

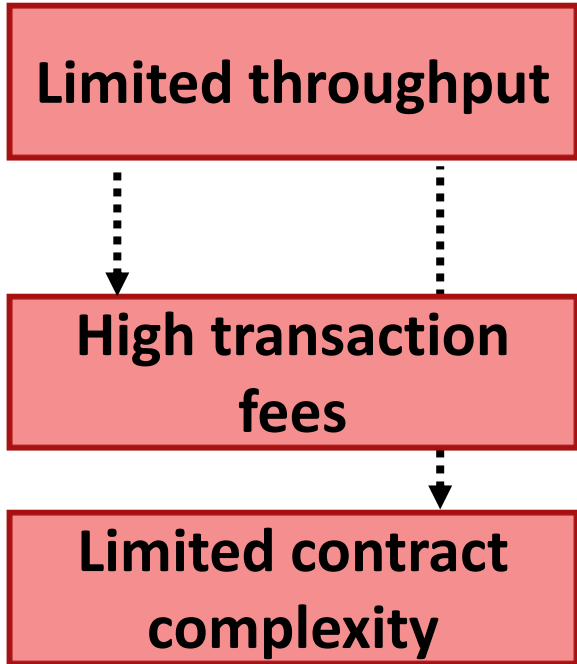
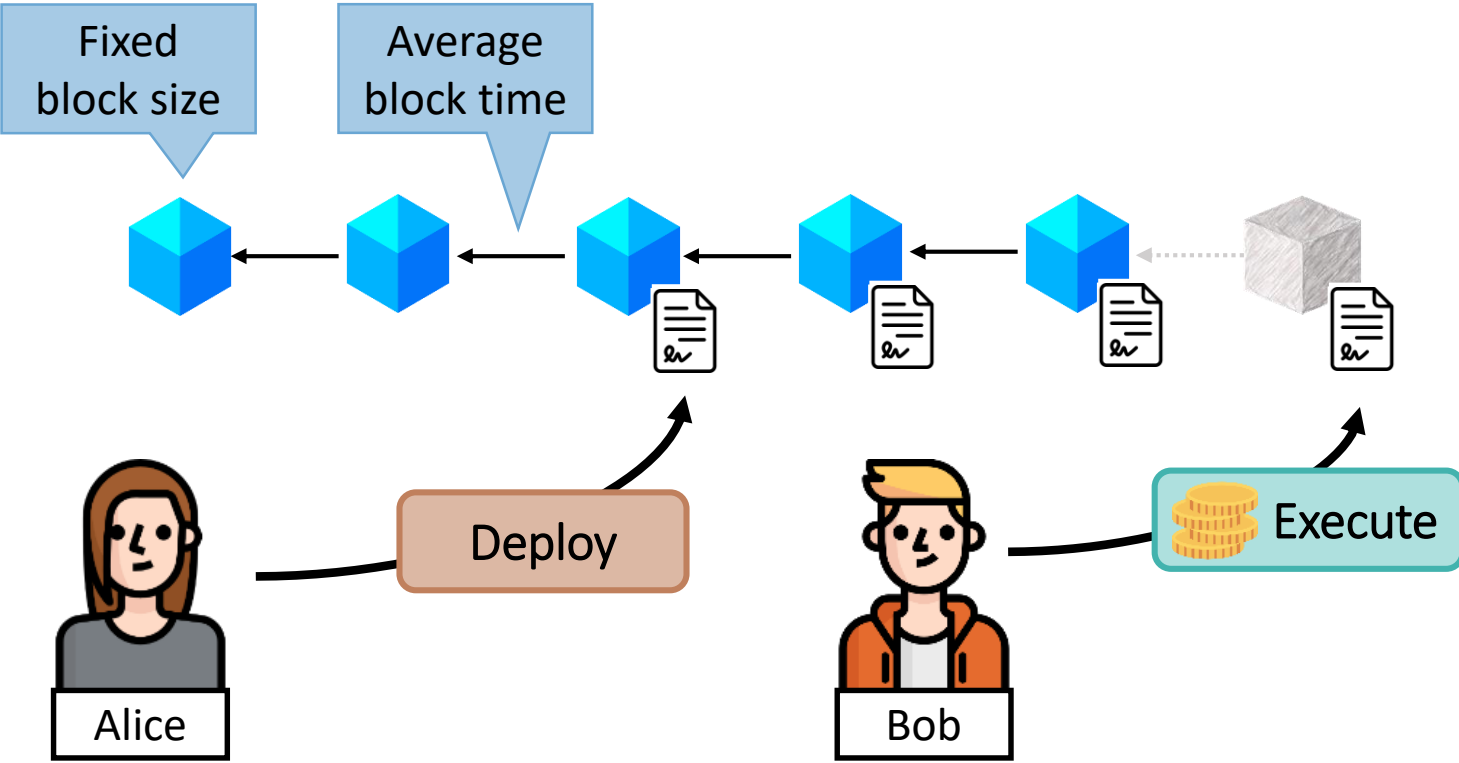
POSE: Practical Off-chain Smart Contract Execution

Tommaso Frassetto, Patrick Jauernig, David Koisser, David Kretzler, Benjamin Schlosser, Sebastian Faust and Ahmad-Reza Sadeghi



