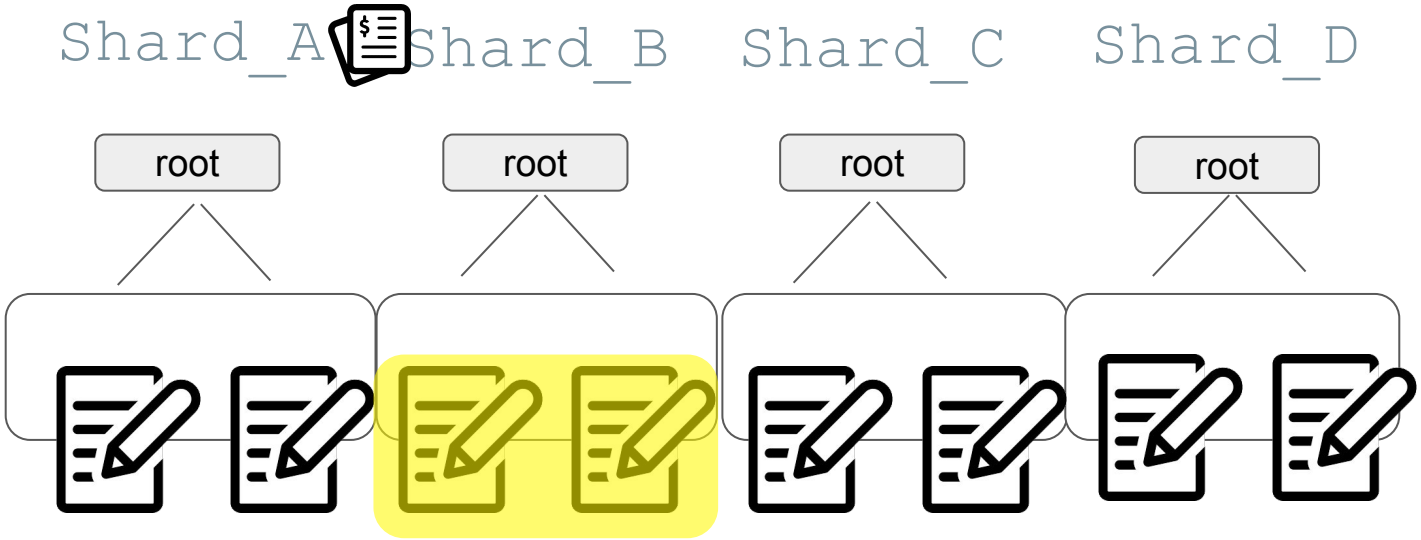
Multiple State Spaces



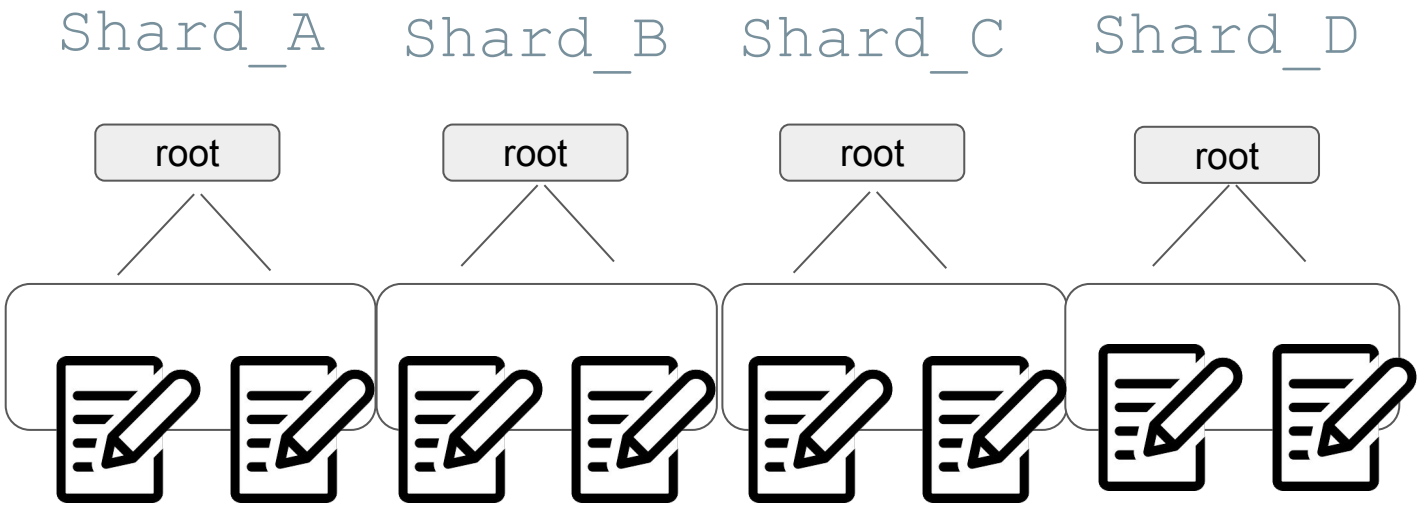
Multiple State Spaces



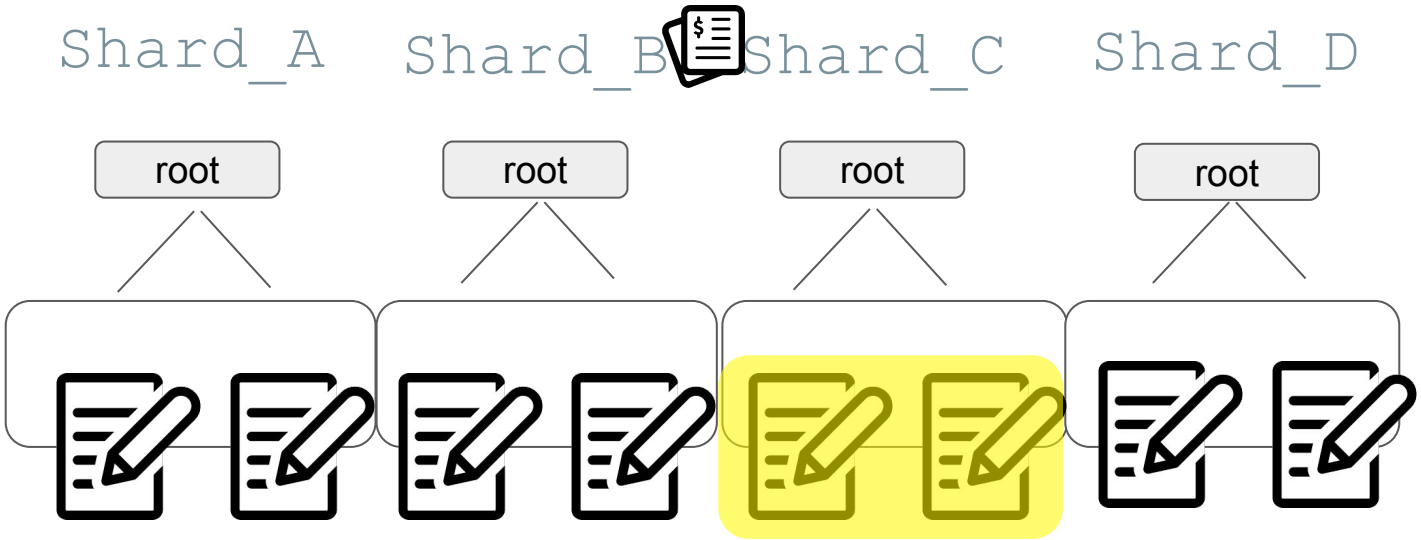
Multiple State Spaces



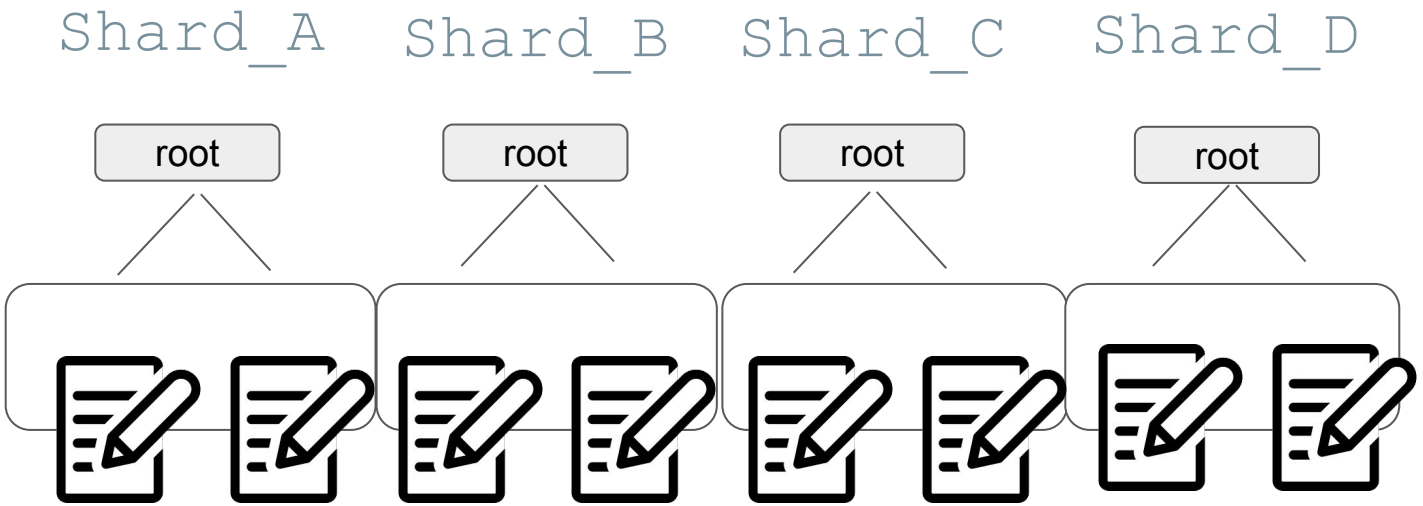
Multiple State Spaces



