Founding Secure Computation on Blockchains

Arka Rai Choudhuri
Johns Hopkins University

Vipul Goyal
Carnegie Mellon University

Abhishek Jain
Johns Hopkins University

Eurocrypt 2019, Darmstadt

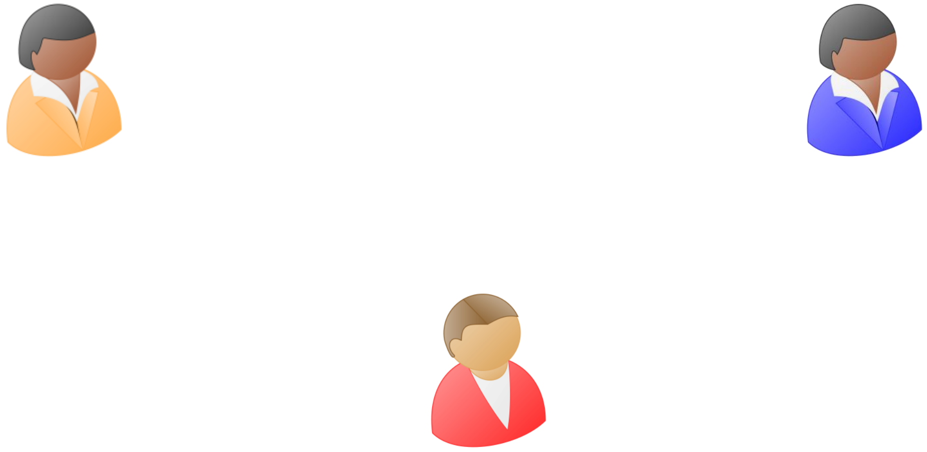
Motivation

Blockchain!

Motivation

Blockchain!

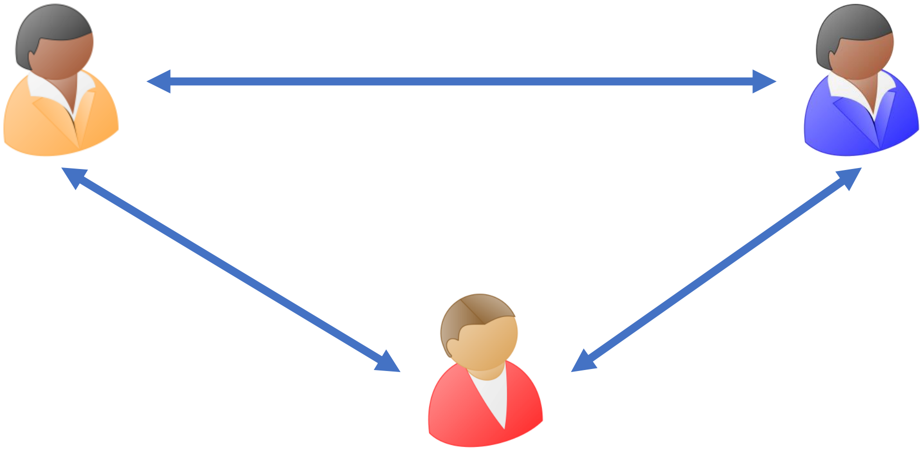
Examine the foundation of secure computation protocols in the context of blockchains.



Motivation

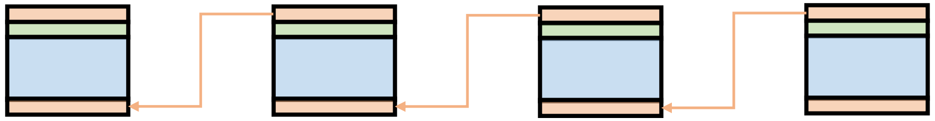
Blockchain!

Examine the foundation of secure computation protocols in the context of blockchains.

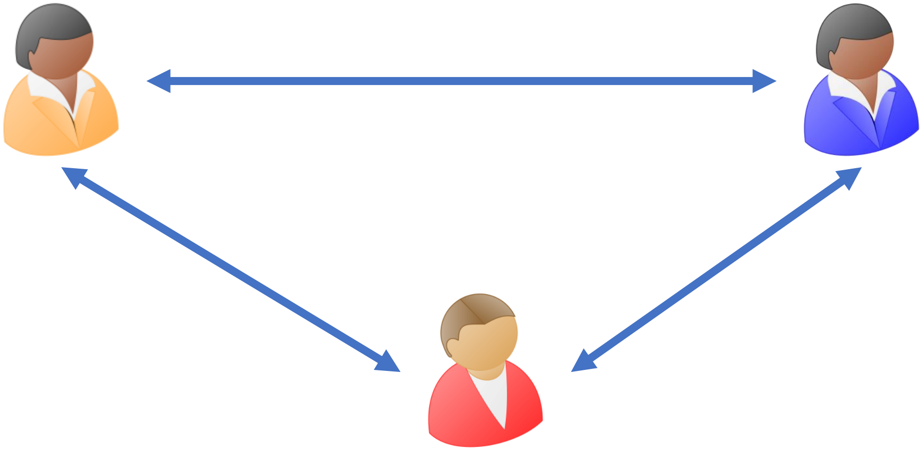


Motivation

Blockchain!



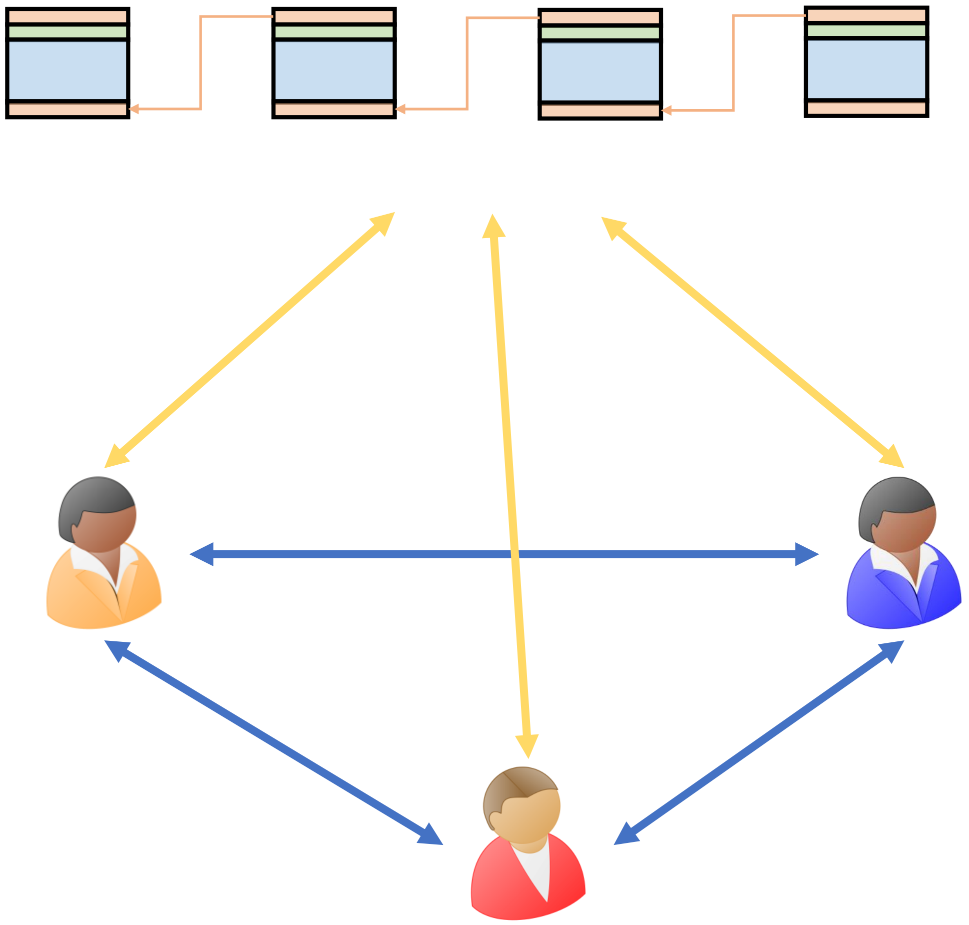
Examine the foundation of secure computation protocols in the context of blockchains.



Motivation

Blockchain!

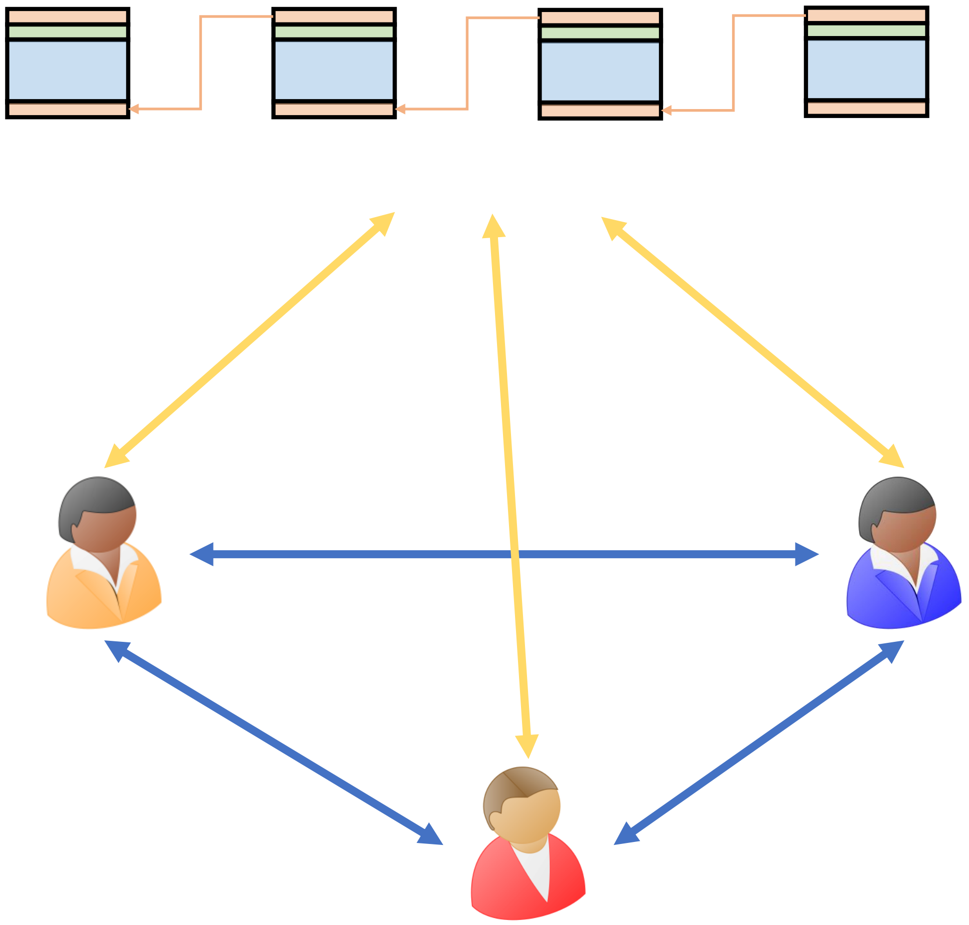
Examine the foundation of secure computation protocols in the context of blockchains.



Motivation

Blockchain!

Examine the foundation of secure computation protocols in the context of blockchains.



What change does this make to the study of protocols in this setting?

Model

Modeling of the blockchain:

Model

Modeling of the blockchain:

[Kiayis-Zhou-Zikas 16, Badertscher-Maurer-Tschudi-Zikas 17, Badertscher-Gazi-Kiayis-Russell-Zikas 18]

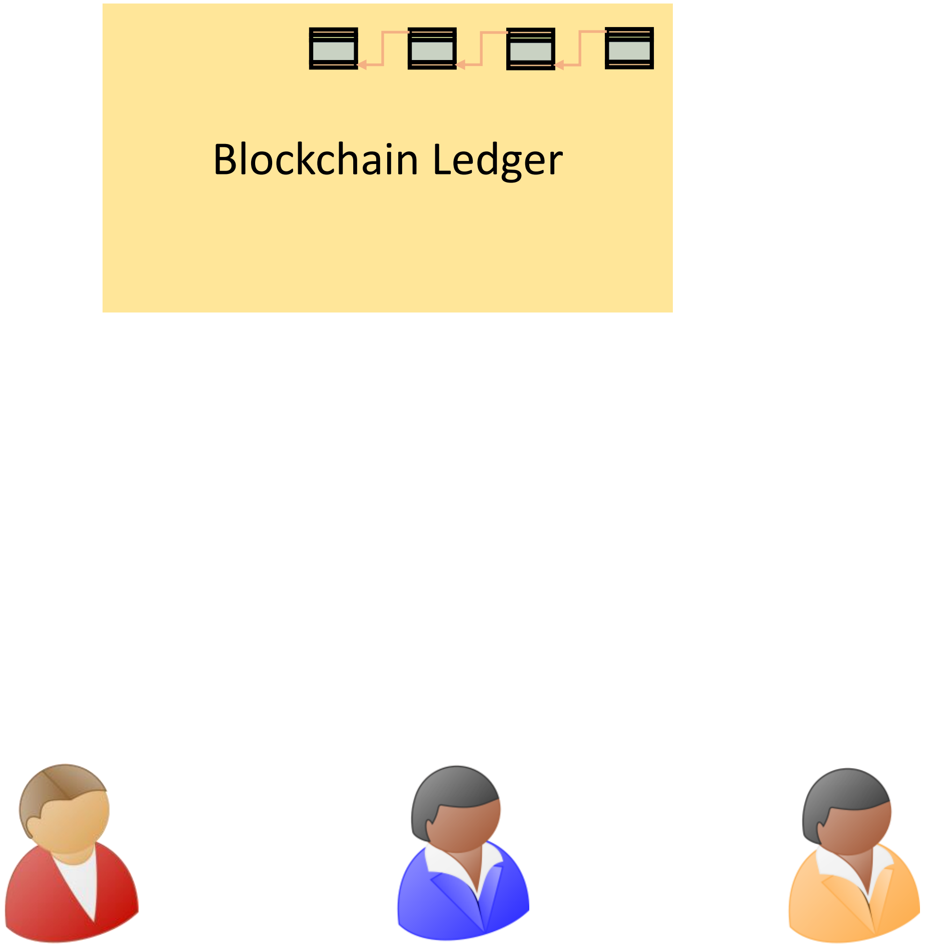
Model

Modeling of the blockchain:

[Kiayis-Zhou-Zikas 16, Badertscher-Maurer-Tschudi-Zikas 17, Badertscher-Gazi-Kiayis-Russell-Zikas 18]

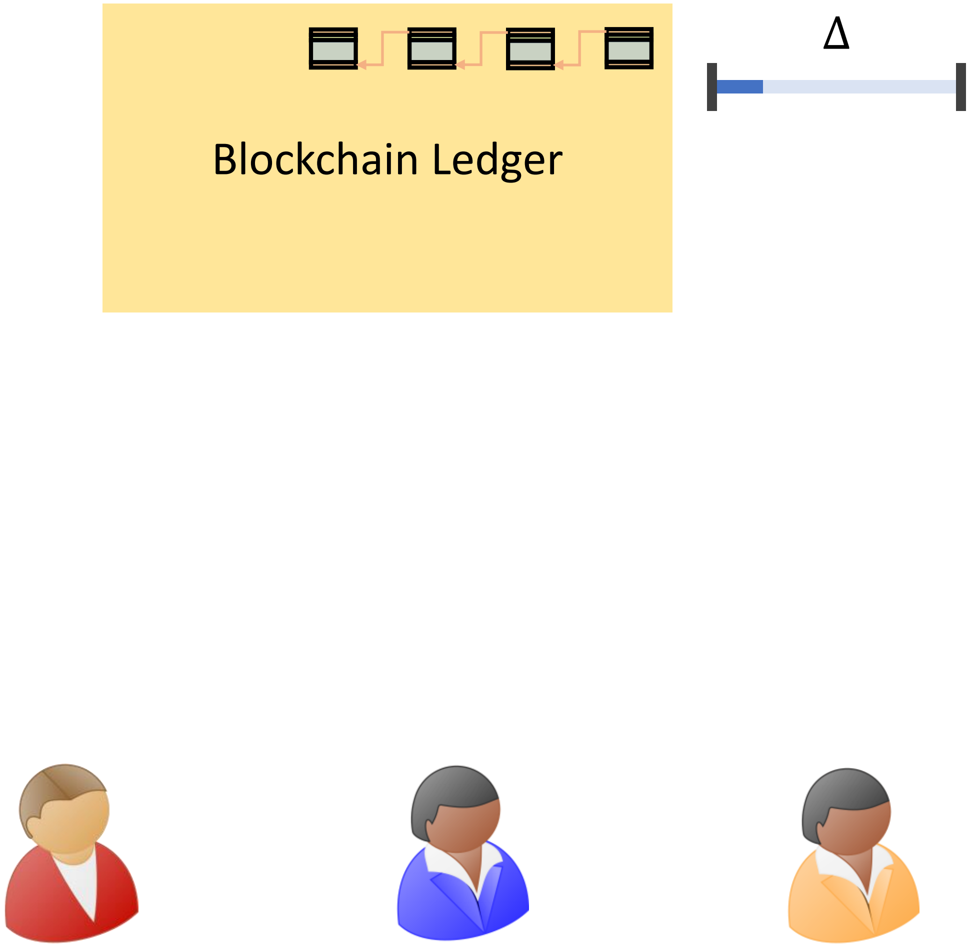
For this talk: simplified model

Simplified Model

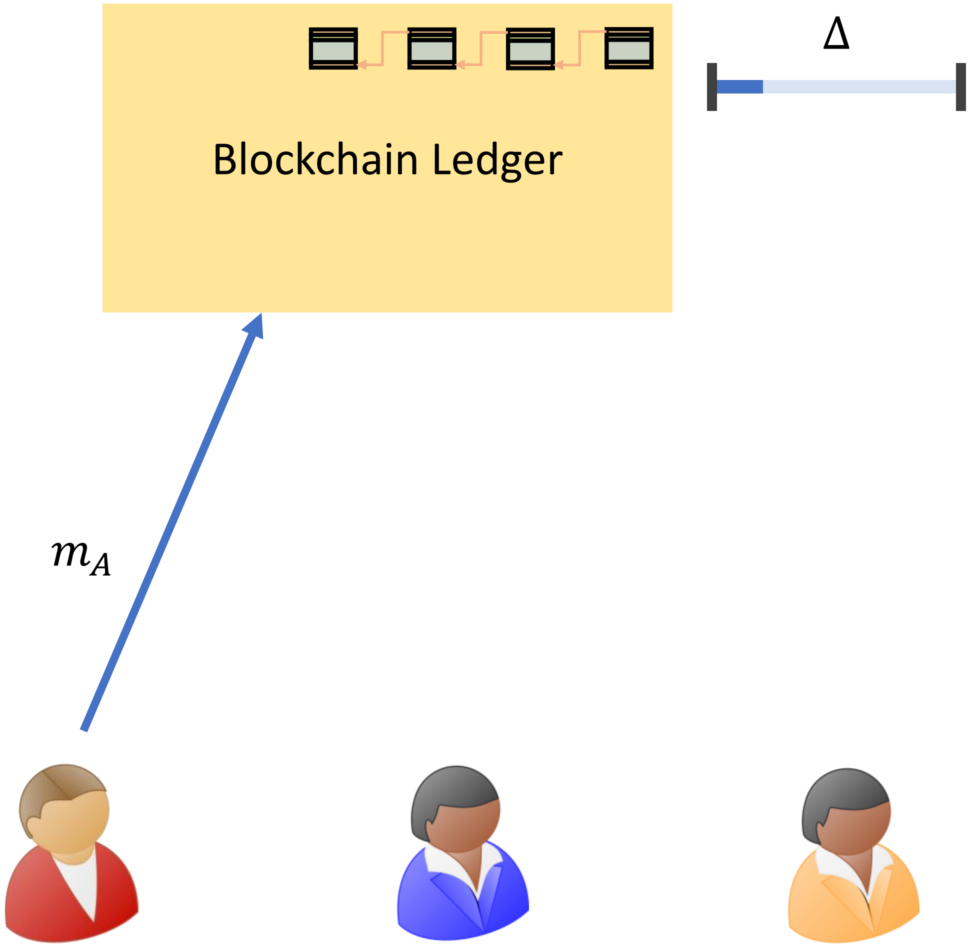


Blockchain Ledger

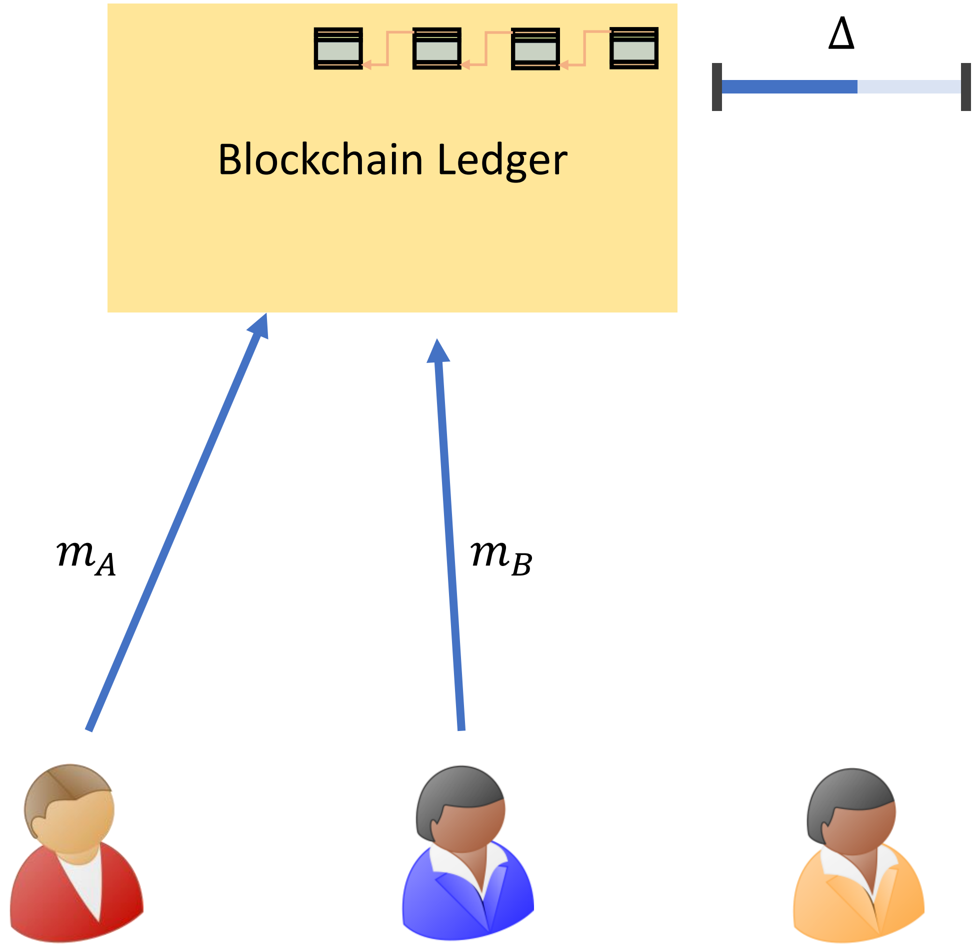
Simplified Model



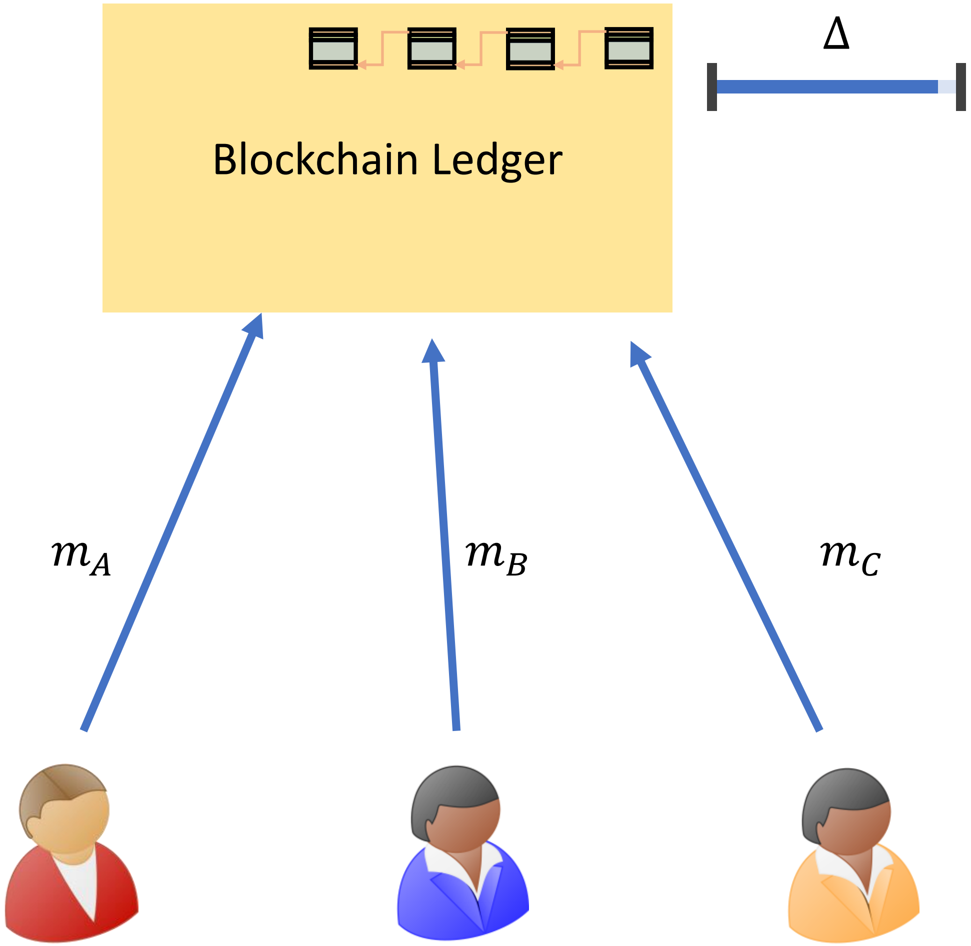
Simplified Model



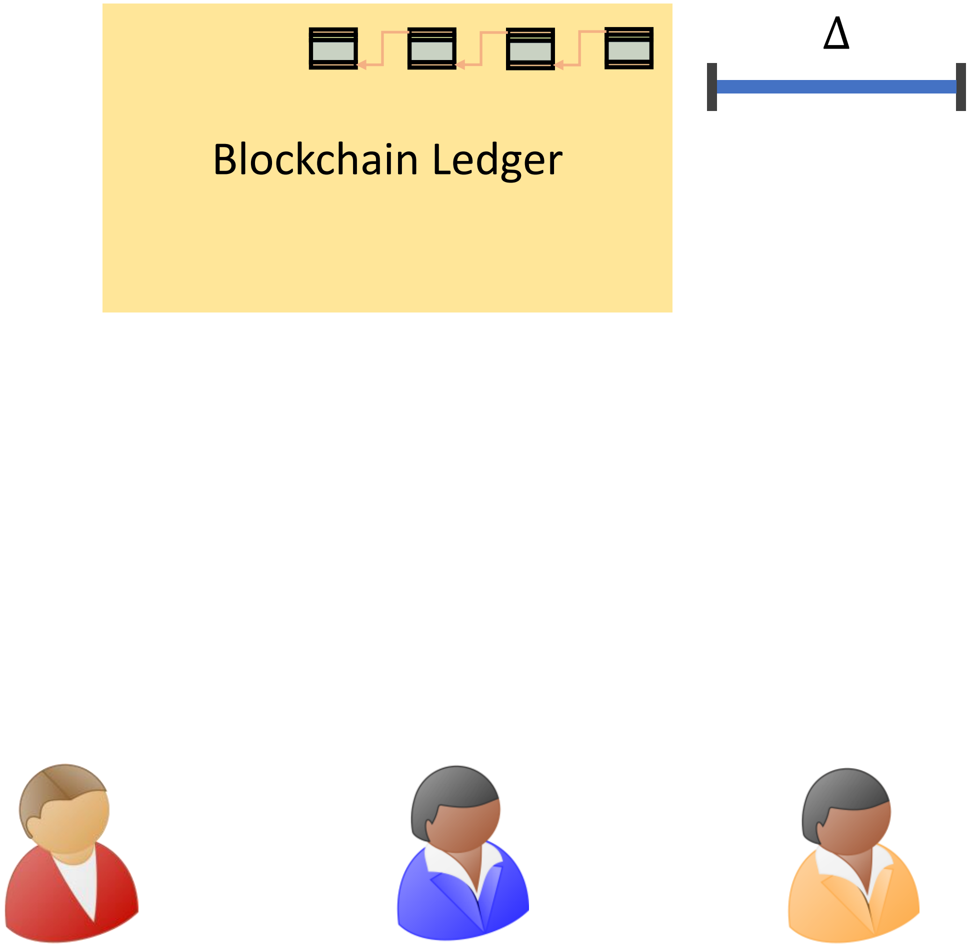
Simplified Model



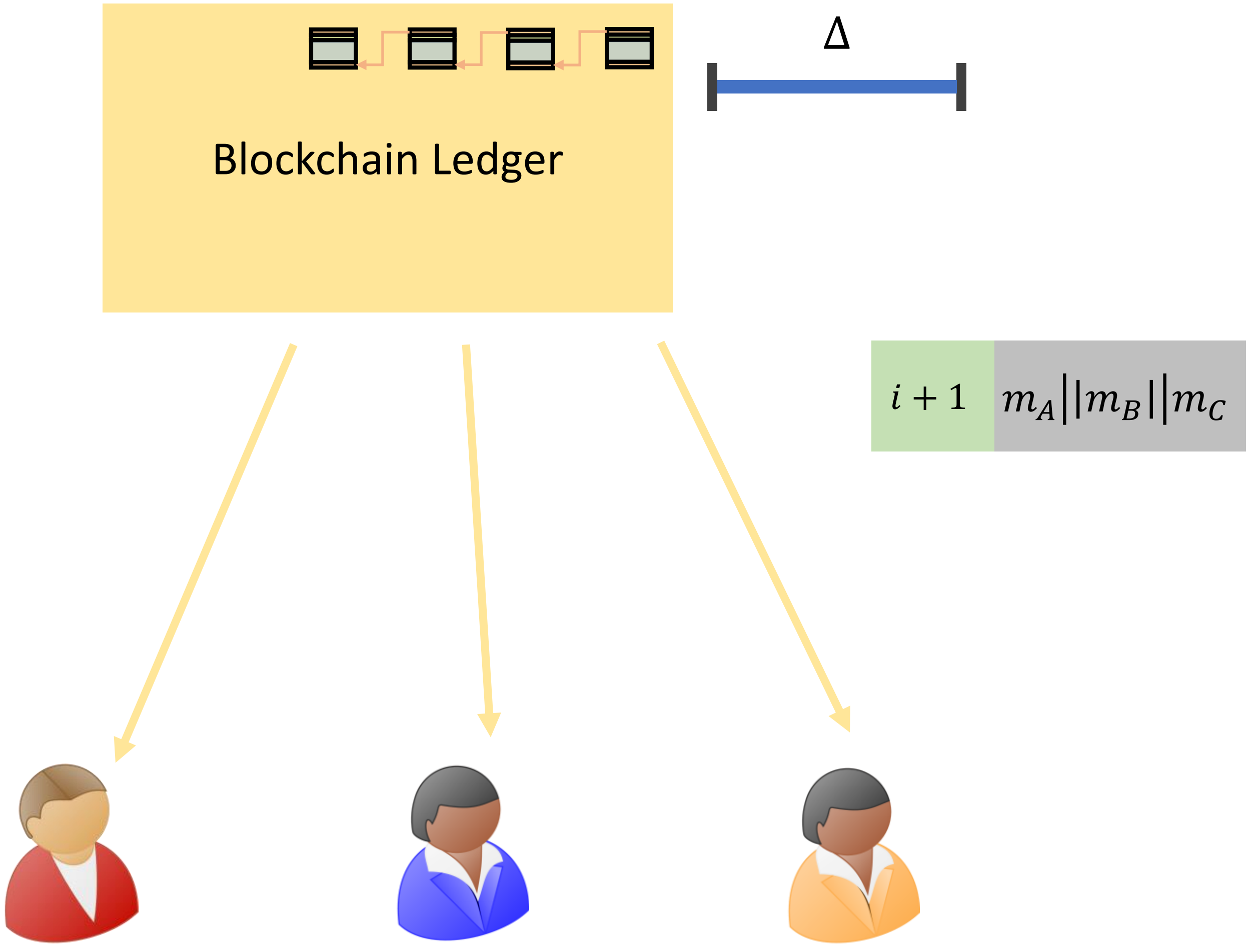
Simplified Model



Simplified Model



Simplified Model



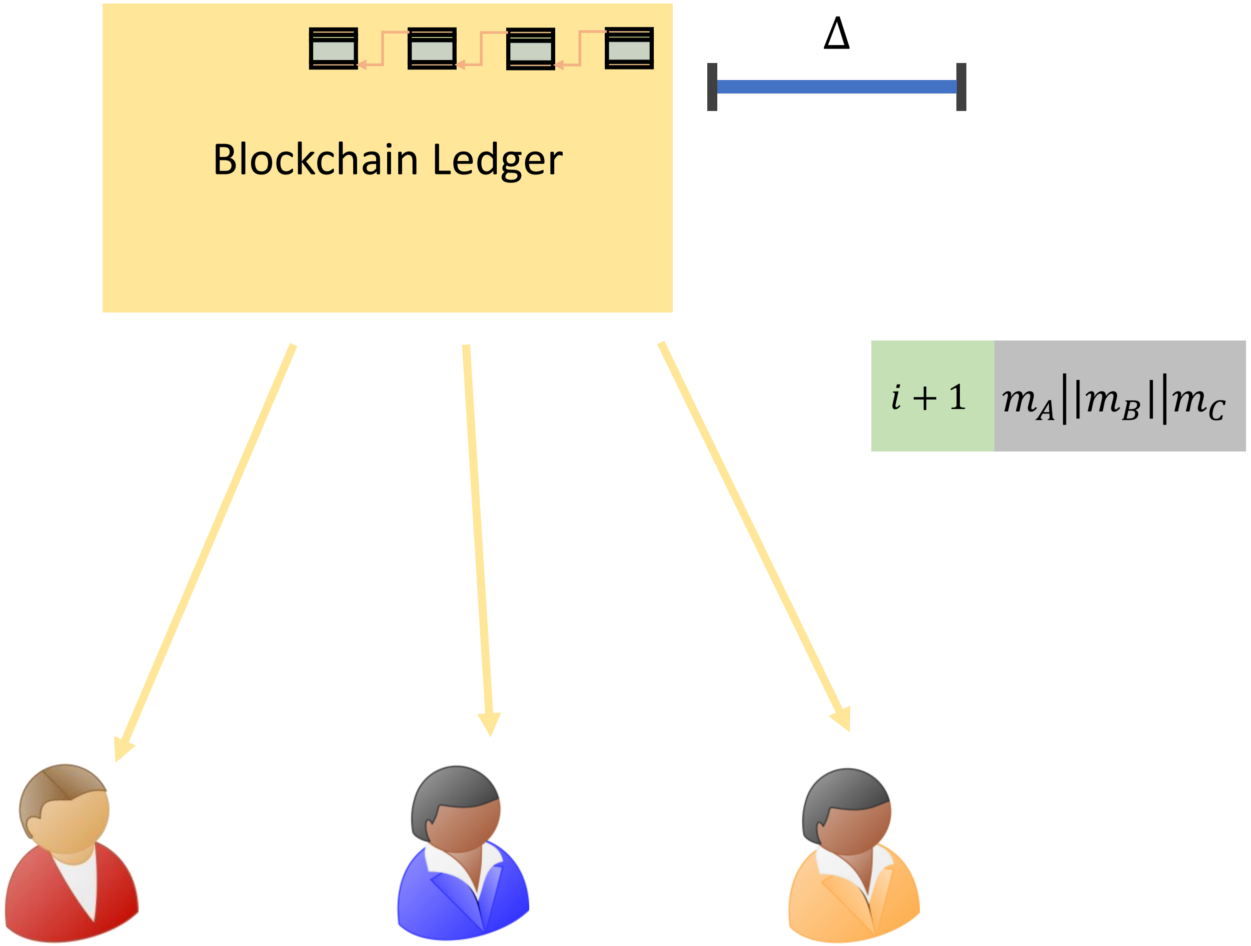
Blockchain Ledger

Δ

$i + 1$ $m_A||m_B||m_C$

Simplified Model

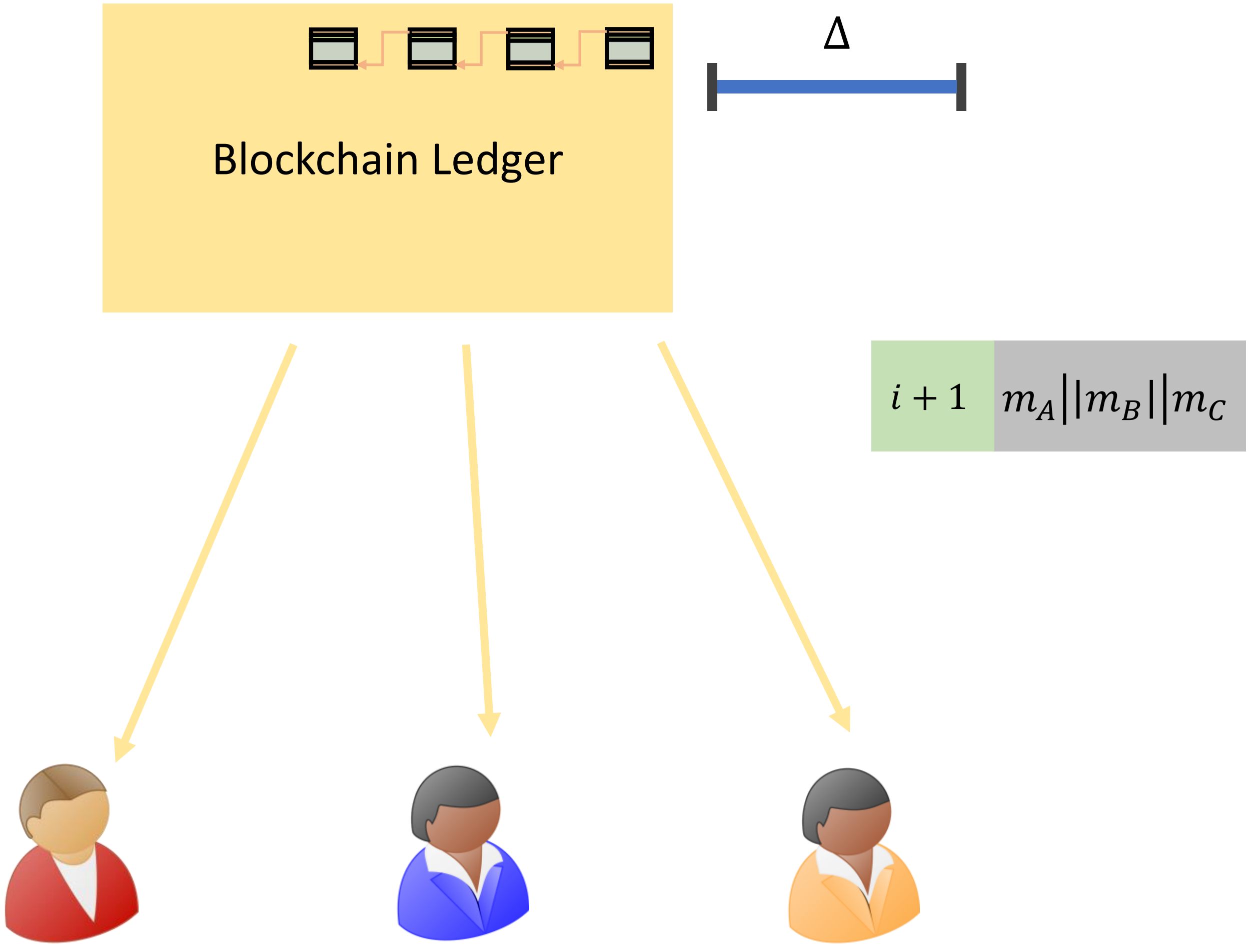
All parties have a consistent view of the blockchain