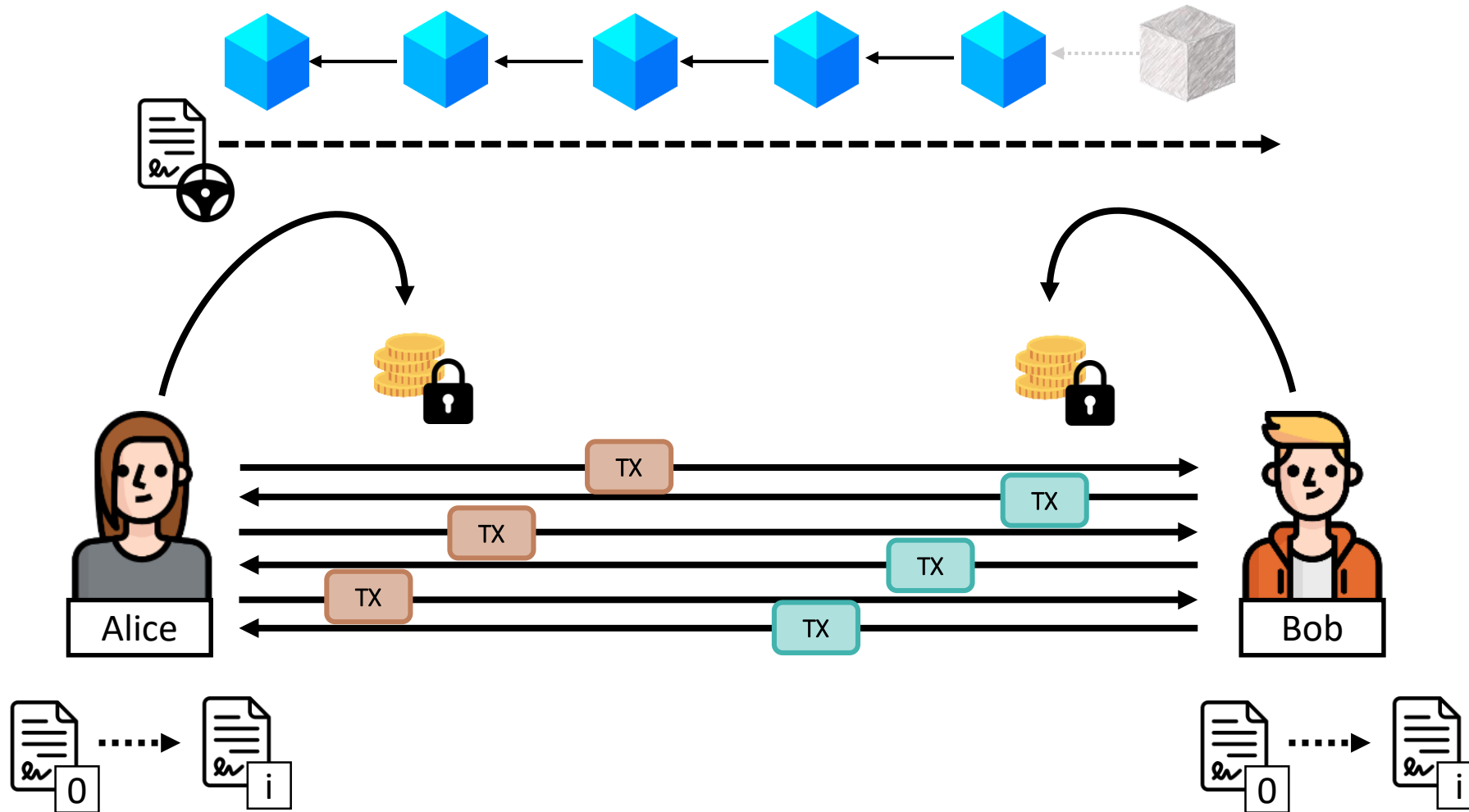
TECHNISCHE
UNIVERSITÄT
DARMSTADT

Blockchain Scalability Issues

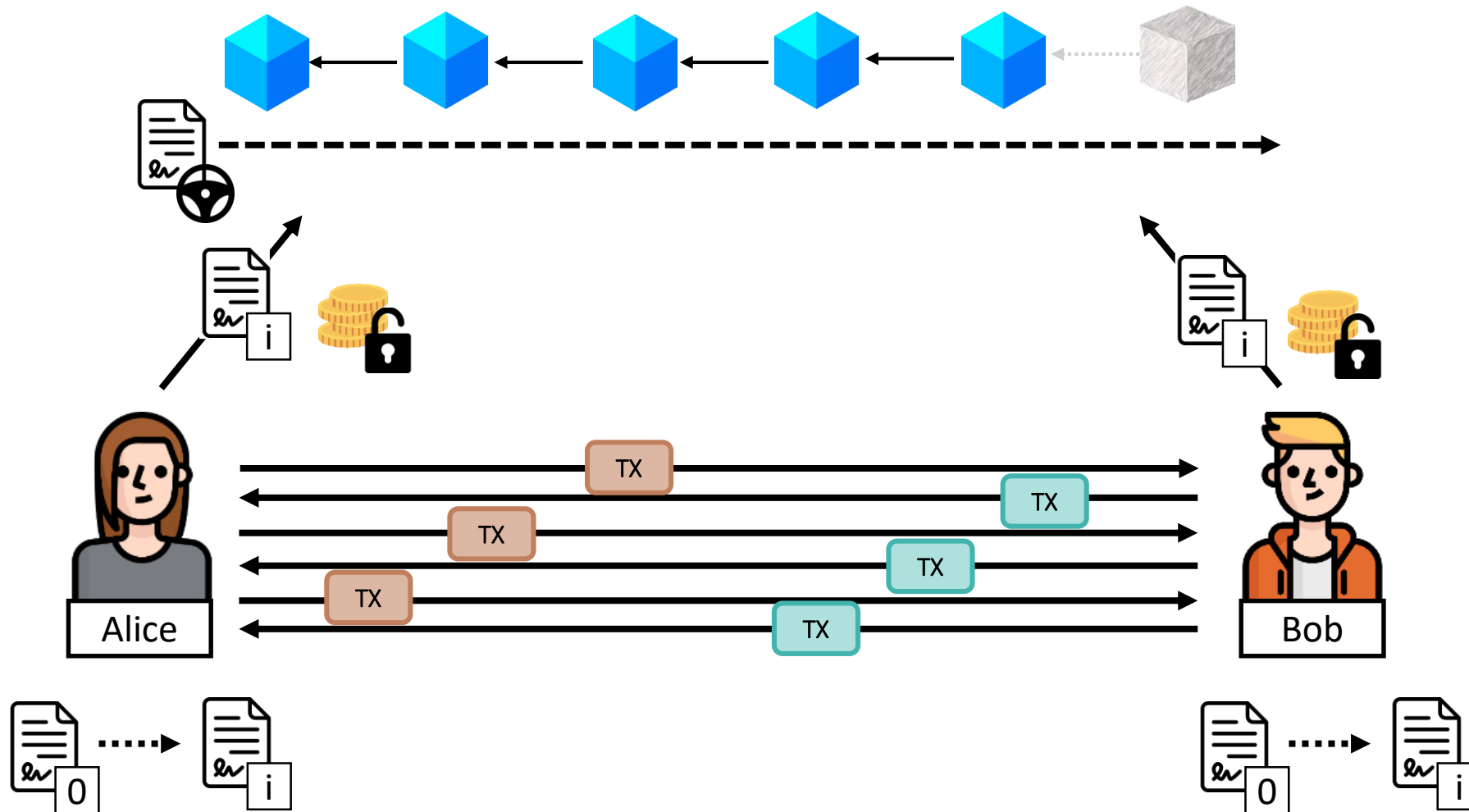


 **Idea:** Off-chain protocols!

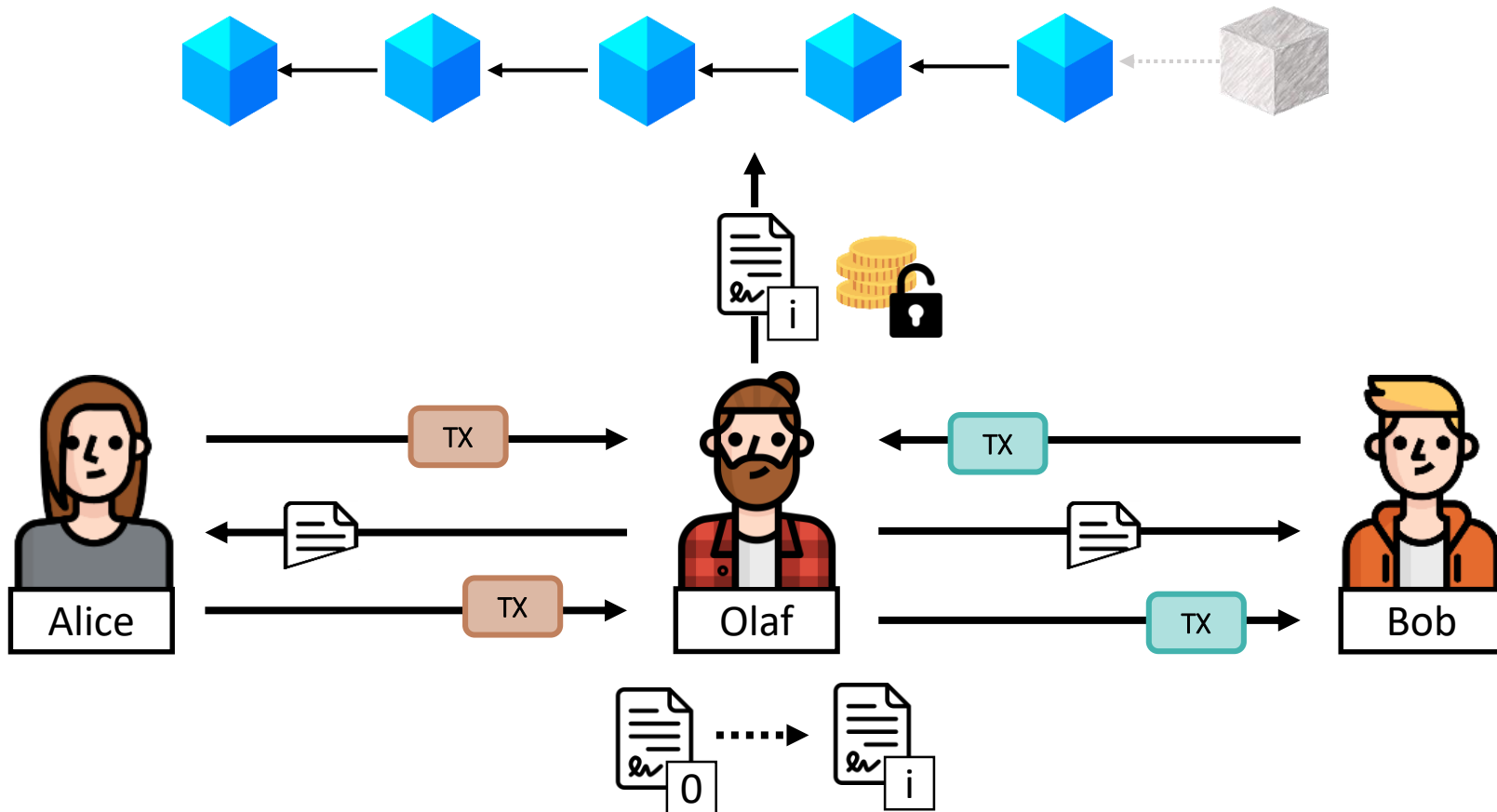
Off-chain Protocols



Off-chain Protocols



... with operators



A lot of different approaches:

- **Academical**

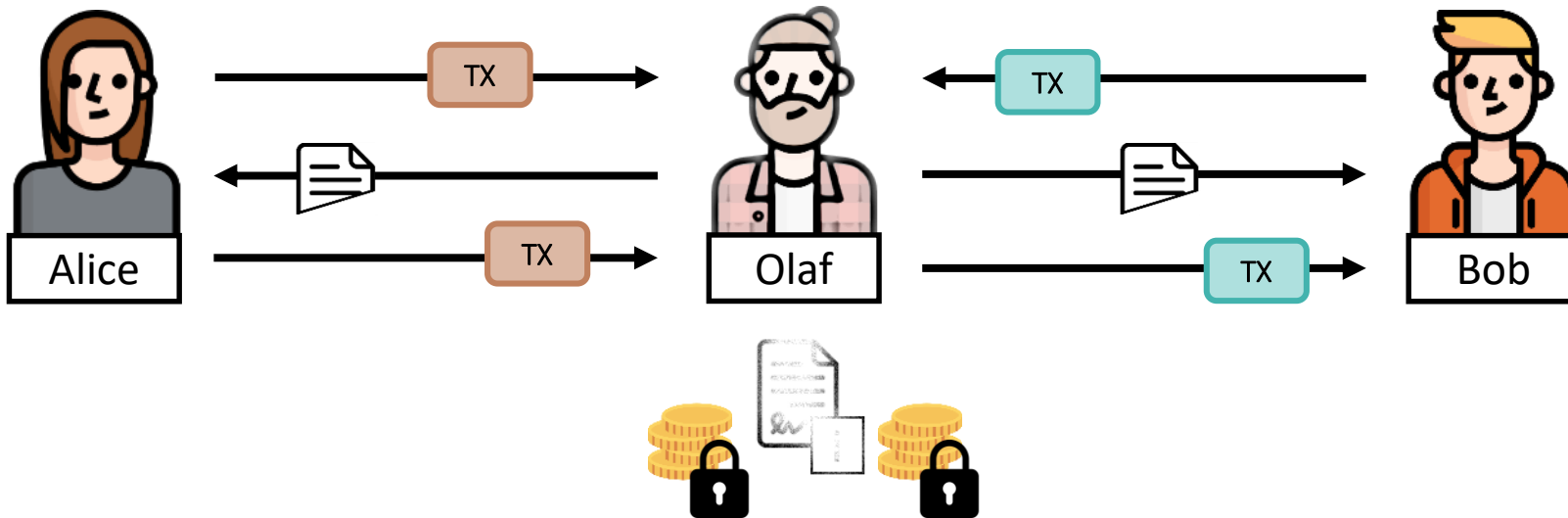
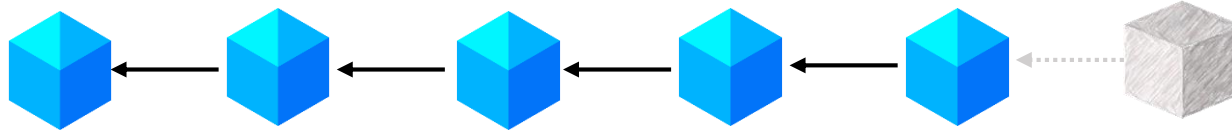
- Sprites (FC'19), Perun (CCS'18), Arbitrum (USENIX'18), Fast Kitten (USENIX'19), Ekiden (EuroS&P'19), CommiTEE (ePrint'20) ...

- **Industrial**

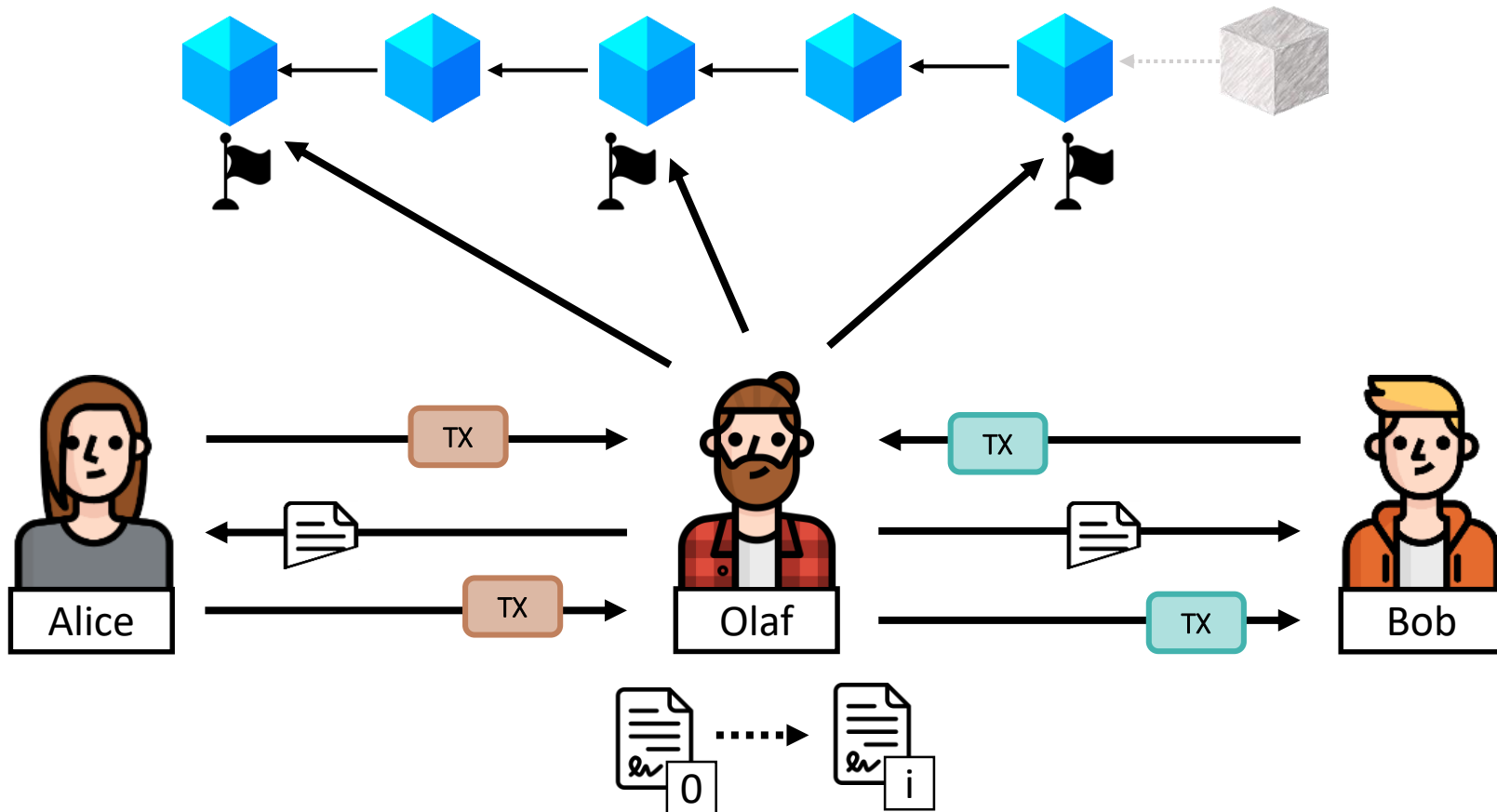


and many more ...