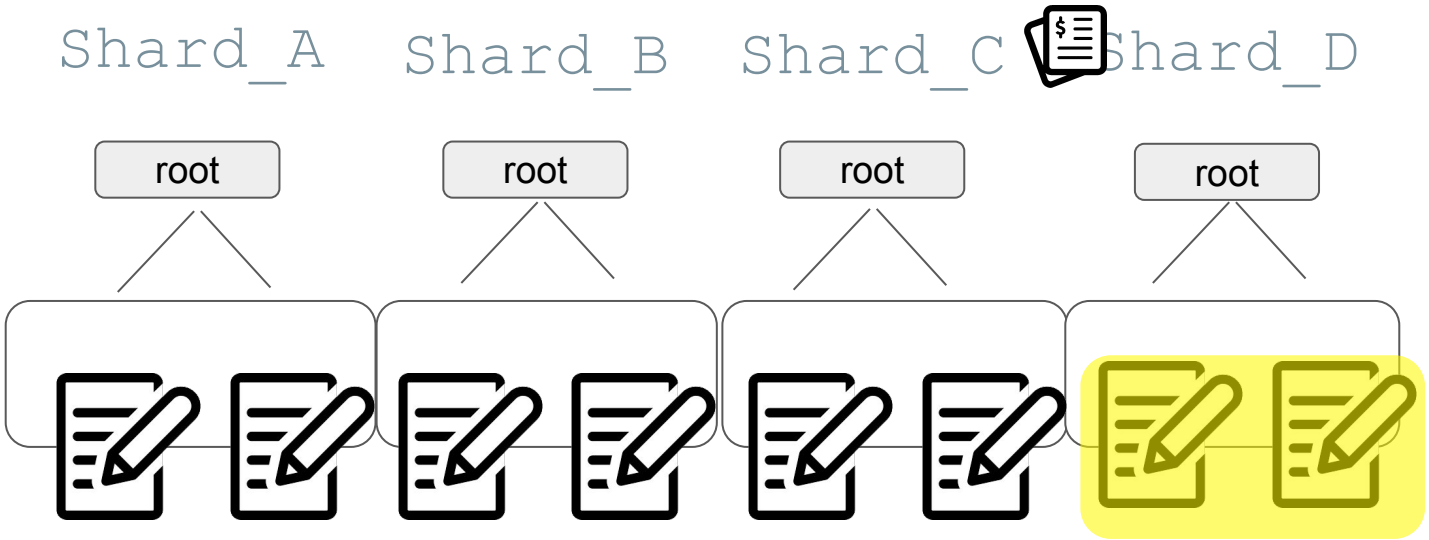
Multiple State Spaces



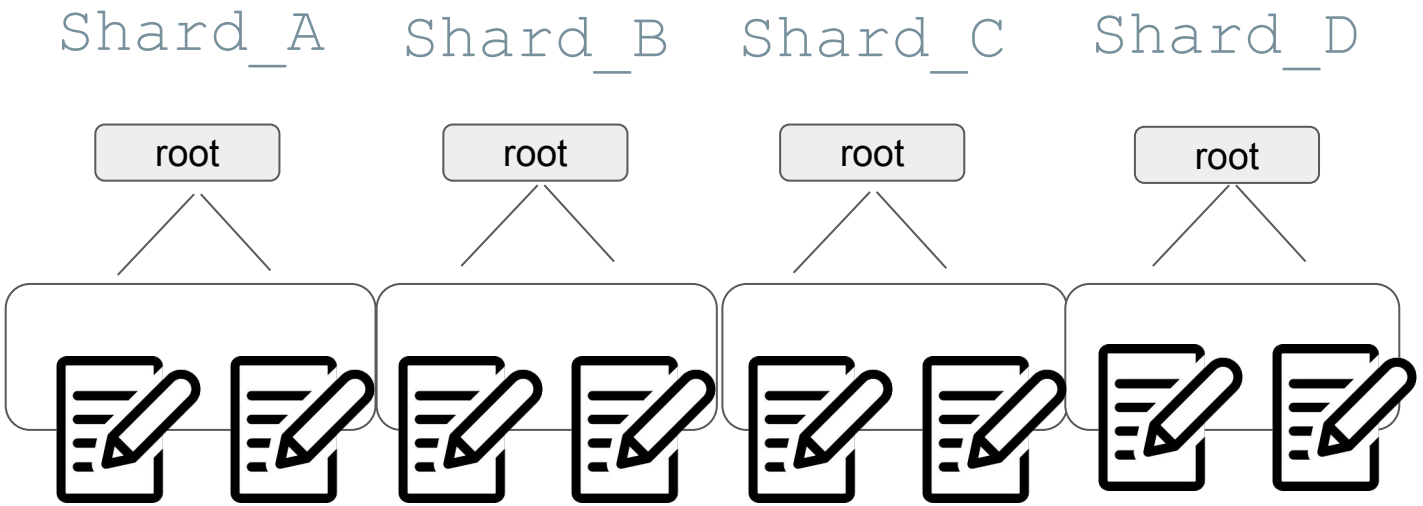
Multiple State Spaces



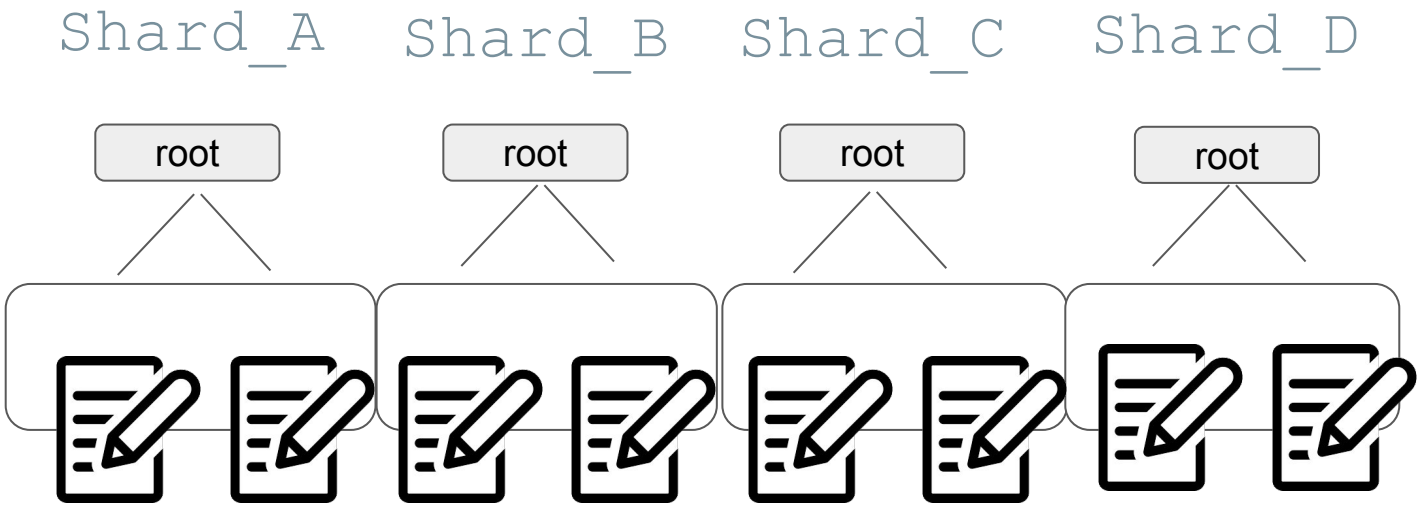
Multiple State Spaces



Multiple State Spaces



Multiple State Spaces

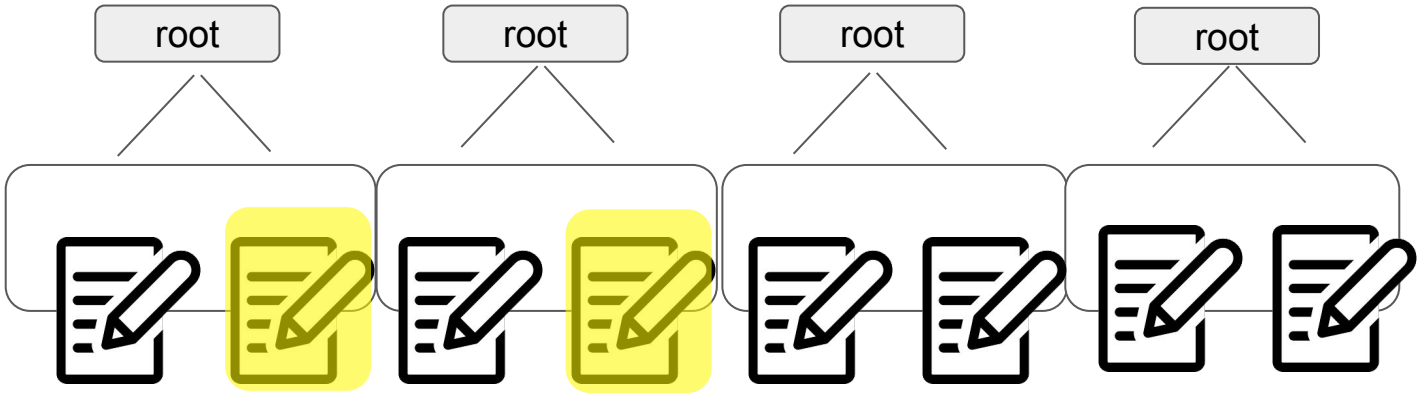