Simplified Model

All parties have a consistent view of the blockchain

A message sent to the oracle is guaranteed to appear on the next block

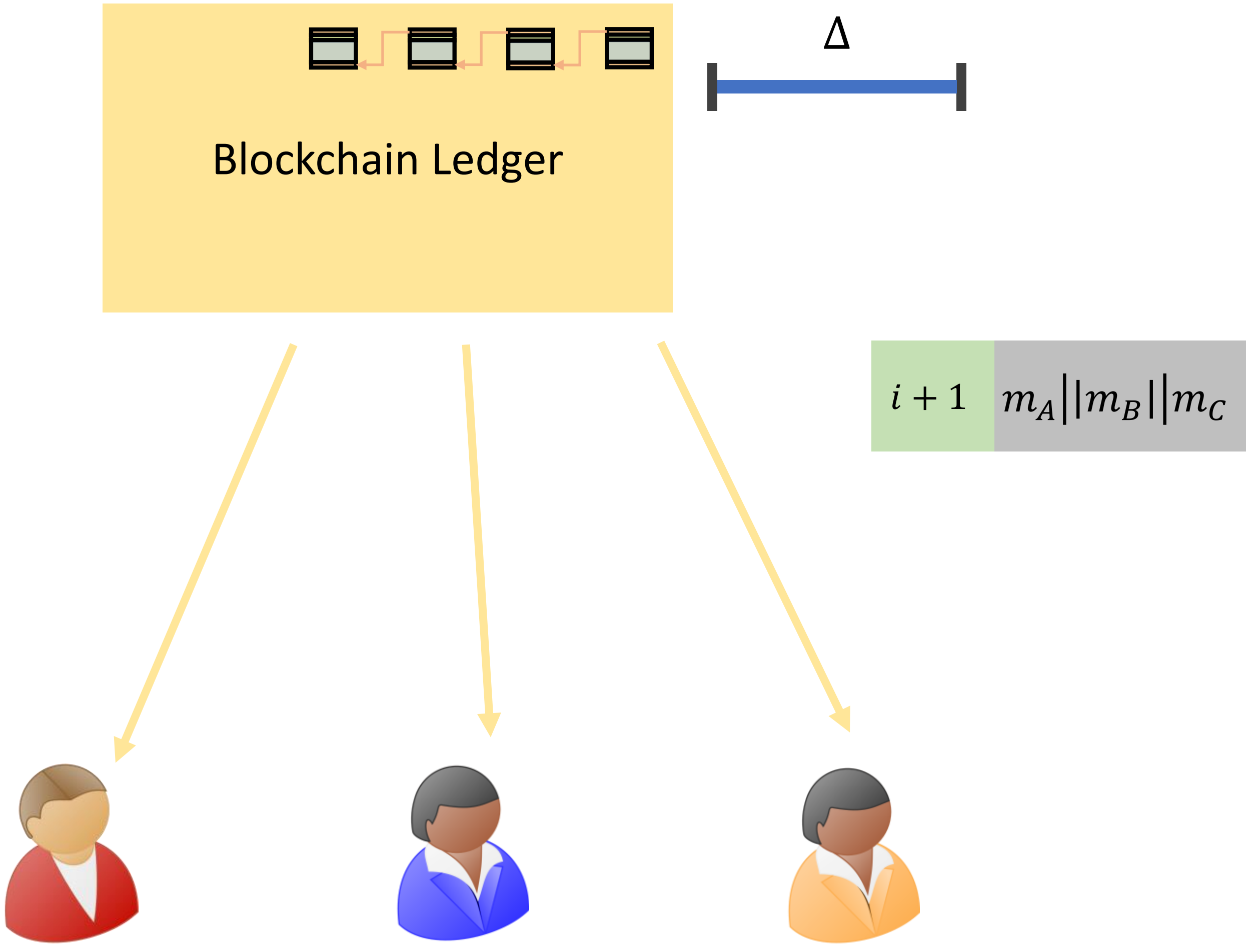


Simplified Model

All parties have a consistent view of the blockchain

A message sent to the oracle is guaranteed to appear on the next block

Only the oracle can create blocks



Blockchain Ledger

Δ

$i+1$ $m_A||m_B||m_C$

Blockchain hybrid model

A party is called blockchain active if it has post and read access to the blockchain

Blockchain hybrid model

A party is called blockchain active if it has post and read access to the blockchain

Simulator has same access to the blockchain

Blockchain hybrid model

A party is called blockchain active if it has post and read access to the blockchain

Simulator has same access to the blockchain

Local access: [Choudhuri-Green-Jain-Kaptchuk-Miers 17, Goyal-Goyal 17]

Our Results

Our Results

Black-box Zero Knowledge Impossible in the presence of blockchain active adversary

Our Results

Black-box Zero Knowledge Impossible in the presence of blockchain active adversary

$\omega(1)$ round Black-box Zero Knowledge in the blockchain hybrid model

Our Results

Black-box Zero Knowledge Impossible in the presence of blockchain active adversary

$\omega(1)$ round Black-box Zero Knowledge in the blockchain hybrid model

$O(1)$ round Black-box Zero Knowledge in the blockchain hybrid model impossible

Our Results

Black-box Zero Knowledge Impossible in the presence of blockchain active adversary

$\omega(1)$ round Black-box Zero Knowledge in the blockchain hybrid model

$O(1)$ round Black-box Zero Knowledge in the blockchain hybrid model impossible

Our Results

Concurrent secure computation possible for all functionalities in the blockchain hybrid model

Our Results

Concurrent secure computation possible for all functionalities in the blockchain hybrid model

Universally composable (UC) commitments impossible in the blockchain hybrid model

Our Results

Concurrent secure computation possible for all functionalities in the blockchain hybrid model

Universally composable (UC) commitments impossible in the blockchain hybrid model

Our Results

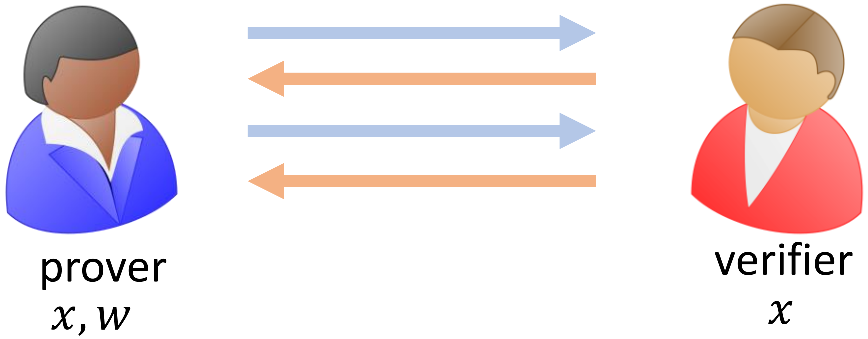
Concurrent secure computation possible for all functionalities in the blockchain hybrid model

Universally composable (UC) commitments impossible in the blockchain hybrid model

Blockchains have both destructive and constructive uses.

Zero Knowledge

Zero Knowledge (ZK)



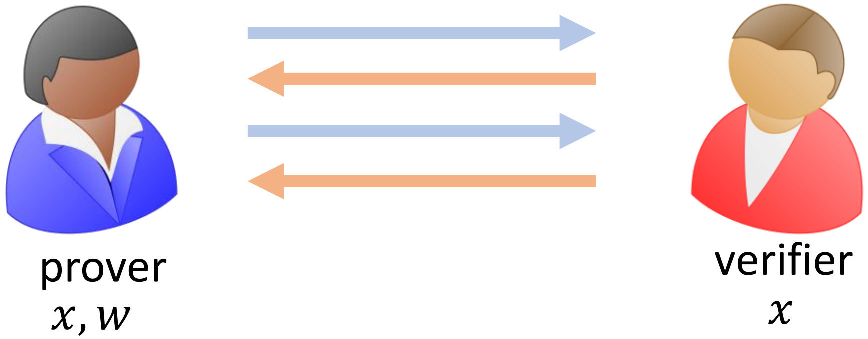
prover

x, w

verifier

x

Zero Knowledge (ZK)



prover

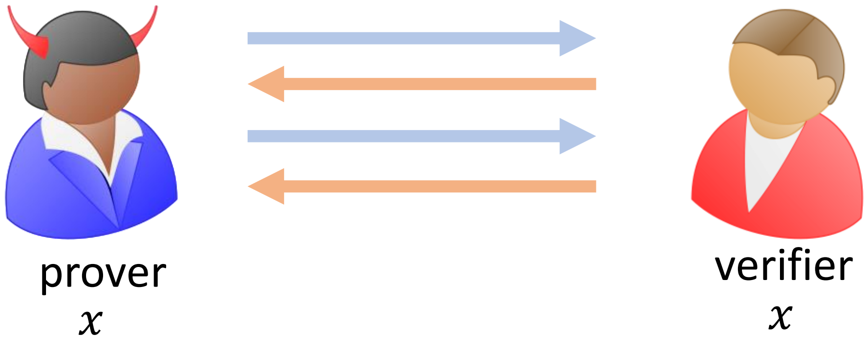
x, w

verifier

x

Completeness

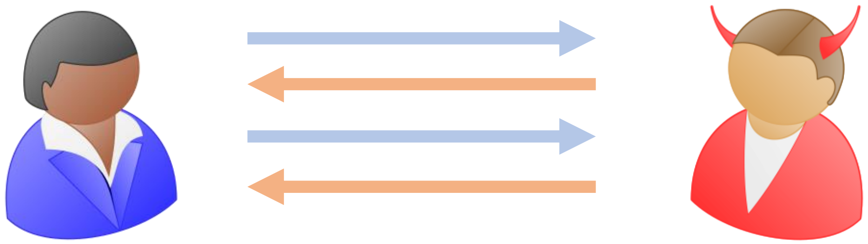
Zero Knowledge (ZK)



Completeness

Soundness

Zero Knowledge (ZK)



prover
x, w

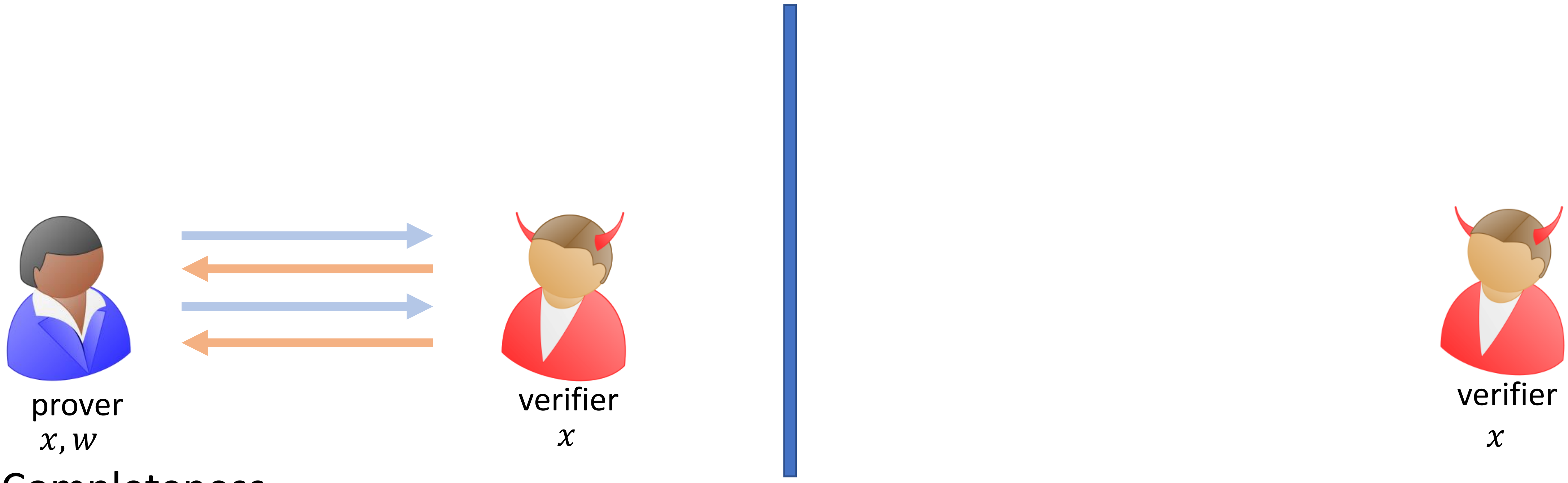
verifier
x

Completeness

Soundness

Zero Knowledge

Zero Knowledge (ZK)



prover

x, w

verifier

x

verifier

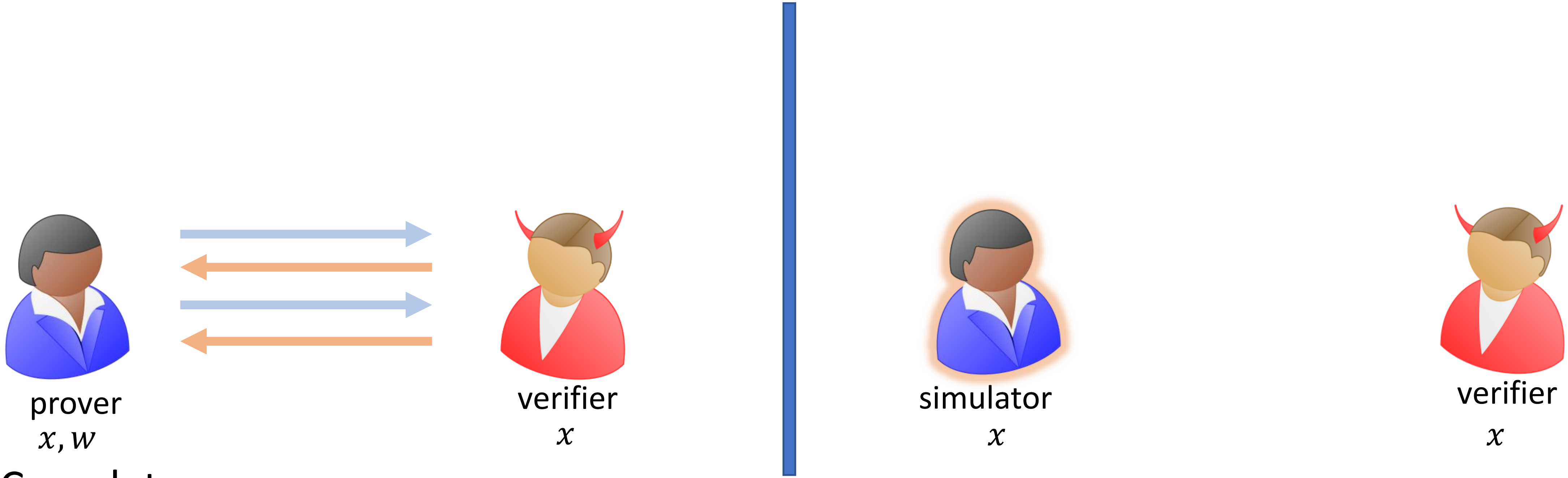
x

Completeness

Soundness

Zero Knowledge

Zero Knowledge (ZK)



prover

x, w

verifier

x

simulator

x

verifier

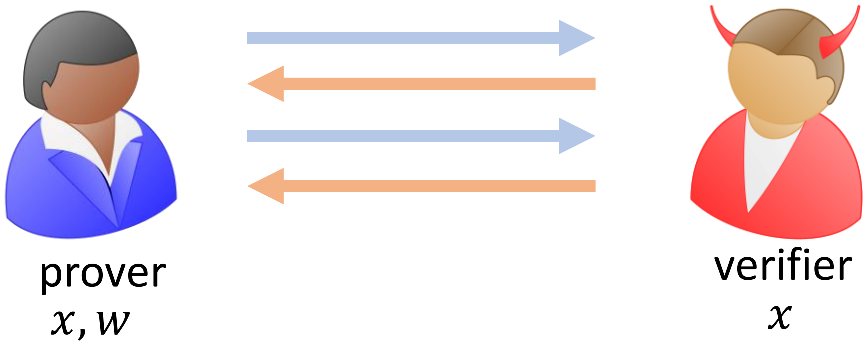
x

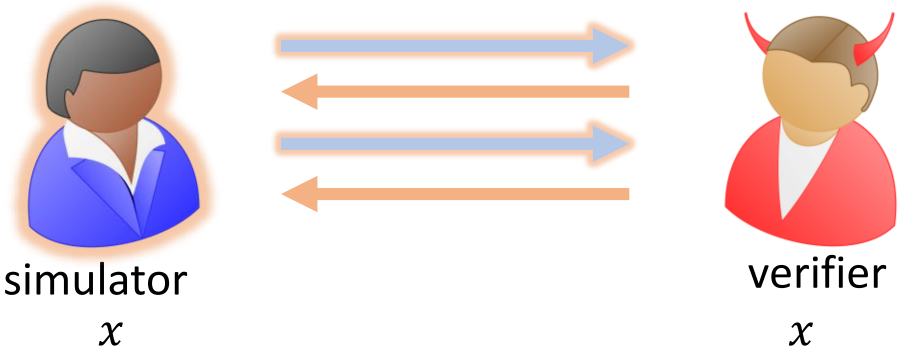
Completeness

Soundness

Zero Knowledge

Zero Knowledge (ZK)



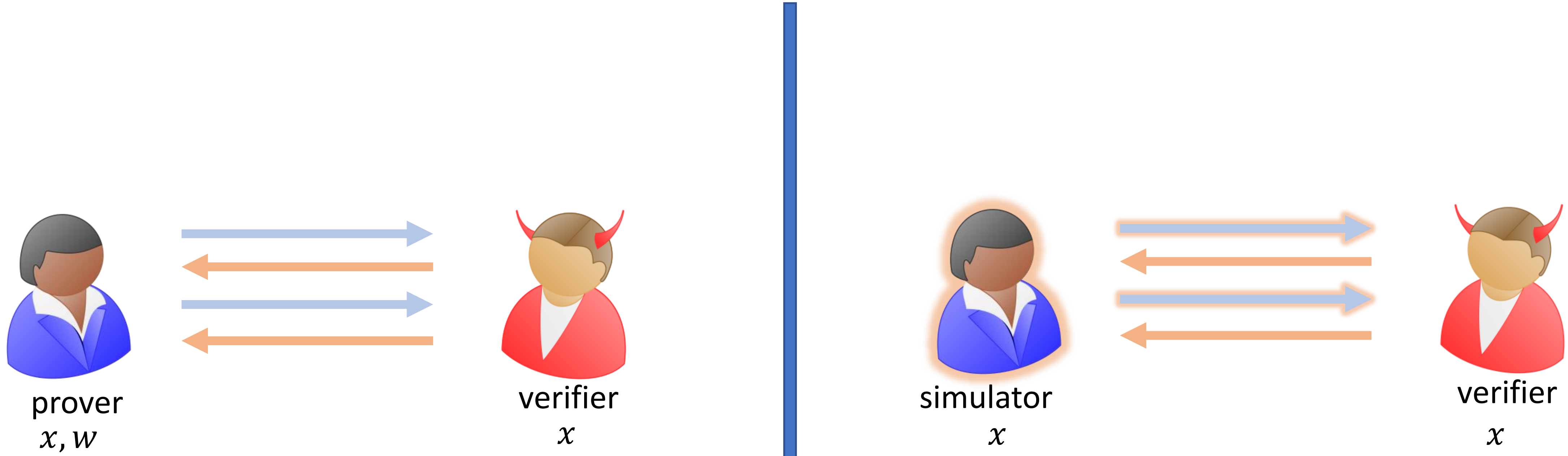


Completeness

Soundness

Zero Knowledge

Zero Knowledge (ZK)

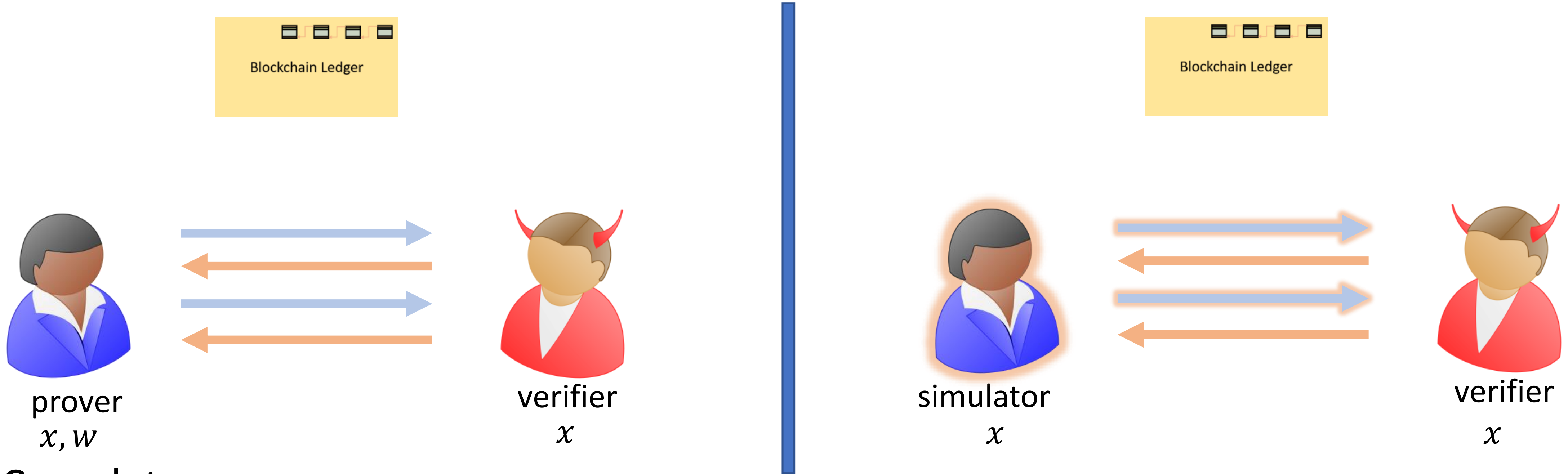


Completeness

Soundness

Zero Knowledge

Zero Knowledge (ZK)



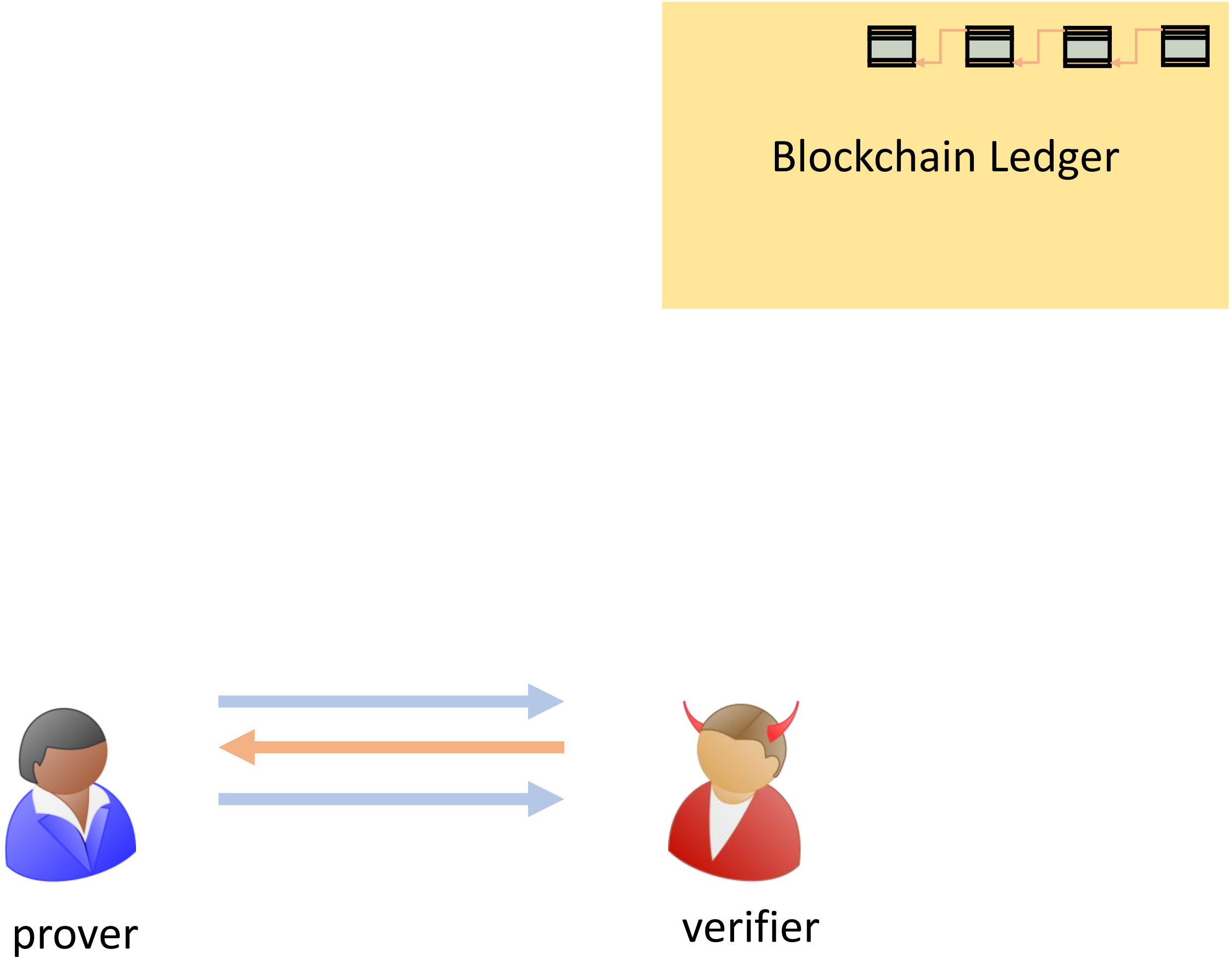
$\approx$

Completeness

Soundness

Zero Knowledge

ZK Impossible against Blockchain Active Adversaries

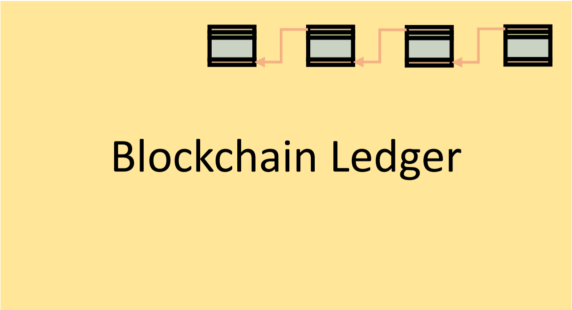


Blockchain Ledger

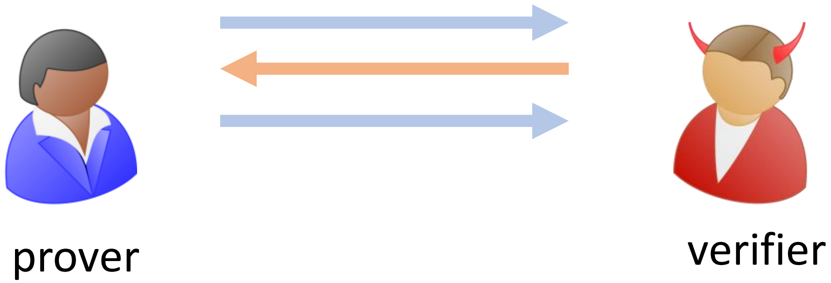
prover

verifier

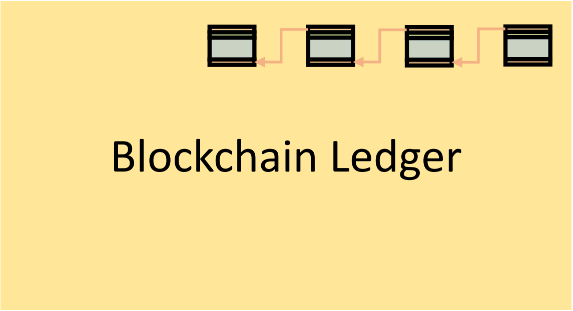
ZK Impossible against Blockchain Active Adversaries



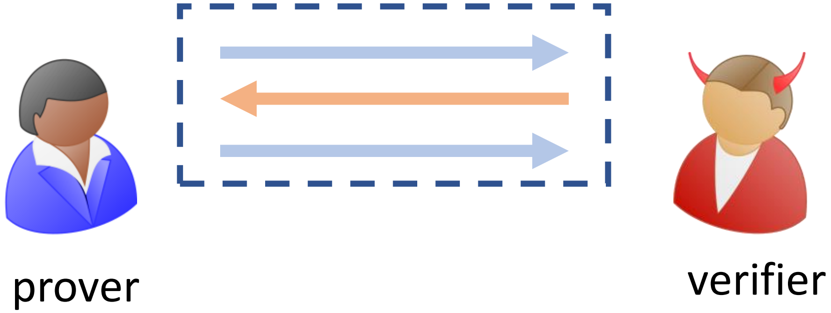
Prevent Simulator from rewinding the verifier.



ZK Impossible against Blockchain Active Adversaries

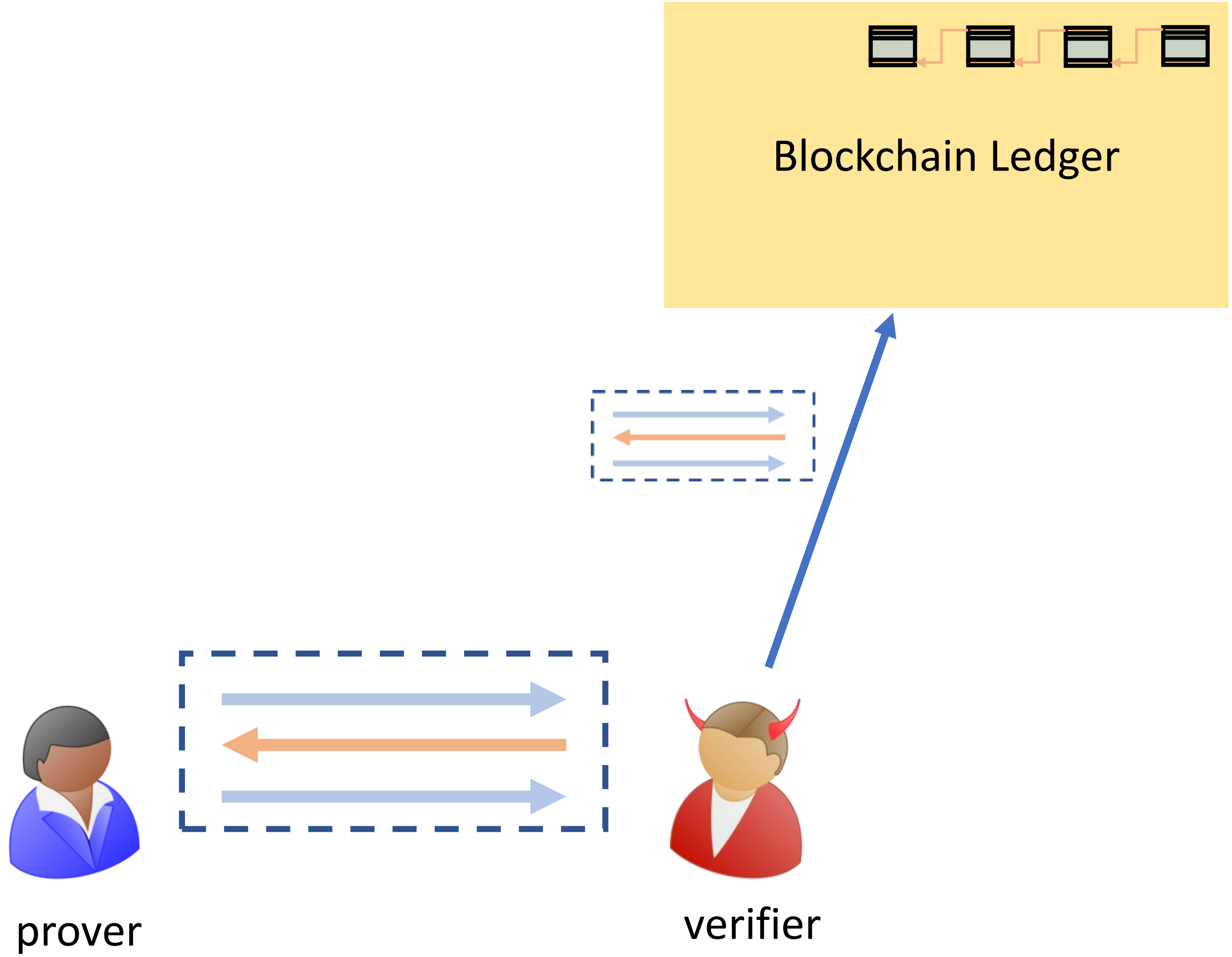


Prevent Simulator from rewinding the verifier.