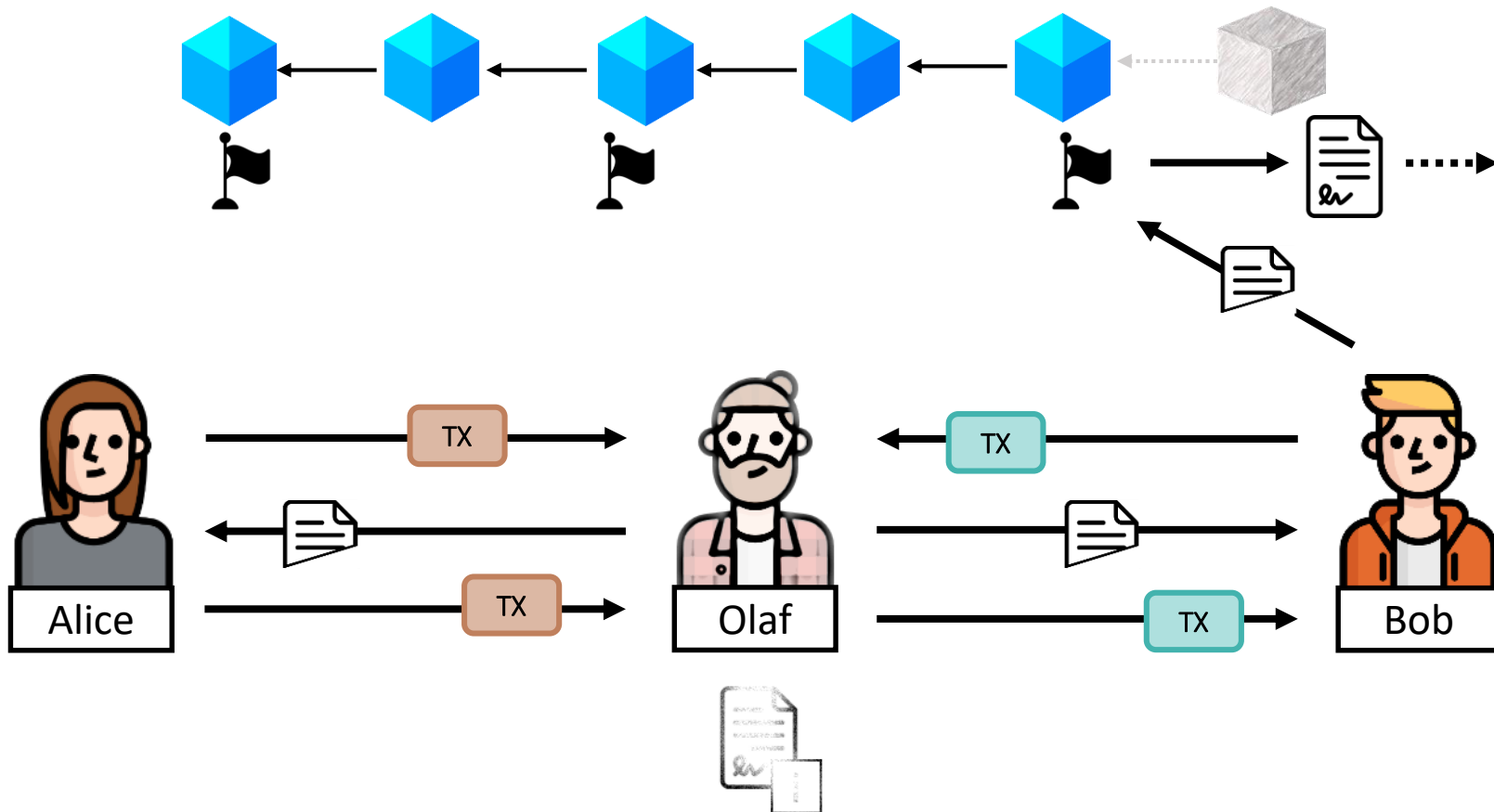
What if a party disappears?



Approach 1: On-chain checkpoints

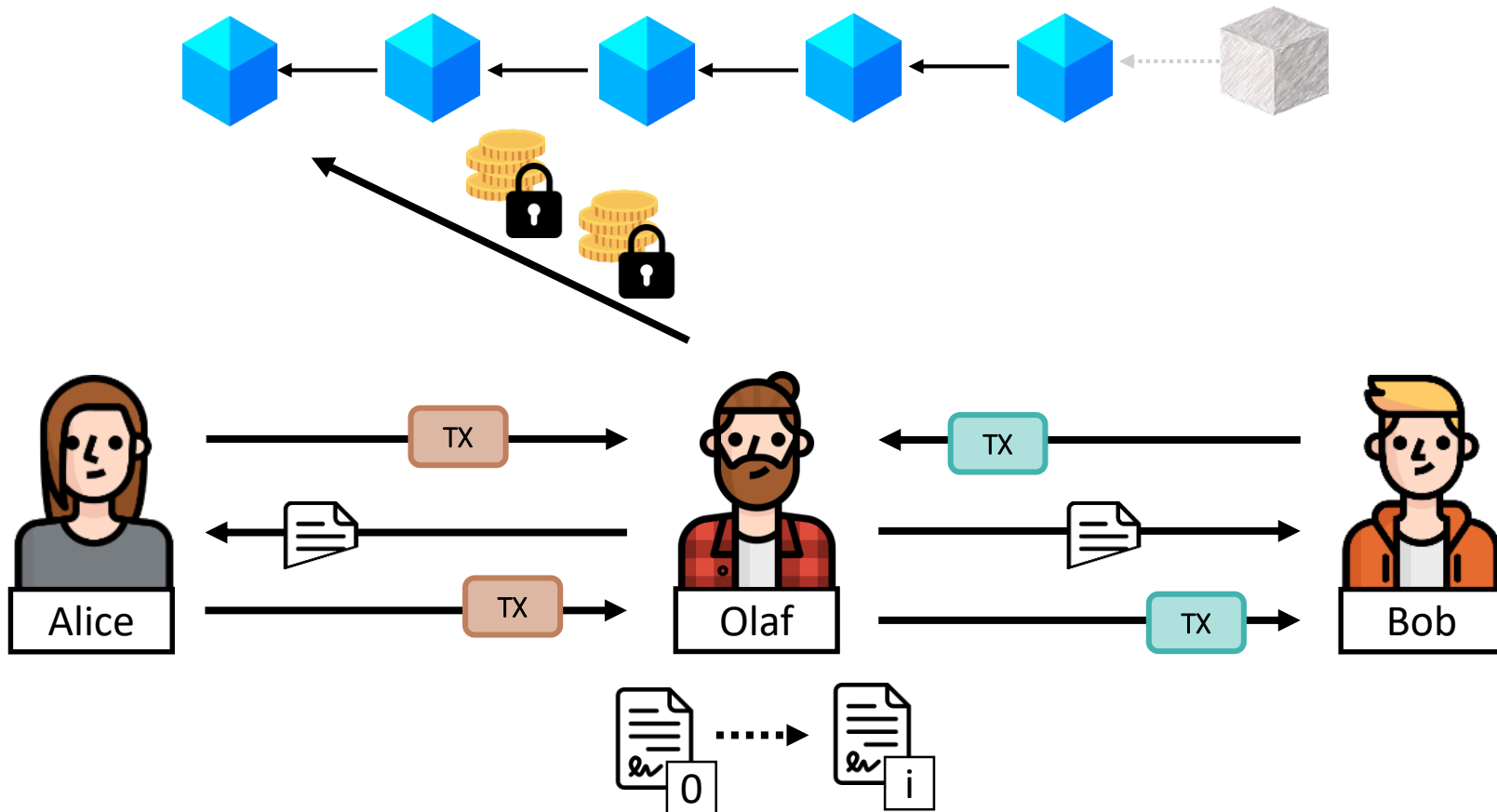


Approach 1: On-chain checkpoints

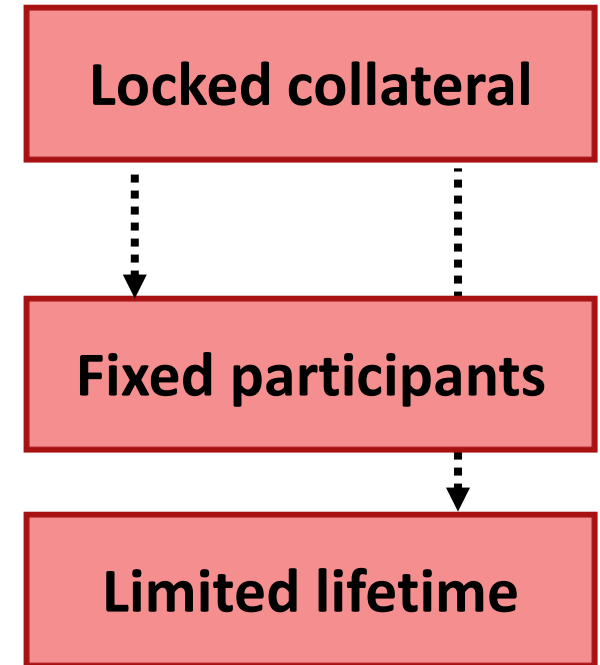
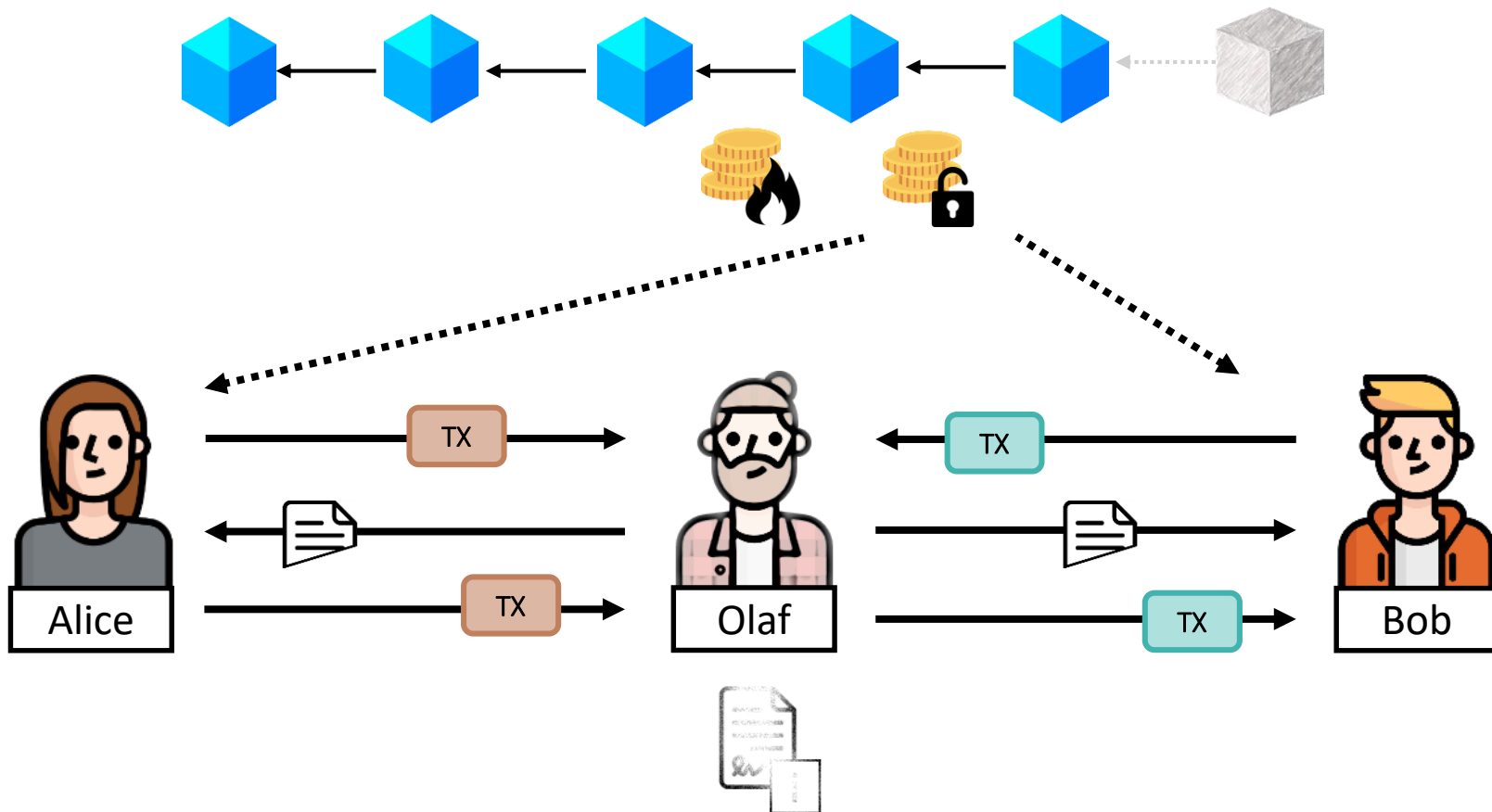