DeFi in ETH2: Why are we worried?

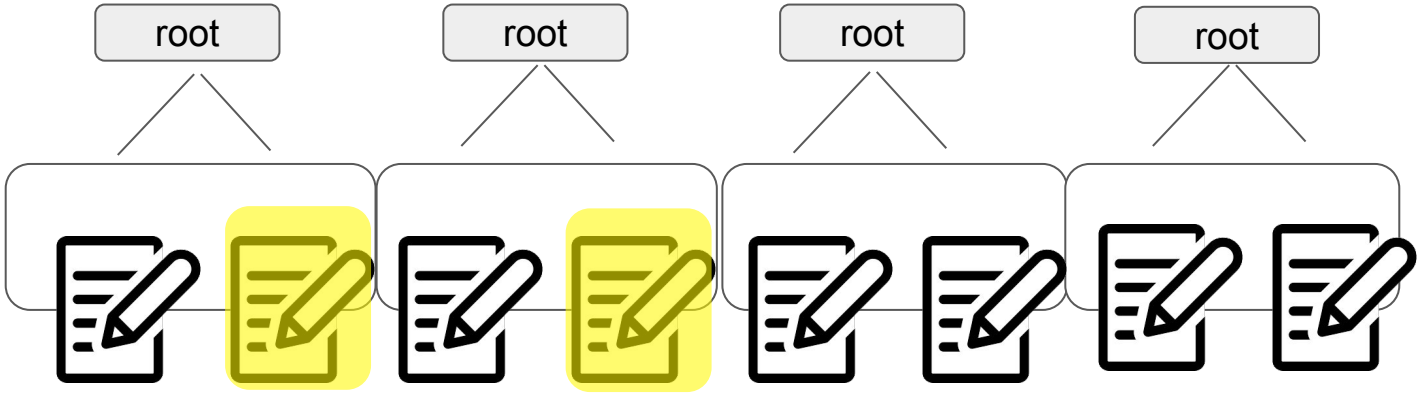
- **Multiple state spaces.**
  - Shard_a, shard_b, shard_c, etc.
  - What do I deploy to?
- **Atomic txs have added complexity.**
  - Do we enforce in L1?
  - Do we enforce in L2?