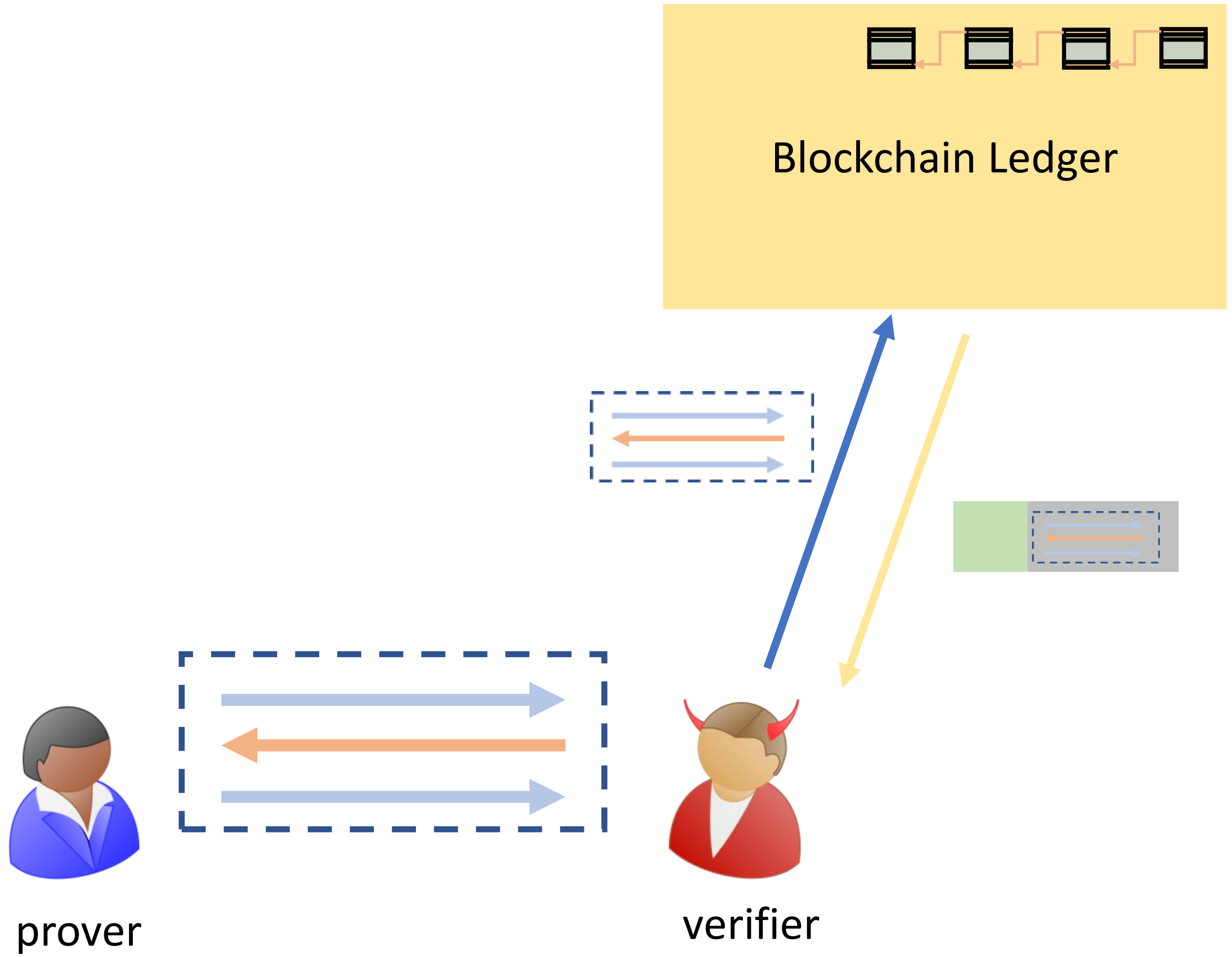
ZK Impossible against Blockchain Active Adversaries

Prevent Simulator from rewinding the verifier.



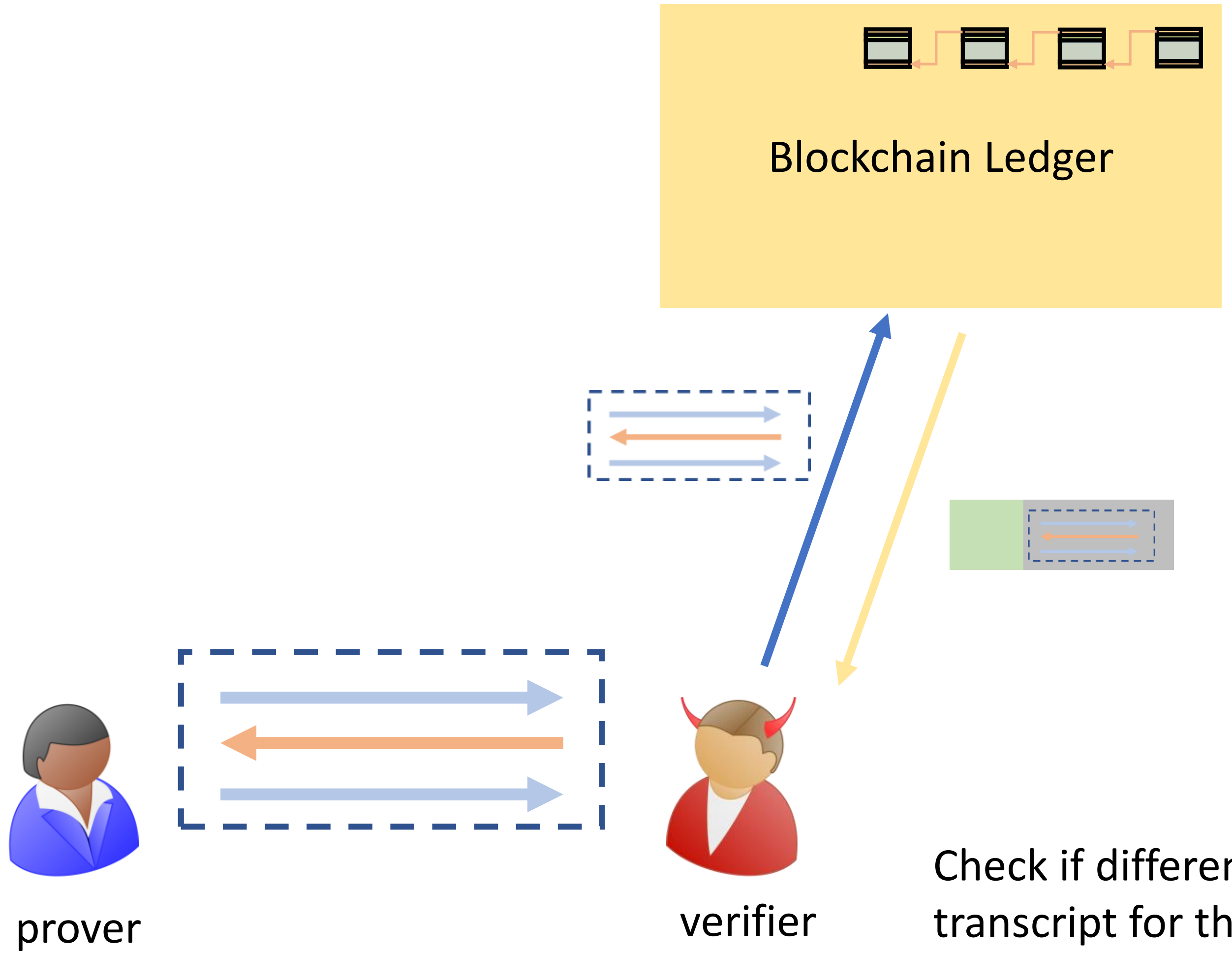
ZK Impossible against Blockchain Active Adversaries

Prevent Simulator from rewinding the verifier.



ZK Impossible against Blockchain Active Adversaries

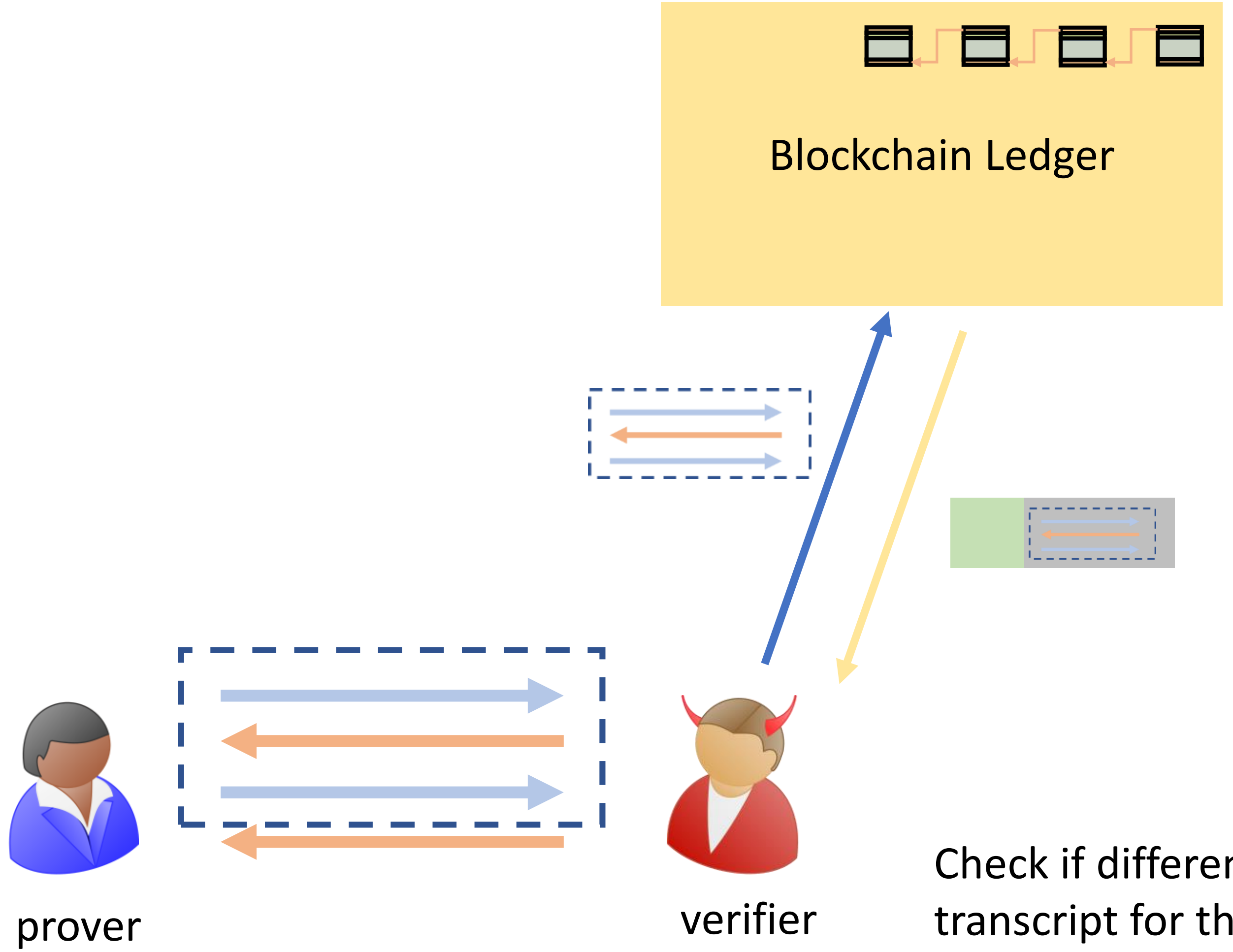
Prevent Simulator from rewinding the verifier.



Check if different transcript for the same session.

ZK Impossible against Blockchain Active Adversaries

Prevent Simulator from rewinding the verifier.



Check if different transcript for the same session.

Achieving ZK in the Blockchain hybrid model

Impossible: Only adversary is blockchain active

Achieving ZK in the Blockchain hybrid model

Impossible: Only adversary is blockchain active

Positive result: All parties are blockchain active

Structure of the Zero Knowledge Protocol

[Prabhakaran-Rosen-Sahai 02]

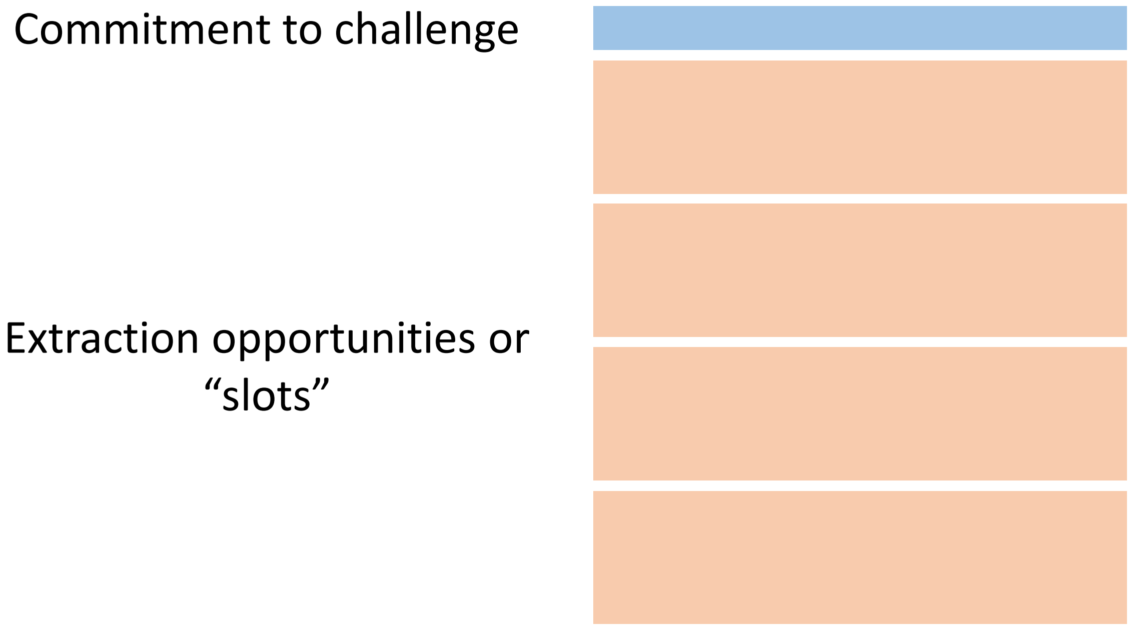
Structure of the Zero Knowledge Protocol

[Prabhakaran-Rosen-Sahai 02]

Commitment to challenge

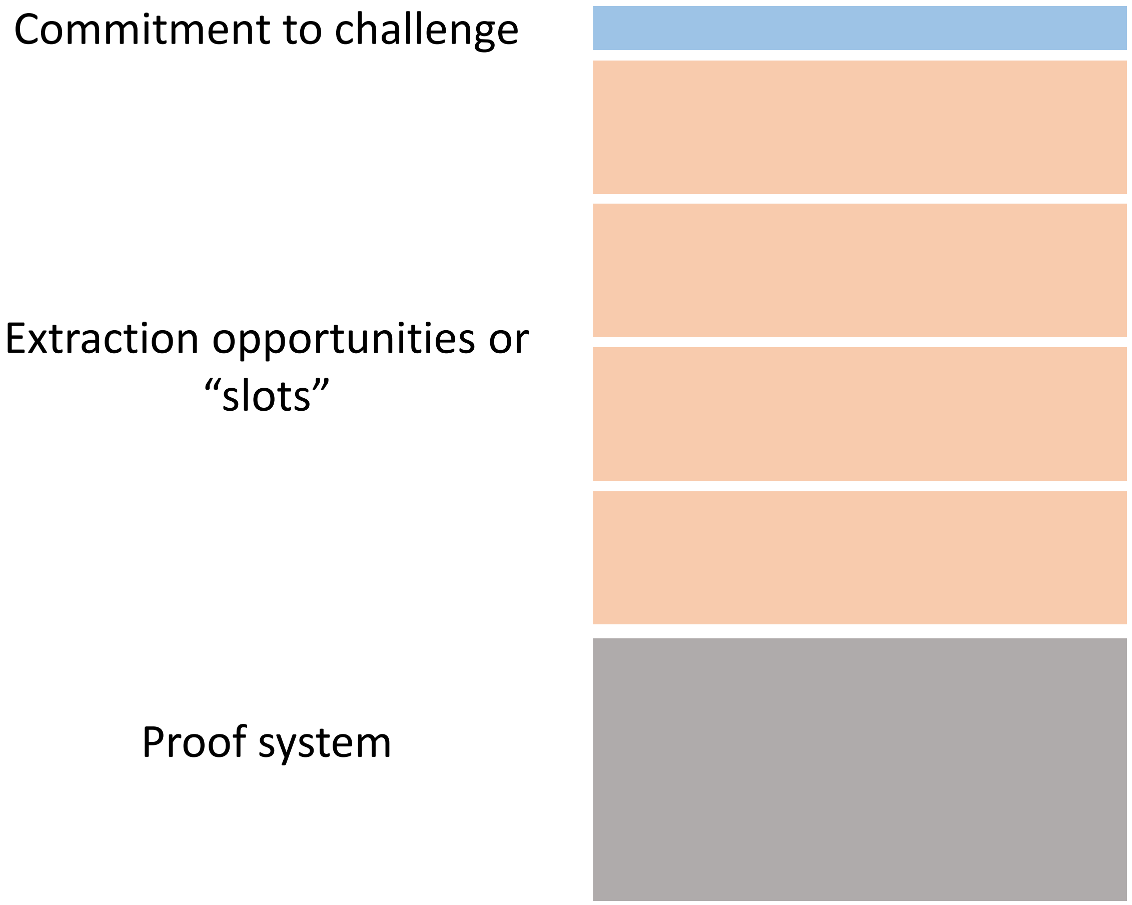
Structure of the Zero Knowledge Protocol

[Prabhakaran-Rosen-Sahai 02]



Structure of the Zero Knowledge Protocol

[Prabhakaran-Rosen-Sahai 02]



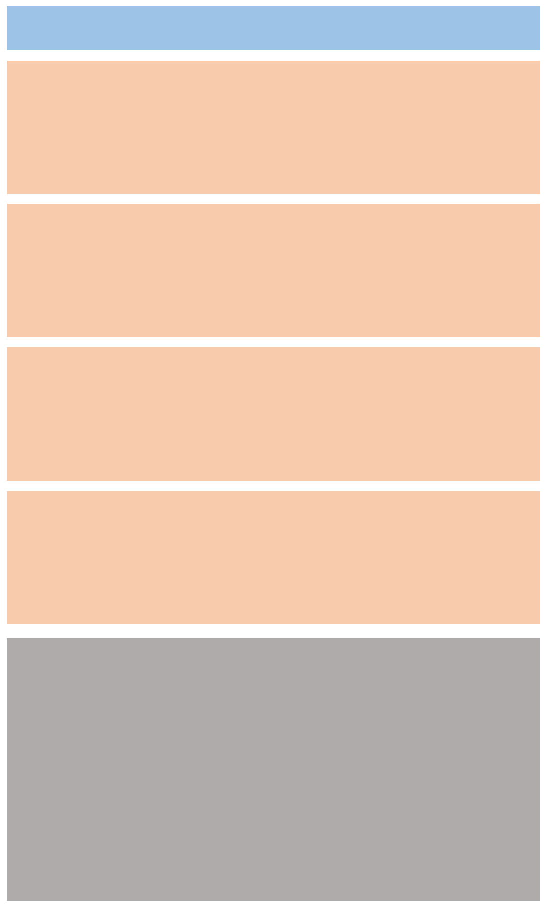
Commitment to challenge

Extraction opportunities or “slots”

Proof system

Structure of the Zero Knowledge Protocol

[Prabhakaran-Rosen-Sahai 02]



Commitment to challenge

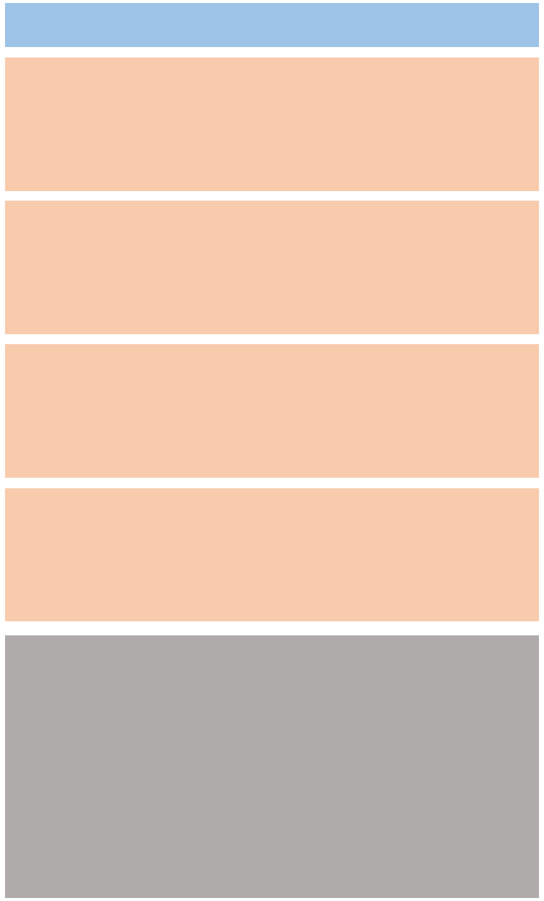
Extraction opportunities or "slots"

Proof system

E.g. Hamiltonicity Proof System

Structure of the Zero Knowledge Protocol

[Prabhakaran-Rosen-Sahai 02]



Commitment to challenge

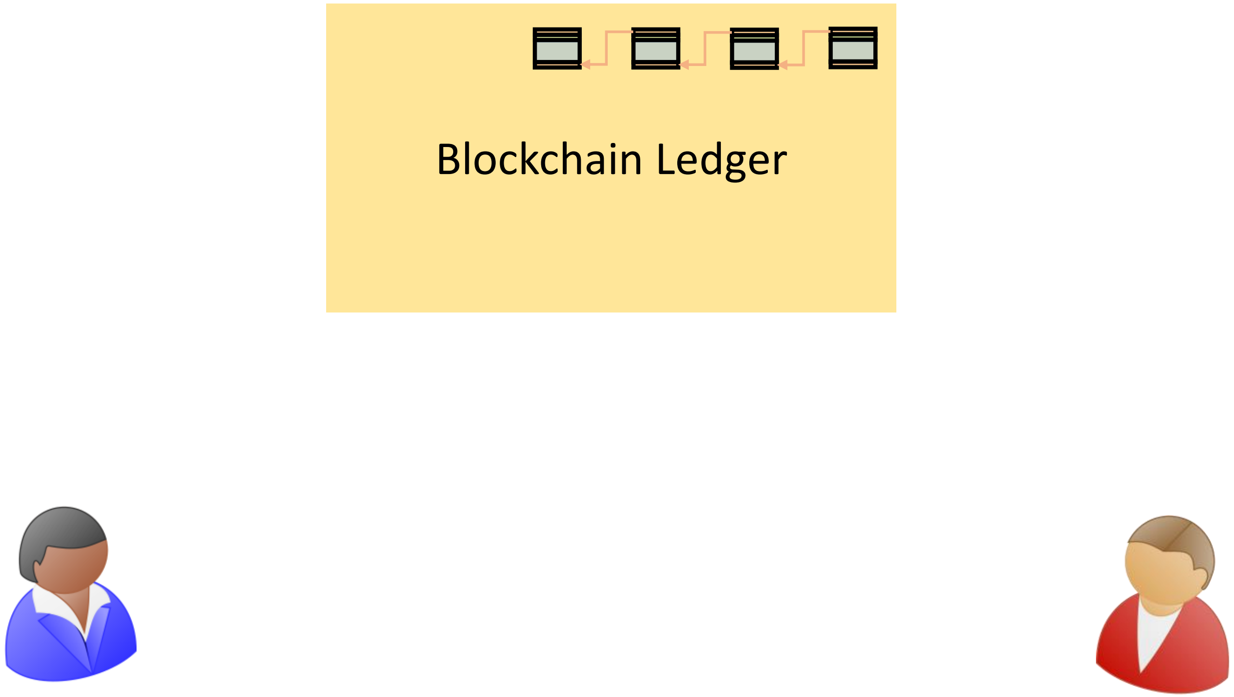
Extraction opportunities or "slots"

Proof system

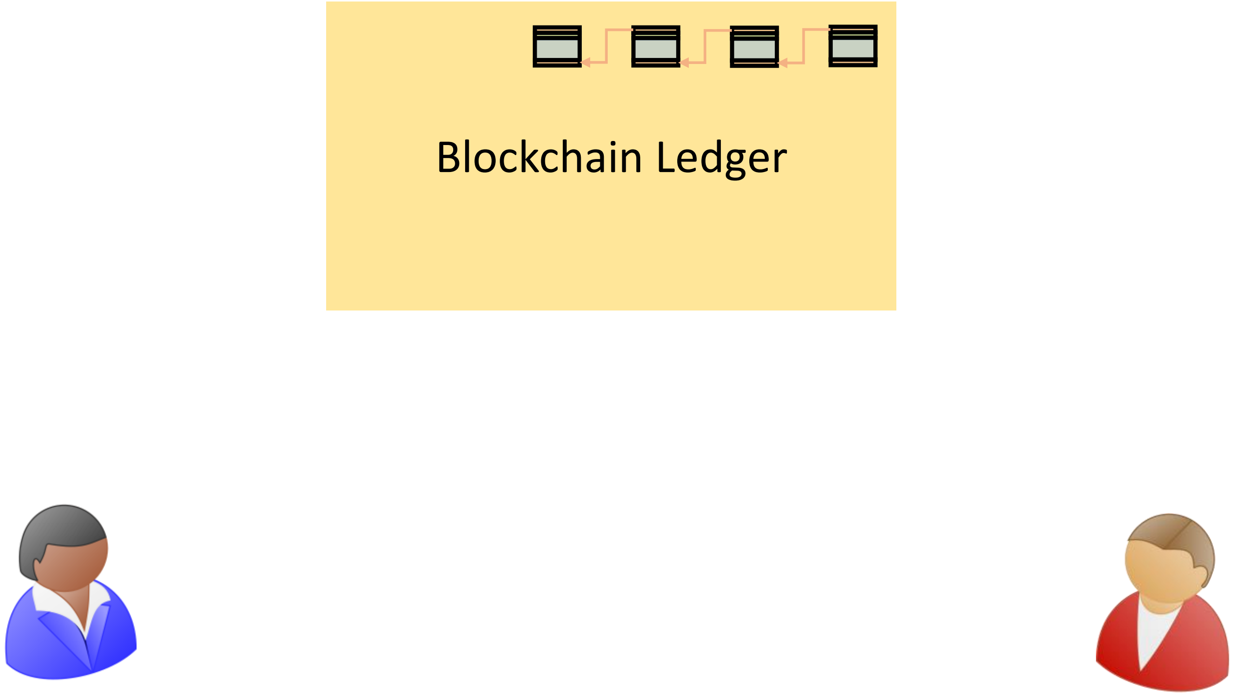
E.g. Hamiltonicity Proof System

Simulation Guarantee:

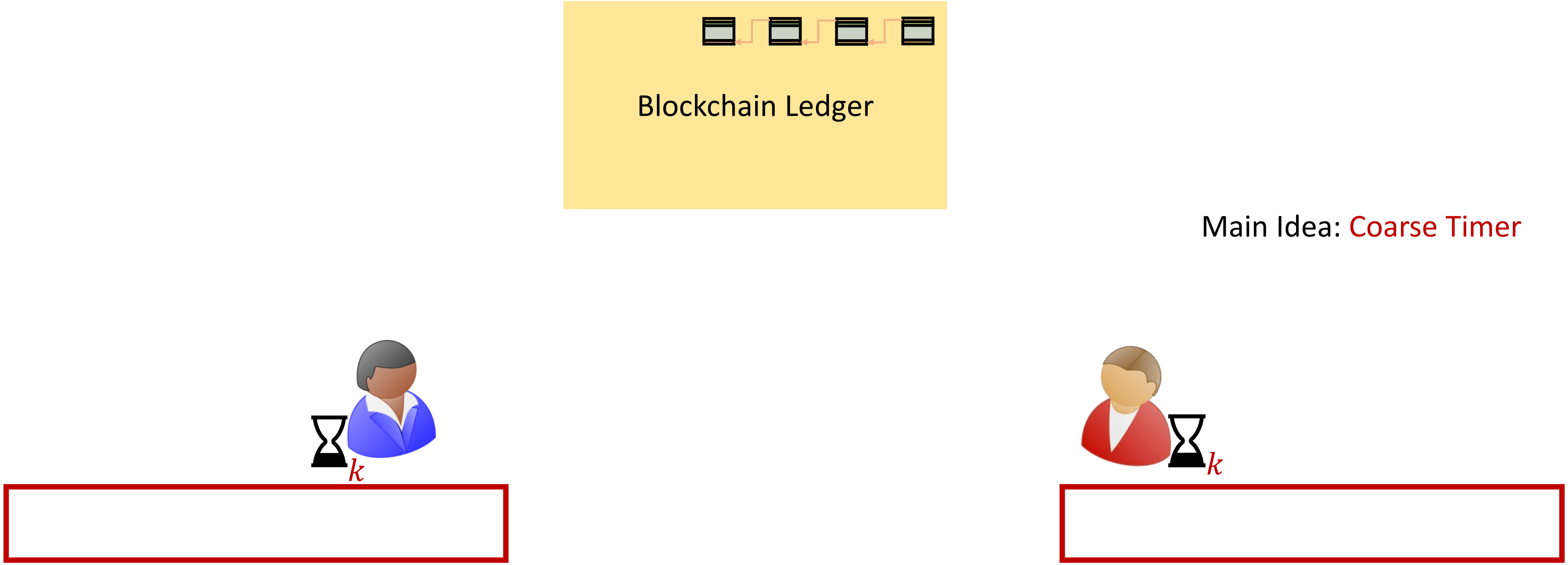
If extraction succeeds in one of the slots, the simulation can be performed in a simple manner without rewinding.