Periodic on-chain tx

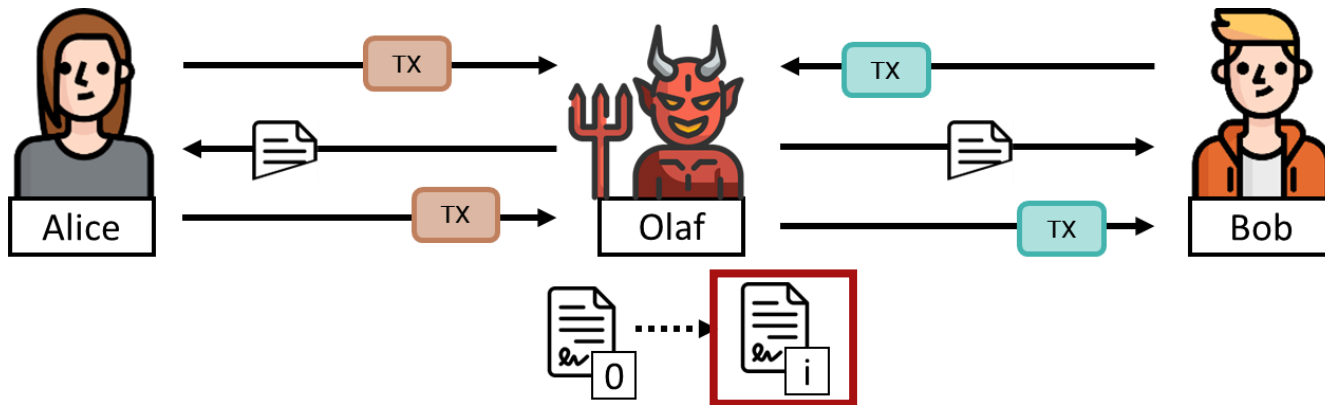
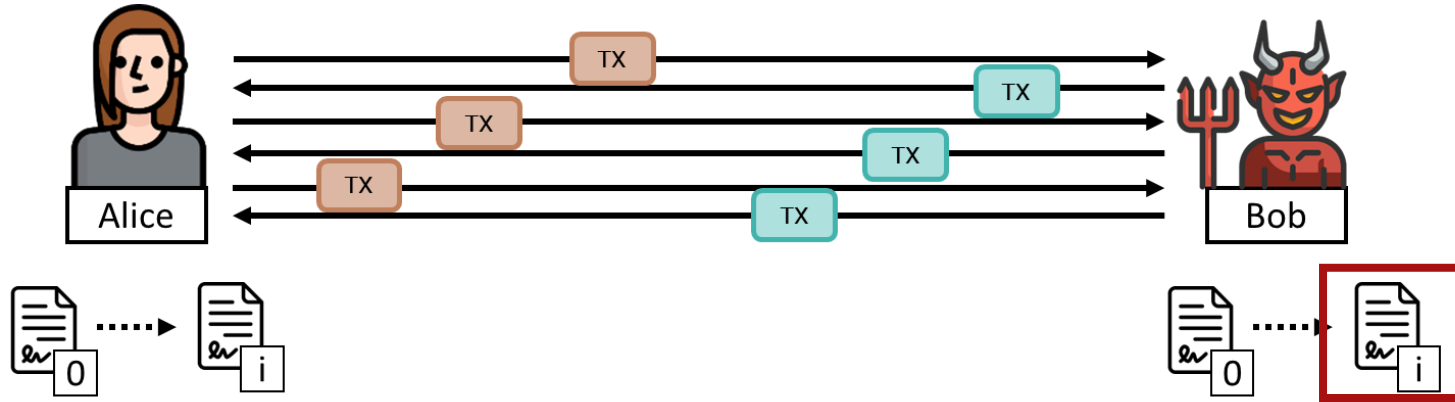
Approach 2: Collateral



Approach 2: Collateral



Contract state is unprotected!



No private state

Limited applications

To sum it up, ...

All known solutions suffer from at least one of the following:

Locked collateral

Periodic on-chain tx

Fixed participants

Limited lifetime

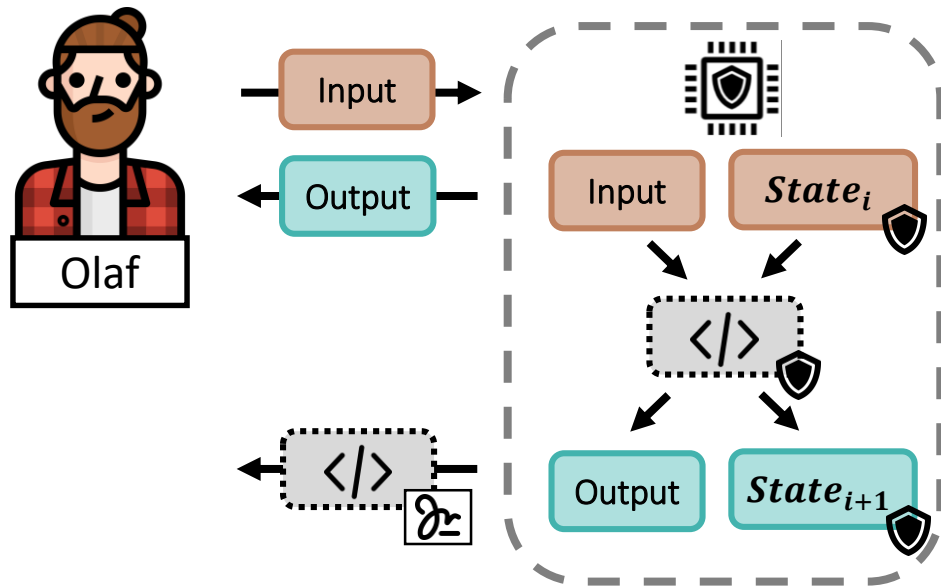
No private state

Contribution

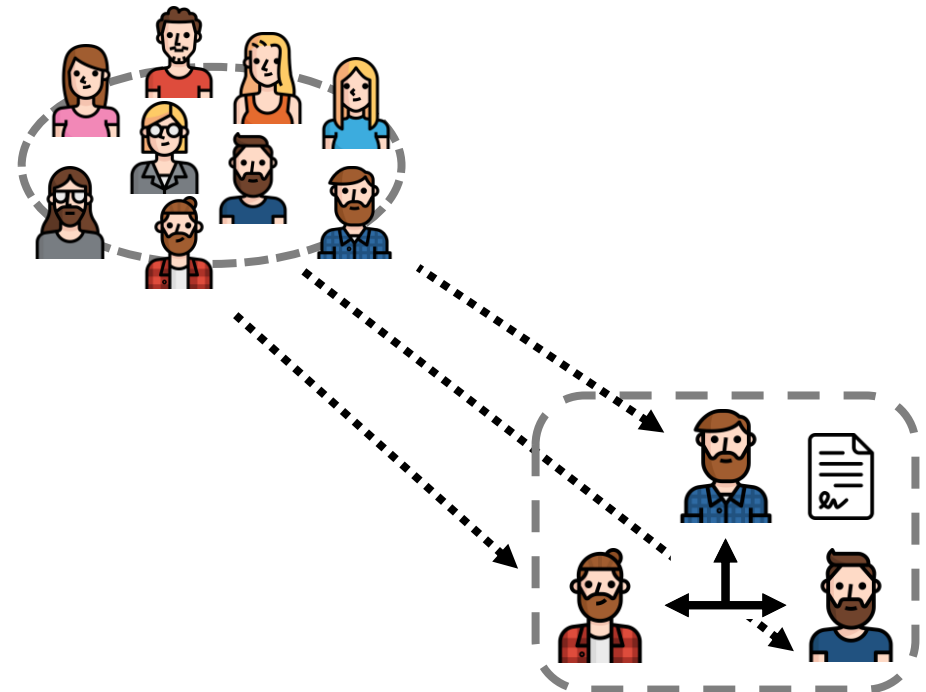
- The POSE protocol:
 - Optimistically completely off-chain
 - No collateral required
 - Arbitrary contract lifetime
 - Open participation
 - Private state
- } **Still providing high liveness guarantees**
- Security analysis
 - Prototype implementation and evaluation