Atomic Cross Shard Communication

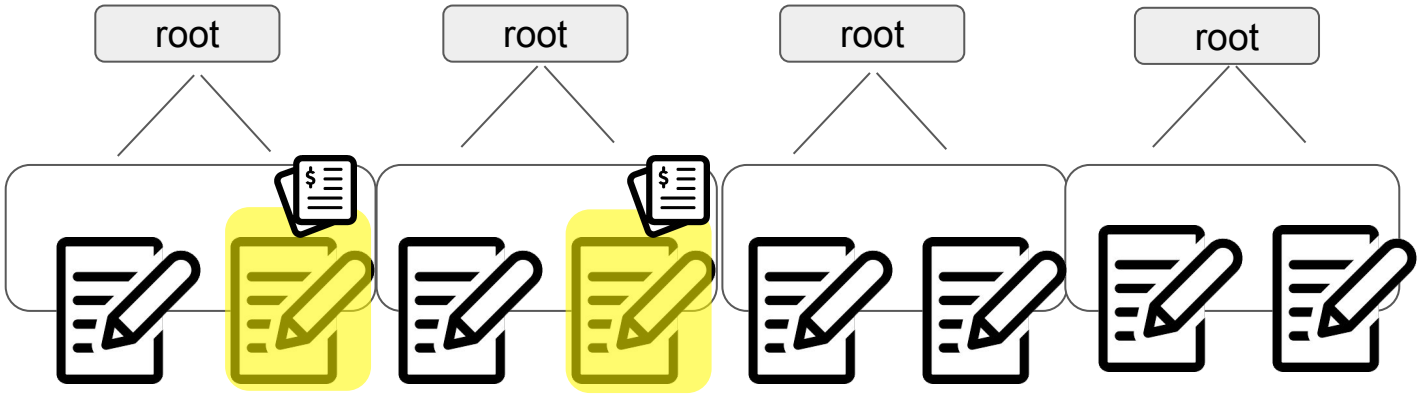
Atomic Cross Shard Communication



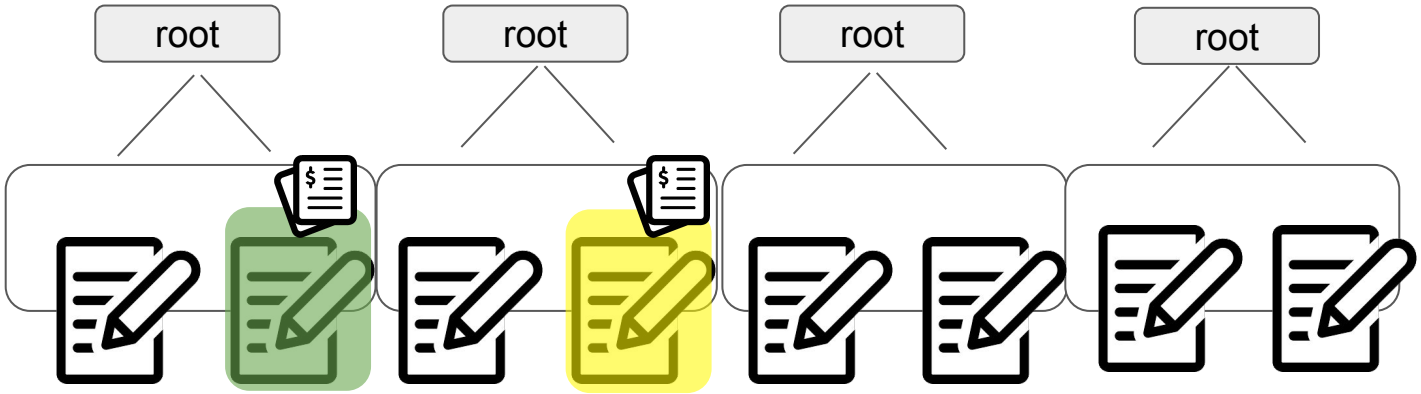
Atomic Cross Shard Communication



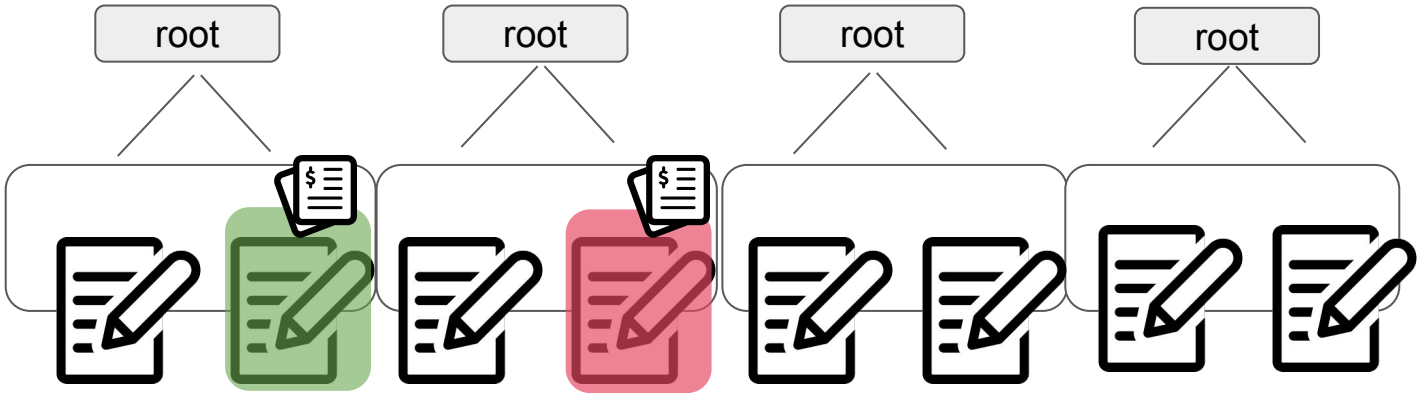
Atomic Cross Shard Communication



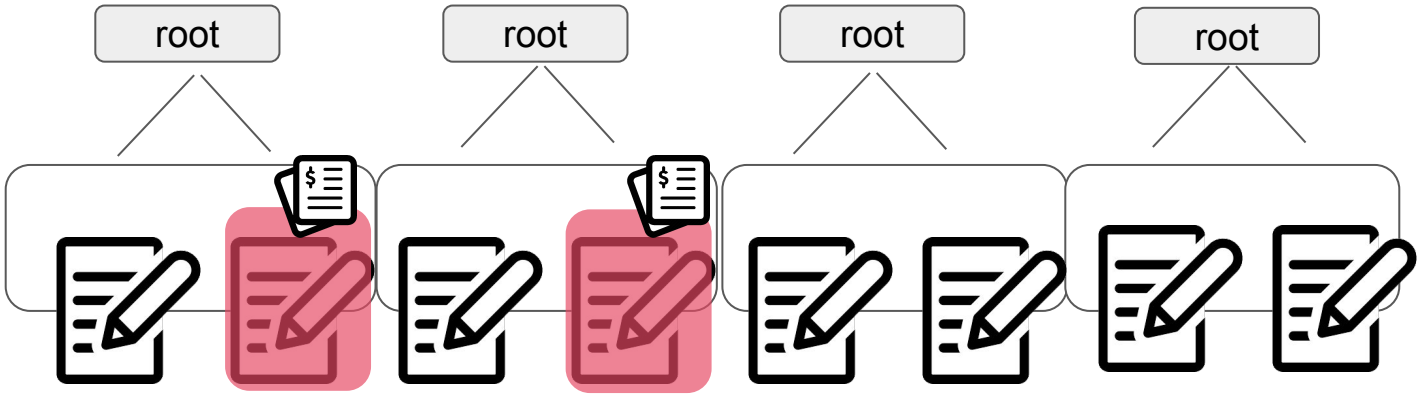
Atomic Cross Shard Communication



Atomic Cross Shard Communication

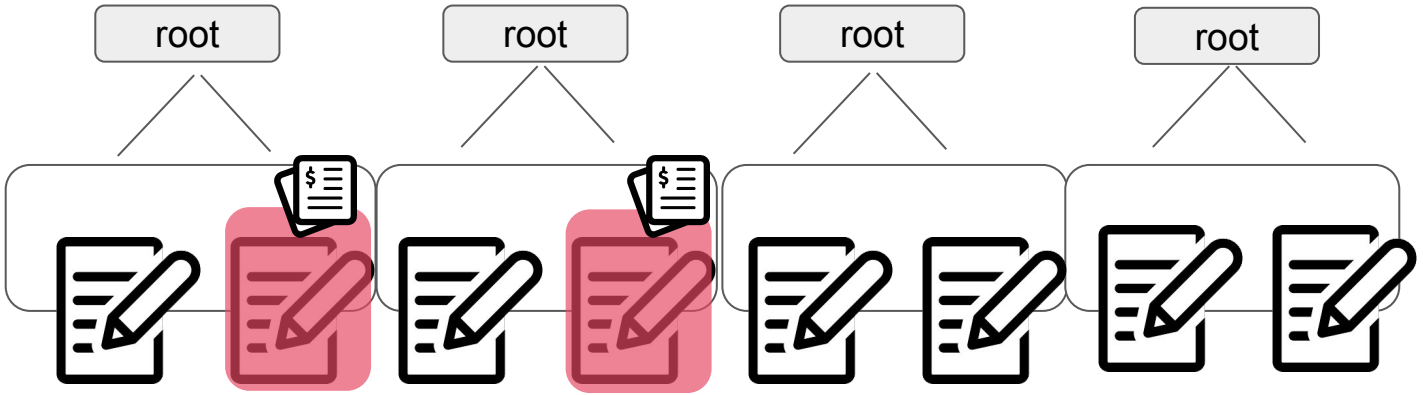


Atomic Cross Shard Communication