Blockchain Ledger

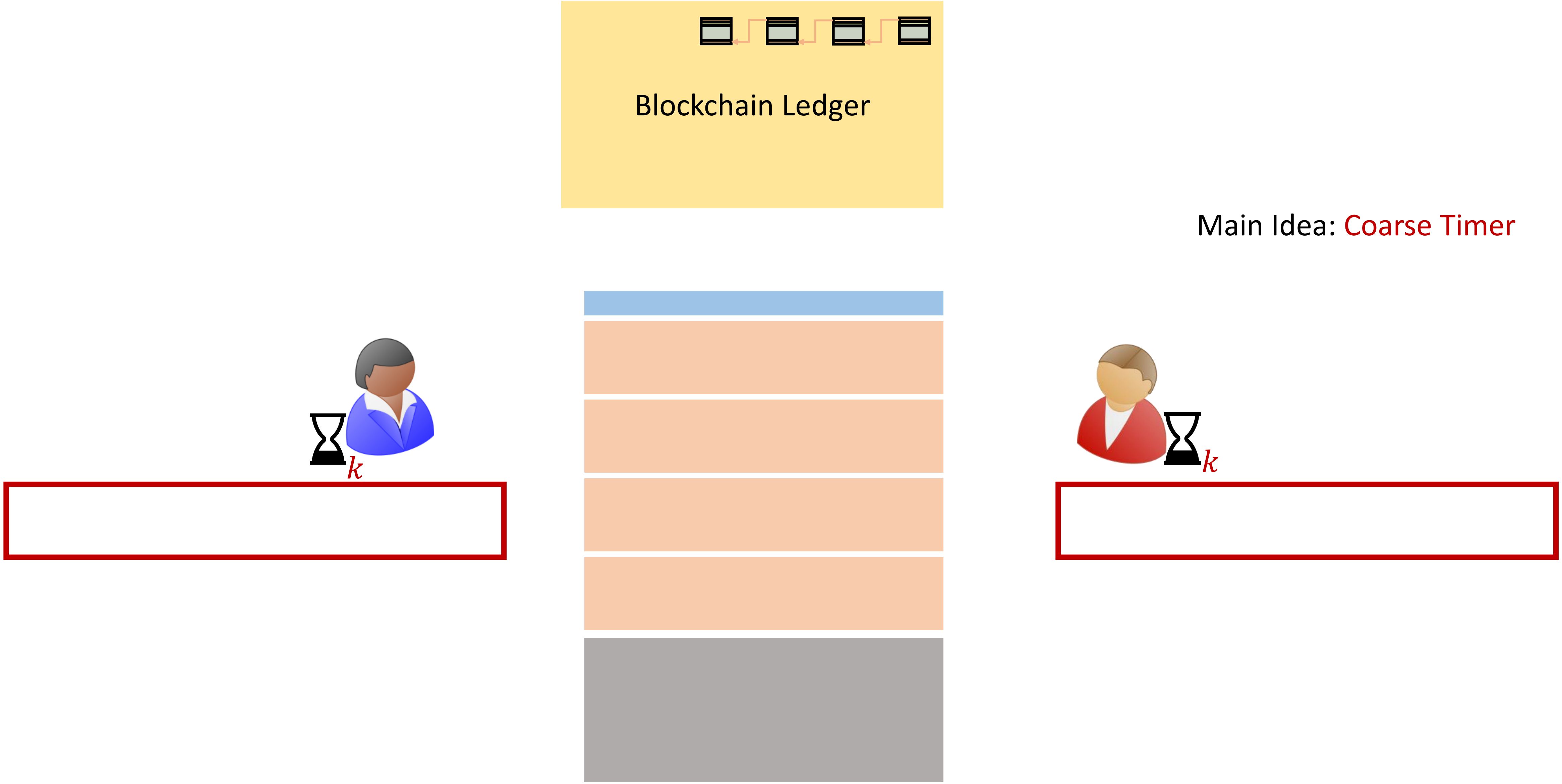


Main Idea: Coarse Timer

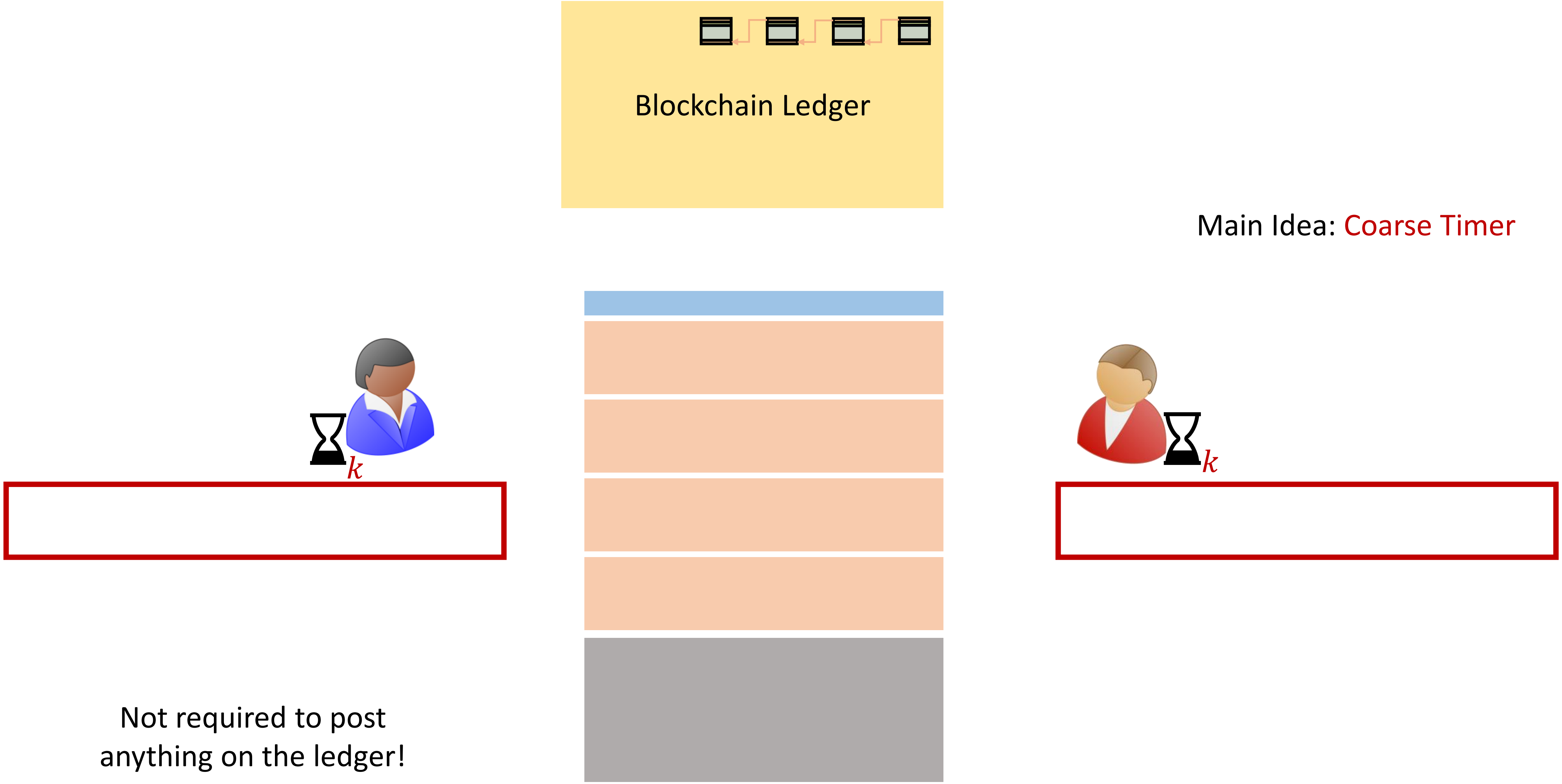


Main Idea: Coarse Timer

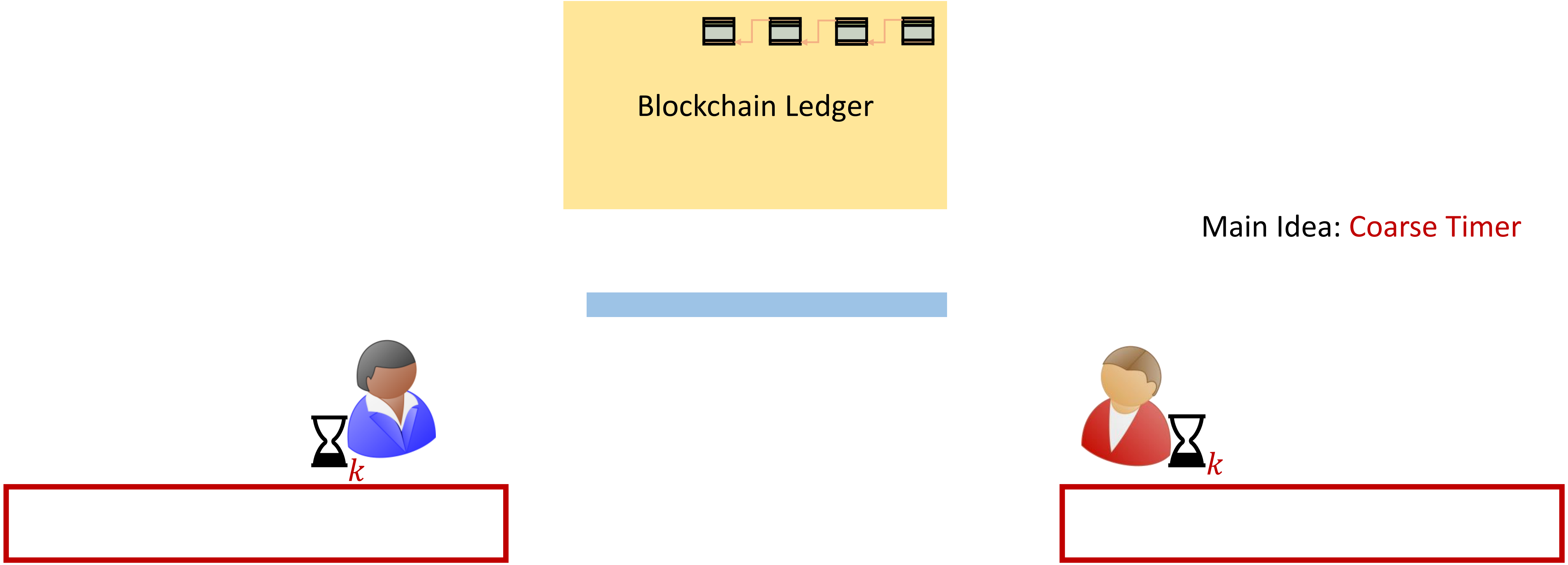
Main Idea: Coarse Timer



Main Idea: Coarse Timer



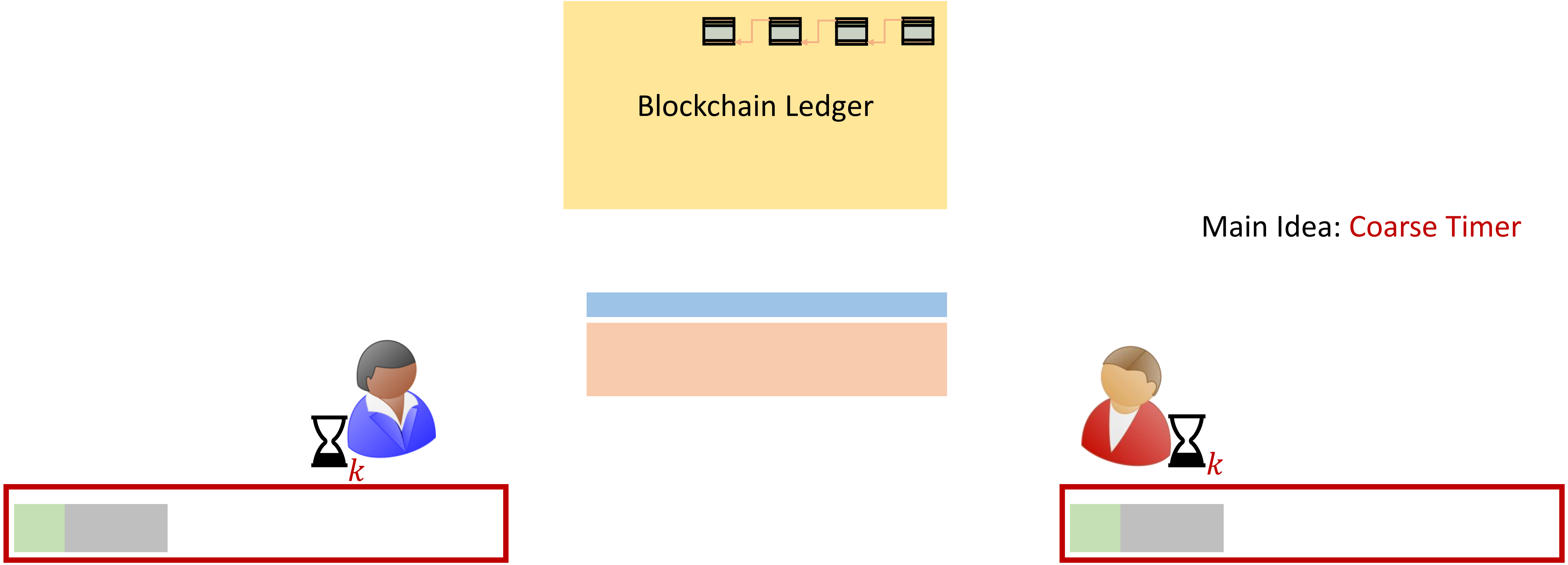
Not required to post
anything on the ledger!

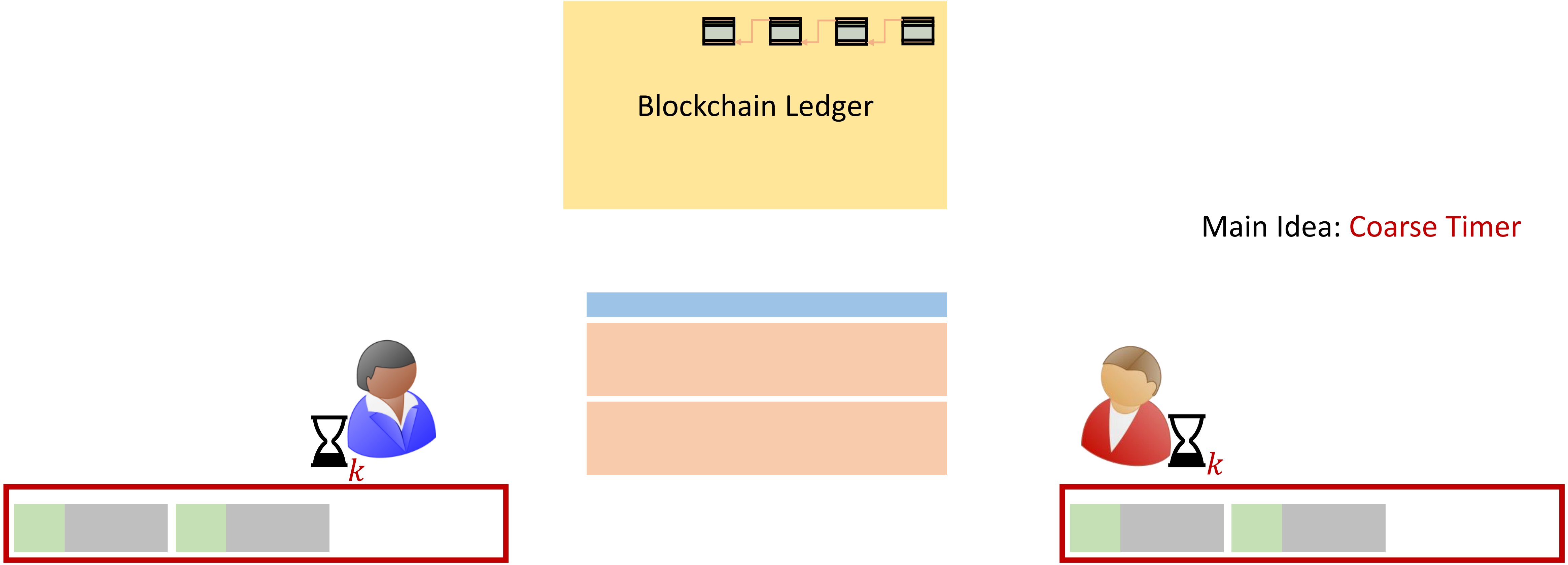


Blockchain Ledger

Main Idea: Coarse Timer

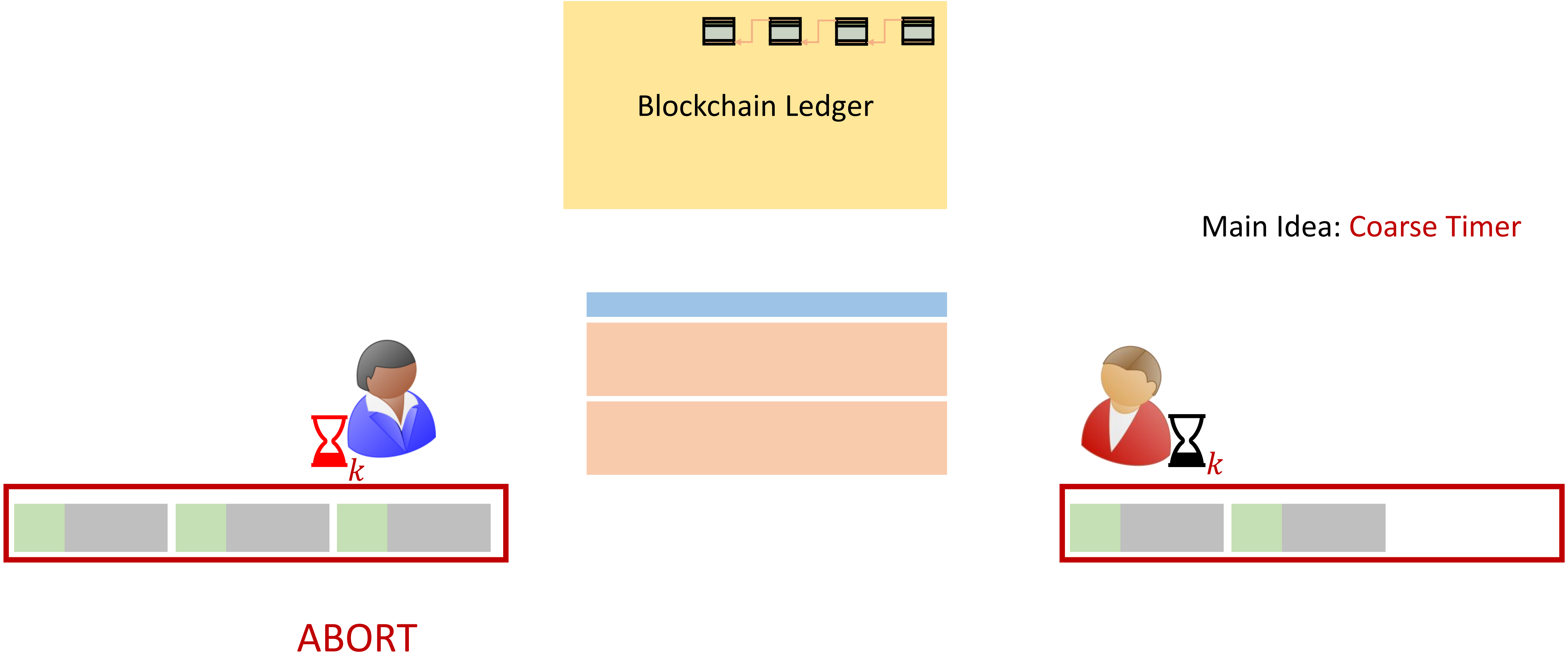
Main Idea: Coarse Timer



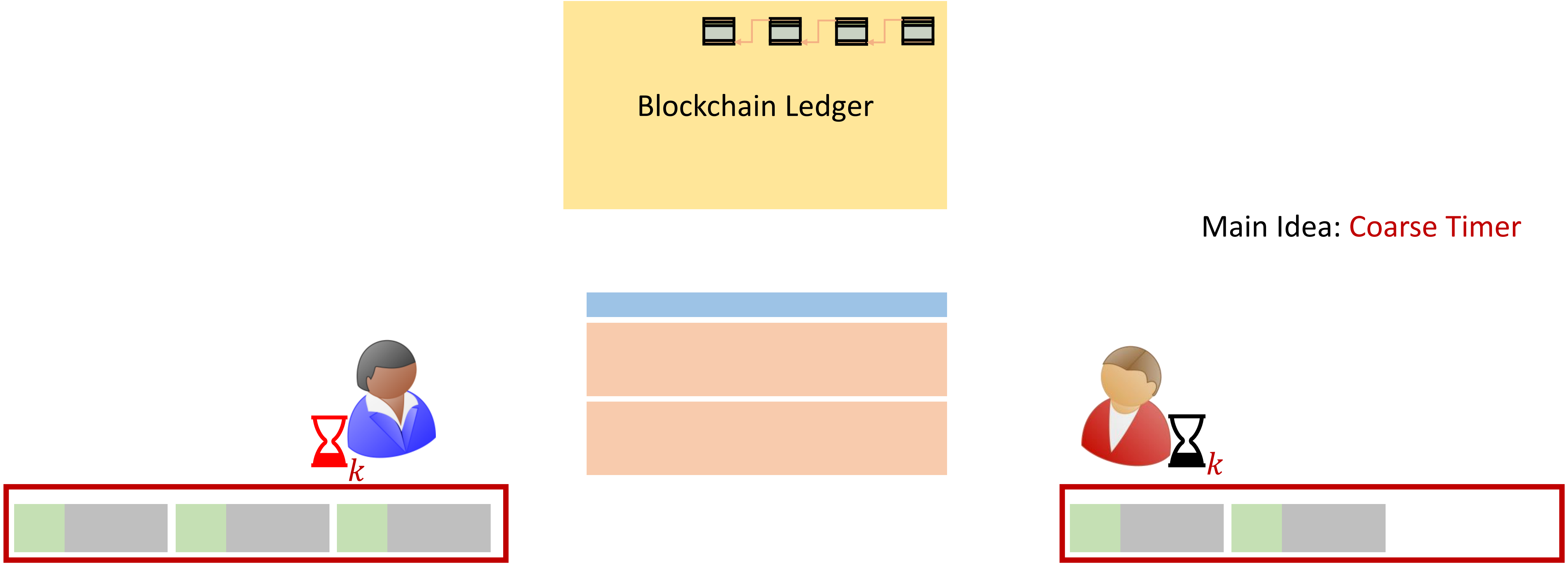


Main Idea: Coarse Timer

Main Idea: Coarse Timer

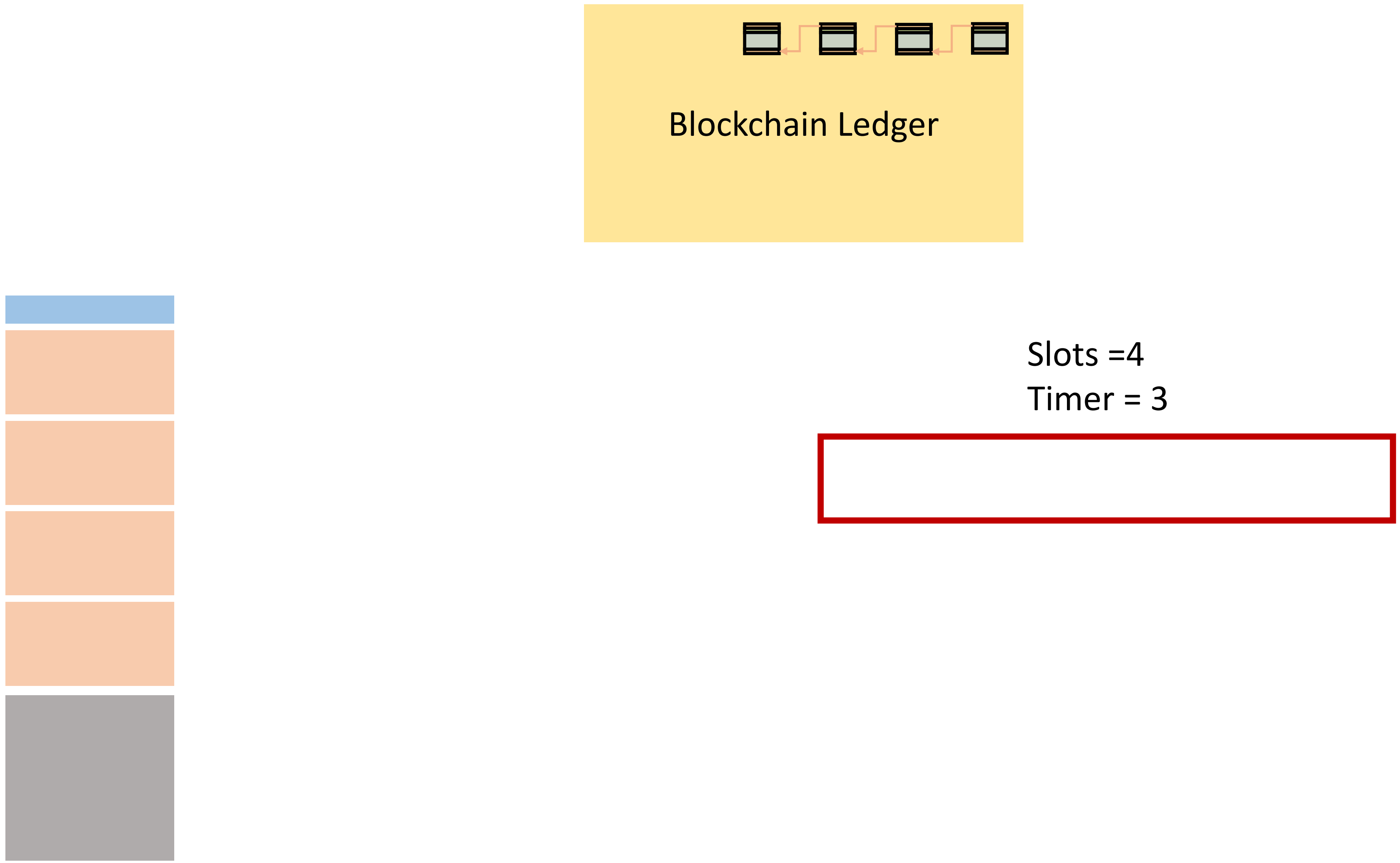


Main Idea: Coarse Timer



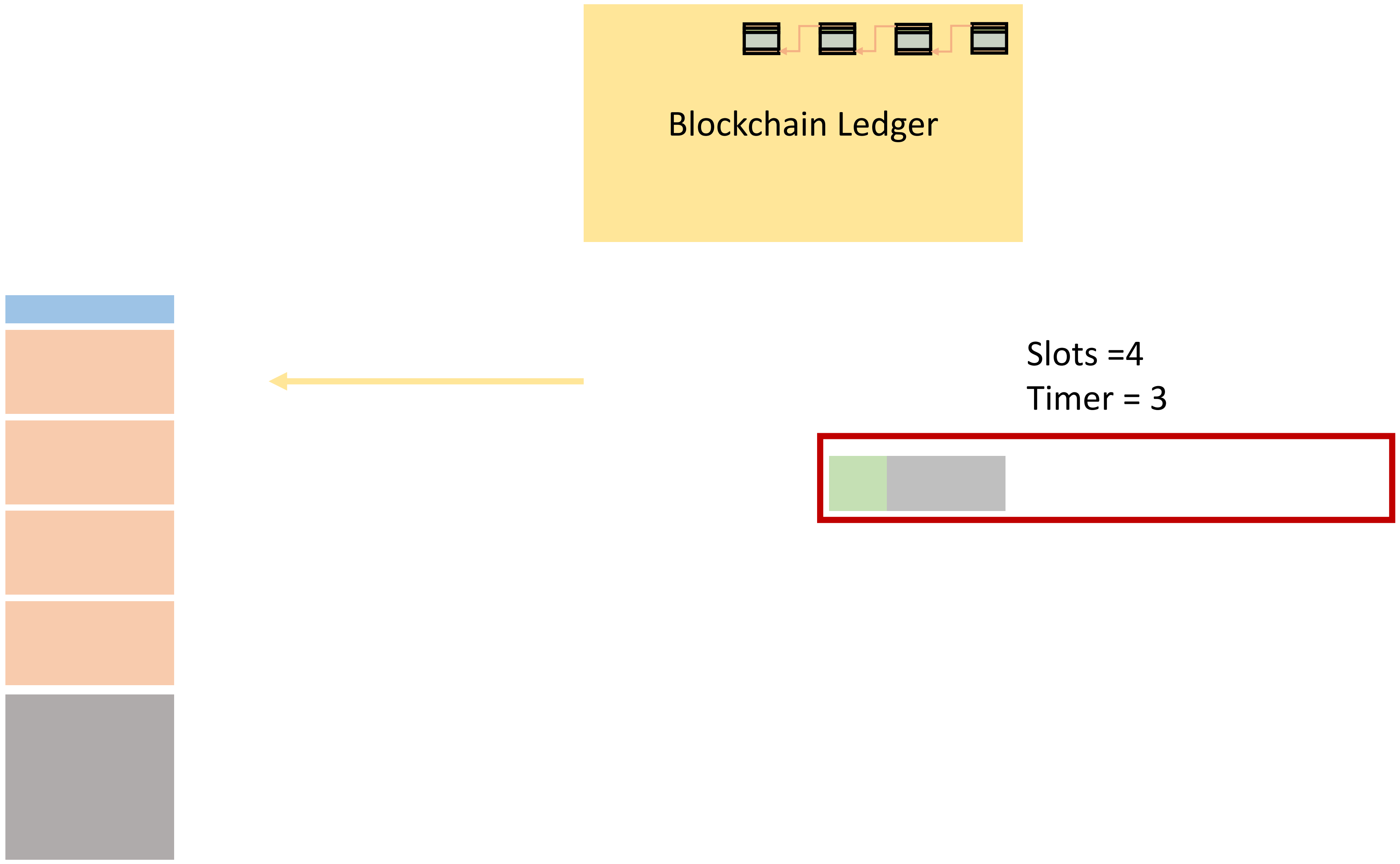
ABORT

Why is this helpful?



Blockchain Ledger

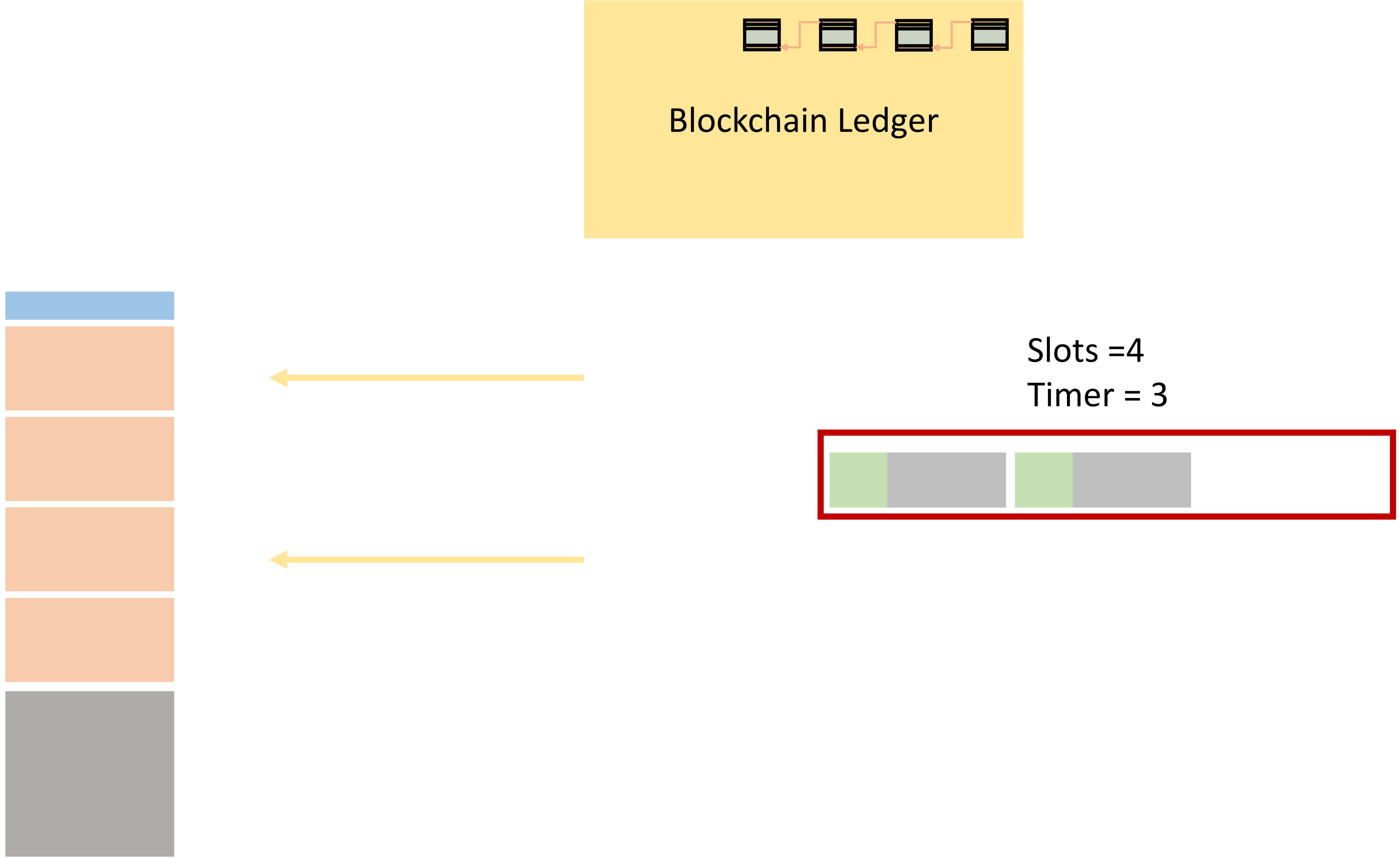
Slots =4
Timer = 3



Blockchain Ledger

Slots =4

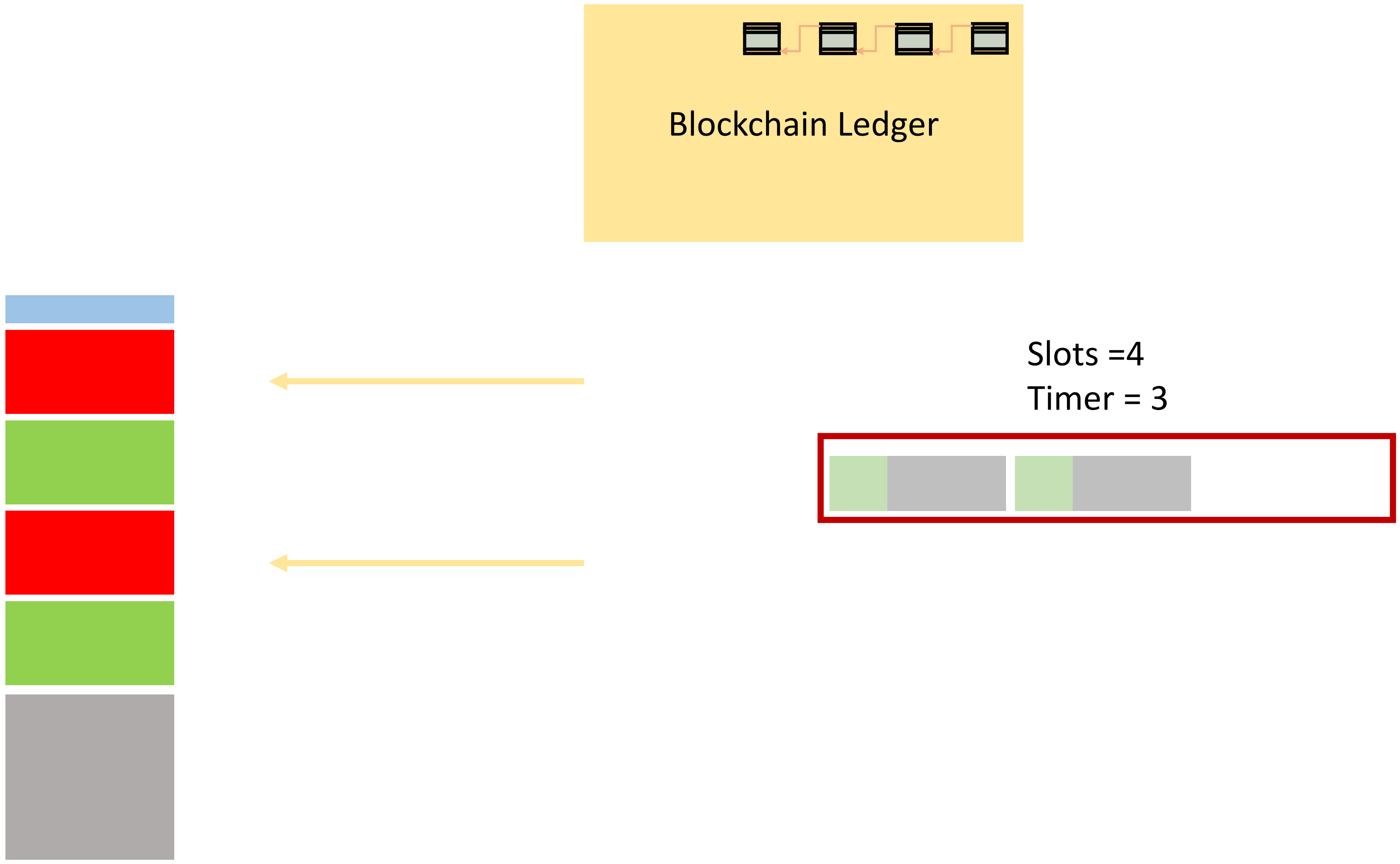
Timer = 3



Blockchain Ledger

Slots =4

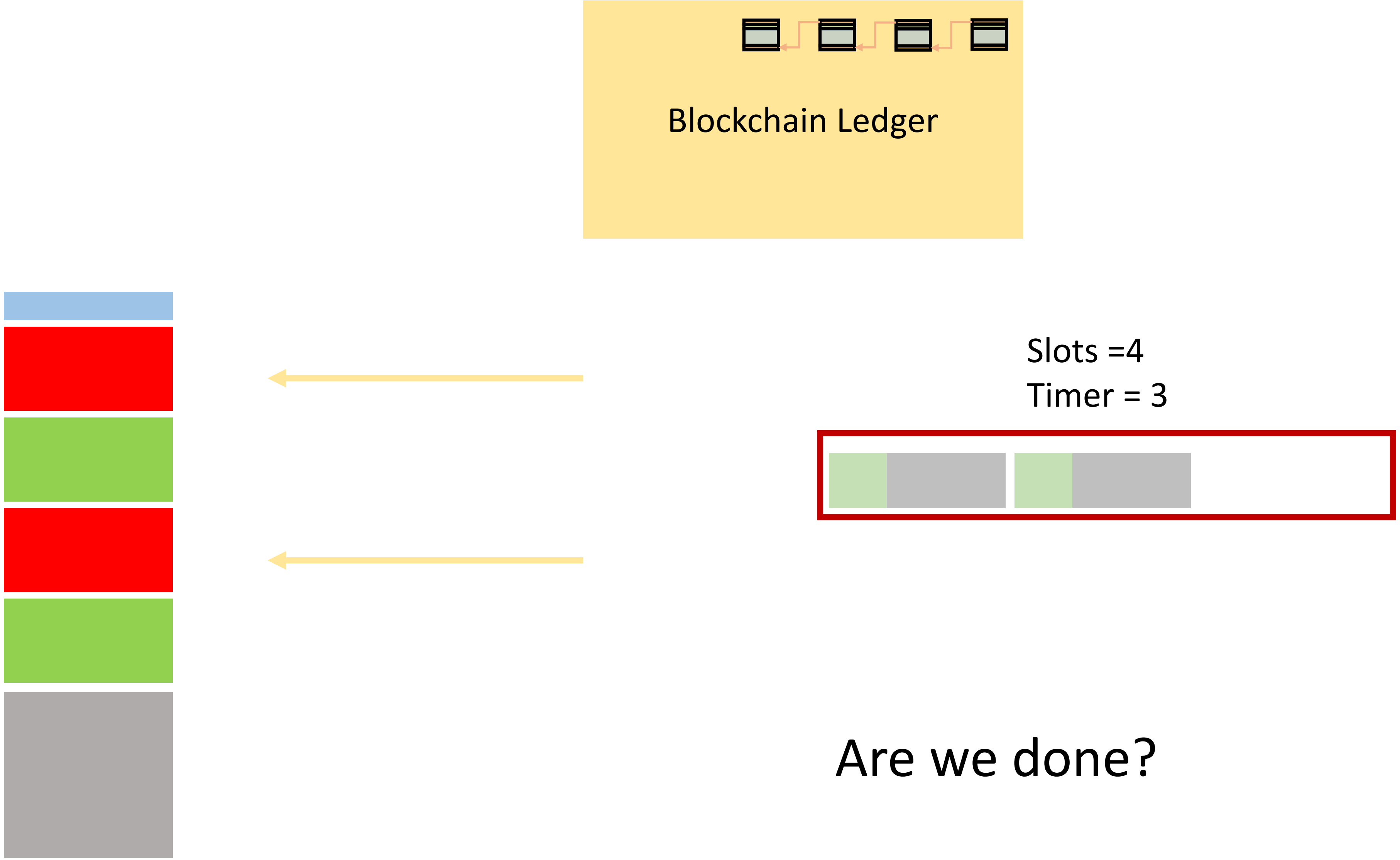
Timer = 3



Blockchain Ledger

Slots =4

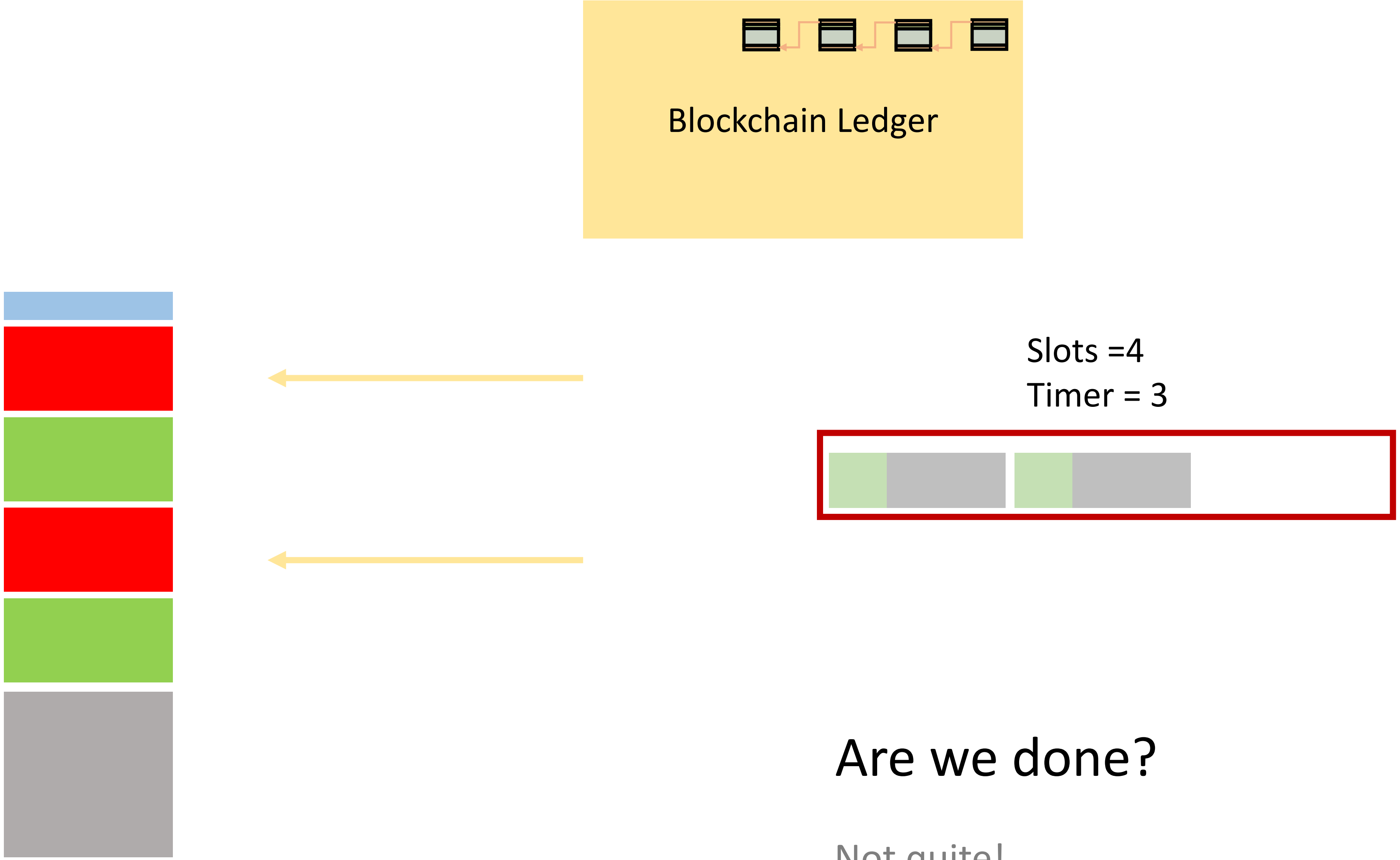
Timer = 3



Blockchain Ledger

Slots =4
Timer = 3

Are we done?