Our Tools

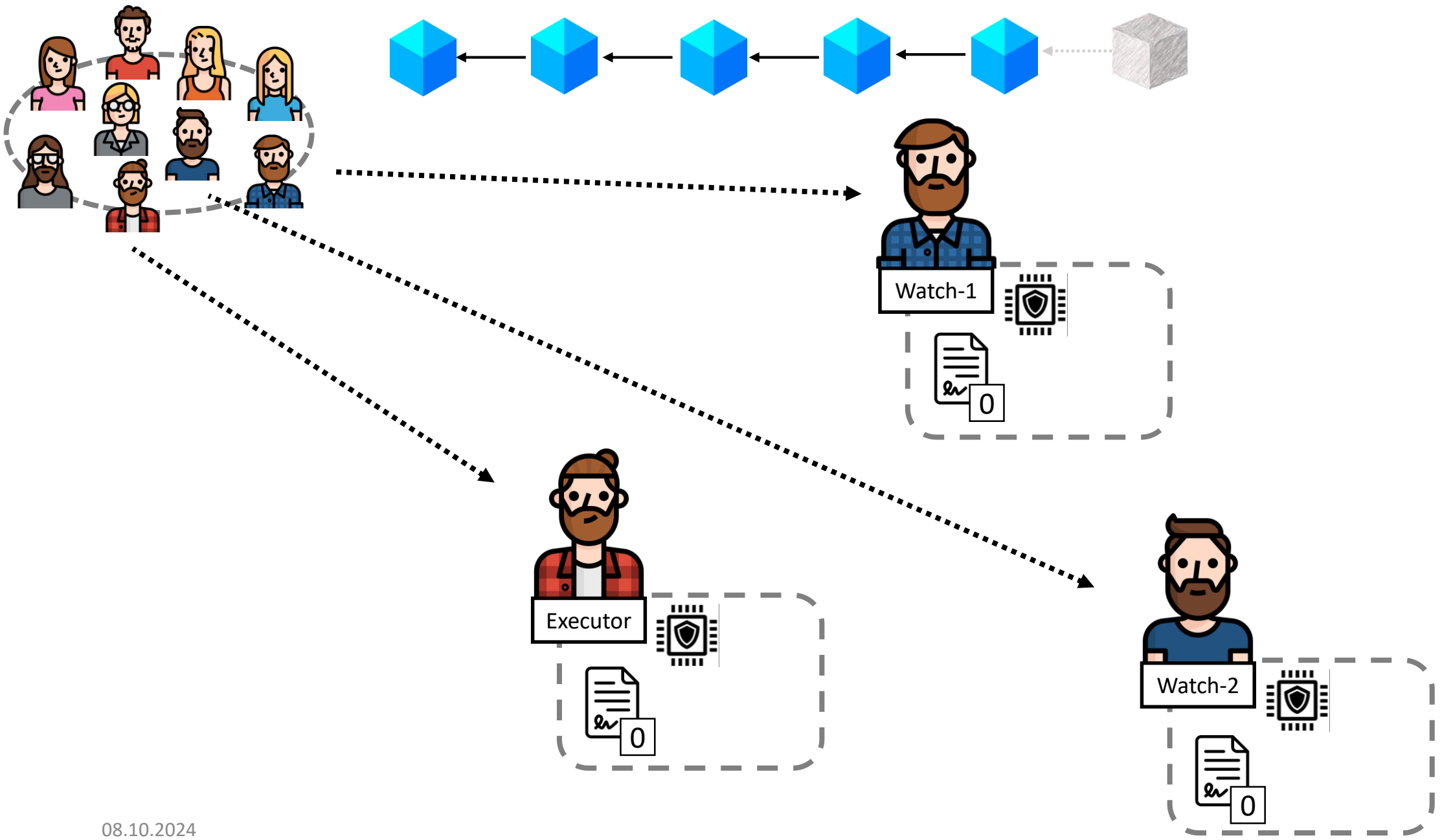
Trusted Execution Environment

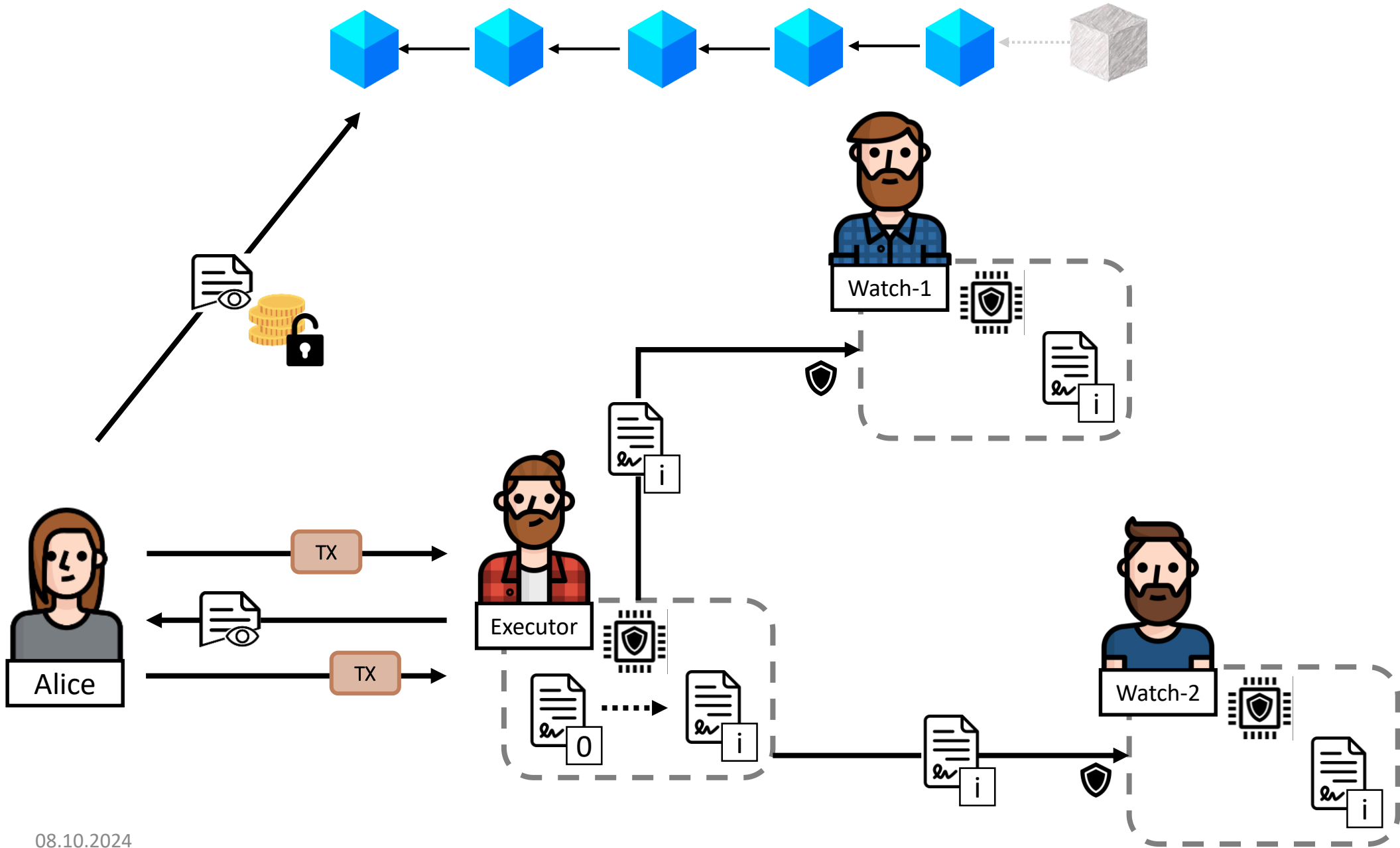


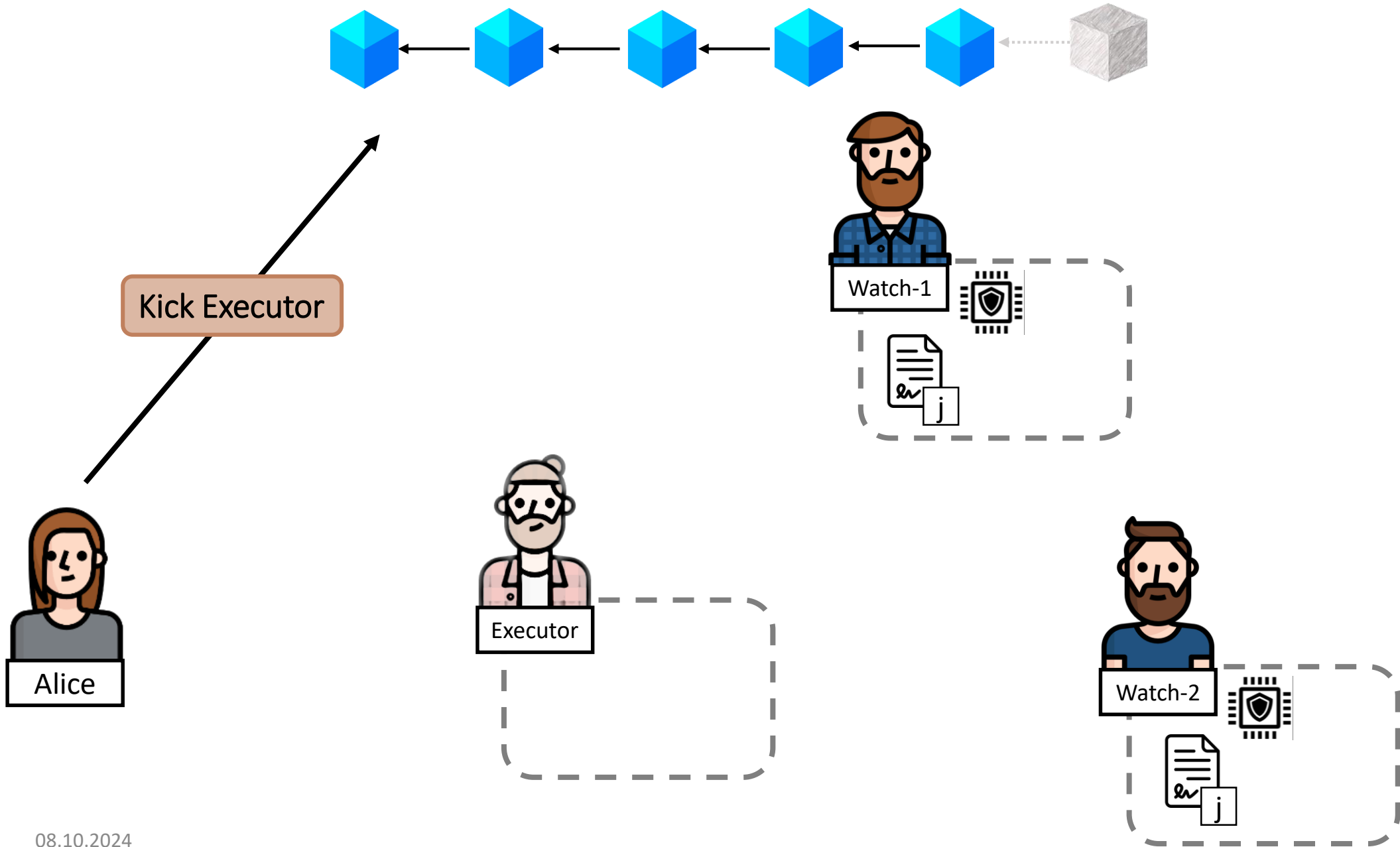
Pooled Execution

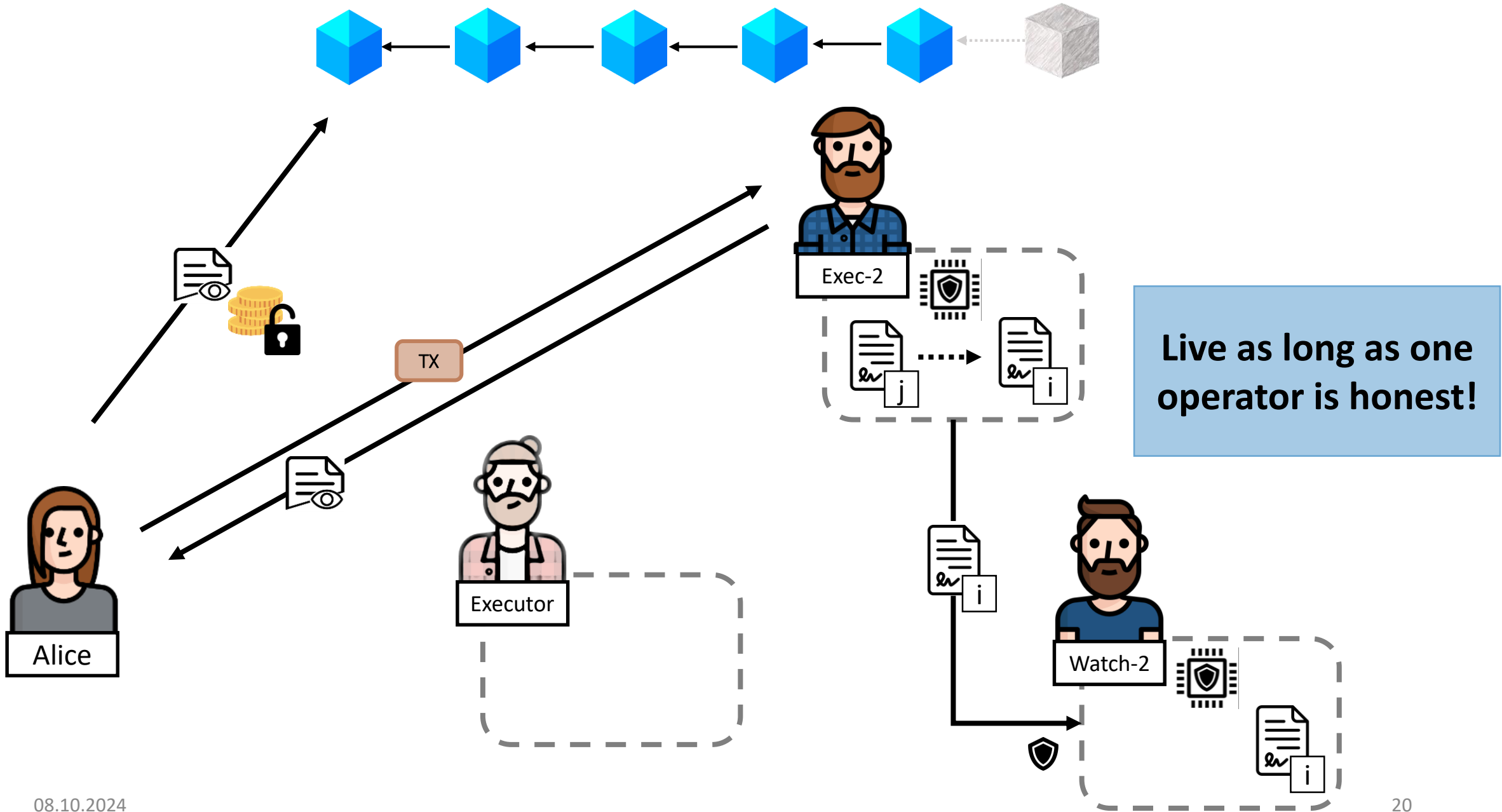


The POSE Protocol in a Nutshell







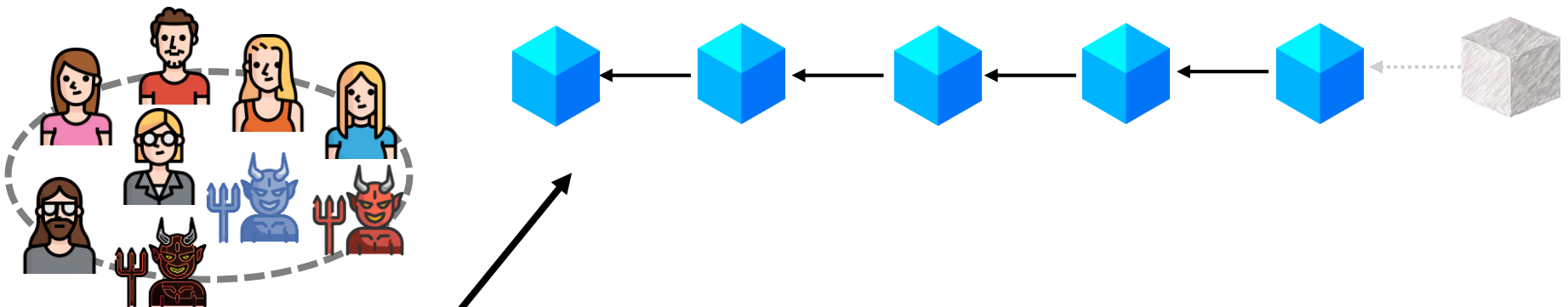





Easy at first, but ...