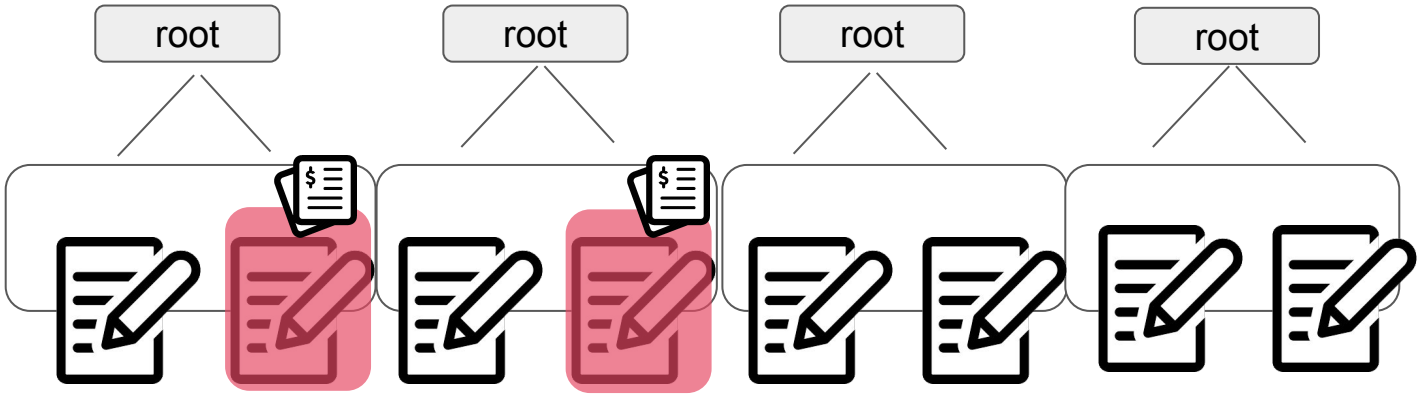
Atomic Cross Shard Communication

Train & hotel problem!



Atomic Cross Shard Communication

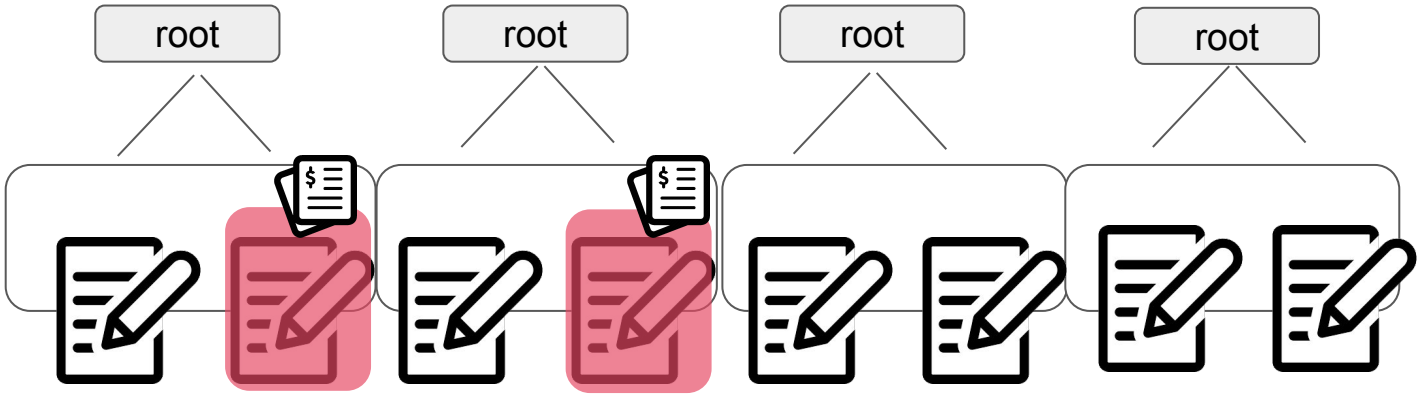
~~Train & hotel problem!~~



Atomic Cross Shard Communication

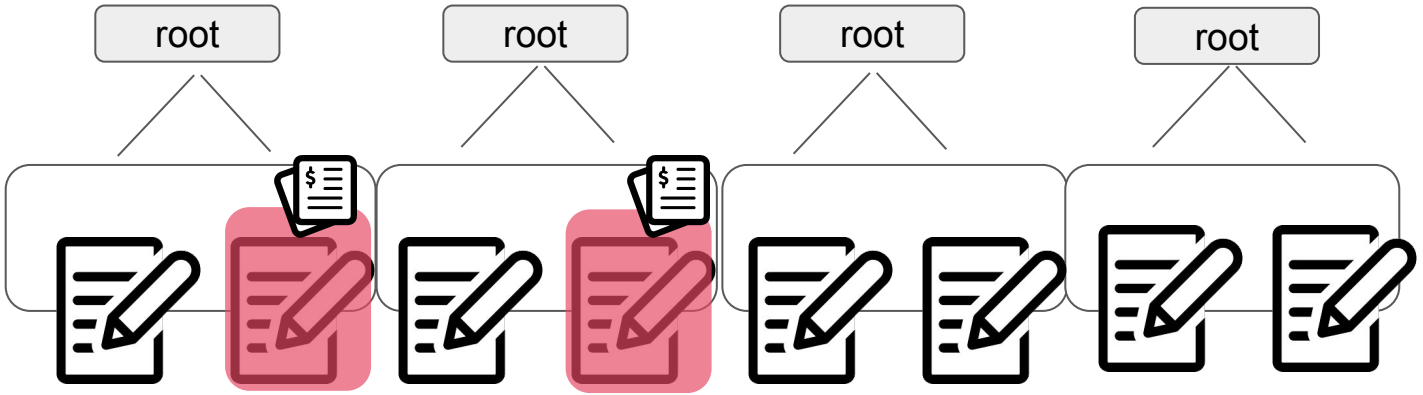
Risk-free arbitrage problem!

~~Train & hotel problem!~~



Atomic Cross Shard Communication

How do we achieve this?



Atomic Cross Shard Communication

How do we achieve this?