Are we done?

Not quite!

Challenge: Timing Leakage

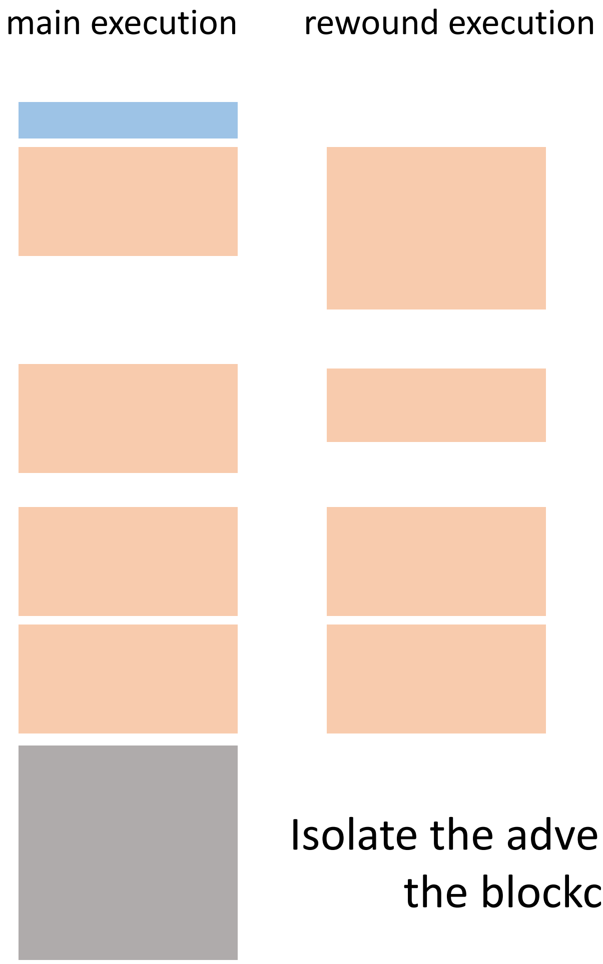
Running time of the simulator larger than running time of adversary.

Challenge: Timing Leakage

Running time of the simulator larger than running time of adversary.

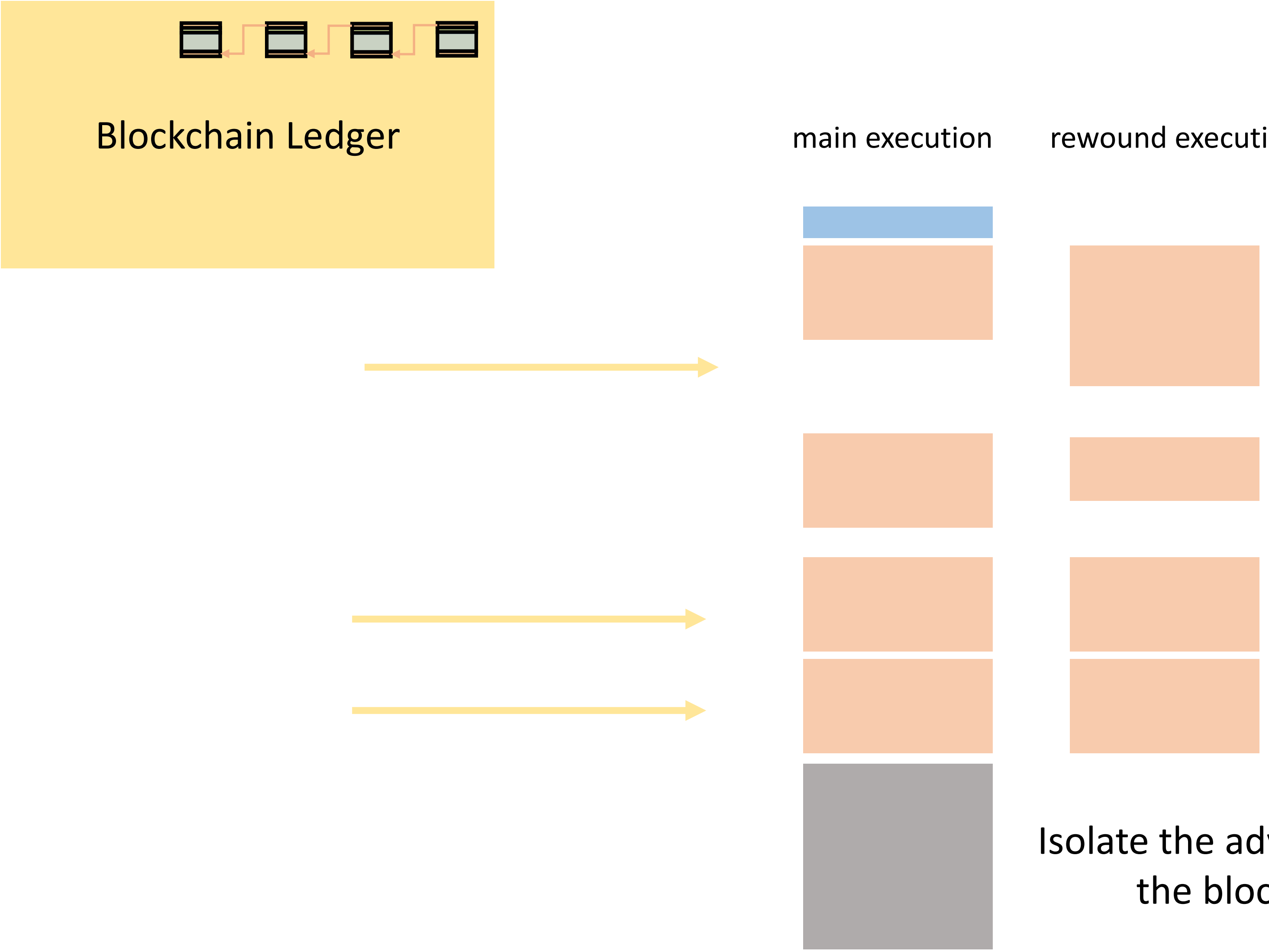
Time that the simulator takes to complete v/s number of computational steps.

Simulate in parallel



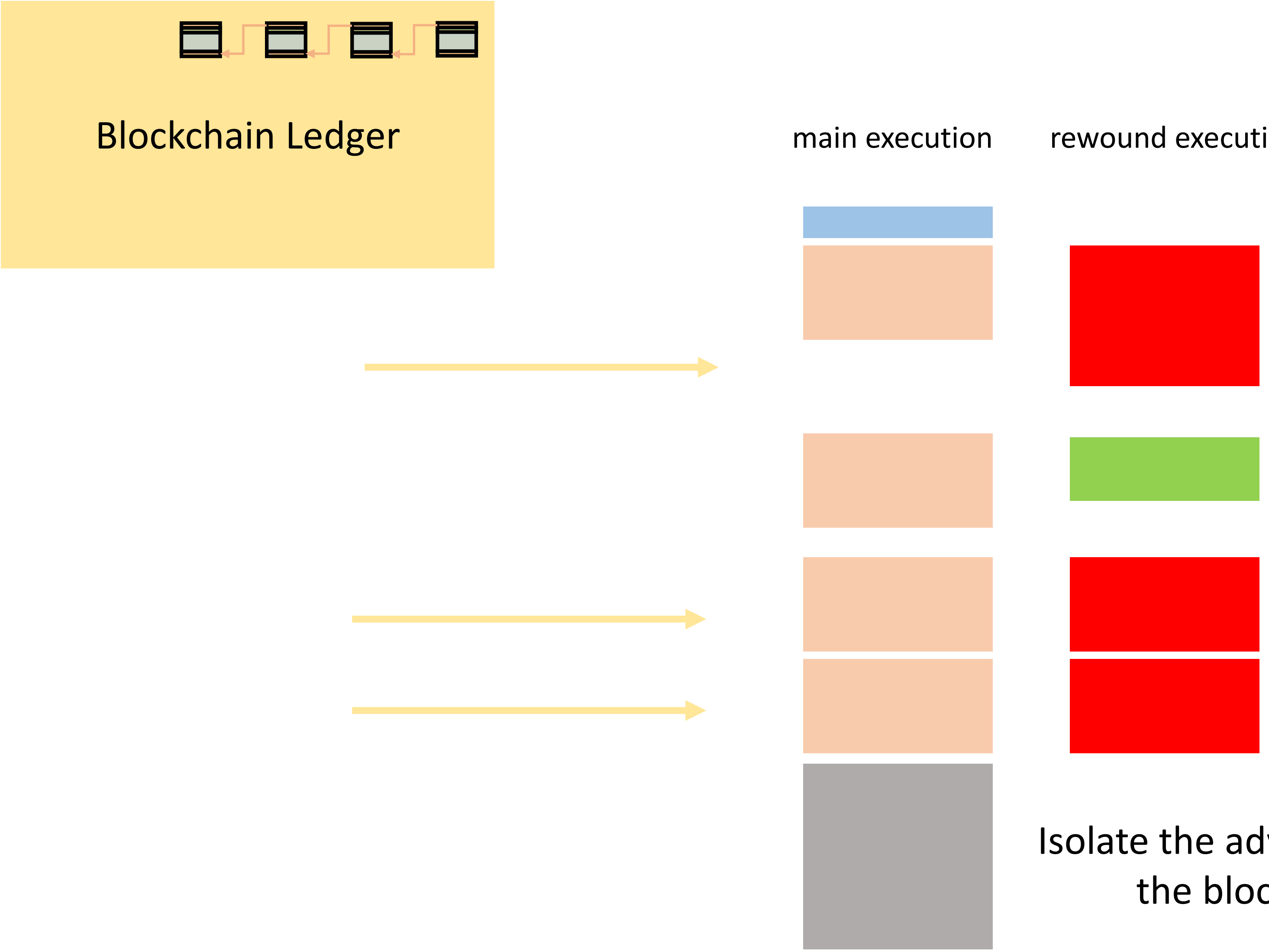
Isolate the adversary from the blockchain

Simulate in parallel



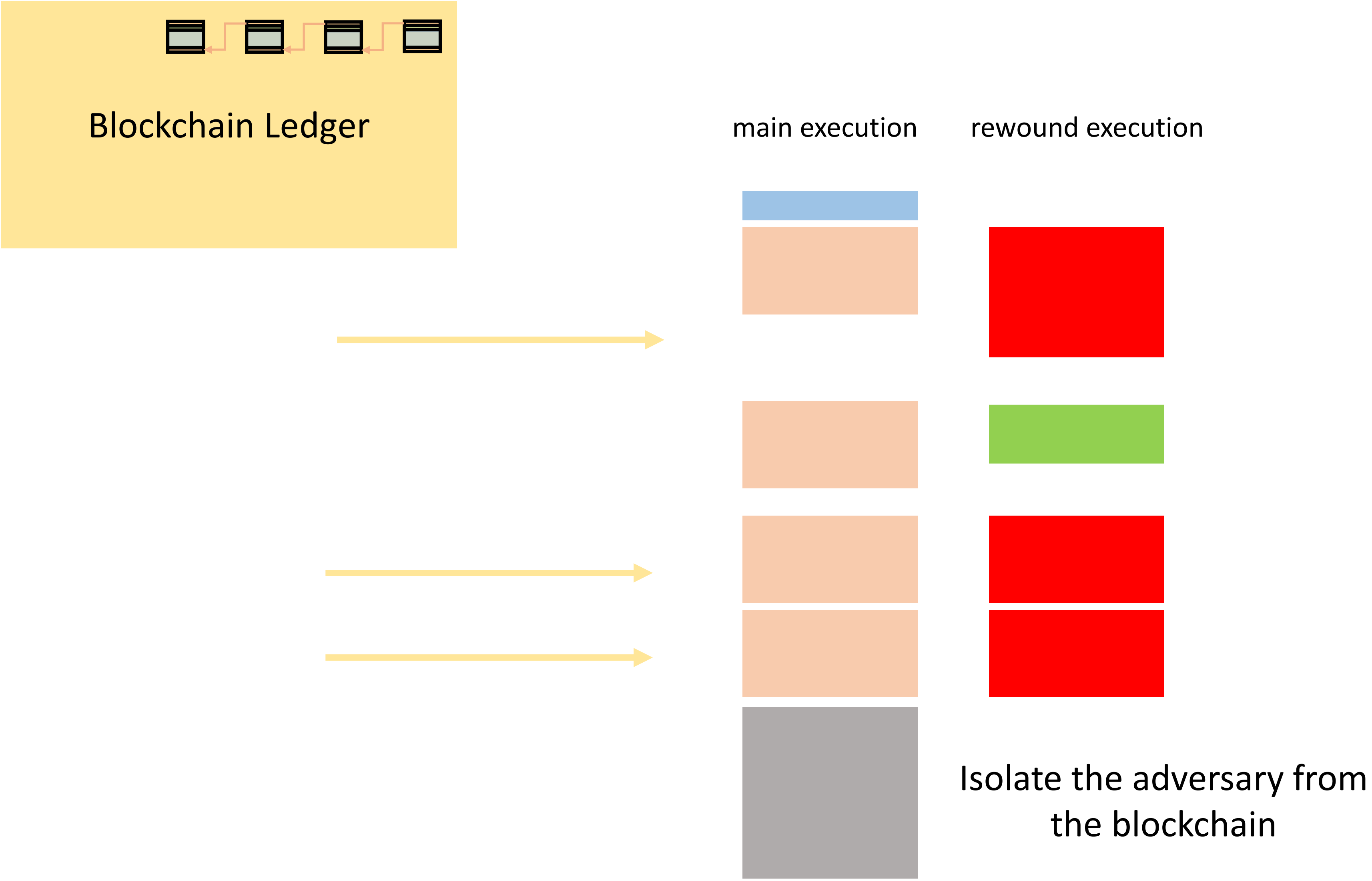
Isolate the adversary from the blockchain

Simulate in parallel



Isolate the adversary from the blockchain

Simulate in parallel



Blockchain Ledger

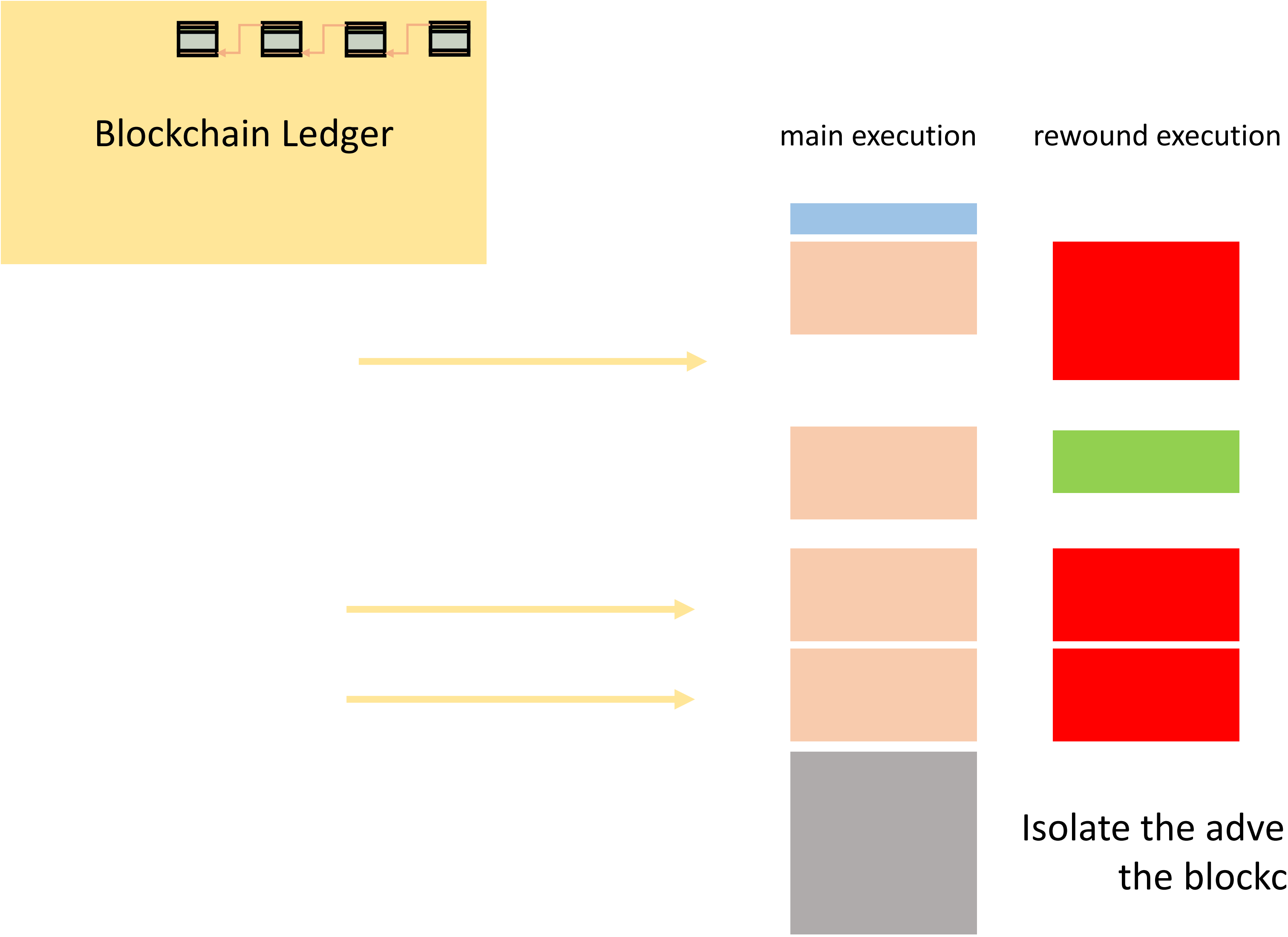
main execution

rewound execution

Isolate the adversary from the blockchain

For extraction to succeed, we need $\omega(1)$ slots.

Simulate in parallel

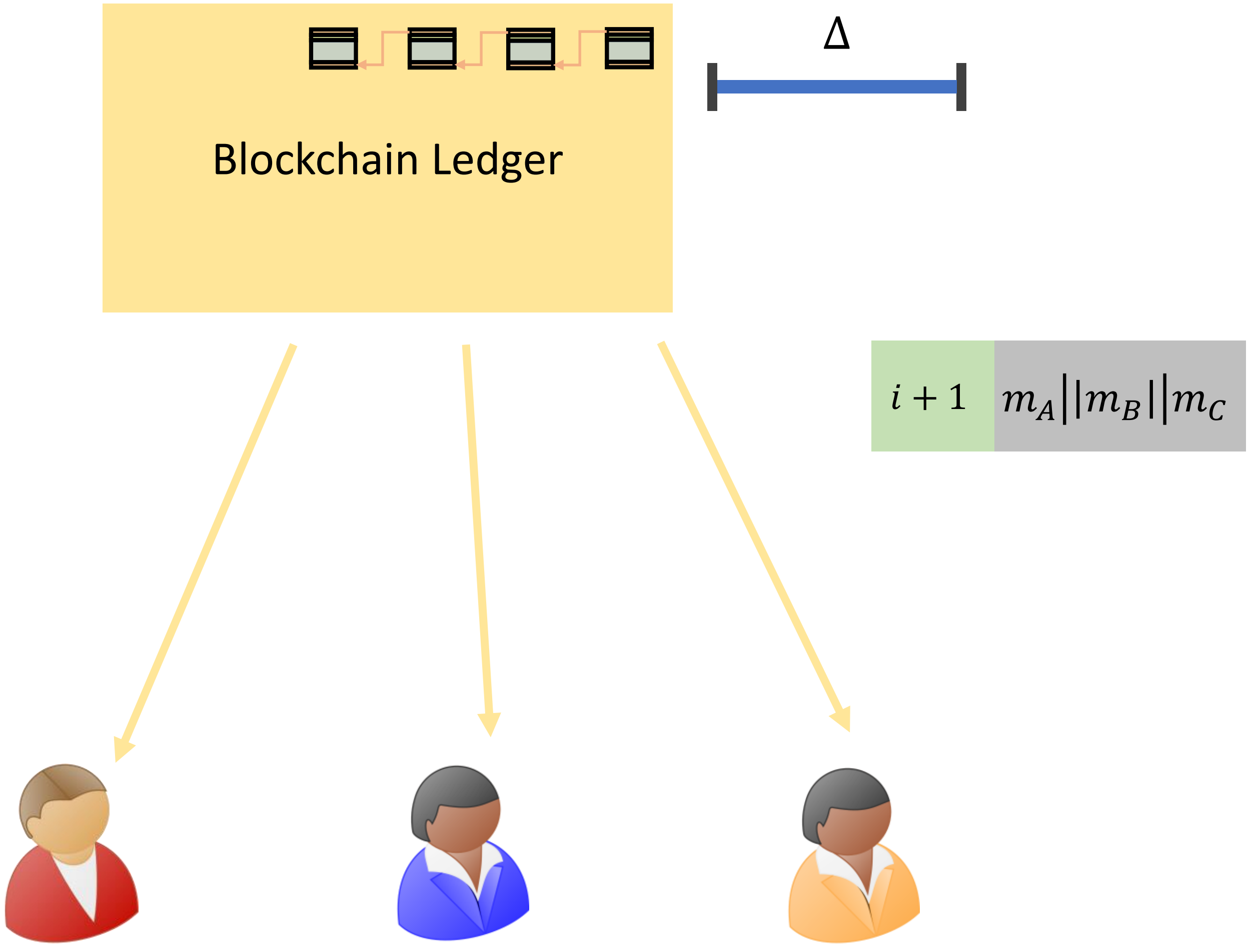


Isolate the adversary from the blockchain

For extraction to succeed, we need $\omega(1)$ slots.

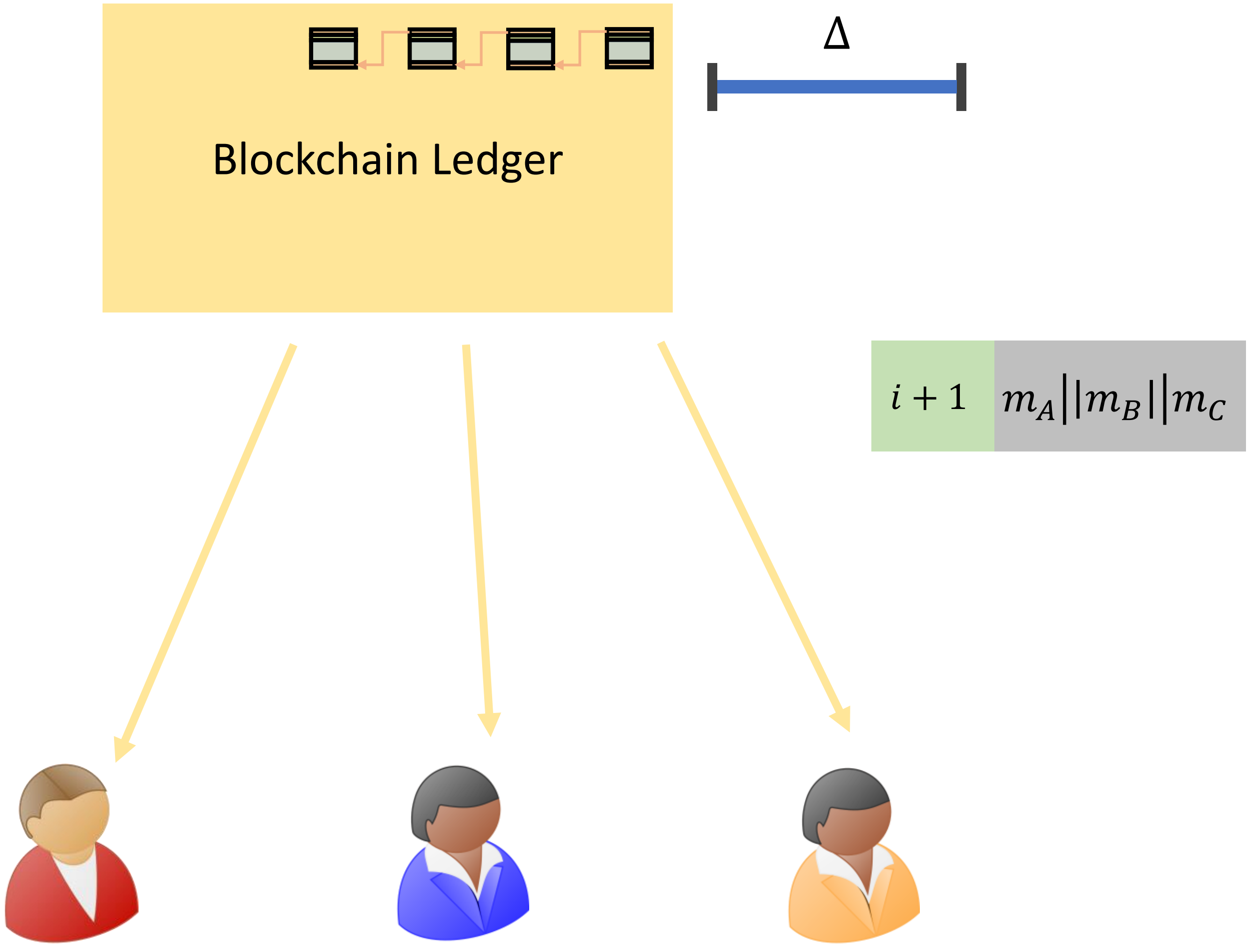
There does not exist an $O(1)$-round ZK argument in the blockchain-hybrid model with black-box simulation.

Simplified Model



Simplified Model

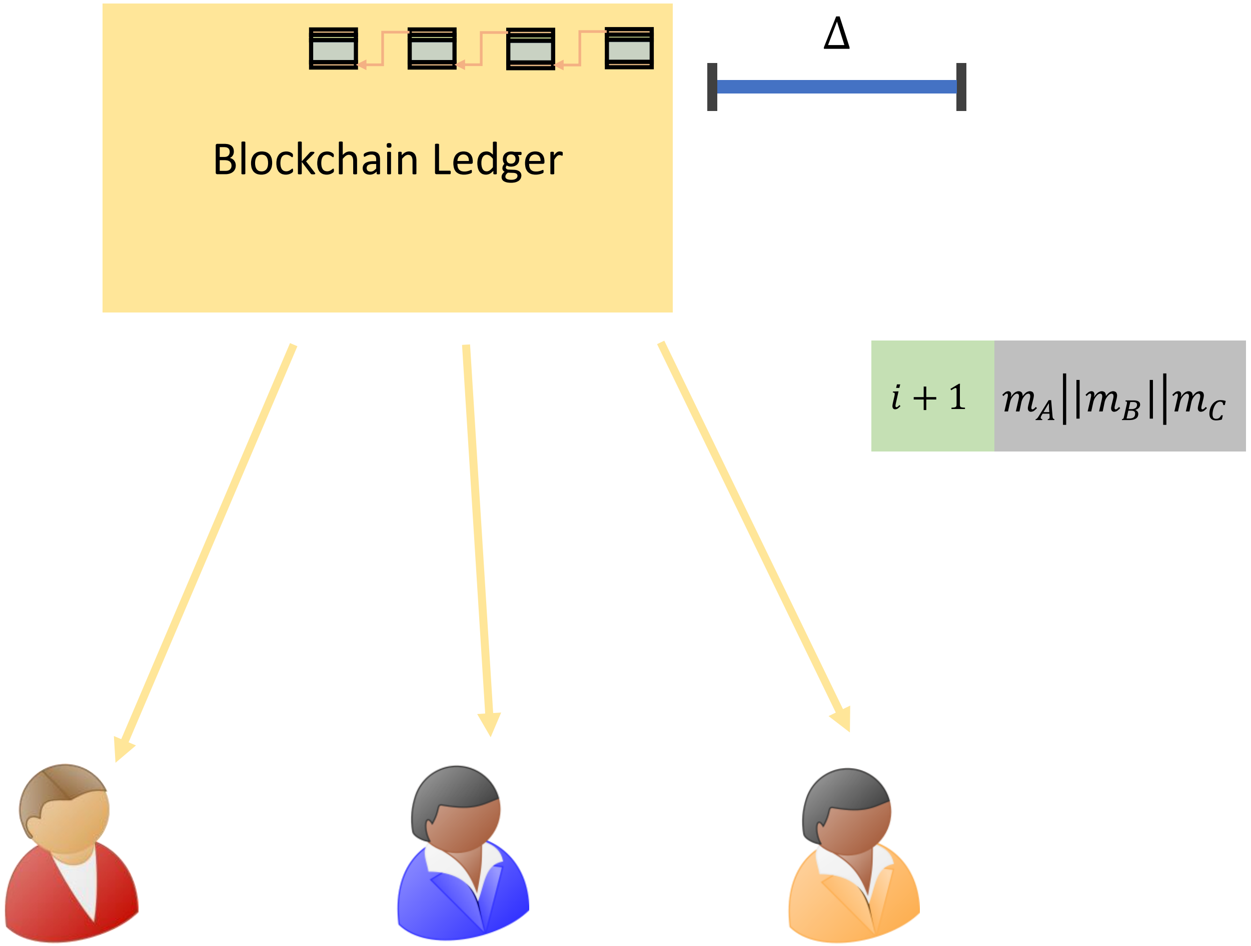
Comparison to Timing Model



Simplified Model

Comparison to Timing Model

Simulator can control the clock.