- How to protect from malicious operators?
- How to setup the pool?
- How to resolve inconsistent state updates?
- How to synchronize with the blockchain (efficiently)?
- How to protect the coin flow?
- ...

Easy at first, but ...

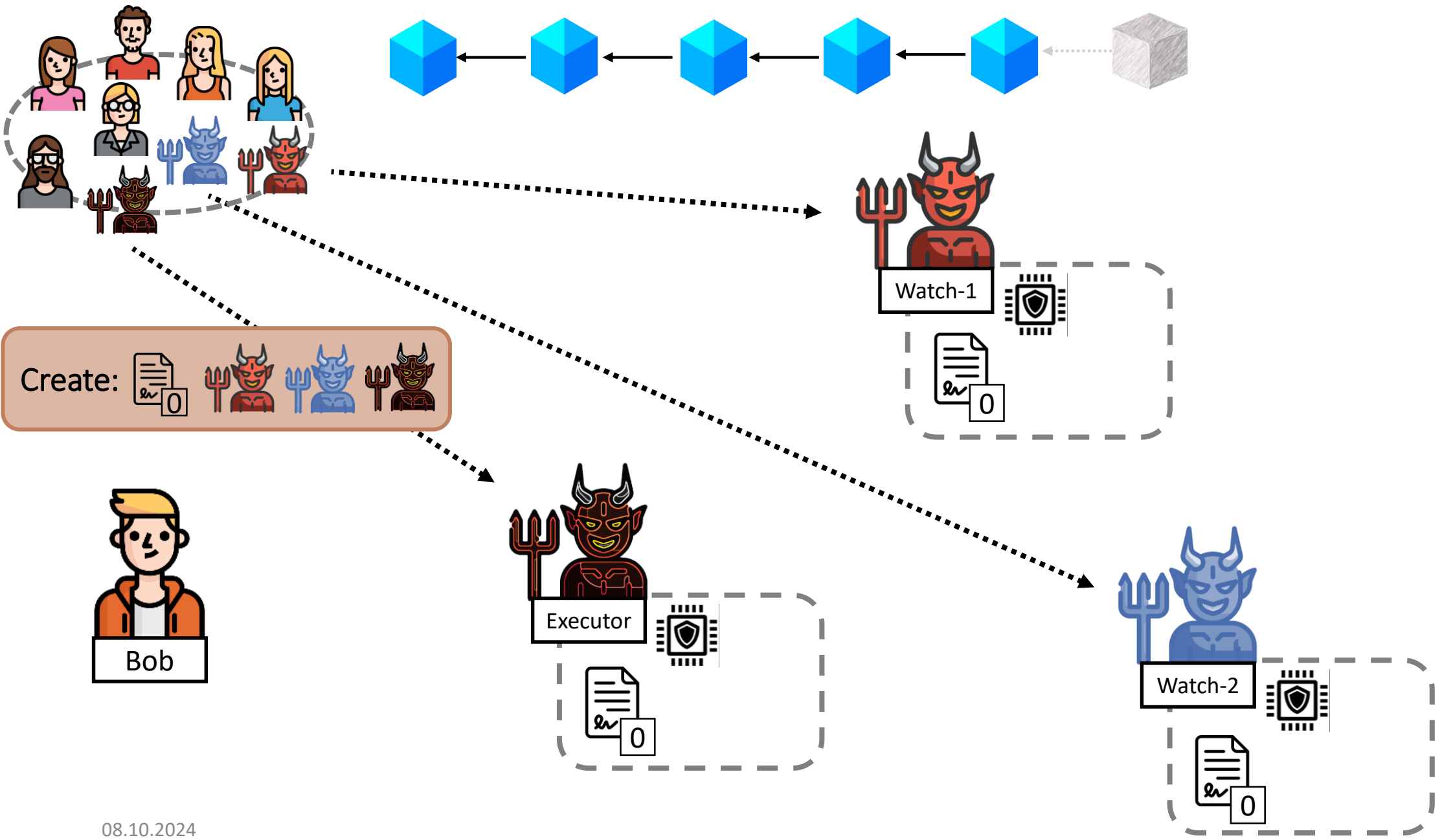
- How to protect from malicious operators?
- **How to setup the pool?**
- How to resolve inconsistent state updates?
- How to synchronize with the blockchain (efficiently)?
- How to protect the coin flow?
- ...

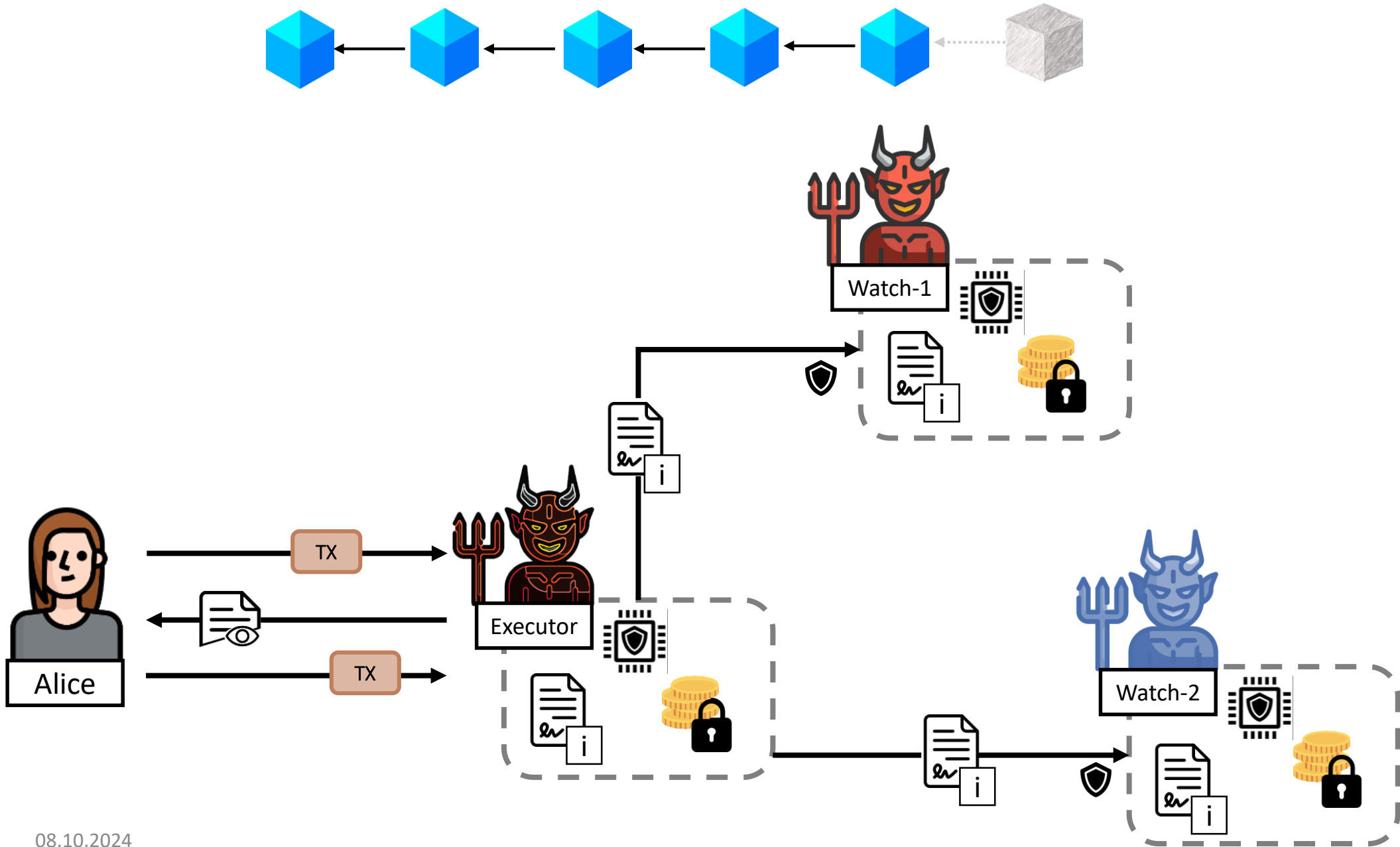


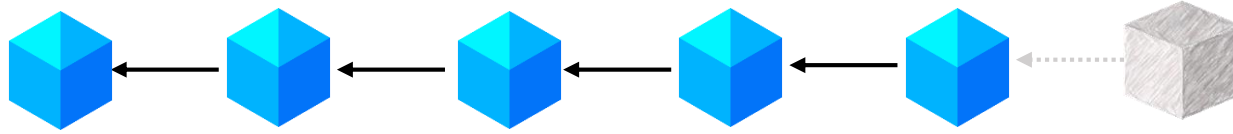
Create:    



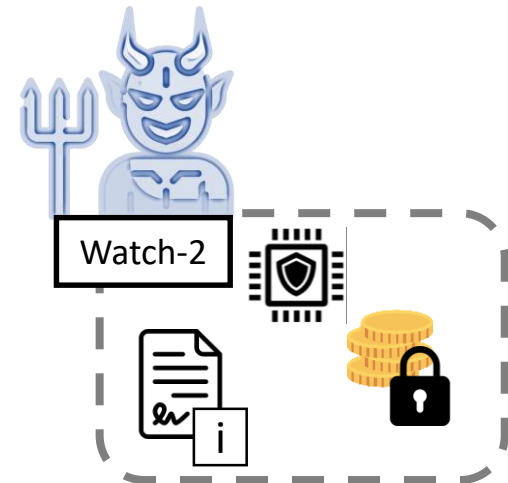
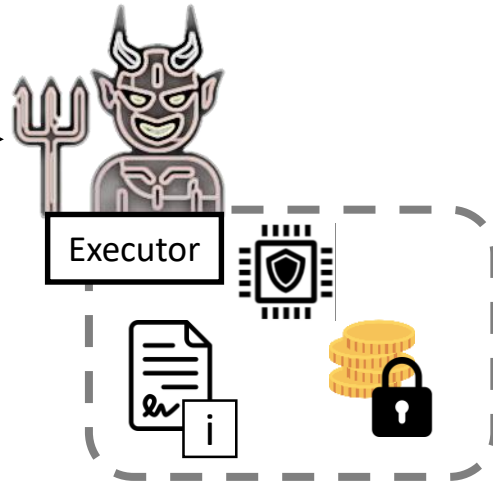
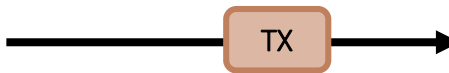
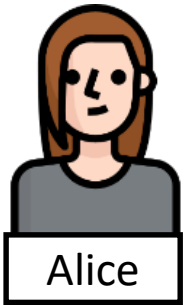
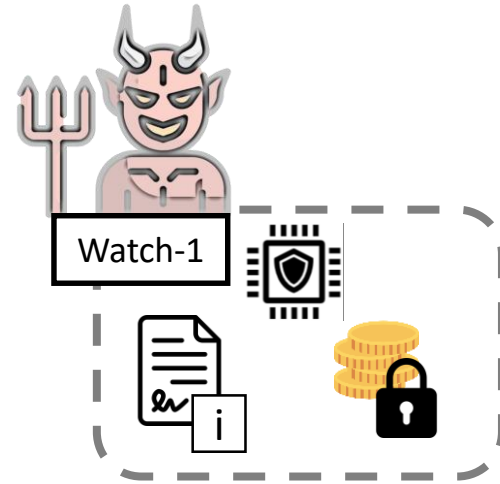
Bob