Atomic Cross Shard Communication

How do we achieve this?

1) Layer 1, or (and) 2) Layer 2

Layer 1 Cross Shard

research

Receipts!

Cross-shard contract yanking

Sharding cross-shard

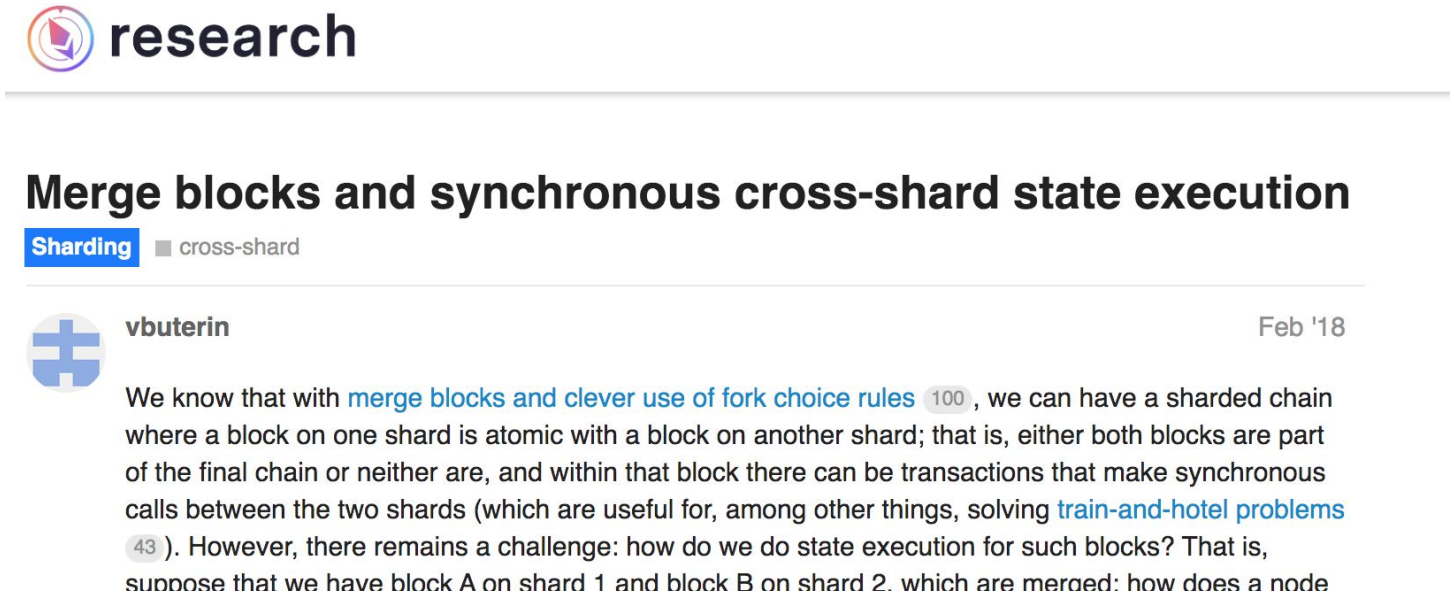
vbuterin 1 Mar '18

Special thanks to Piper Merriam for helping to come up with this idea

This is a generalization of cross-shard locking schemes and similar techniques to enabling cross-shard activity to solve train-and-hotel problems. Philosophically speaking, “locking” a contract that exists on shard A effectively means freezing its state on shard A, saving the state into a receipt, importing the contract to shard B, performing some operation between that contract and some other object on shard B, then using another receipt to send the contract back to shard A where it then continues its existence.

We can simplify and generalize this by changing the mechanism from “locking” to “yanking”. We add an opcode to the EVM, YANK (one stack argument: `target_shard`), which deletes the contract from the state and issues a receipt that contains the state of the contract and the `target_shard`; this receipt can then be processed in the `target_shard` to instantiate the same contract in that shard. Contracts are free to specify their own conditions for when they get yanked.

Honorable Mention: Merge Blocks



Layer 2 Cross Shard

research

Fast cross-shard transfers via optimistic receipt roots

Sharding ■ phase-2 ■ cross-shard

vbuterin 4 Apr 21

TLDR: given a base layer blockchain that offers slow asynchronous cross-shard transactions, we can emulate fast asynchronous cross-shard transactions as a layer on top.

This is done by storing “conditional states”, for example if Bob has 50 coins on shard B, and Alice sends 20 coins to Bob from shard A, but shard B does not yet know the state of shard A and so cannot fully authenticate the transfer, Bob’s account state temporarily becomes “70 coins if the transfer from Alice is genuine, else 50 coins”. Clients that have the ability to authenticate shard A and shard B can be sure of

Layer 2 Cross Shard

research

Fast cross-shard transfers via optimistic receipt roots

Sharding ■ phase-2 ■ cross-shard

vbuterin 4 Apr 21

TLDR: given a base layer blockchain that offers slow asynchronous cross-shard transactions, we can emulate fast asynchronous cross-shard transactions as a layer on top.

This is done by storing “conditional states”, for example if Bob has 50 coins on shard B, and Alice sends 20 coins to Bob from shard A, but shard B does not yet know the state of shard A and so cannot fully authenticate the transfer, Bob’s account state temporarily becomes “70 coins if the transfer from Alice is genuine, else 50 coins”. Clients that have the ability to authenticate shard A and shard B can be sure of

Honorable Mentions ...

Honorable Mentions ...

research

Phase 2 pre-spec: cross-shard mechanics

Eth2 phase 2 ■ phase-2 ■ cross-shard

vbuterin 5 Feb 11

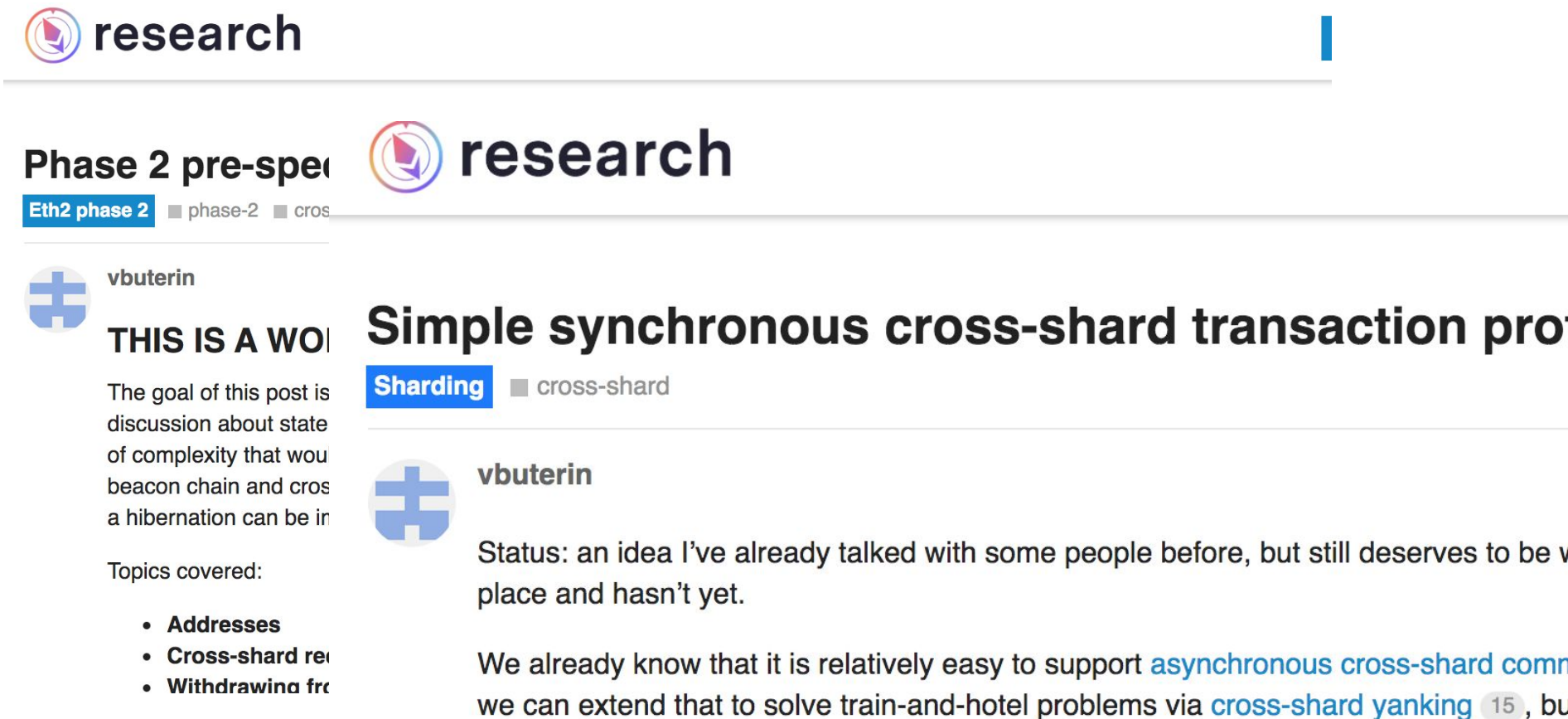
THIS IS A WORK IN PROGRESS!