Blockchain Ledger

Δ

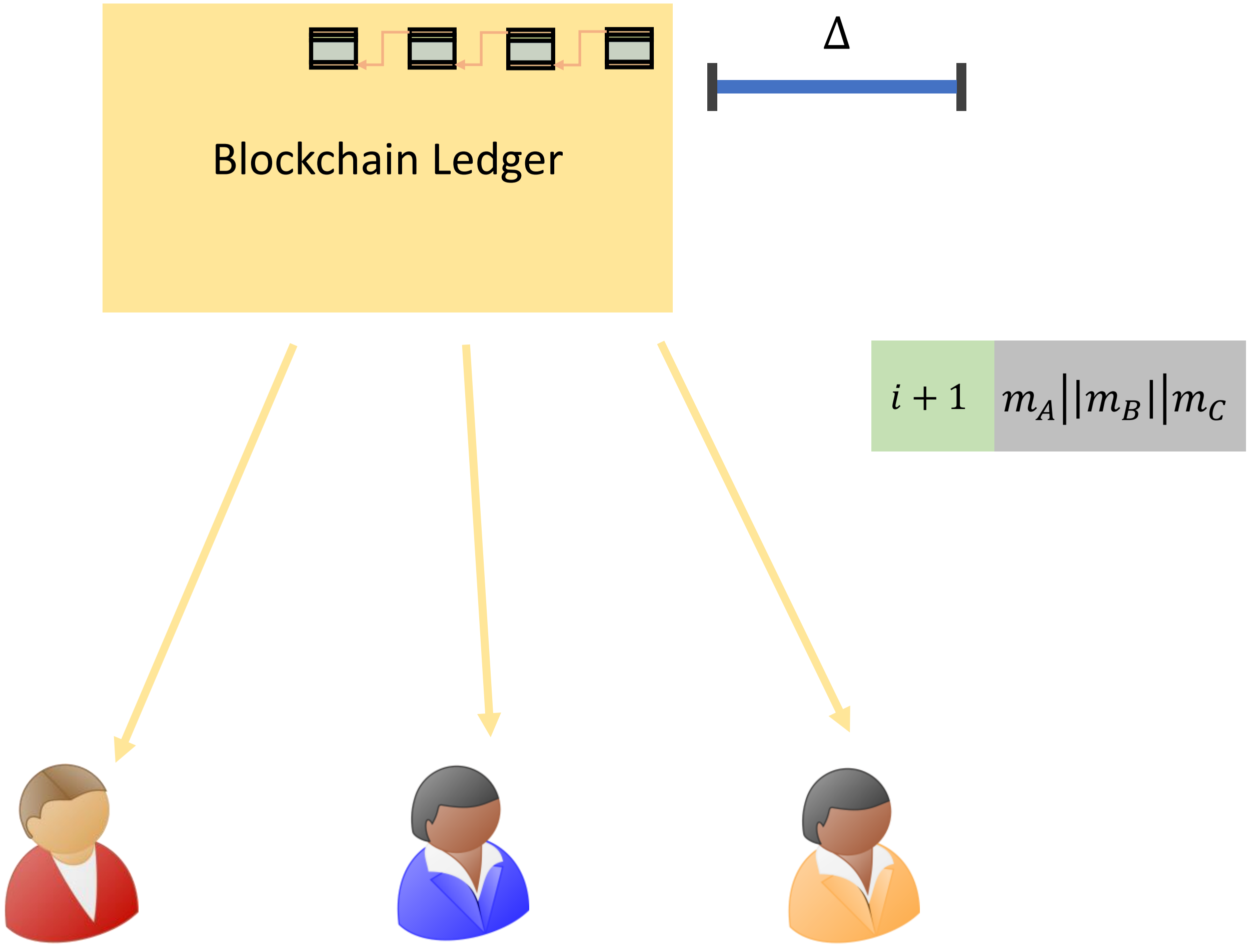
$i+1$ $m_A||m_B||m_C$

Simplified Model

Comparison to Timing Model

Simulator can control the clock.

unforgeable clock



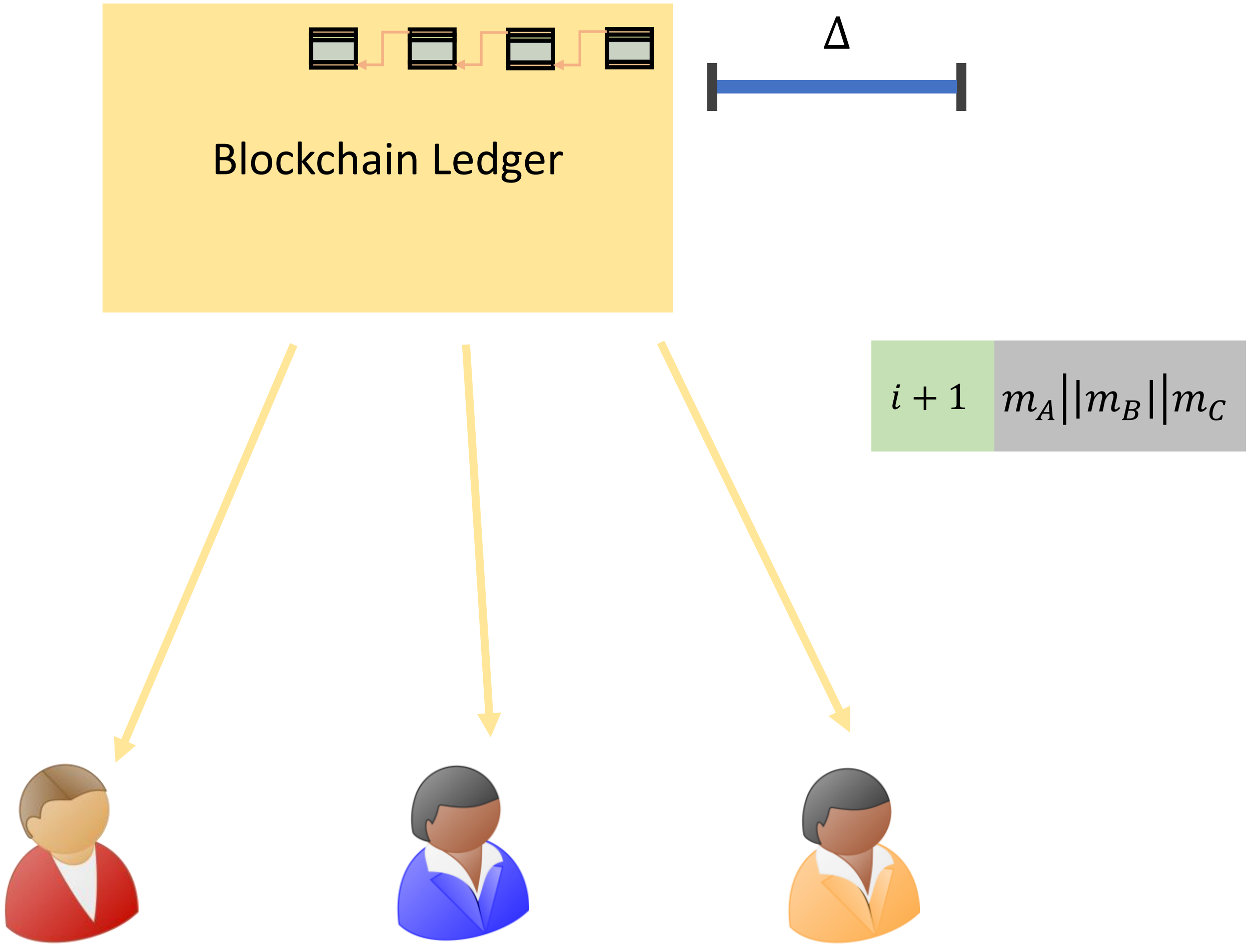
Simplified Model

Comparison to Timing Model

Simulator can control the clock.

unforgeable clock

Adversary can be rewound at any point.



Simplified Model

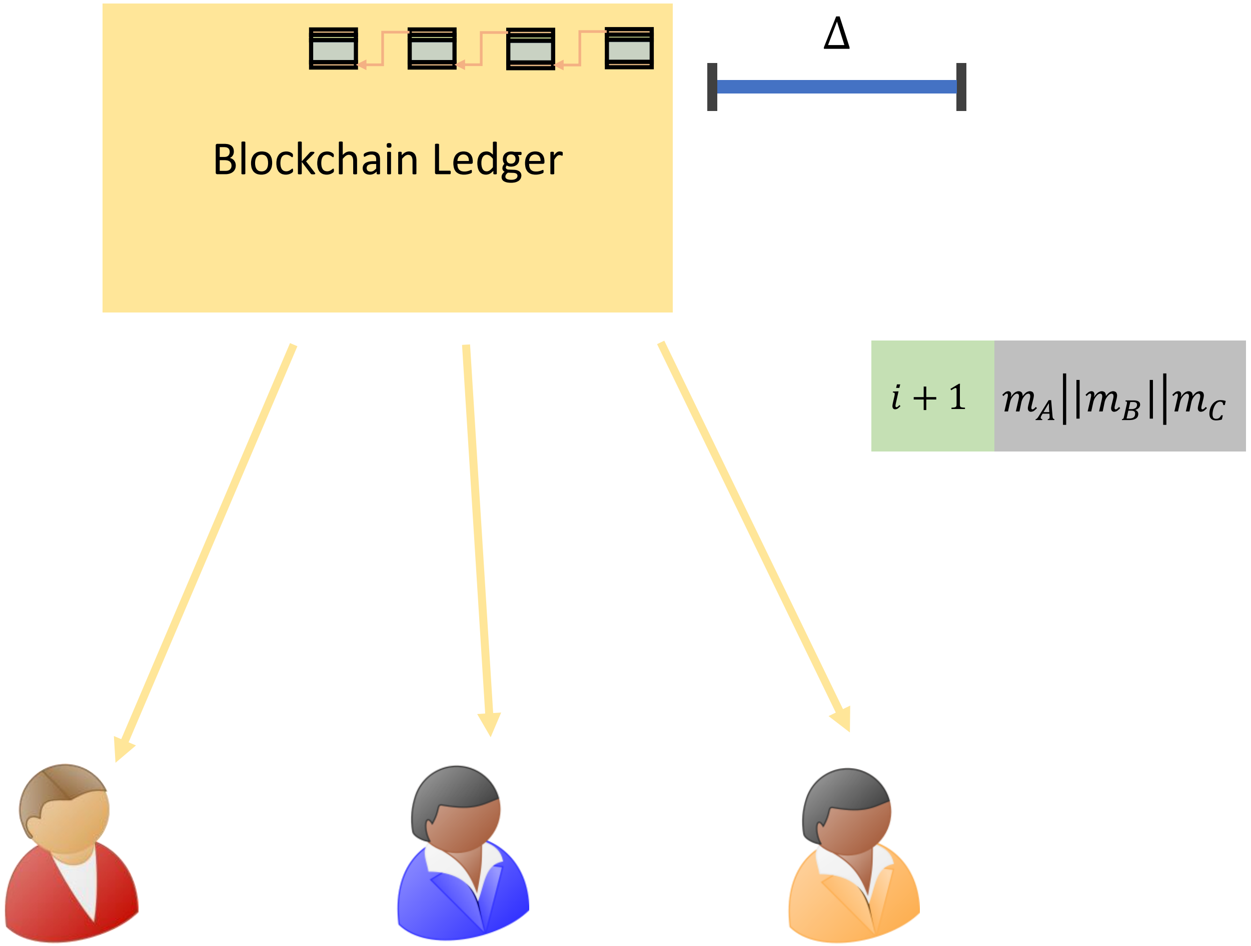
Comparison to Timing Model

Simulator can control the clock.

unforgeable clock

Adversary can be rewound at any point.

