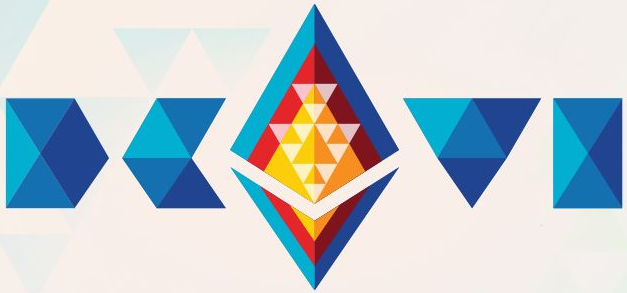


Challenges of Parallelizability

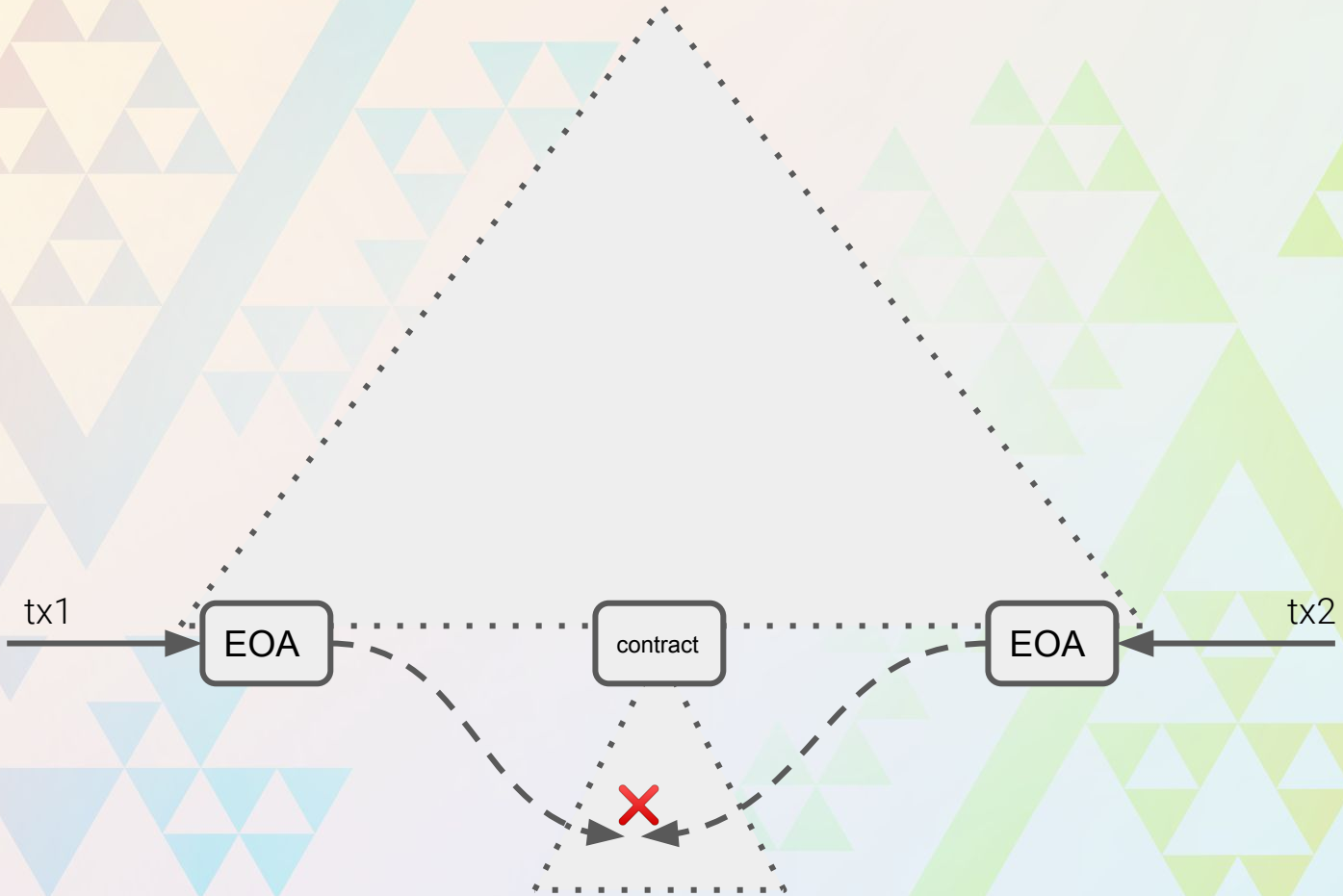
Under Ethereum's Execution Model

Péter Garamvölgyi

Infra @ Scroll

Section 1

Challenge I. **Dependencies**


tx1
EOA
contract
EOA
tx2

Ethereum’s dependencies can limit its parallel speedup at **as little as 4x**