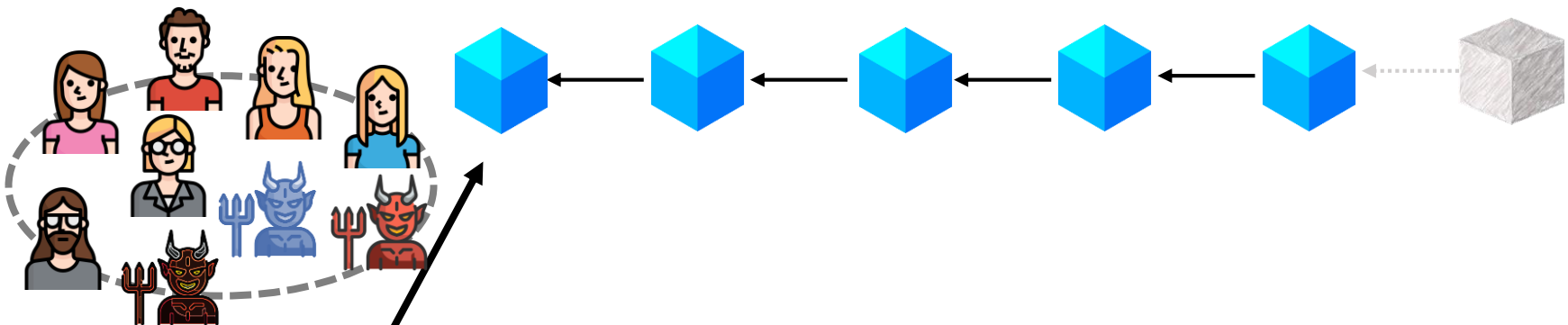






State & funds of contract are lost!

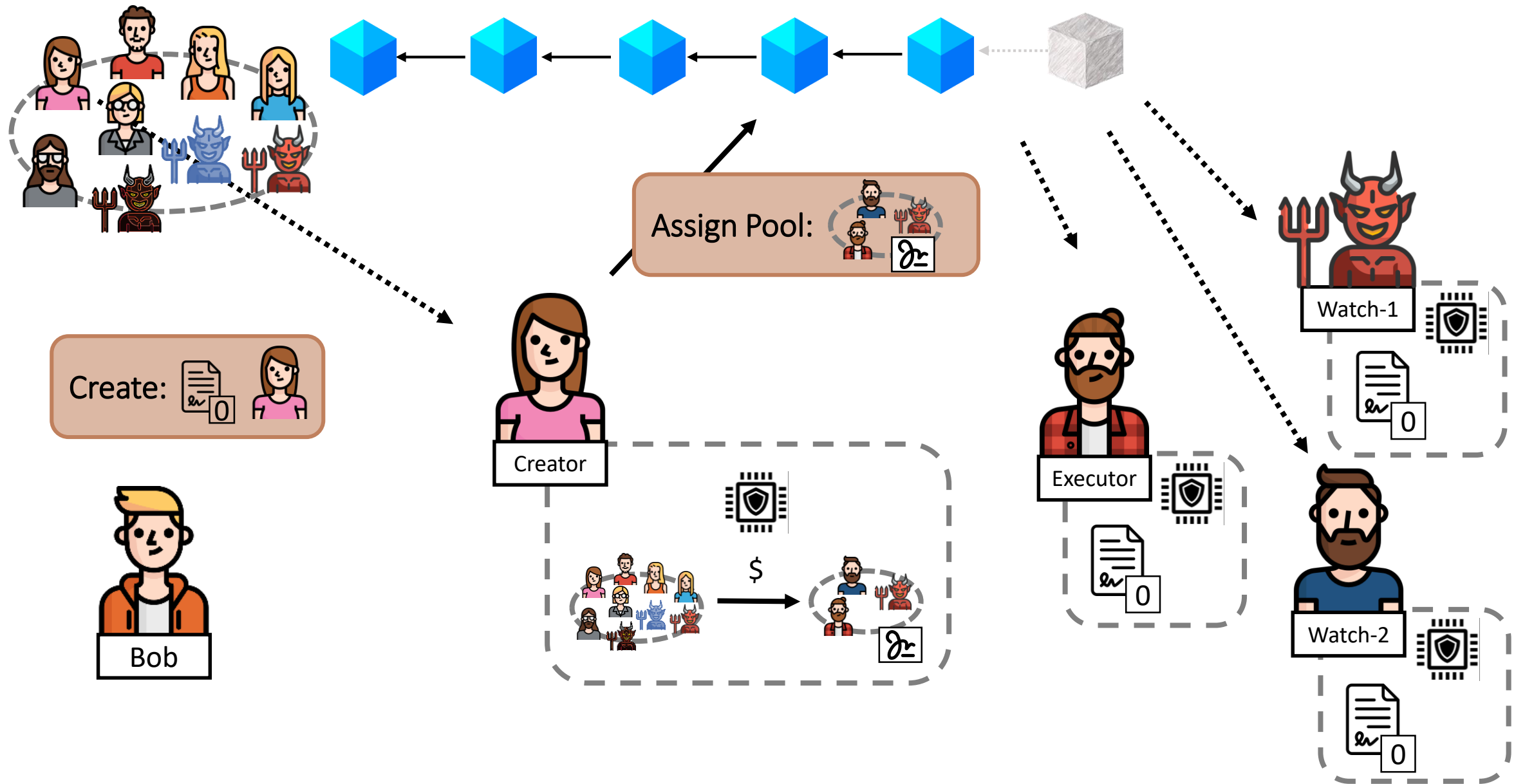


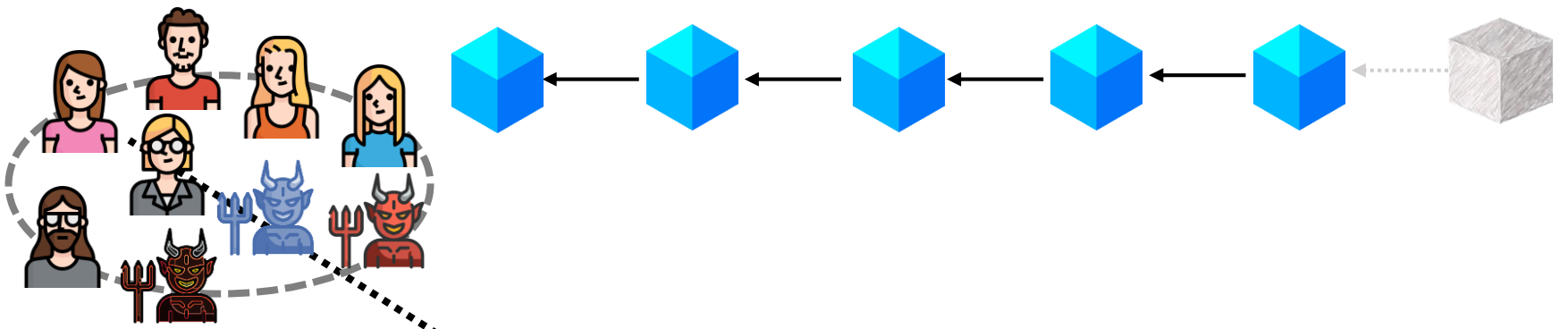


Create:  



Bob

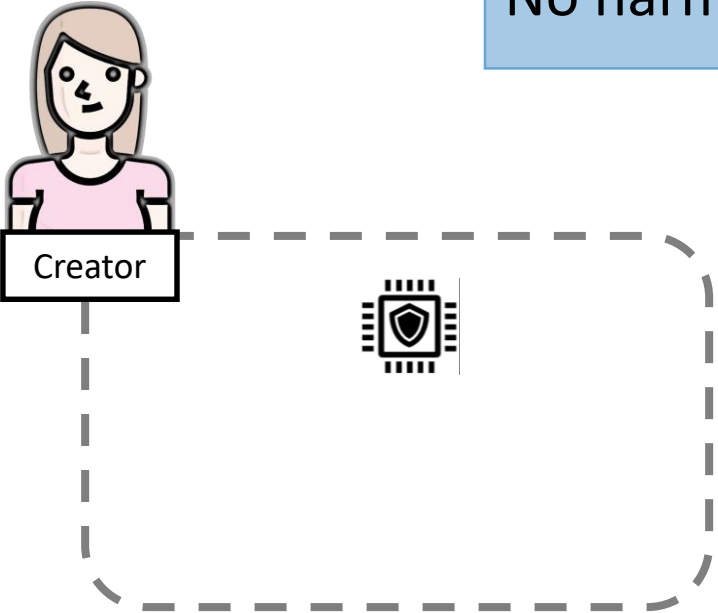




No harm → Try again

Create:  


Bob



Liveness guarantees

$$\Pr[\textit{Contract crash}] = \sum_{i=0}^{s-1} \binom{m-i}{n-i} < \left(\frac{m}{n}\right)^s$$

n : # operators
 m : #malicious operators
 s : pool size

Liveness guarantees

$$\Pr[\textit{Contract crash}] = \sum_{i=0}^{s-1} \binom{m-i}{n-i} < \left(\frac{m}{n}\right)^s$$