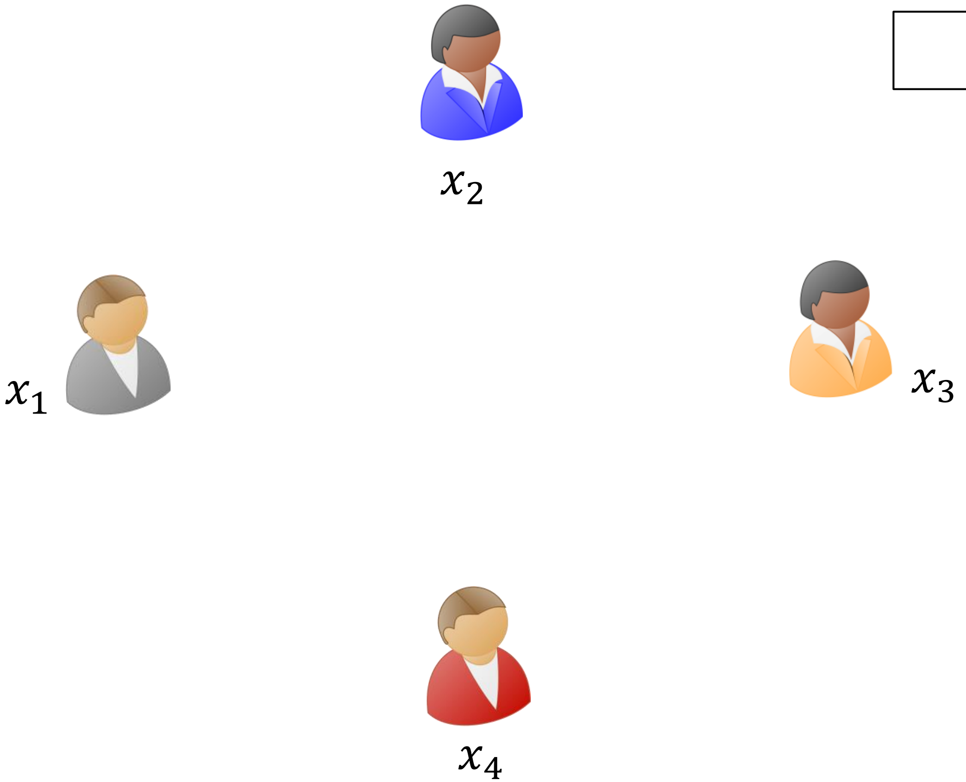
new rewinding techniques



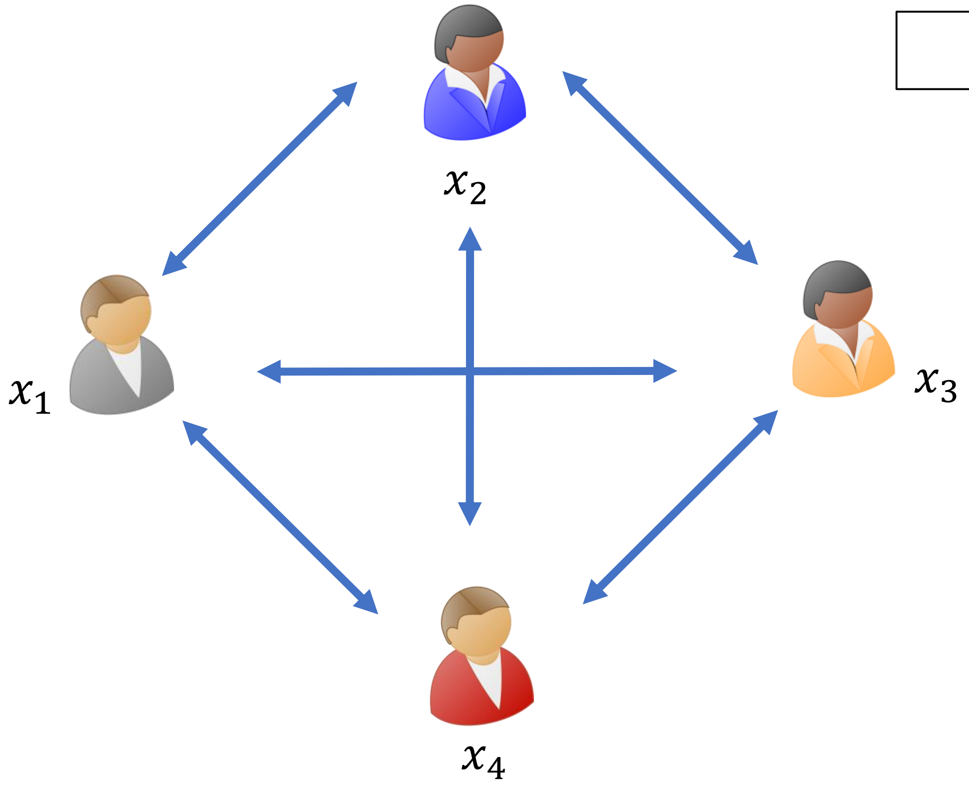
Concurrent Self Composition

Secure Computation

$$\boxed{y = f(x_1, x_2, x_3, x_4)}$$



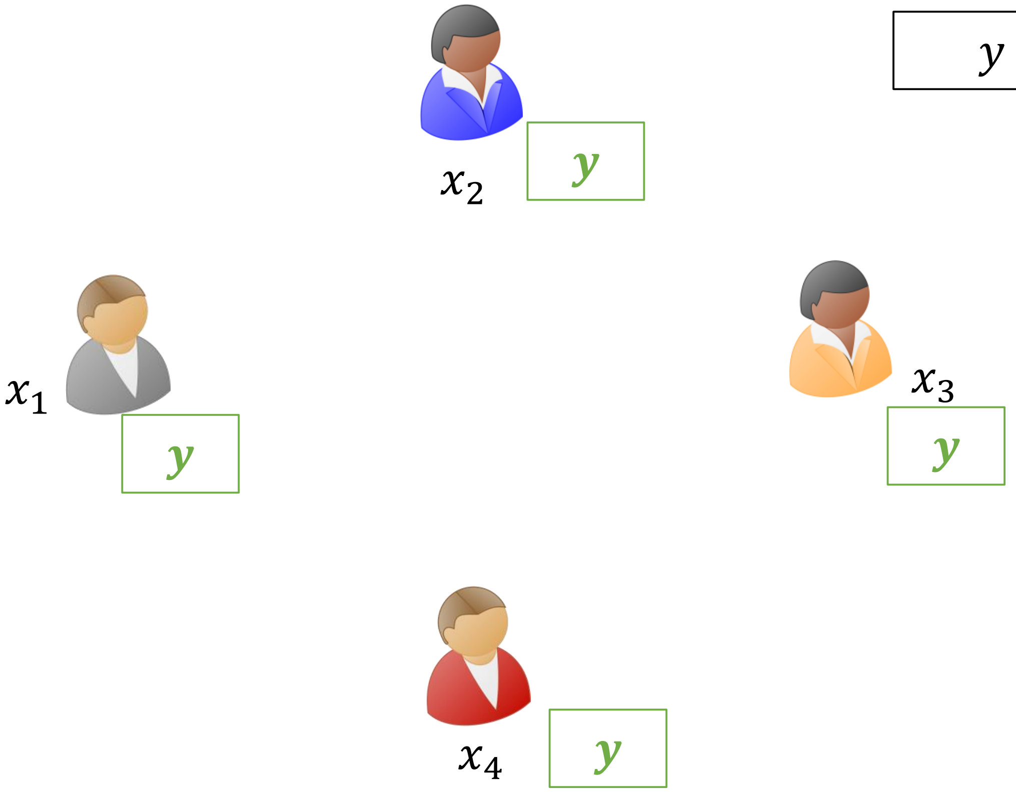
Secure Computation



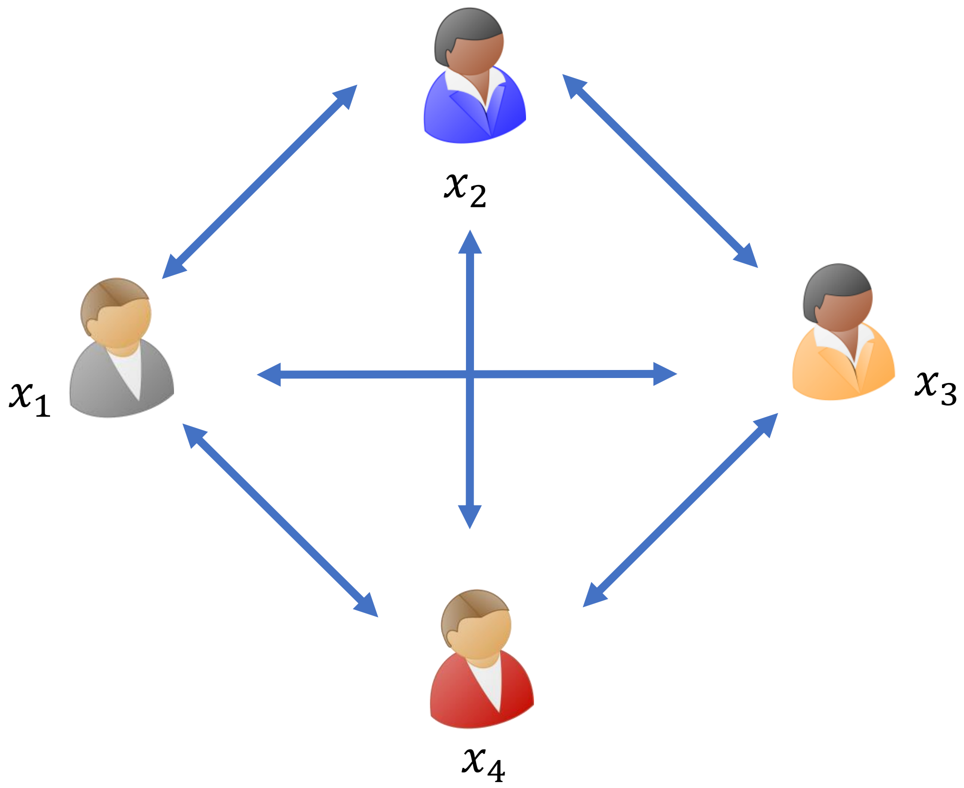
$$\boxed{y = f(x_1, x_2, x_3, x_4)}$$

Secure Computation

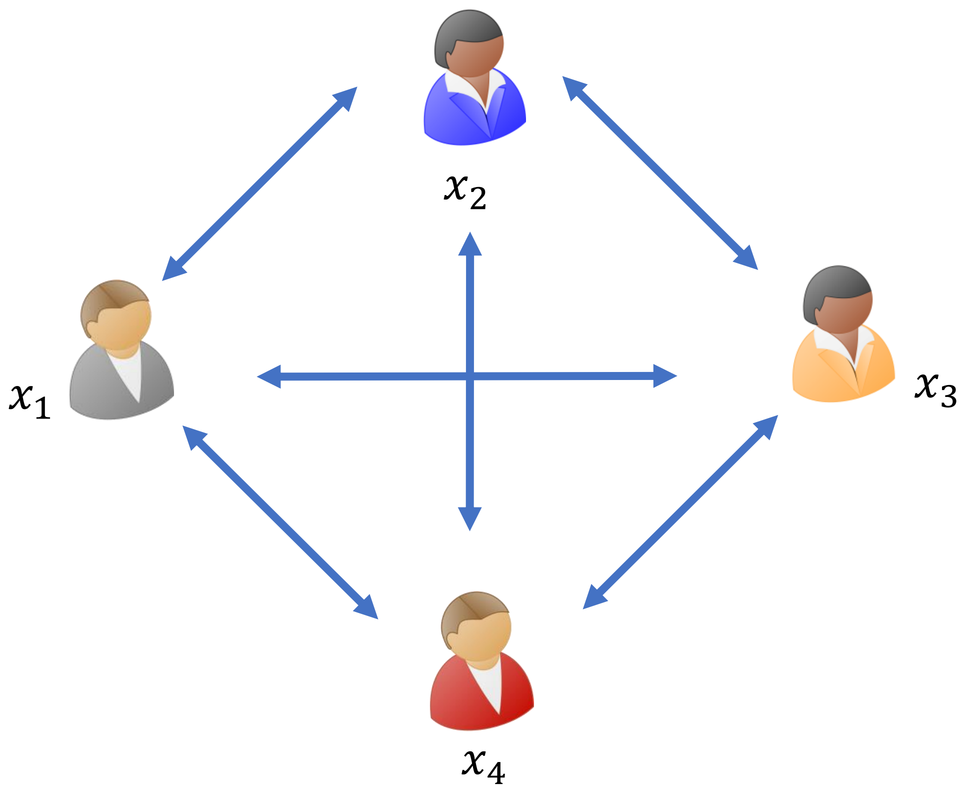
$$\boxed{y = f(x_1, x_2, x_3, x_4)}$$

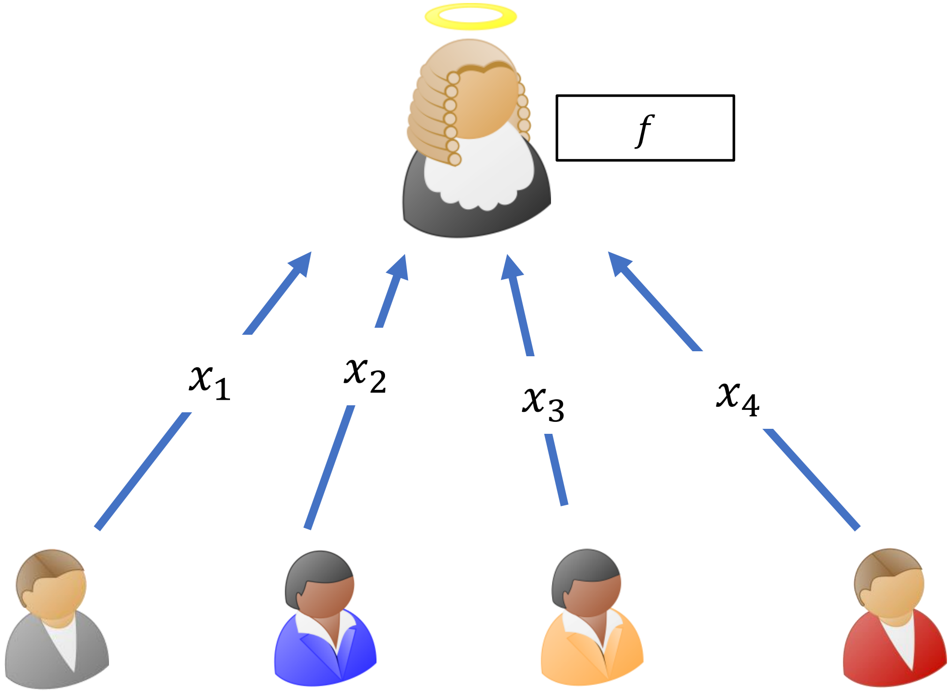


Security

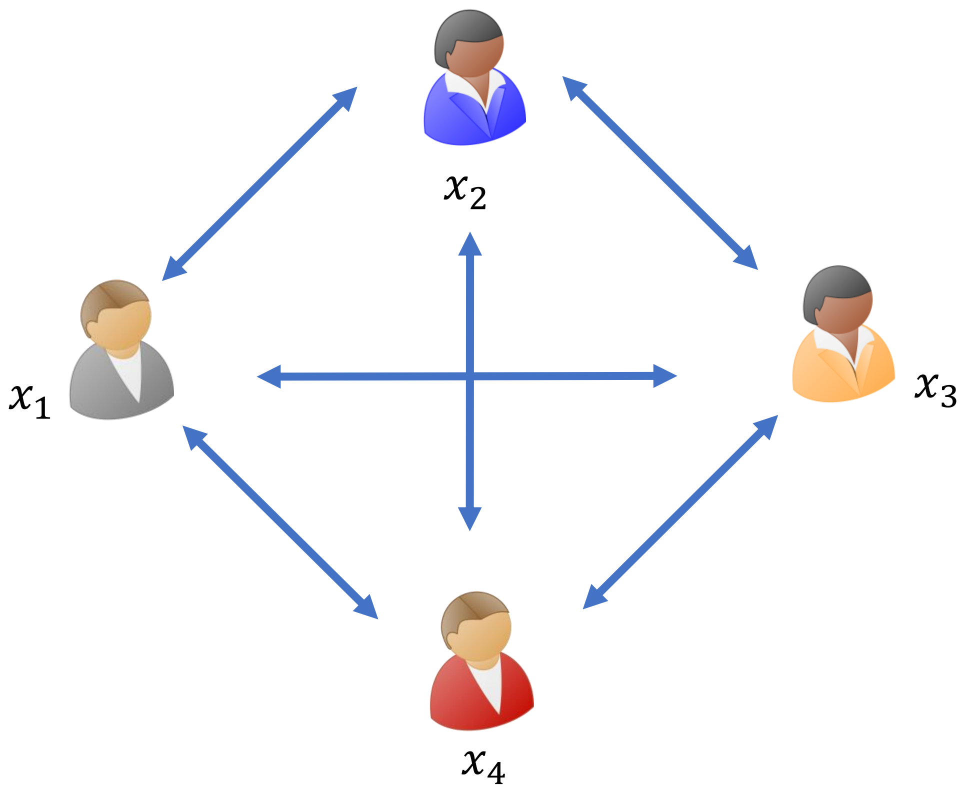


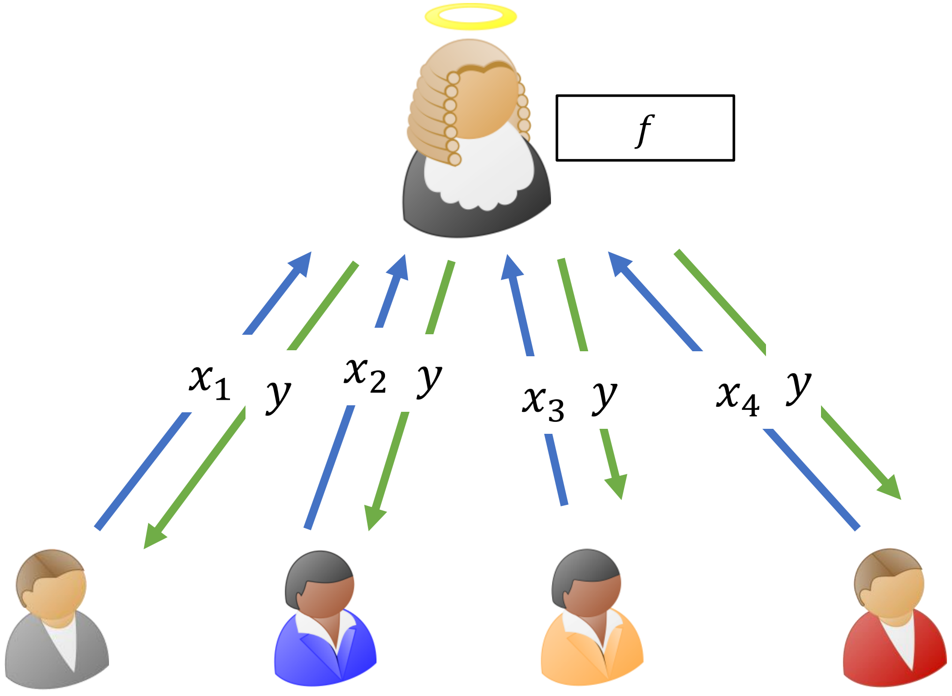
Security



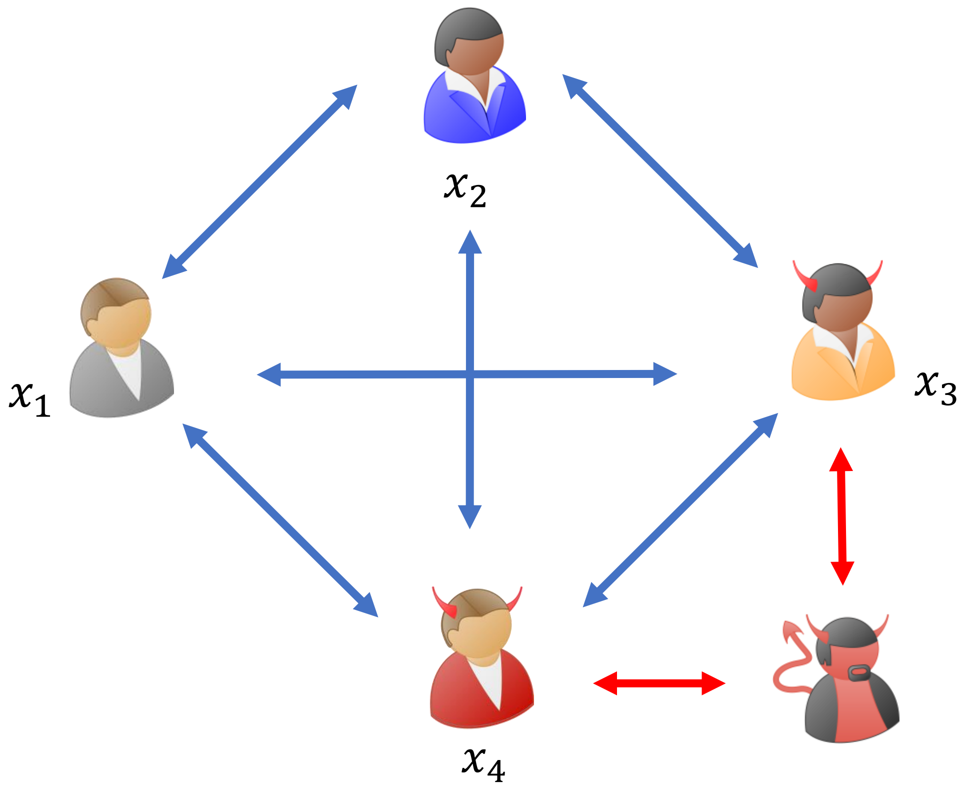


Security

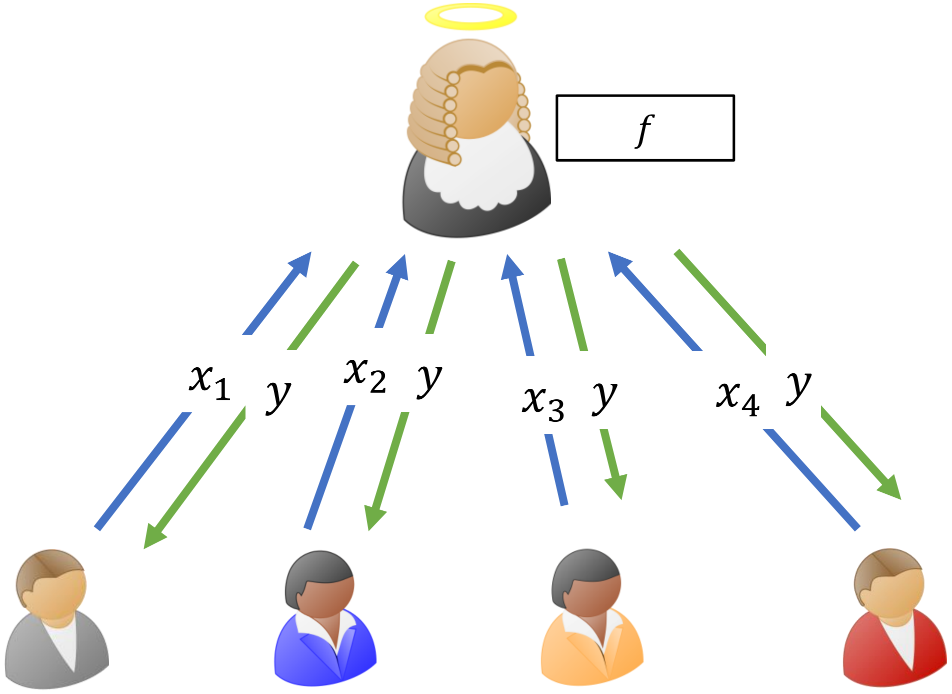




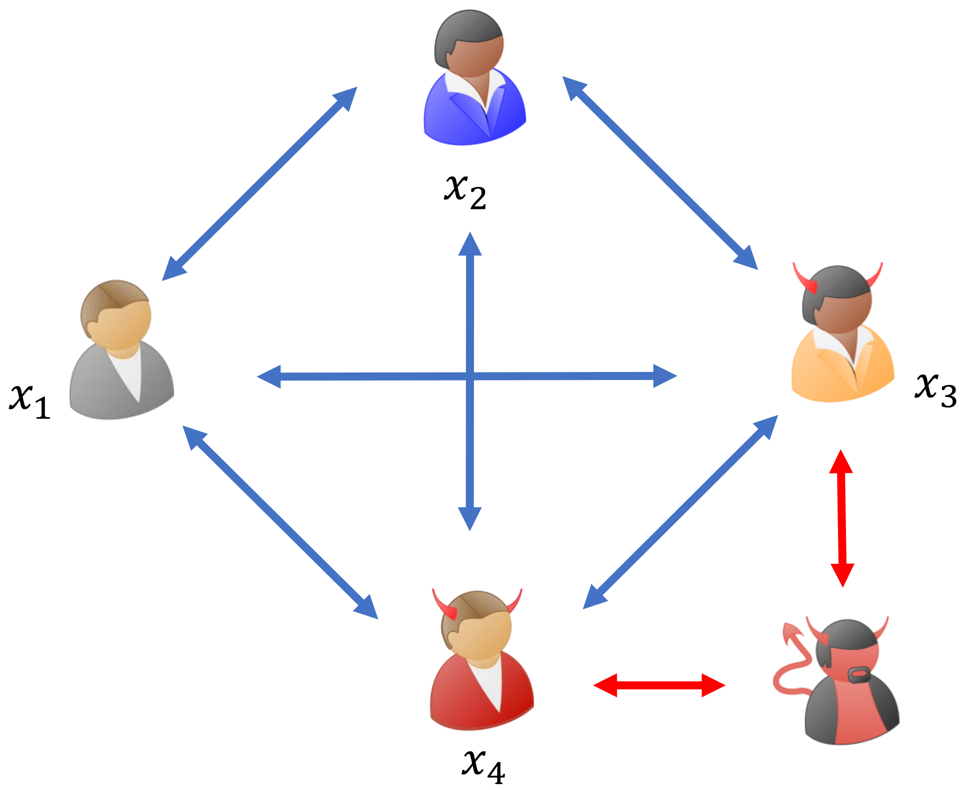
Security

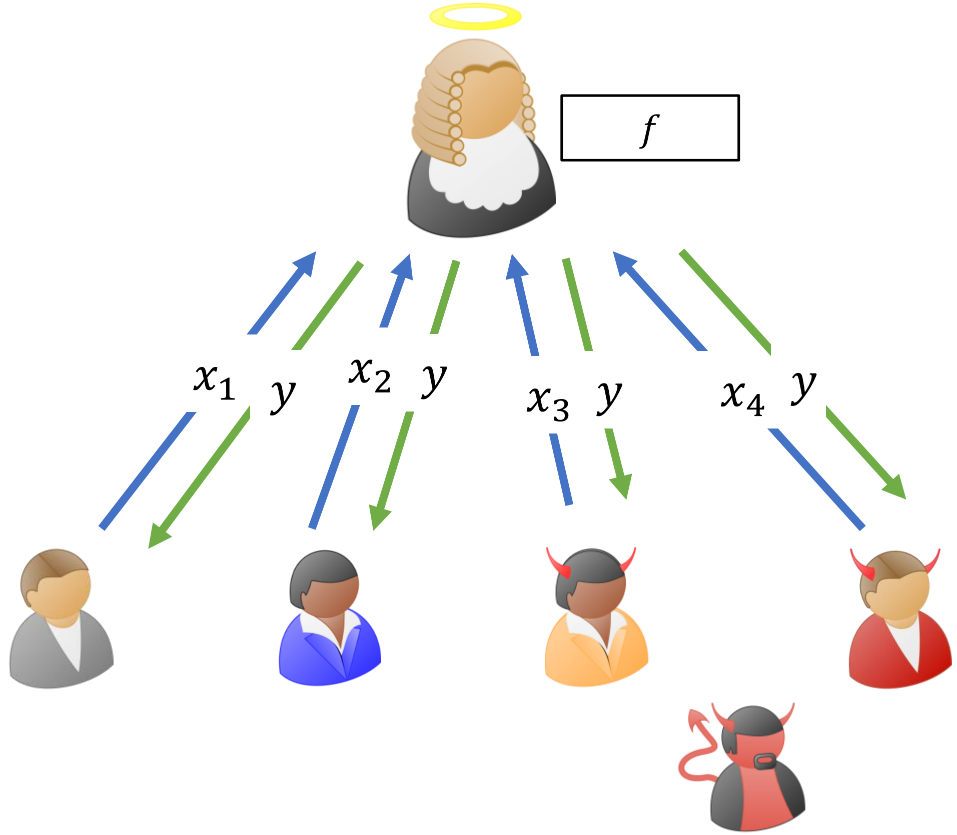


Real World



Security

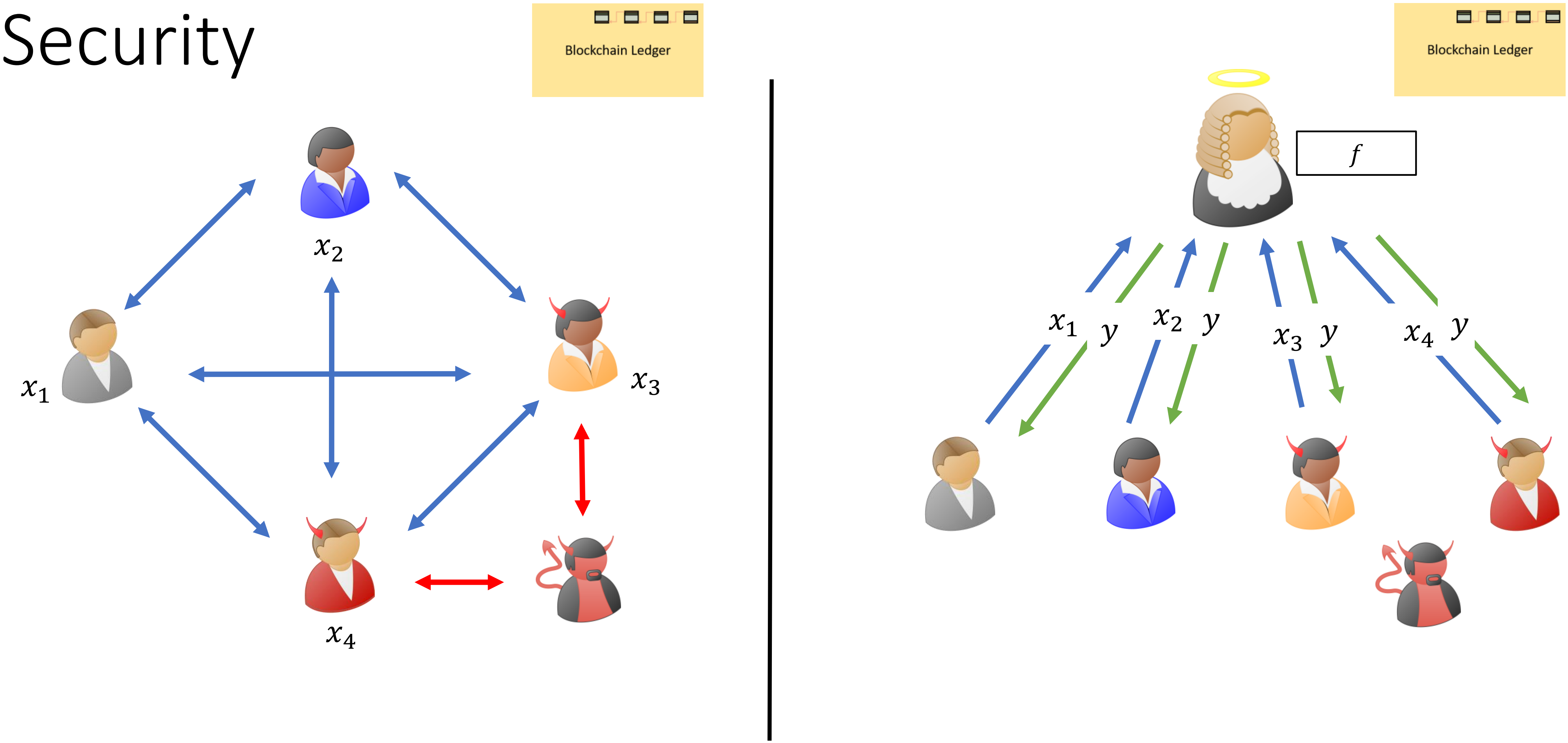




Real World

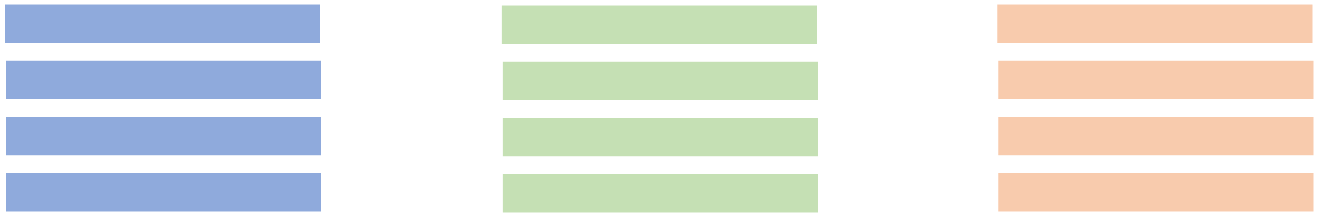
Ideal World

Security



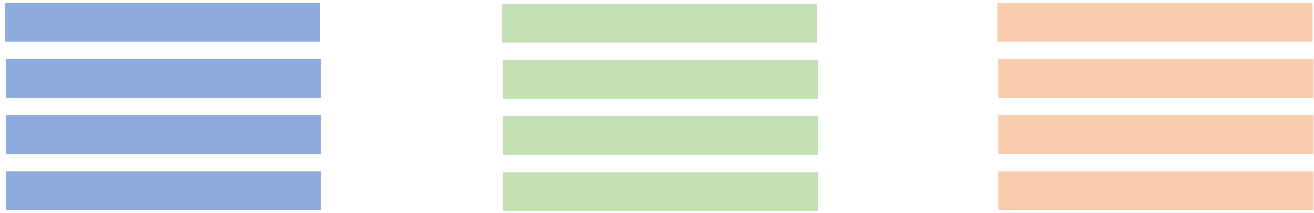
Real World ≈ Ideal World

Concurrent Secure Computation



Protocol transcripts

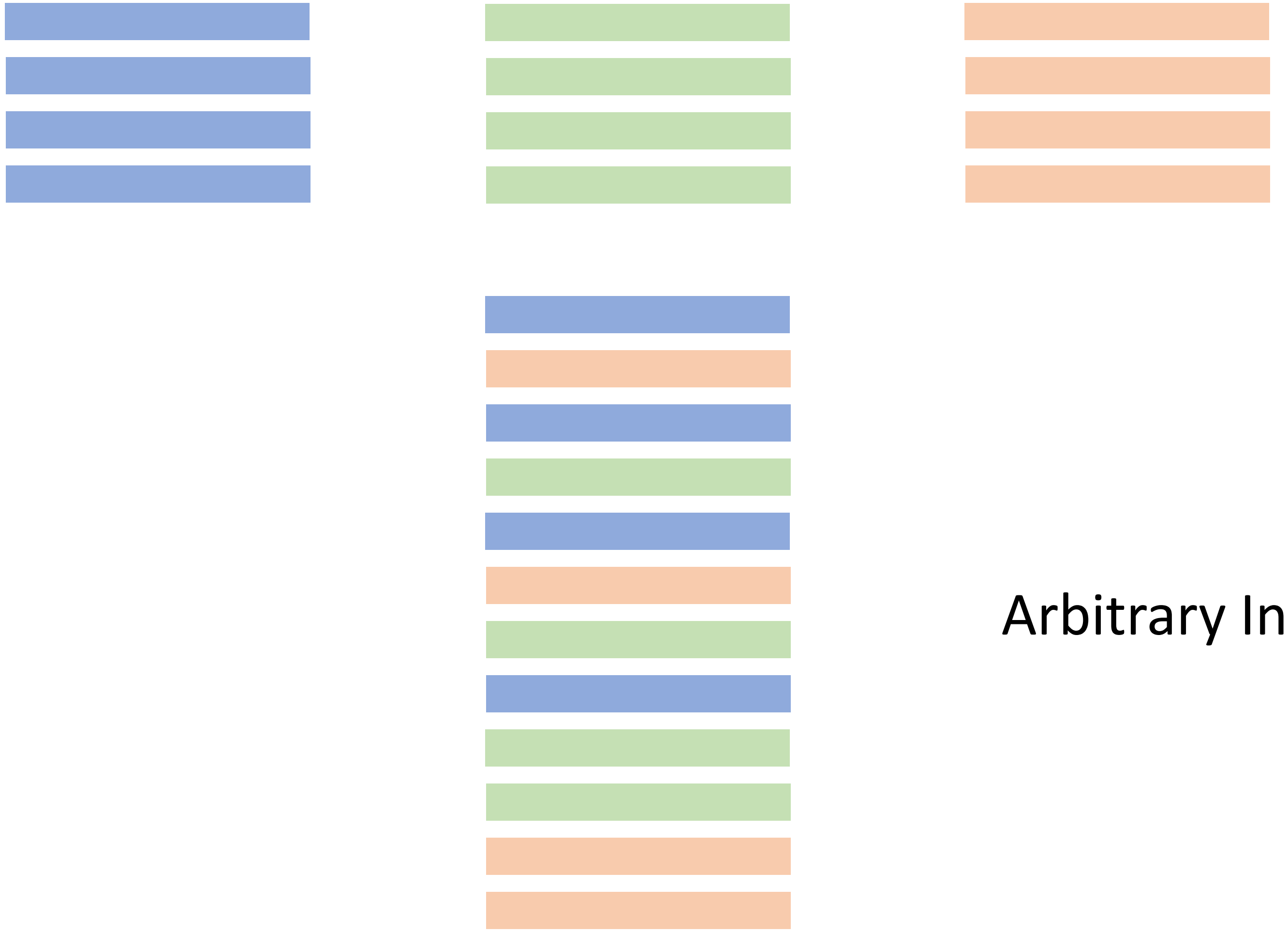
Concurrent Secure Computation



Protocol transcripts

Self Composition

Concurrent Secure Computation

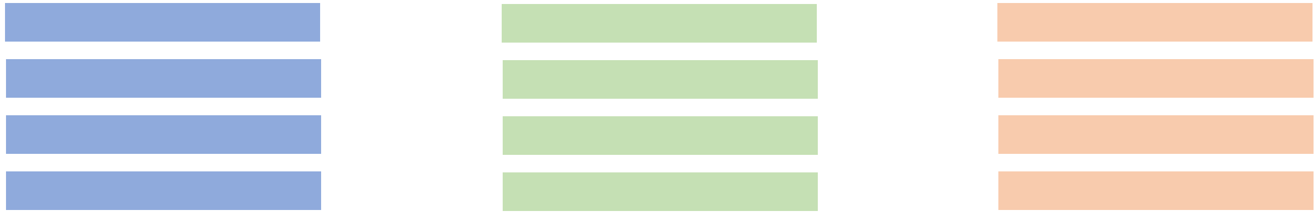


Protocol transcripts

Self Composition

Arbitrary Interleaving

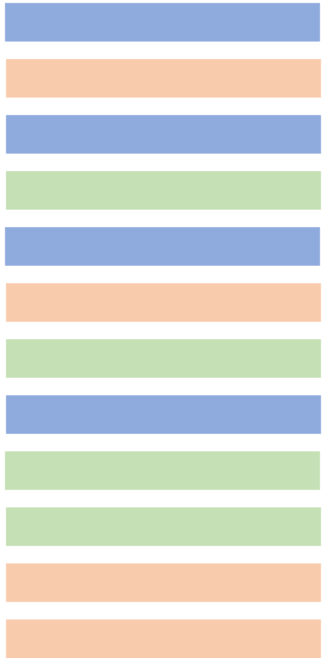
Concurrent Secure Computation



Protocol transcripts

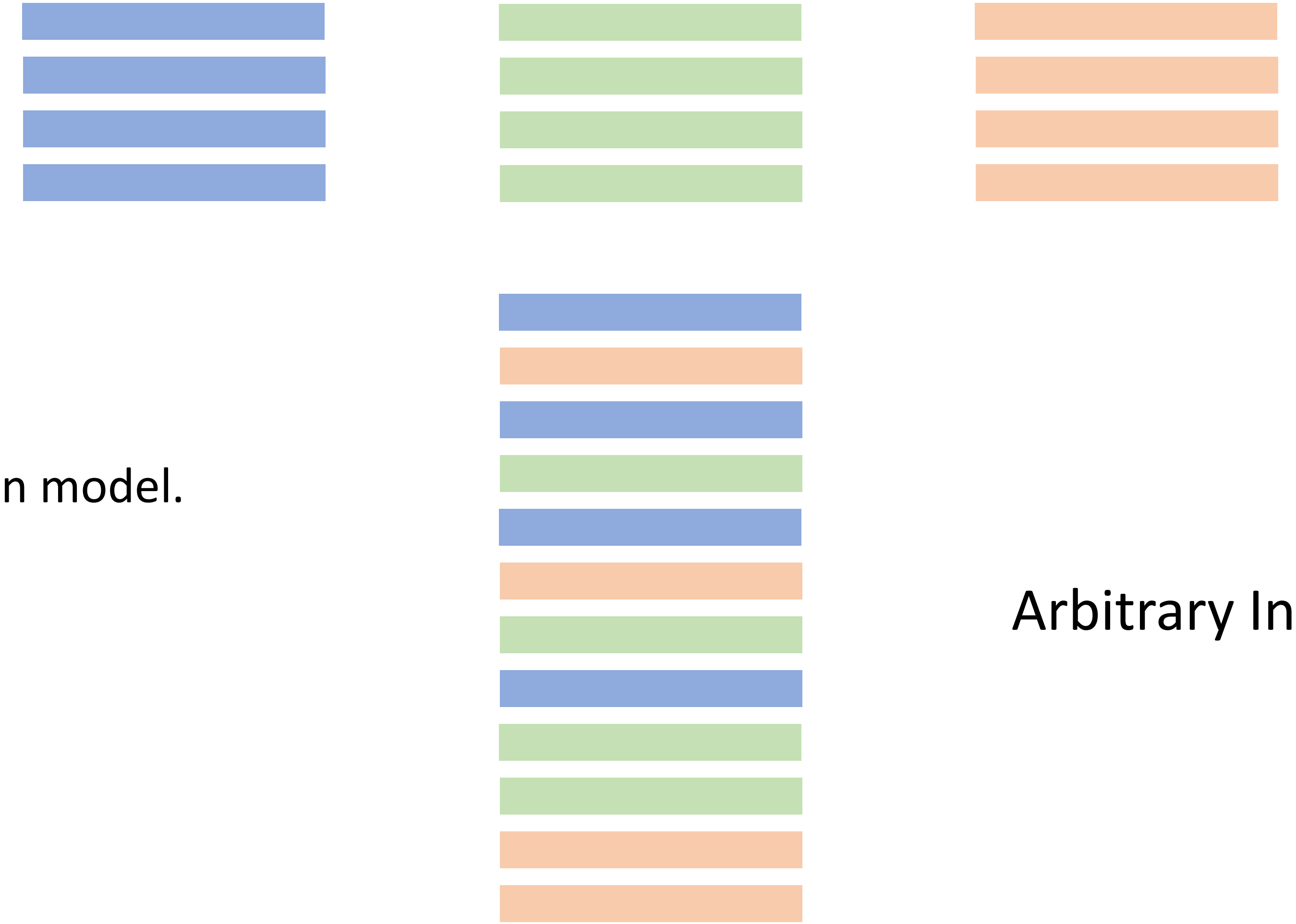
Self Composition

Impossible in the plain model.



Arbitrary Interleaving

Concurrent Secure Computation



Protocol transcripts

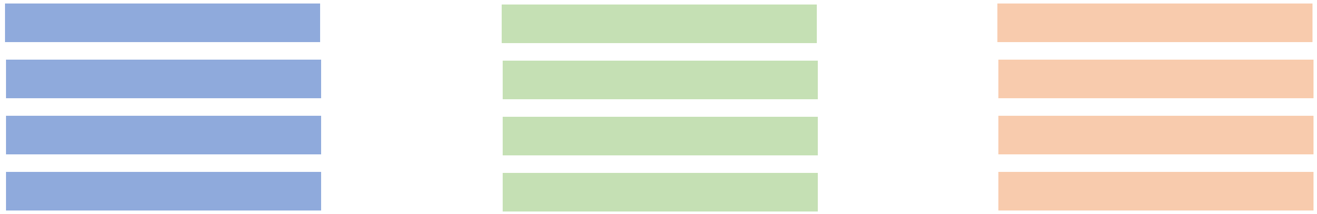
Self Composition

Impossible in the plain model.

Prior work:

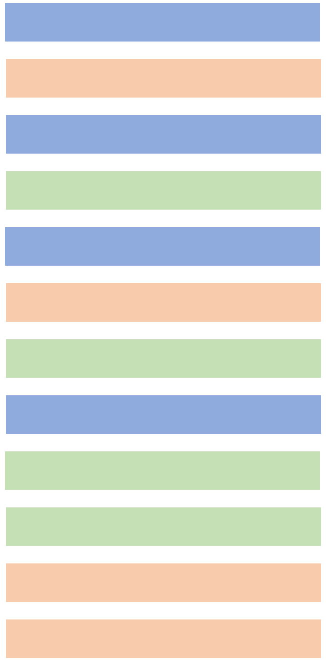
Arbitrary Interleaving

Concurrent Secure Computation



Protocol transcripts

Self Composition



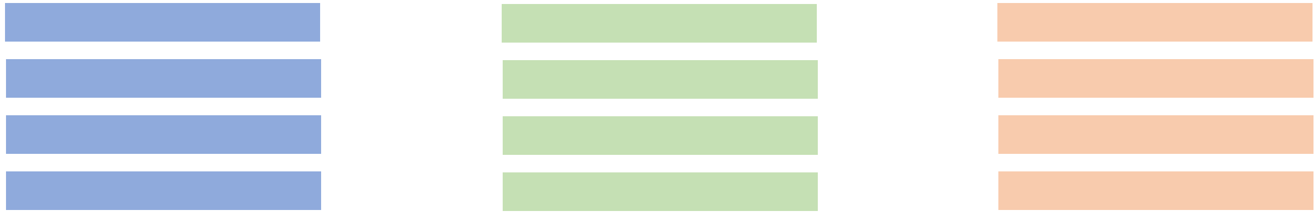
Impossible in the plain model.

Prior work:

Weaker security notion

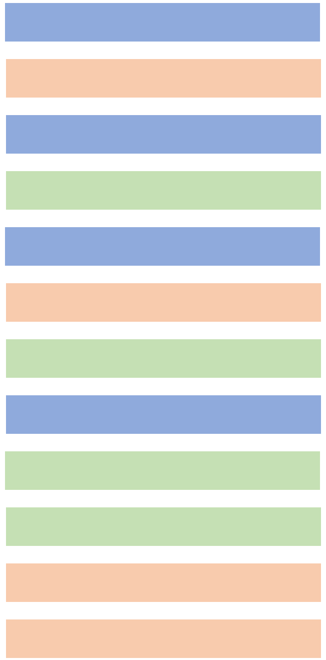
Arbitrary Interleaving

Concurrent Secure Computation



Protocol transcripts

Self Composition



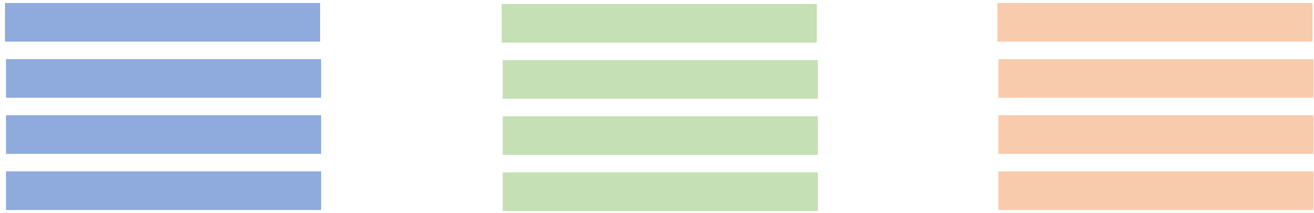
Arbitrary Interleaving

Impossible in the plain model.

Prior work:

- Weaker security notion
- Trust Assumptions

Concurrent Secure Computation



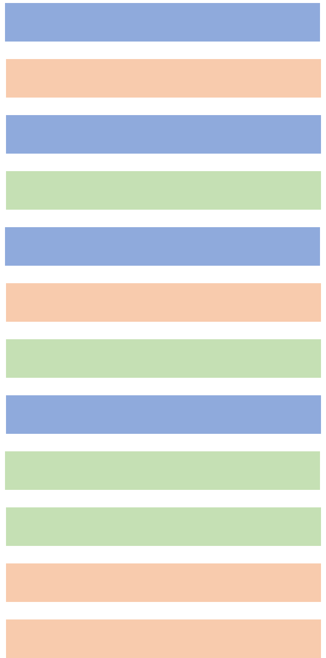
Protocol transcripts

Self Composition

Impossible in the plain model.

Prior work:

- Weaker security notion
- Trust Assumptions



Arbitrary Interleaving

Decentralized trust assumption

Concurrent Secure Computation Impossible

Concurrent Secure Computation Impossible

Black-box impossible

[Lindell 04]

Concurrent Secure Computation Impossible

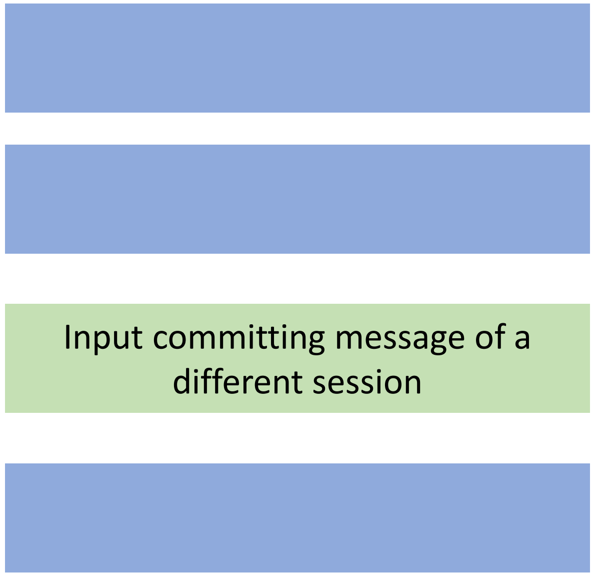
Black-box impossible

[Lindell 04]

Non-black box, and varied settings:

[Barak-Prabhakaran-Sahai 06]
[Goyal 12]
[Agrawal-Goyal-Jain-Prabhakaran-Sahai 12]
[Garg-Kumarasubramanian-Ostrovsky-Visconti 12]

Concurrent Secure Computation Impossible



Input committing message of a different session

Black-box impossible

[Lindell 04]

Non-black box, and varied settings:

[Barak-Prabhakaran-Sahai 06]

[Goyal 12]

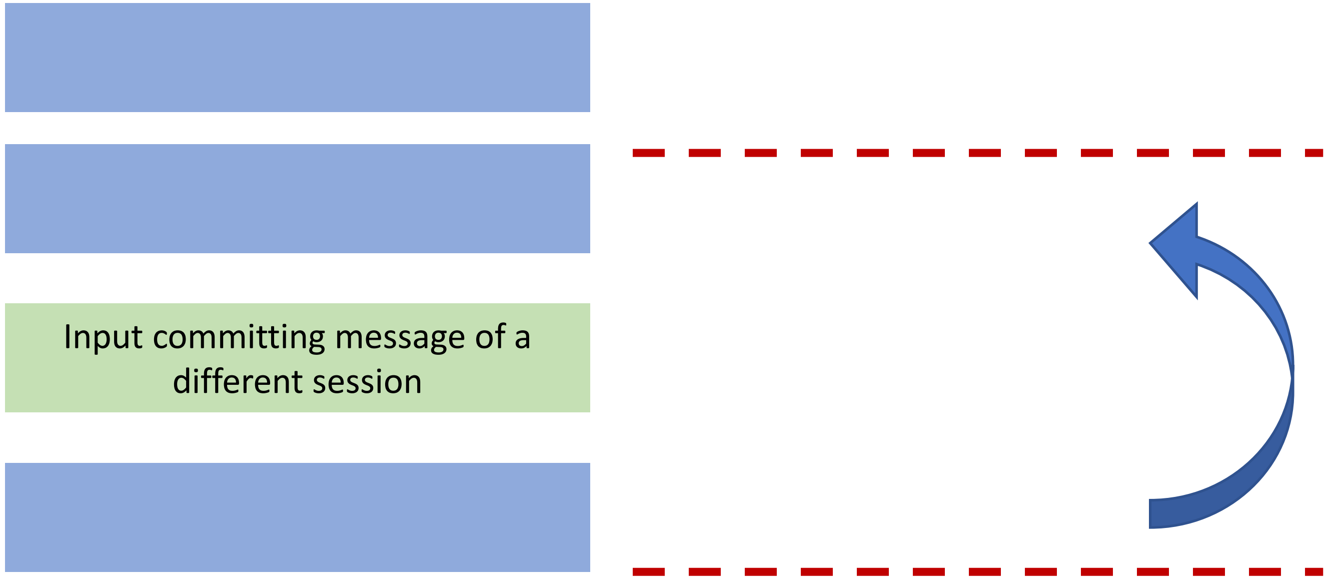
[Agrawal-Goyal-Jain-Prabhakaran-Sahai 12]

[Garg-Kumarasubramanian-Ostrovsky-Visconti 12]

Concurrent Secure Computation Impossible

Black-box impossible
[Lindell 04]

Non-black box, and varied settings:
[Barak-Prabhakaran-Sahai 06]
[Goyal 12]
[Agrawal-Goyal-Jain-Prabhakaran-Sahai 12]
[Garg-Kumarasubramanian-Ostrovsky-Visconti 12]



Input committing message of a different session

Structure of Concurrently Secure Computation

Weaker models

[Goyal-Jain-Ostrovsky 10, Goyal-Gupta-Jain 13, Canetti-Goyal-J 15]

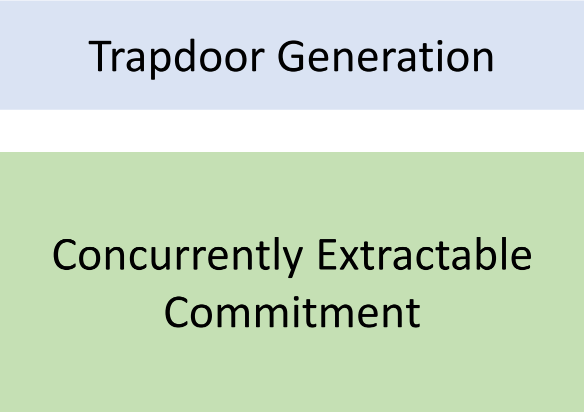
Structure of Concurrently Secure Computation

Trapdoor Generation

Weaker models

[Goyal-Jain-Ostrovsky 10, Goyal-Gupta-Jain 13, Canetti-Goyal-J 15]

Structure of Concurrently Secure Computation



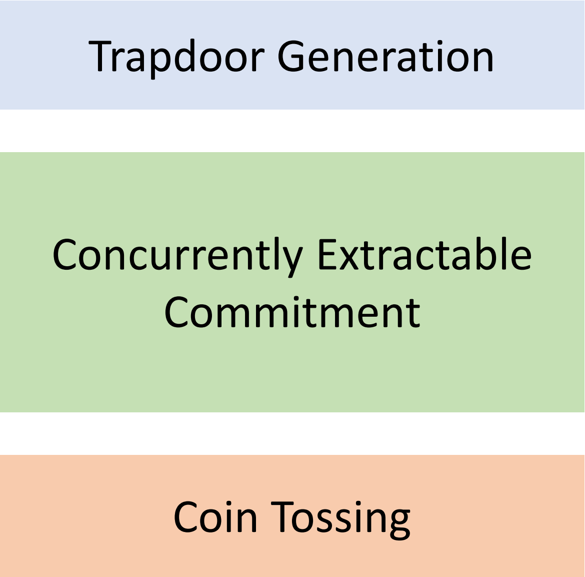
Trapdoor Generation

Concurrently Extractable Commitment

Weaker models

[Goyal-Jain-Ostrovsky 10, Goyal-Gupta-Jain 13, Canetti-Goyal-J 15]

Structure of Concurrently Secure Computation



Trapdoor Generation

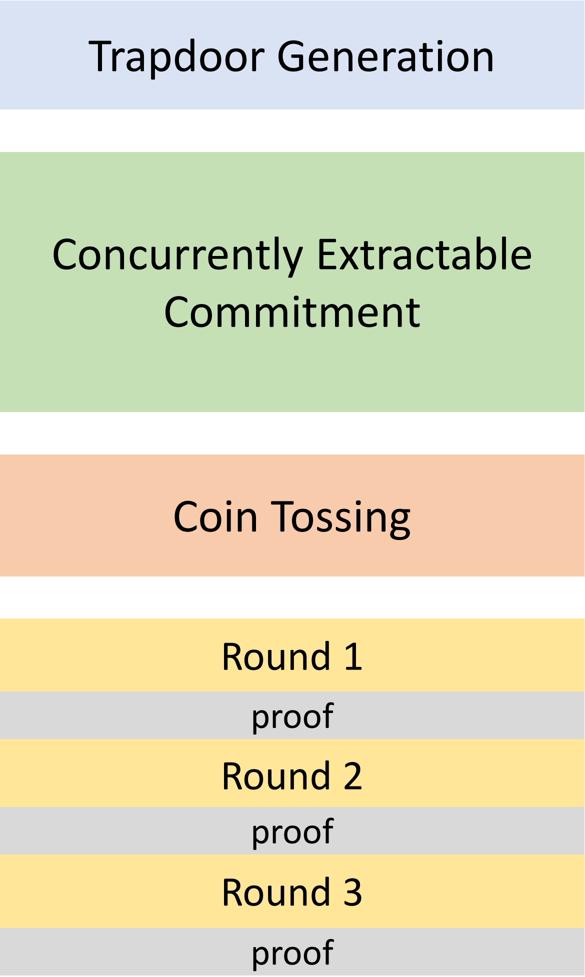
Concurrently Extractable Commitment

Coin Tossing

Weaker models

[Goyal-Jain-Ostrovsky 10, Goyal-Gupta-Jain 13, Canetti-Goyal-J 15]

Structure of Concurrently Secure Computation



Trapdoor Generation

Concurrently Extractable Commitment

Coin Tossing

Round 1

proof

Round 2

proof

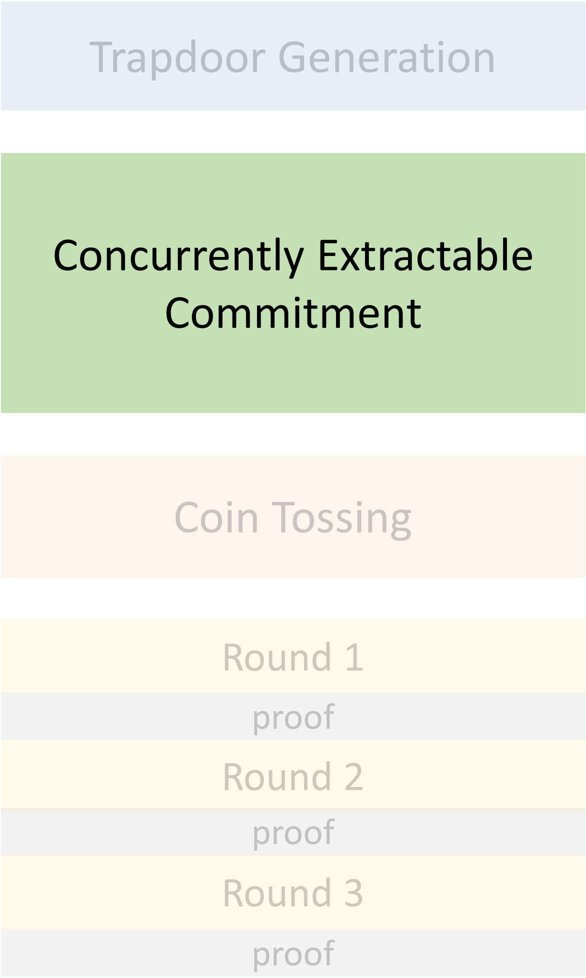
Round 3

proof

Weaker models

[Goyal-Jain-Ostrovsky 10, Goyal-Gupta-Jain 13, Canetti-Goyal-J 15]

Structure of Concurrently Secure Computation



Trapdoor Generation

Concurrently Extractable Commitment

Coin Tossing

Round 1

proof

Round 2

proof

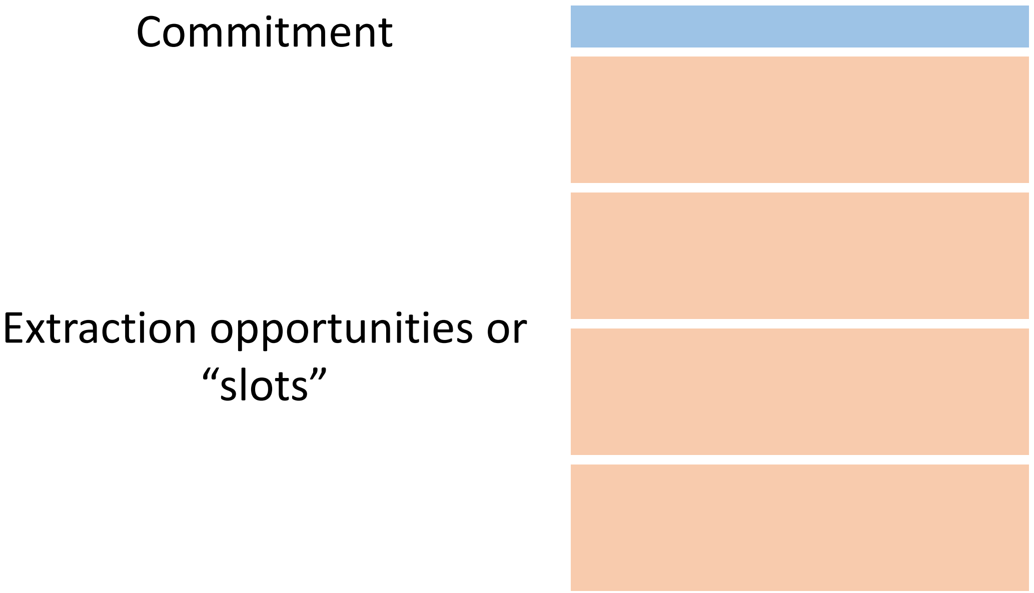
Round 3

proof

Weaker models

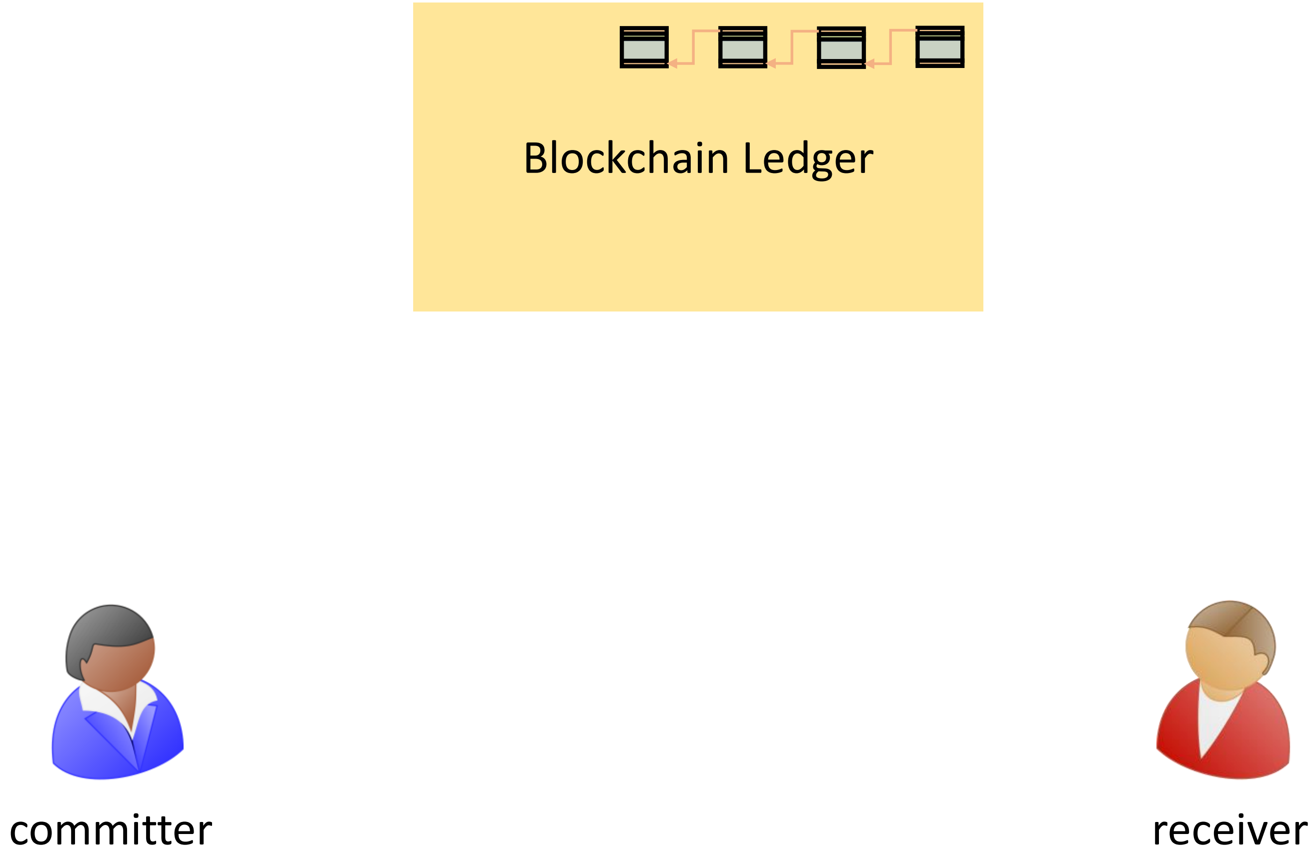
[Goyal-Jain-Ostrovsky 10, Goyal-Gupta-Jain 13, Canetti-Goyal-J 15]

