Elements of DeFi

https://web3.princeton.edu/elements-of-defi/

Professor Pramod Viswanath

Princeton University

Lecture 14

Wrapped tokens and bridges

Interconnecting Blockchains

Last lecture: Attack Surface due to Lending

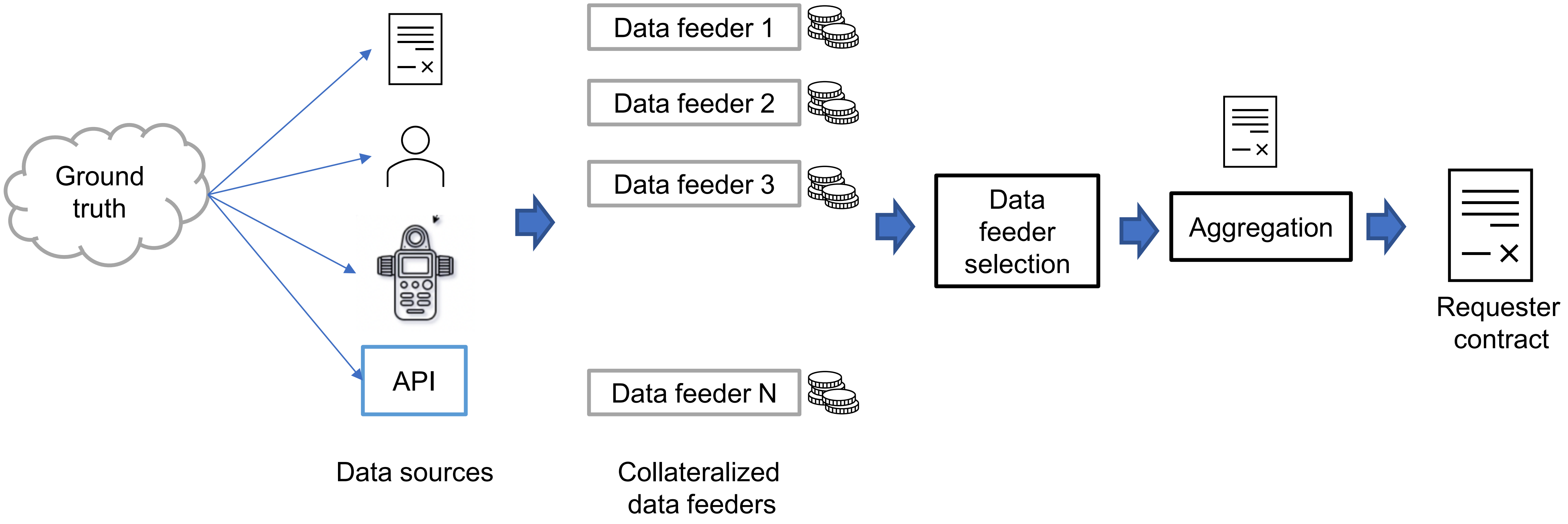
- Attacker can be attacked:
    - Sandwich
    - Poisoned sandwich
- Attacks can be done cheaply:
    - Flash loan attacks
    - Euler labs attack and the dangers of contagion
- Attacks can destabilize the trust platform:
    - Shorting attacks in PoS protocols: **dangers to consensus**
    - Lending vs Staking tradeoff in PoS protocols: danger even without byzantine agents

This Lecture: Wrapped tokens and bridges

- Importing data from other blockchains
  - Wrapped tokens
- General bridge architecture
  - Design space
  - Desired properties
- Bridge designs
- Blockchain interoperability not via bridges