The goal of this post is to provide a rough outline of what phase 2 might look like, to help make the discussion about state and execution more concrete as well as to give us an idea of the level and types of complexity that would be involved in implementing it. This section focuses on withdrawals from the beacon chain and cross-shard calls (which use the same mechanism); rent is unspecified but note that a hibernation can be implemented as a forced cross-shard call to the same shard.

Topics covered:

- **Addresses**
- **Cross-shard receipt creation and inclusion**
- **Withdrawing from the beacon chain**

Honorable Mentions ...



Honorable Mentions ...

research

research

Phase

Eth2 phase

vb

T

Th
di
of
be
a

To

Synchronous cross-shard transactions with consolidated concurrency co and consensus (or how I rediscovered Chain Fibers)

Sharding cross-shard

cdetrio Jun '18

I'd like to argue against a widespread perception that under the initial sharding protocol, the only way to do cross-shard transactions will be with asynchronous messages (i.e. contract locking/yanking, and so on). I do agree that if we expect the least amount of innovation from protocol engineers, then asynchronous transactions are what we'll get (asynchronous transactions are the solution if protocol engineers shirk it off and leave it as a problem to be solved by users, contract authors and dapp devs). But it would be a shame to accept a "worse is better" fate. Instead, I want to highlight some insight why solving the problem at the protocol layer, enabling synchronous cross-shard transactions that would

we can extend that to solve train-and-hotel problems via cross-shard yanking 15, bu

Honorable Mentions ...

research

Layer 2 state schemes

Sharding

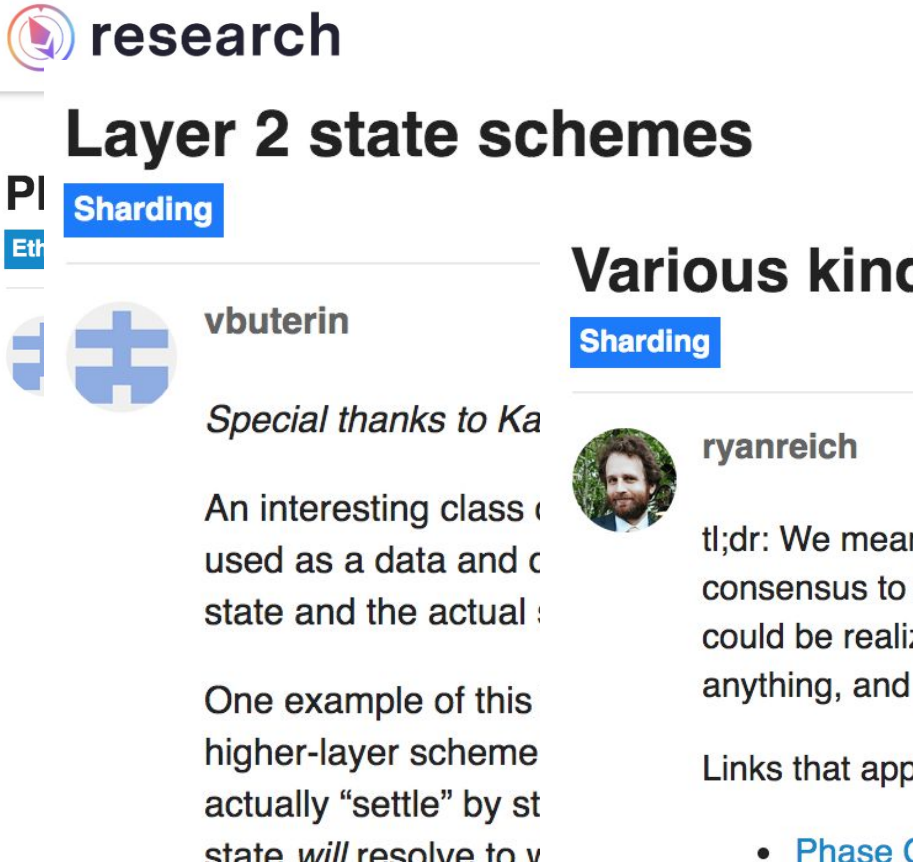
vbuterin Jul

Special thanks to Karl Floersch and Ben Jones for ideas around Plasma with on-chain publication

An interesting class of constructions that is starting to appear for eth2 is that where the blockchain is used as a data and computation layer, but there is not a trivial direct mapping between the blockchain state and the actual state (eg. account balances) that users care about.

One example of this is A layer 2 computing model using optimistic state roots 51, where we create a higher-layer scheme to pretend that cross-shard transactions happen much more quickly than they actually "settle" by storing dependency graphs as part of the state and letting clients figure out what the state *will* resolve to when all the dependencies settle

Honorable Mentions ...



research

Layer 2 state schemes

Sharding

vbuterin

Special thanks to Ka

An interesting class
used as a data and
state and the actual

One example of this
higher-layer scheme
actually "settle" by st
state *will* resolve to

Various kinds of scaling, execution models, a

Sharding

ryanreich

tl;dr: We mean a lot of things when we say "scalability", and we can't ignore
consensus to support a particular execution model scalably, we are missing
could be realized by formulating a more ideal execution model. Ultimately, w
anything, and gain cool stuff like multi-chain communication and separate ex

Links that appear below:

- Phase One and Done 7

Honorable Mentions ...

TOO MUCH!!

So what’s the difference?

- **Layer 1**
  - Simple(r)
  - Can be slower (based on cross-link time)
- **Layer 2**
  - Complex gas payments!
    - Can require DEX between shards
  - Extremely flexible

Agenda

1. What do we mean by tightly coupled Ethereum?
2. What are our options for tightly coupling ETH2?
3. How will DeFi look on scalable Ethereum?

Agenda

1. What do we mean by tightly coupled Ethereum?
2. What are our options for tightly coupling ETH2?
3. How will DeFi look on scalable Ethereum?