Commitment: Structure



Commitment

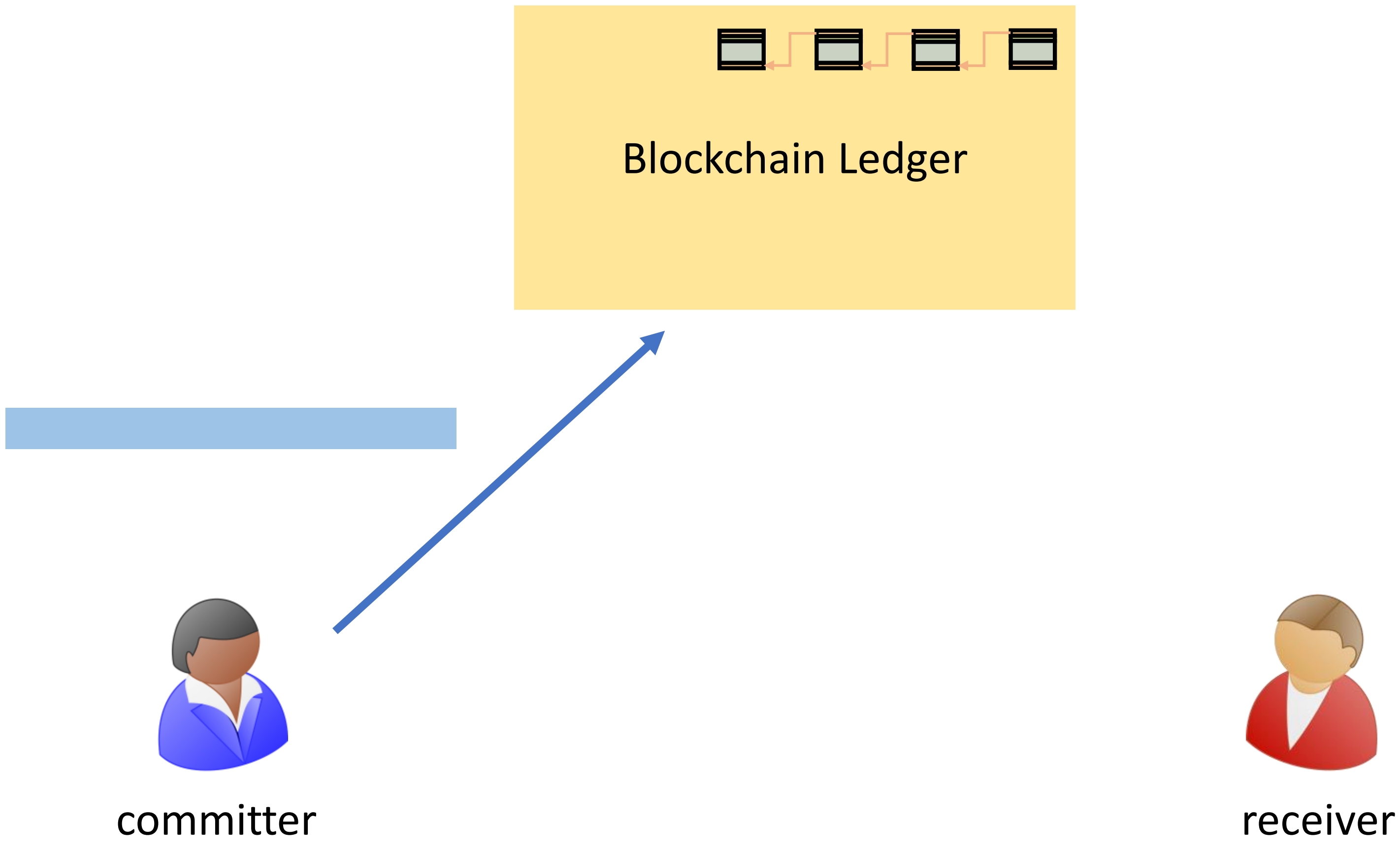
Extraction opportunities or
“slots”



Blockchain Ledger

committer

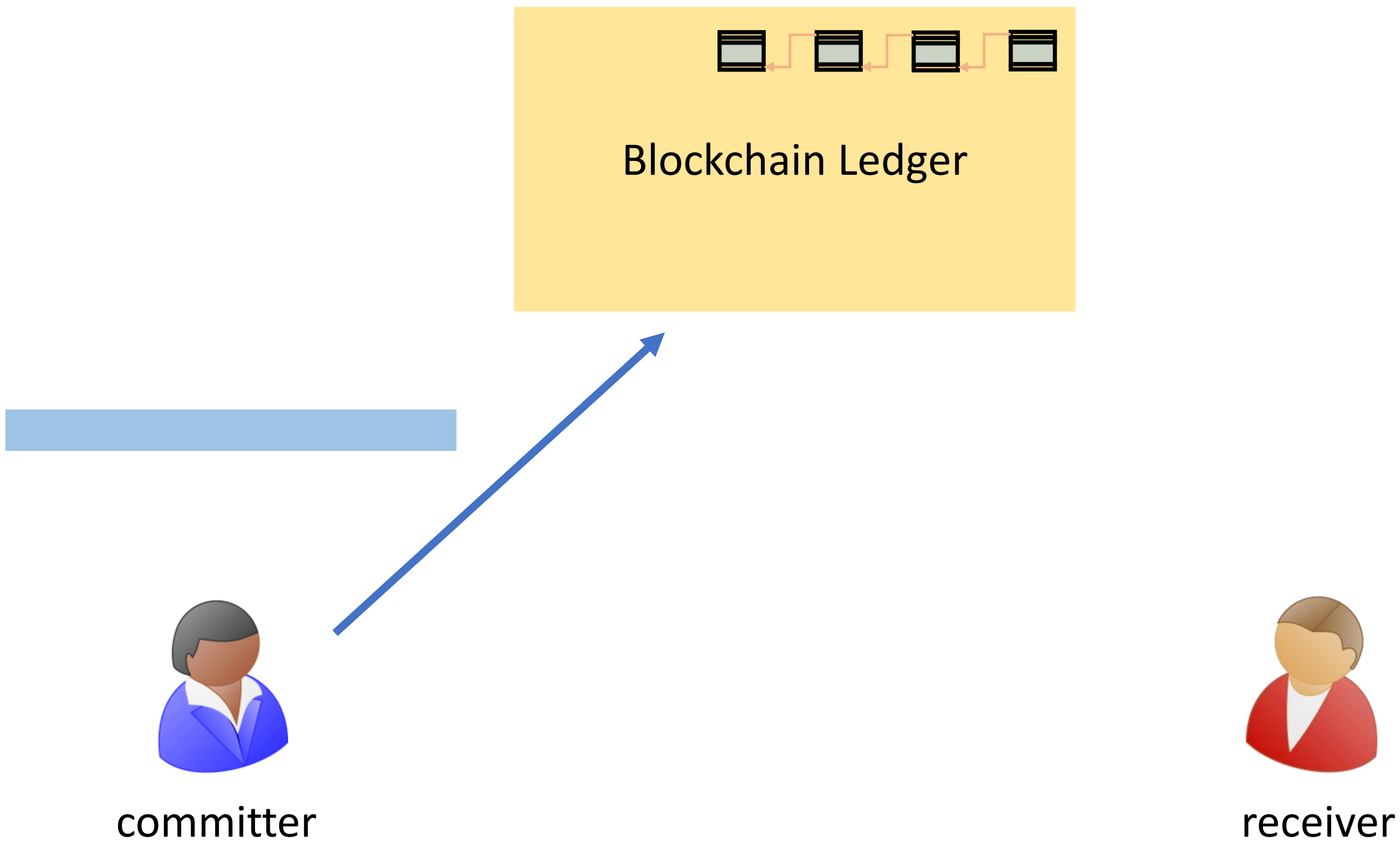
receiver



Blockchain Ledger

committer

receiver

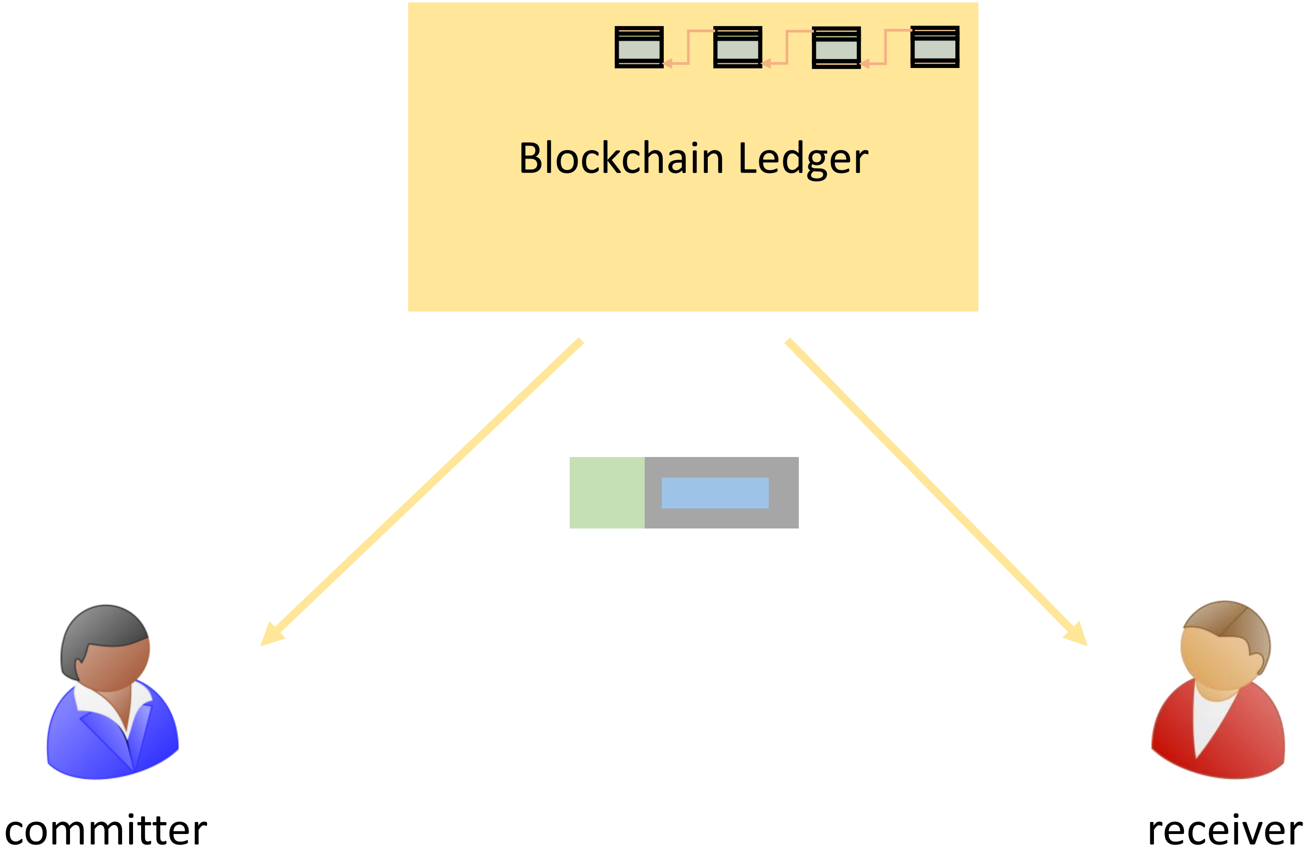


Blockchain Ledger

committer

receiver

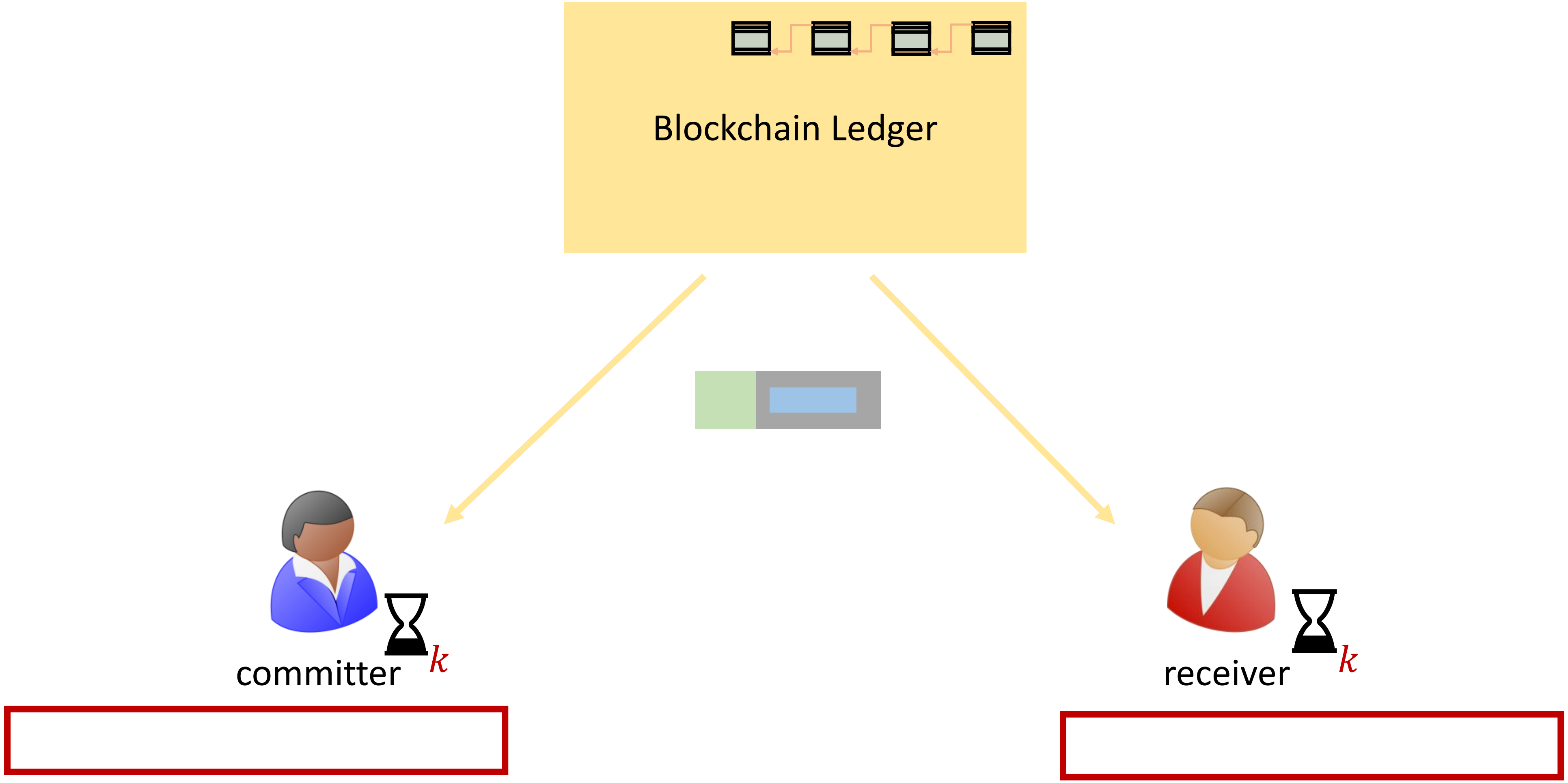
Commit to the blockchain



Blockchain Ledger

committer

receiver



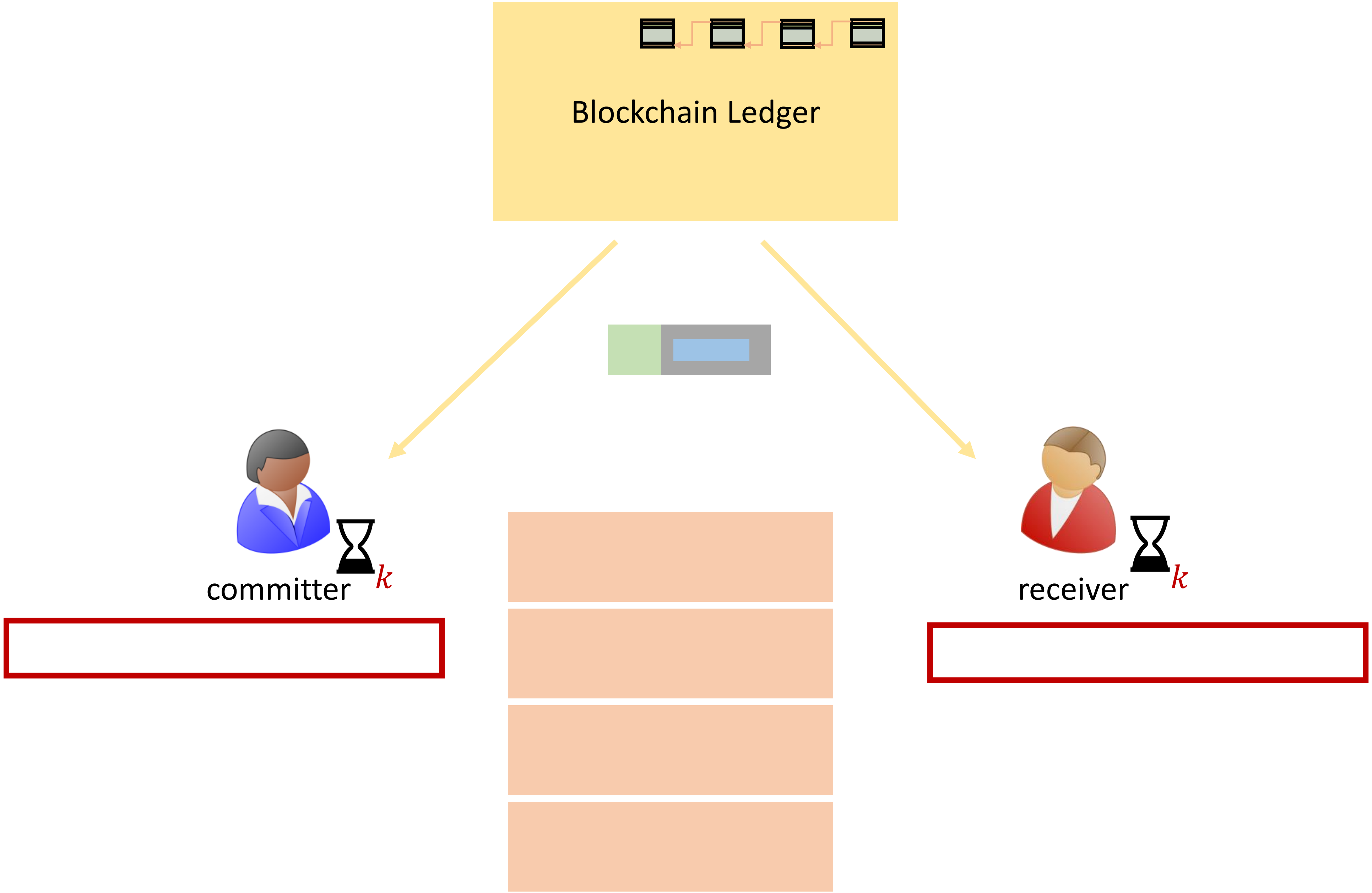
Blockchain Ledger

committer

k

receiver

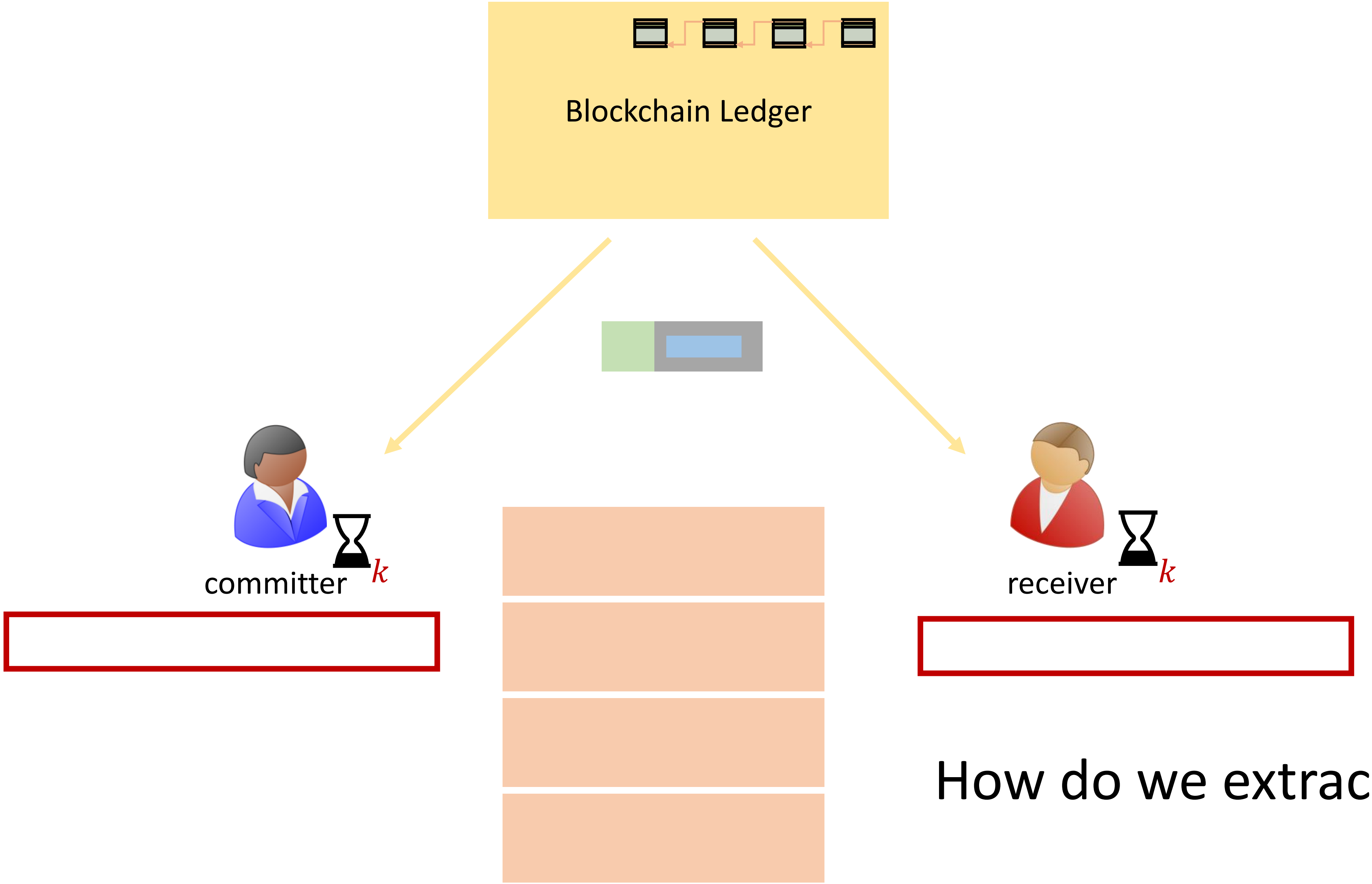
k



Blockchain Ledger

committer k

receiver k



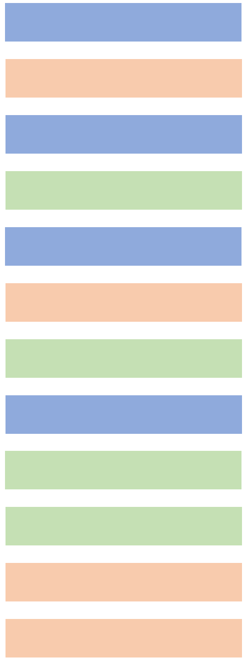
Blockchain Ledger

committer k

receiver k

How do we extract?

Extraction: Idea

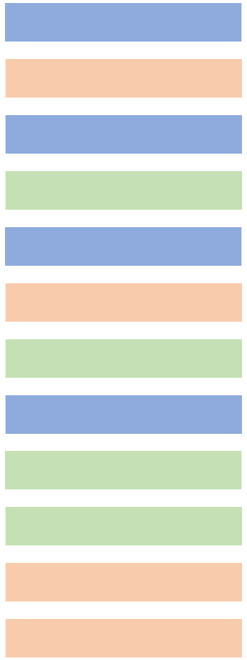


Extraction: Idea

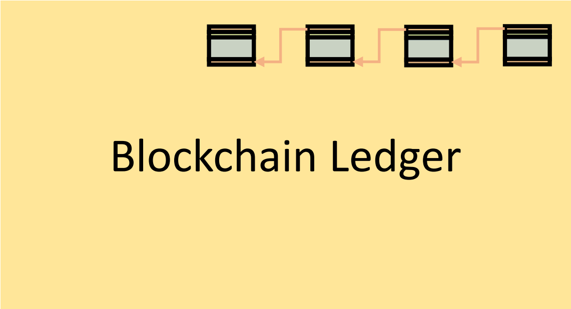
Robust Extraction Lemma

[Goyal-Lin-Pandey-Pass-Sahai15]

Extraction in the presence of constant number of external messages.



Extraction: Idea

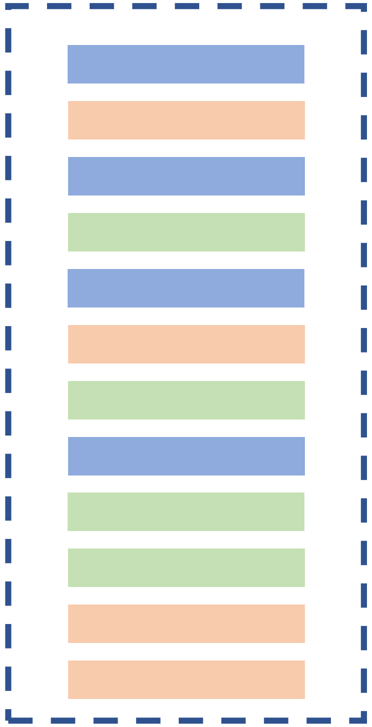


Blockchain Ledger

Robust Extraction Lemma

[Goyal-Lin-Pandey-Pass-Sahai15]

Extraction in the presence of constant number of external messages.

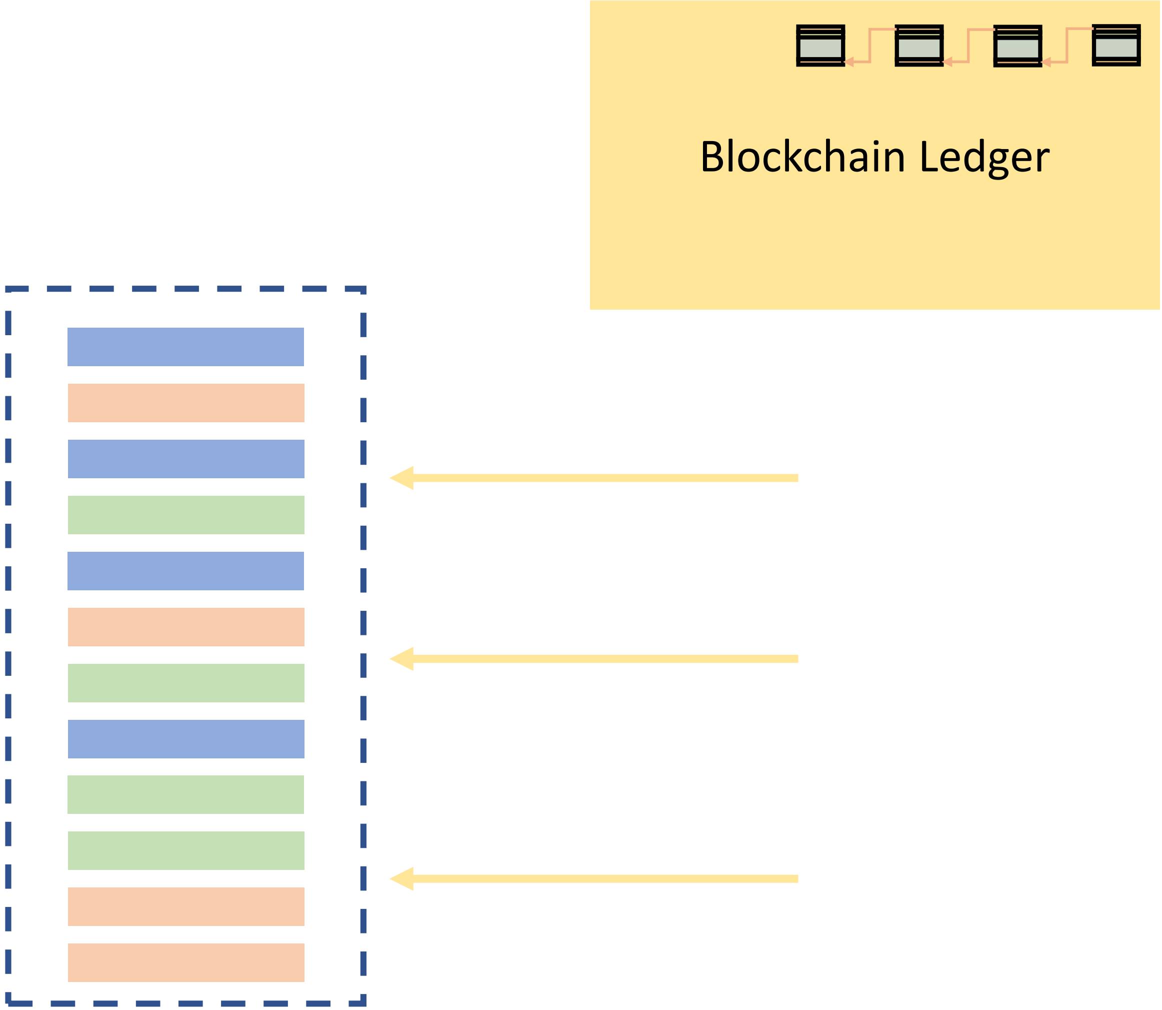


Extraction: Idea

Robust Extraction Lemma

[Goyal-Lin-Pandey-Pass-Sahai15]

Extraction in the presence of constant number of external messages.

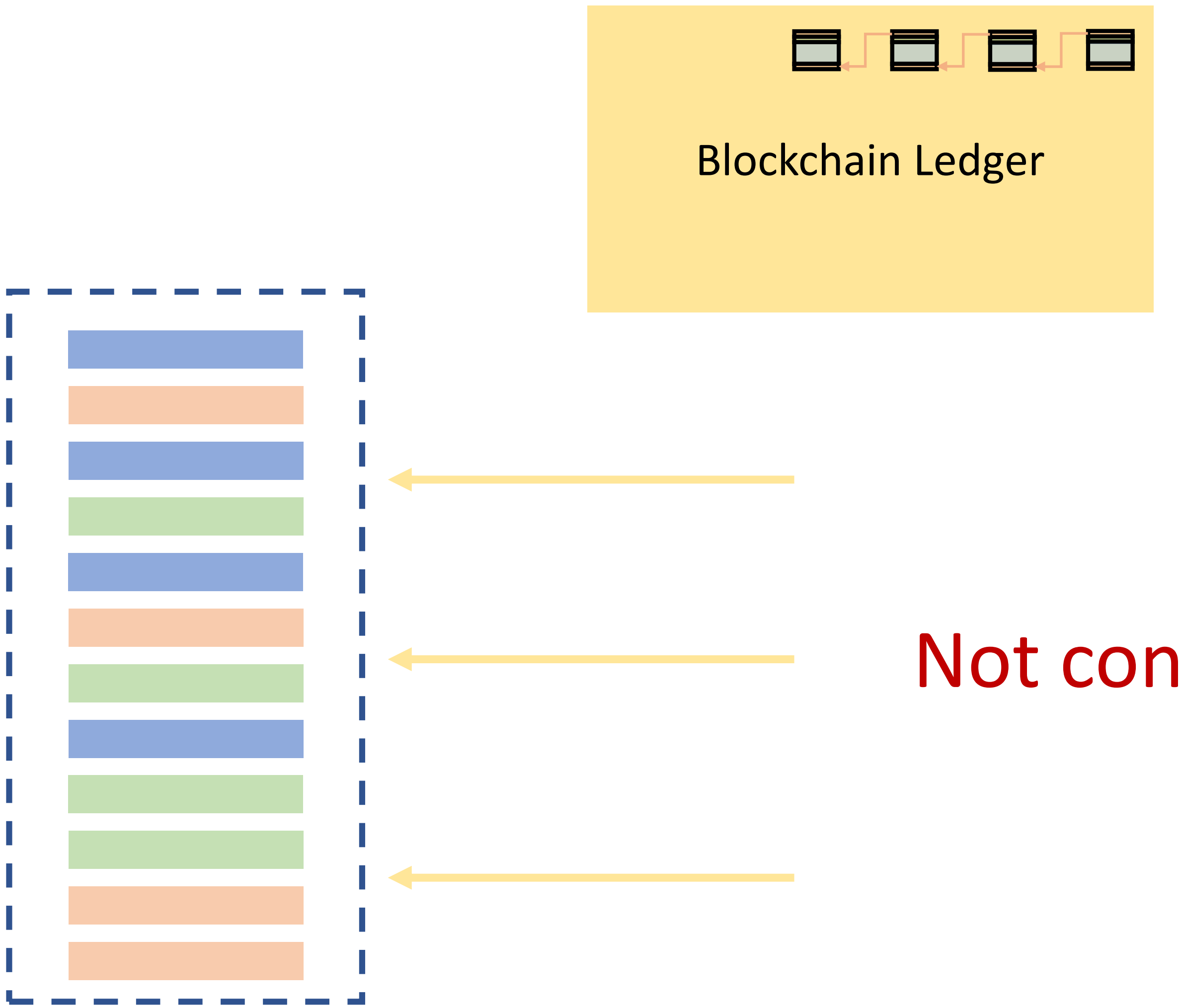


Extraction: Idea

Robust Extraction Lemma

[Goyal-Lin-Pandey-Pass-Sahai15]

Extraction in the presence of constant number of external messages.



Not constant

Conclusion

Blockchains have both destructive and constructive uses in the context of secure computation.

Thank you. Questions?

https://ia.cr/2019/253
achoud@cs.jhu.edu