Recap: Oracles

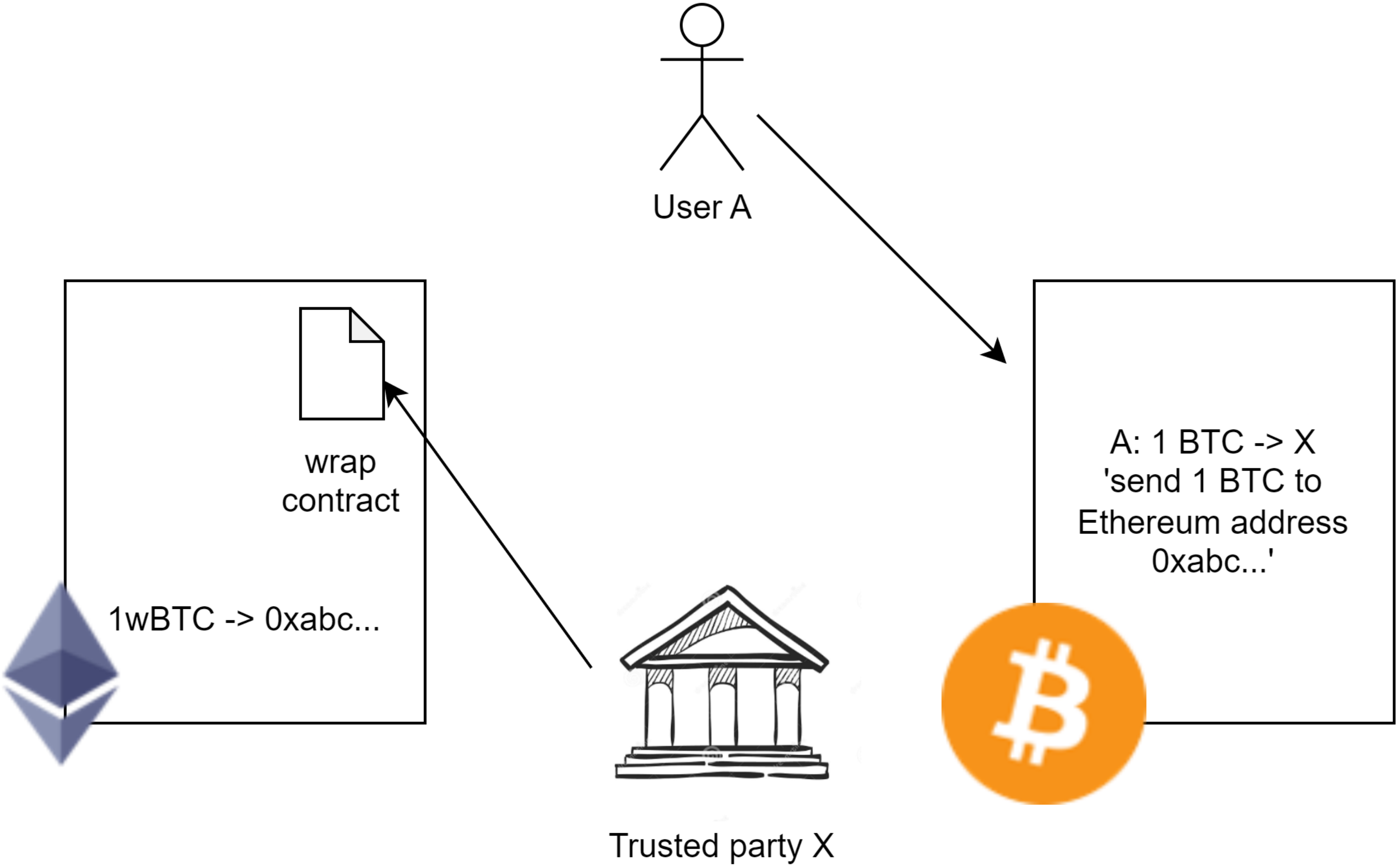
- Oracles are a general-purpose fabric connecting blockchains with other off chain systems