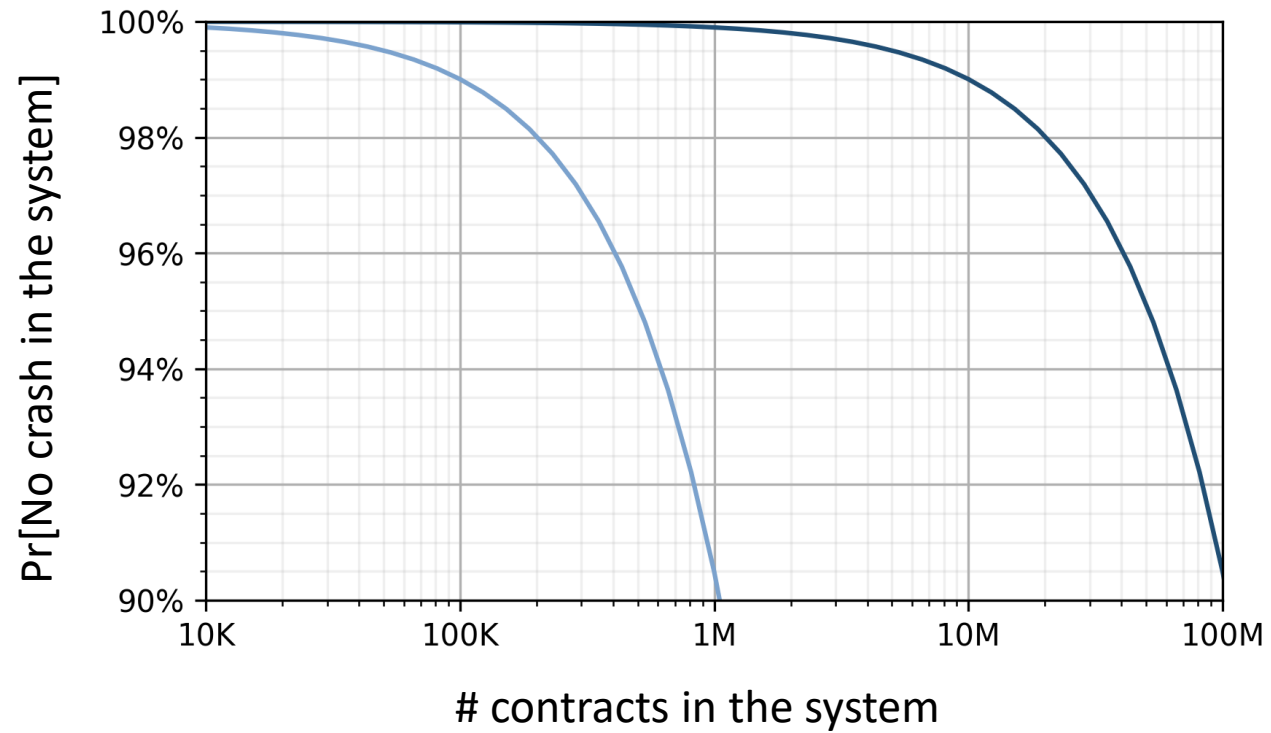
n : # operators

m : #malicious operators

s : pool size

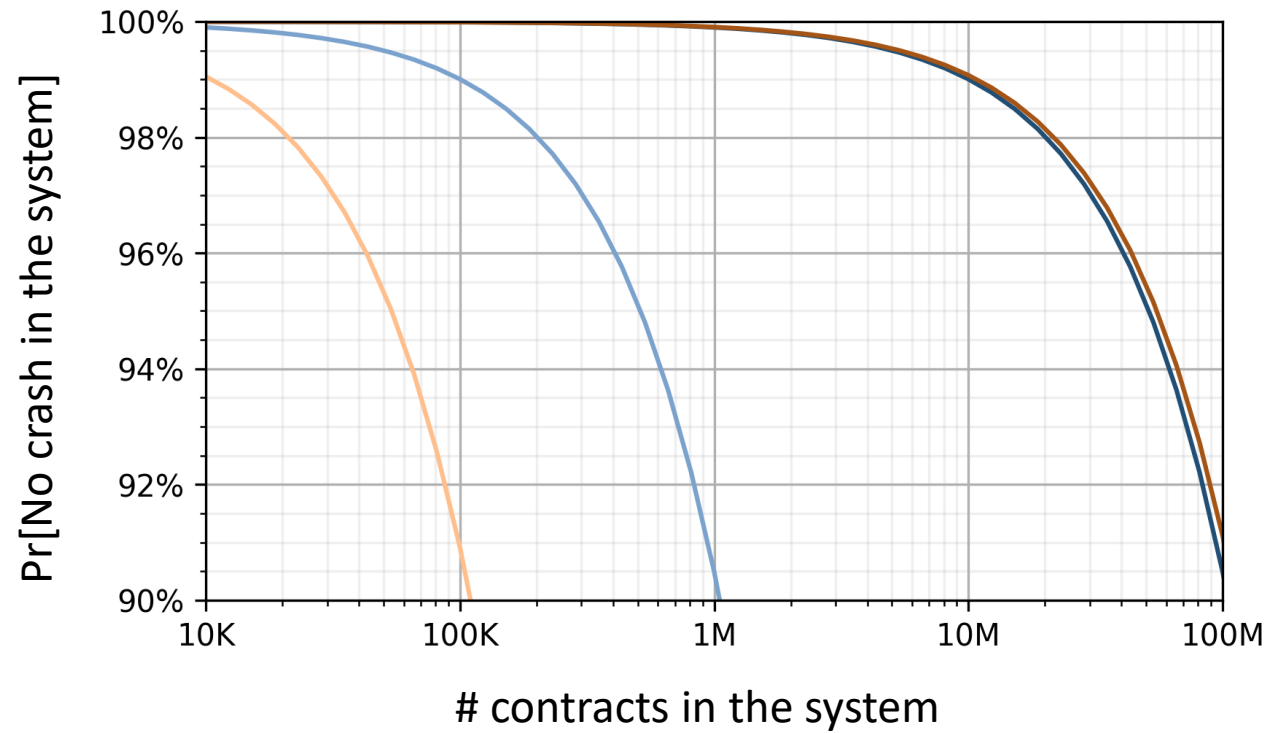
$\left(\frac{m}{n}\right)$: fraction of malicious operators

Liveness guarantees



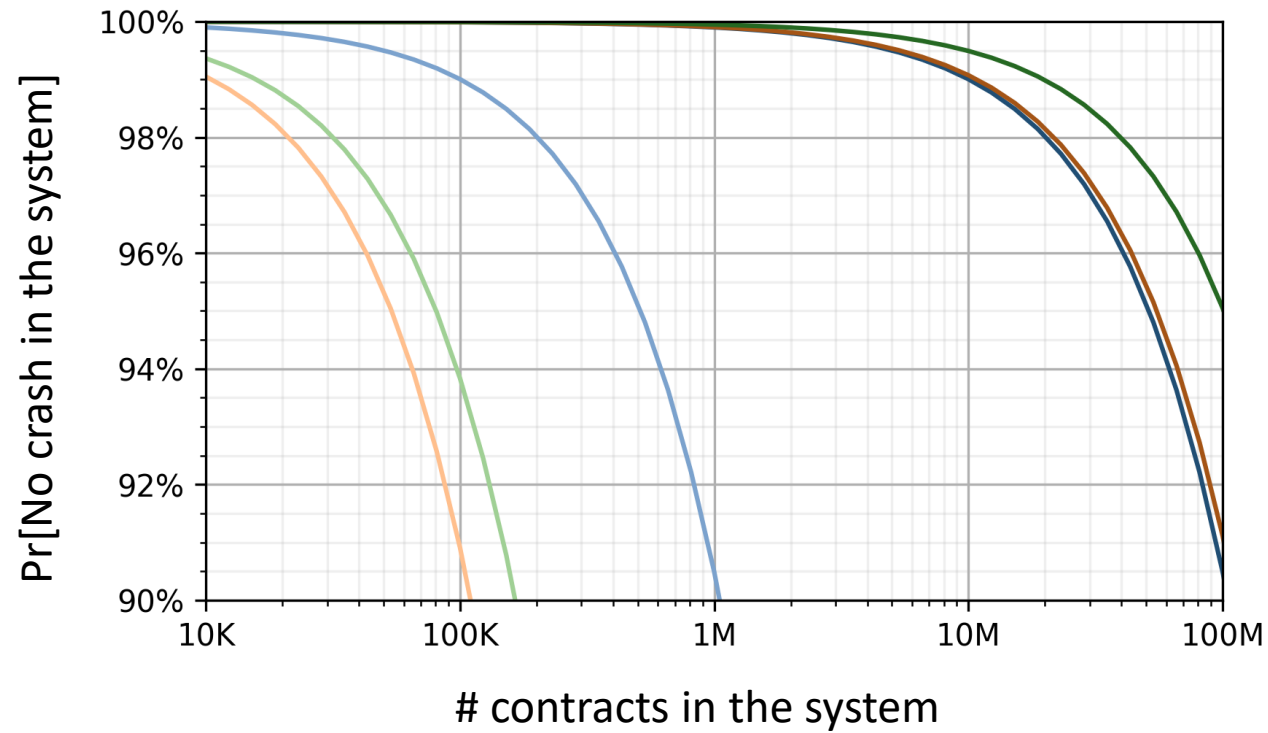
- 10% malicious operators
 - $s = 7$
 - $s = 8$

Liveness guarantees



- 10% malicious operators
 - $s = 7$
 - $s = 8$
- 50% malicious operators
 - $s = 20$
 - $s = 30$

Liveness guarantees



- 10% malicious operators
 - $s = 7$
 - $s = 8$
- 50% malicious operators
 - $s = 20$
 - $s = 30$
- 70% malicious operators
 - $s = 40$
 - $s = 60$

Some evaluation results

- Multiple Case studies

- Rock Paper Scissors
- Poker
- Federated Machine Learning
- Quicksort

} **Too complex for on-chain execution.**

- Blockchain Fees

- Contract Creation \approx CryptoKitten Creation

Method	Cost	
	Gas	USD
registerEnclave	175 910	13.23
initCreation	198 436	14.91
finalizeCreation	79 545	5.98
deposit	37 255	2.80
withdraw	36 997	2.78
challengeExecutor	54 654	4.11
executorResponse	51 478	3.87
executorTimeout	53 327	4.01
challengeWatchdogsCreation	231 286	17.38
challengeWatchdog	131 362	9.87
watchdogResponse	36 257	2.72
watchdogTimeout	52 142	3.92
simple Ether transfer*	21 000	1.58
create CryptoKitty*	250 000	18.78

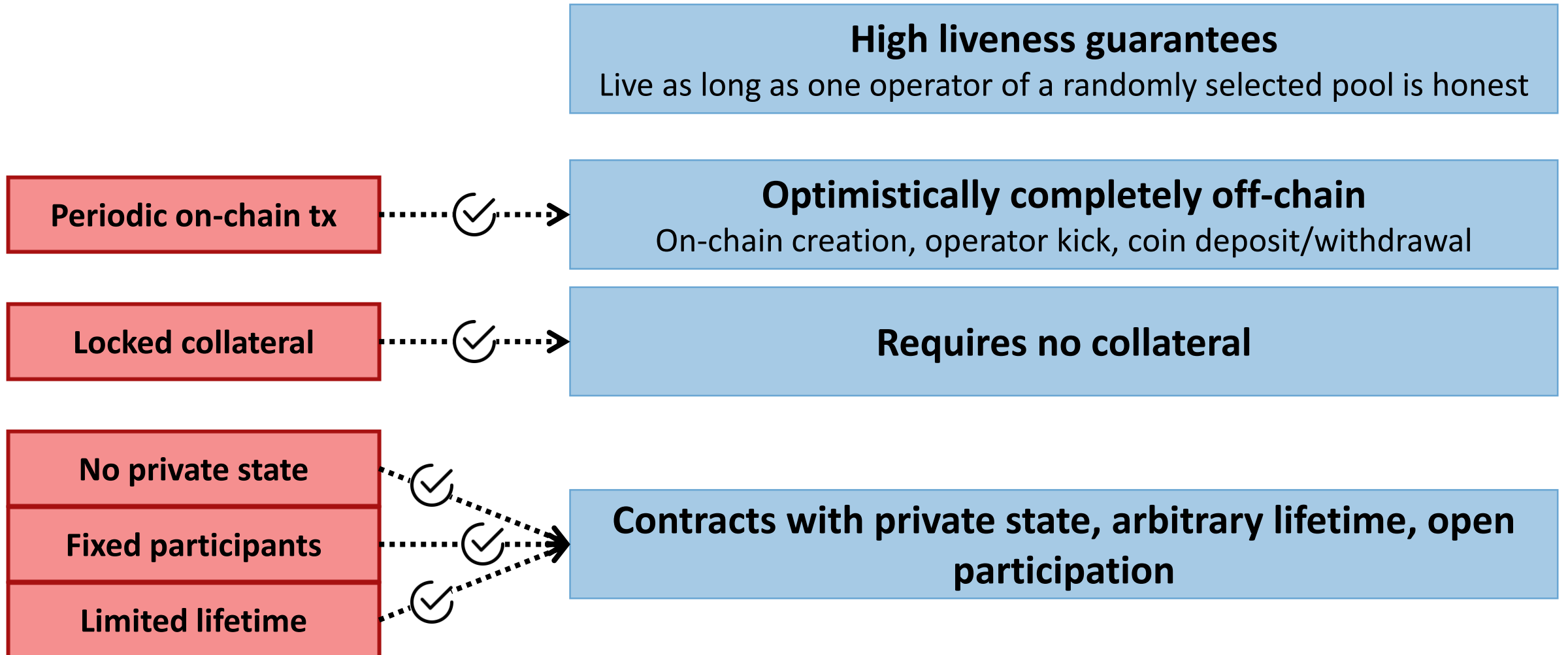
Some results

- Multiple Case studies
 - Rock Paper Scissors
 - Poker
 - Federated Machine Learning
 - Quicksort
- Blockchain Fees
 - Contract Creation \approx CryptoKitten Creation
 - Withdrawal/Deposit $\approx 2 * \text{Ether transfer}$
- Runtime
 - Milliseconds compared to minutes in Ethereum

Method	Cost	
	Gas	USD
registerEnclave	175 910	13.23
initCreation	198 436	14.91
finalizeCreation	79 545	5.98
deposit	37 255	2.80
withdraw	36 997	2.78
challengeExecutor	54 654	4.11
executorResponse	51 478	3.87
executorTimeout	53 327	4.01
challengeWatchdogsCreation	231 286	17.38
challengeWatchdog	131 362	9.87
watchdogResponse	36 257	2.72
watchdogTimeout	52 142	3.92
simple Ether transfer*	21 000	1.58
create CryptoKitty*	250 000	18.78

Action	Execution time [ms]
setupEnclave	189
createAttestationReport	367
teardownEnclave	25
playRockPaperScissors	32
playPoker	199
aggregateMLModels	238
performQuicksort	20
updateWatchdogPokerState	17

Conclusion



Any questions?

arxiv.org/abs/2210.07110

David Kretzler: david.kretzler@tu-darmstadt.de