ERC20 transfers to the same sender

```
balances[_to] += _value;
```

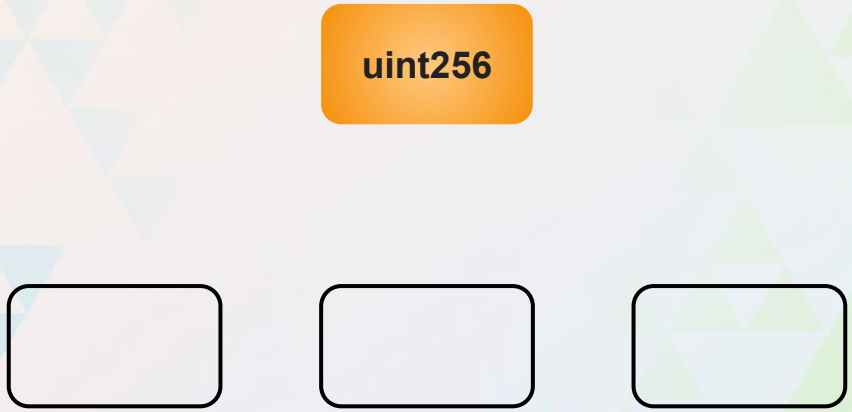
Uniswap swaps involving the same token pair

```
reserve0 = uint112(balance0);
```

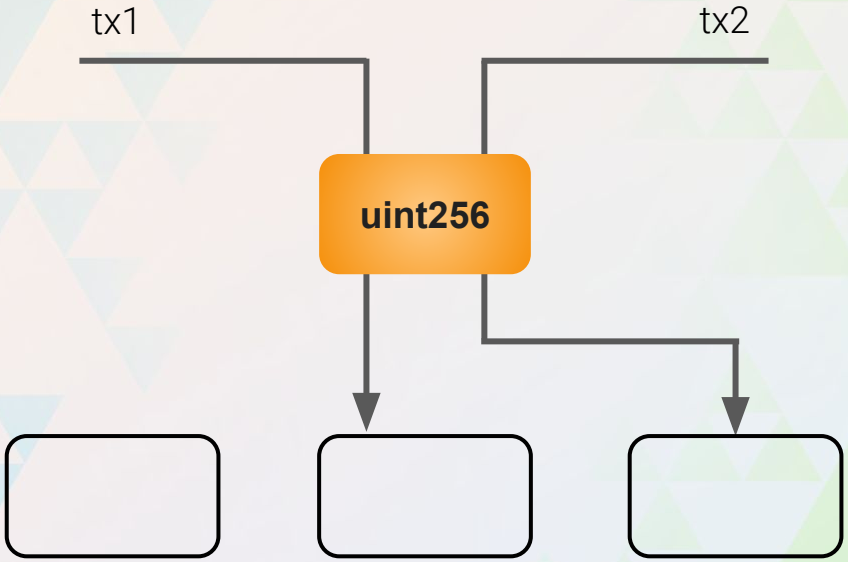
NFT minting transactions

```
_balances[to] += 1;
totalSupply += 1;
```