What is the future?

What is the future?

What is the future?

What is the future?

Layer 1

Layer 2

What is the future?

Layer 1

Layer 2

What is the future?

Layer 1

Simplest cross shard tx scheme adopted.

Layer 2

What is the future?

Layer 1

Simplest cross shard tx scheme adopted.

Layer 2

What is the future?

Layer 1

Simplest cross shard tx scheme adopted.

Layer 2

Optimistic rollup & plasma adopt fun L2 cross shard TXs schemes.

What is the future?

Layer 1

Don’t ask permission.

You won’t need forgiveness.

Layer 2

Optimistic rollup & plasma adopt fun L2 cross shard TXs schemes.

What is the future?

Layer 1

Don't ask permission.

You won't need

Layer 2

Optimistic rollup & plasma adopt fun L2 cross shard TXs schemes.

Wait, what's "Optimistic Rollup"?

What is the future?

So glad you asked! ;)

Layer 1

Don't [illegible]ion.heme
[illegible]pted.

You won't need

Layer 2

Optimistic rollup & plasma adopt fun L2 cross shard TXs schemes.

Wait, what's "Optimistic Rollup"?

Optimistic Rollup! Unipig! Yay!

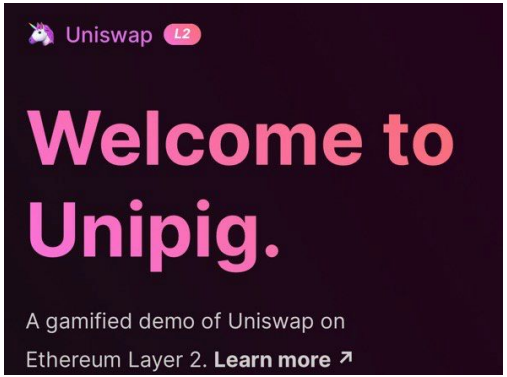
Optimistic Rollup! Unipig! Yay!

Optimistic Rollup! Unipig! Yay!

Optimistic Rollup! Unipig! Yay!

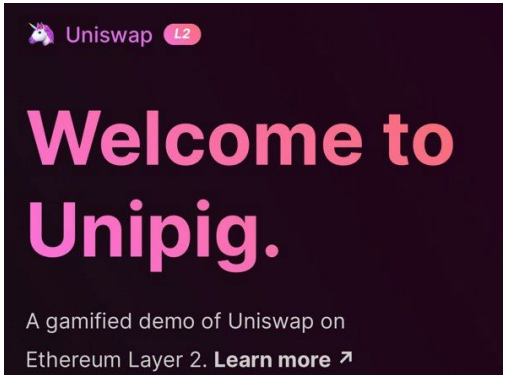
Uniswap

Optimistic Rollup! Unipig! Yay!



Uniswap

Optimistic Rollup! Unipig! Yay!



Uniswap

Uniswap implemented on Ethereum L2 using **Optimistic Rollup**!

DeFi in L2!

DeFi in L2!

Chat with us
if you want to
join Uniswap &
expand into
L2!

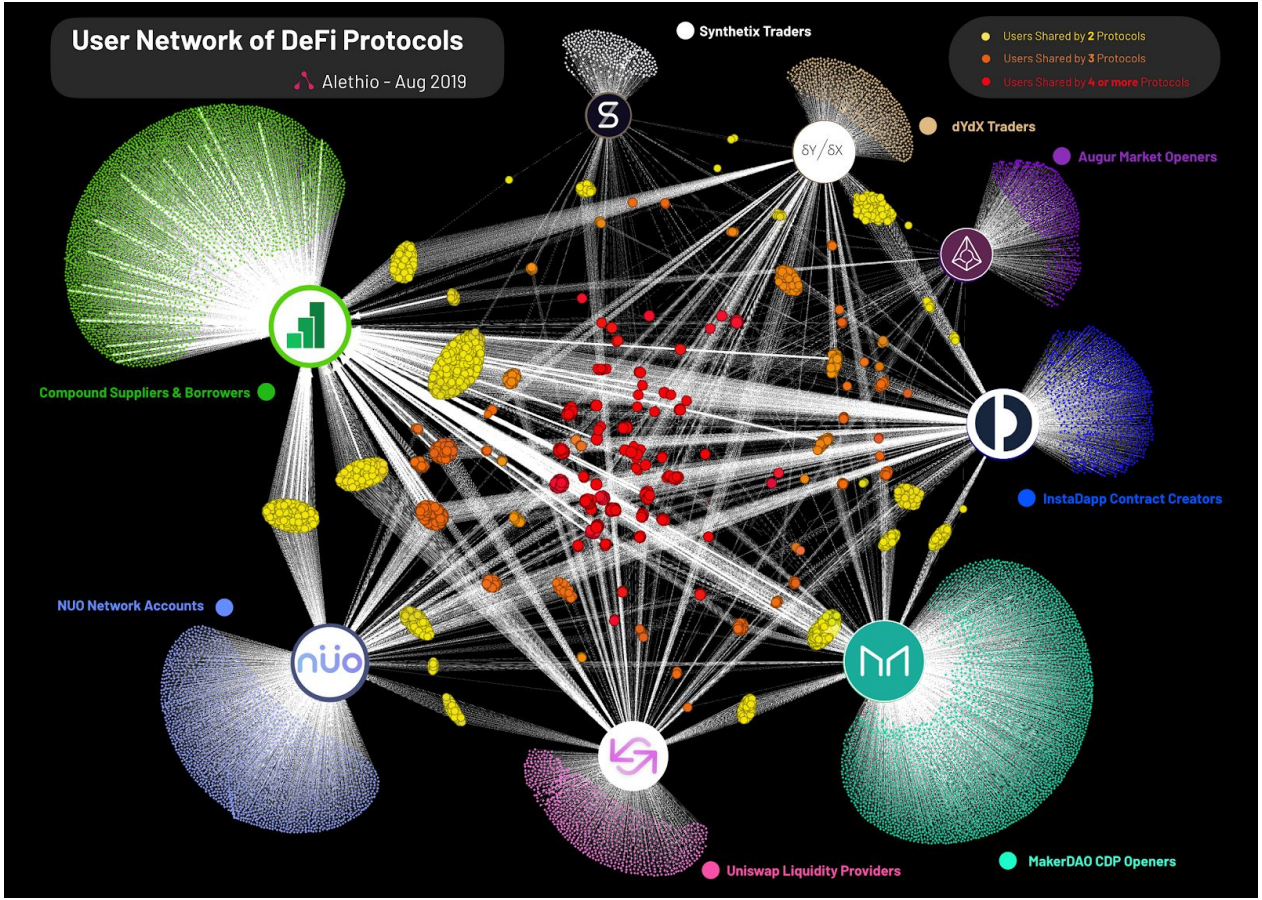
**We're all in
this together!**

DeFi in L2!

Chat with us if you want to join Uniswap & expand into L2!

We're all in this together!

DeFi in L2!


User Network of DeFi Protocols
Alethio - Aug 2019
Synthetix Traders
Users Shared by 2 Protocols
Users Shared by 3 Protocols
Users Shared by 4 or more Protocols
dYdX Traders
Augur Market Openers
Compound Suppliers & Borrowers
InstaDapp Contract Creators
NUO Network Accounts
Uniswap Liquidity Providers
MakerDAO CDP Openers

DeFi in L2!

Thank you!

plasma.group

Thank you!

plasma.group

Thank you!

plasma.group