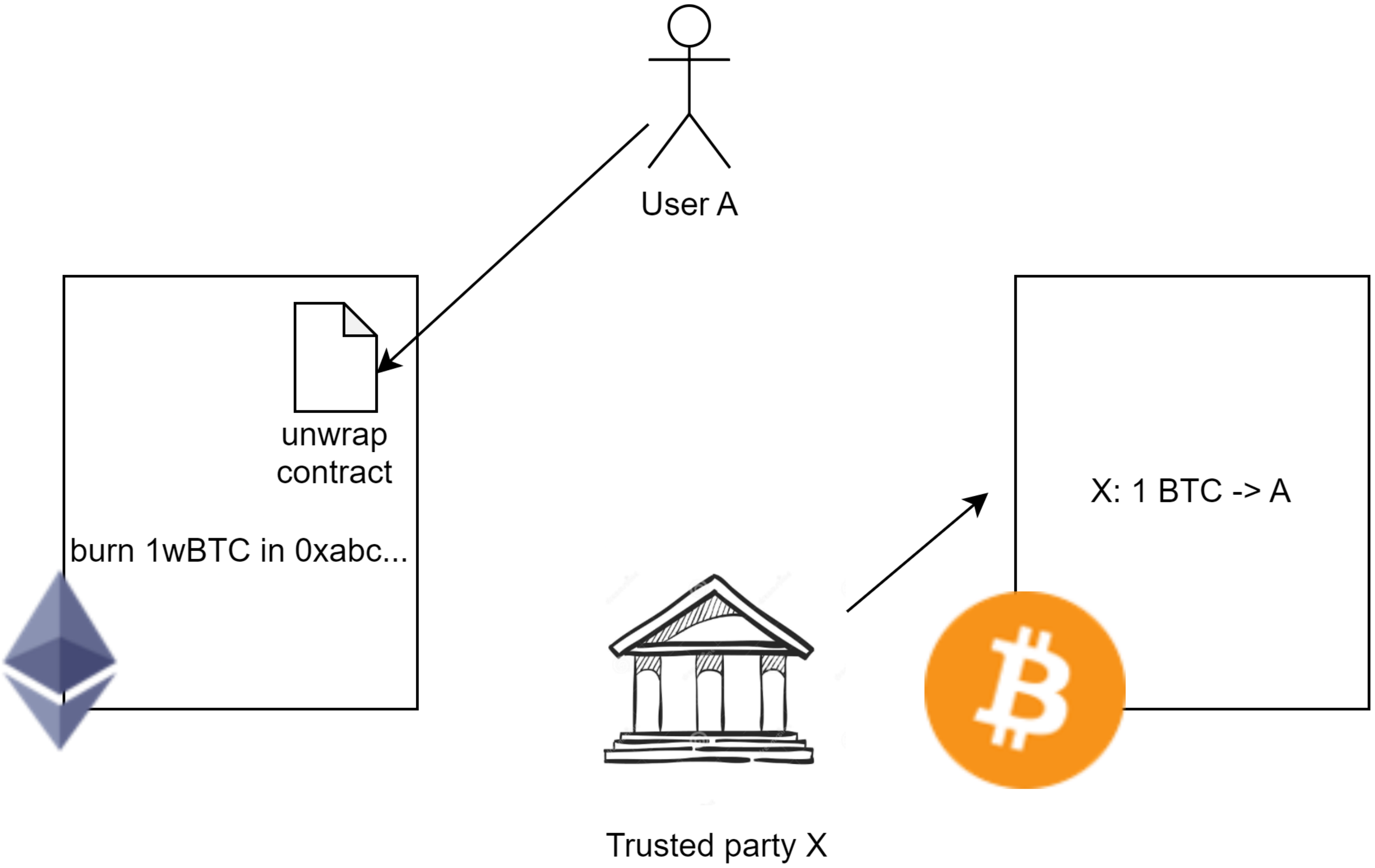
Ground truth from other blockchains

- Need for blockchain interoperability
- Increased liquidity
  - Smooth exchange of assets between different blockchain networks
- Enhanced functionality
  - Access to smart contracts, enhanced privacy, or improved scalability on other blockchains

Wrapped token as an example



Wrapped token as an example

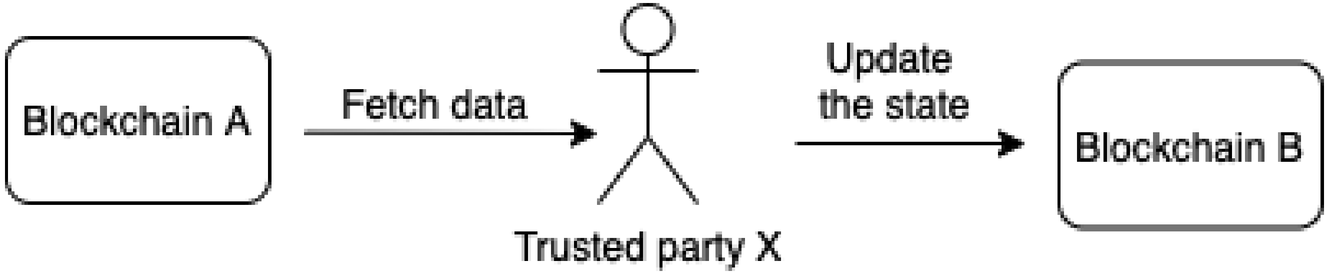


Bridges enabled DeFi applications

- Cross chain token exchange
- Cross-chain collateralized loans
- Cross-chain yield harvesting