Reducing dependencies 1: **sharded counters**



Reducing dependencies 1: **sharded counters**

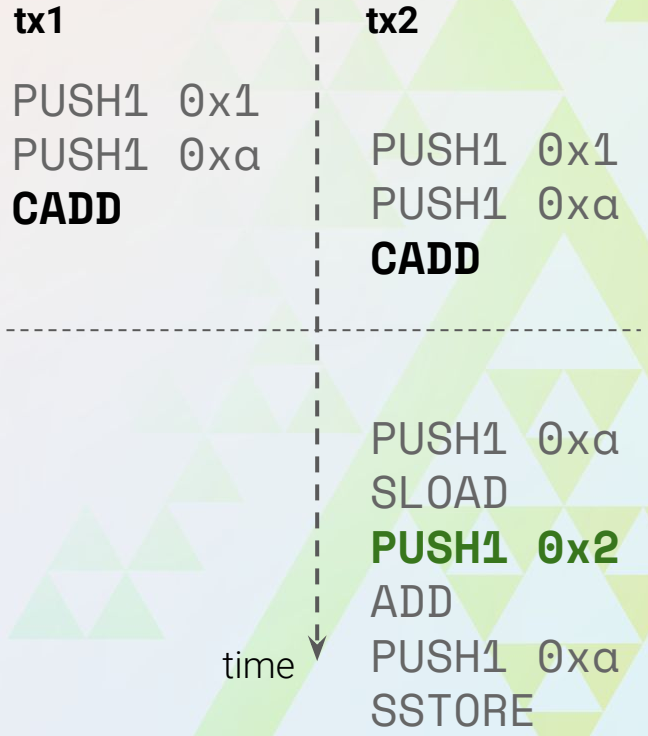


Reducing dependencies 2: **lazy add opcode**

```
function foo() {
  counter += 1;
}
```

Reducing dependencies 2: **lazy add opcode**

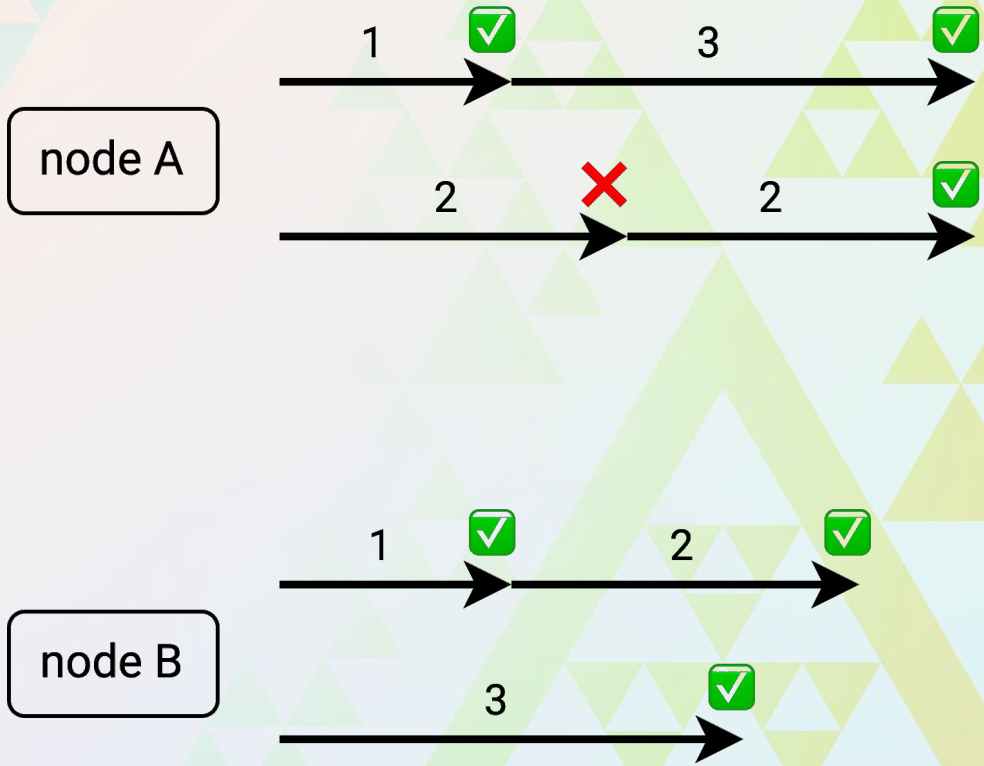
```
function foo() {
  counter += 1;
}
```

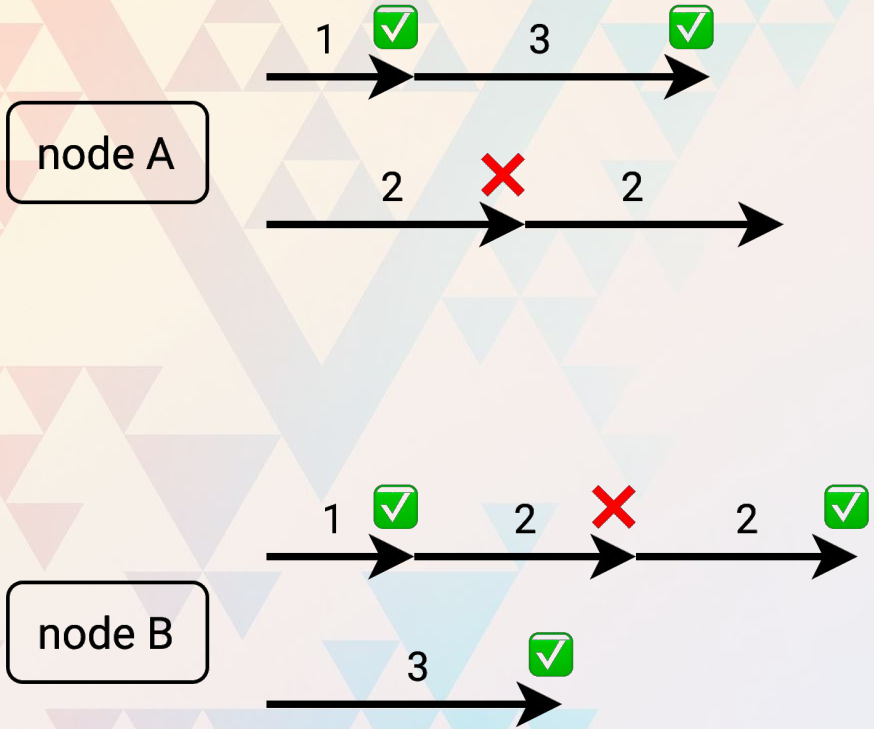


Section 2

Challenge II. **Determinism**

parallel schedules might be different on two nodes, even if their results are the same.



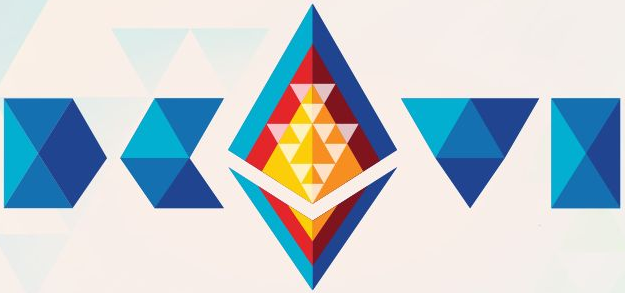


OCC-DA: OCC with Deterministic Aborts

a framework for fully deterministic parallel schedules.

it’s hard to get parallelization on the EVM right

-- but under higher throughput it will be worth it



Thank you!

Péter Garamvölgyi

Infra @ Scroll

peter@scroll.io

thegaram33

https://arxiv.org/pdf/2201.03749.pdf