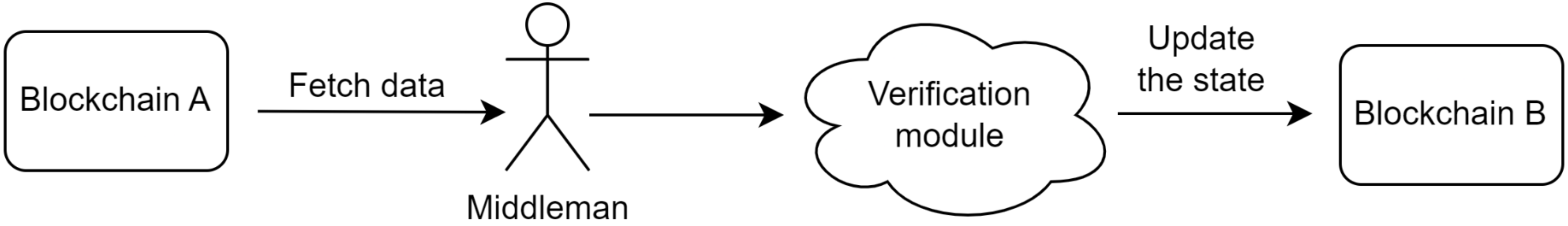
To enable these applications, we need a general-purpose messaging protocols among blockchains, aka cross-chain **bridges.**

Strawman bridge design



- Issues: single point of failure; no verification of data

General bridge architecture



data = cross-chain message + “proof of consensus”

Design space

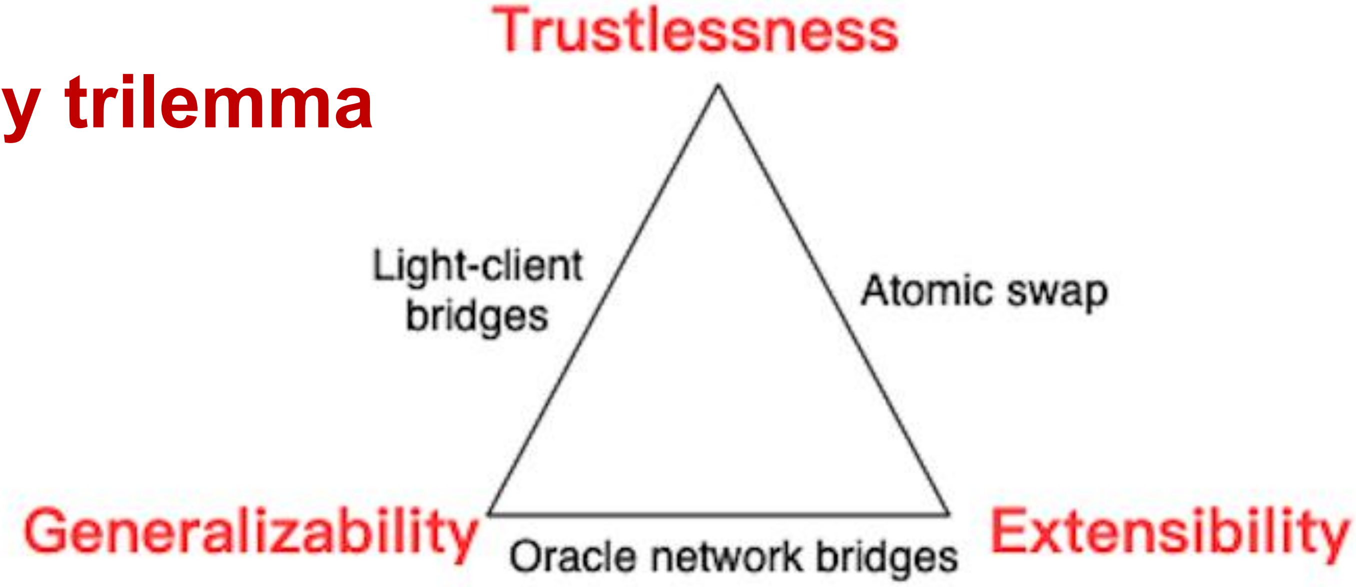
- Middleman
    - Relayer: simply relay the data
    - Validator: validate and sign the data
    - Any oracle network design

- Verification module
    - On-chain light client
    - Signature verification
    - Oracle contract

Desired properties

- **Trustlessness:** equivalent security to the underlying blockchains
- **Extensibility:** able to be supported on any blockchain
- **Generalizability:** capable of handling arbitrary applications

Interoperability trilemma



Bridge designs

- Oracle-network bridges
  - Layer 0: Oracle and relayer cooperate to verify data
  - Axelar: full consensus for verifying data

- Light-client bridges
  - Cosmos IBC: light clients embedded in Cosmos-SDK
  - ZkBridge: verifier contract checks the validity of the data via ZKP

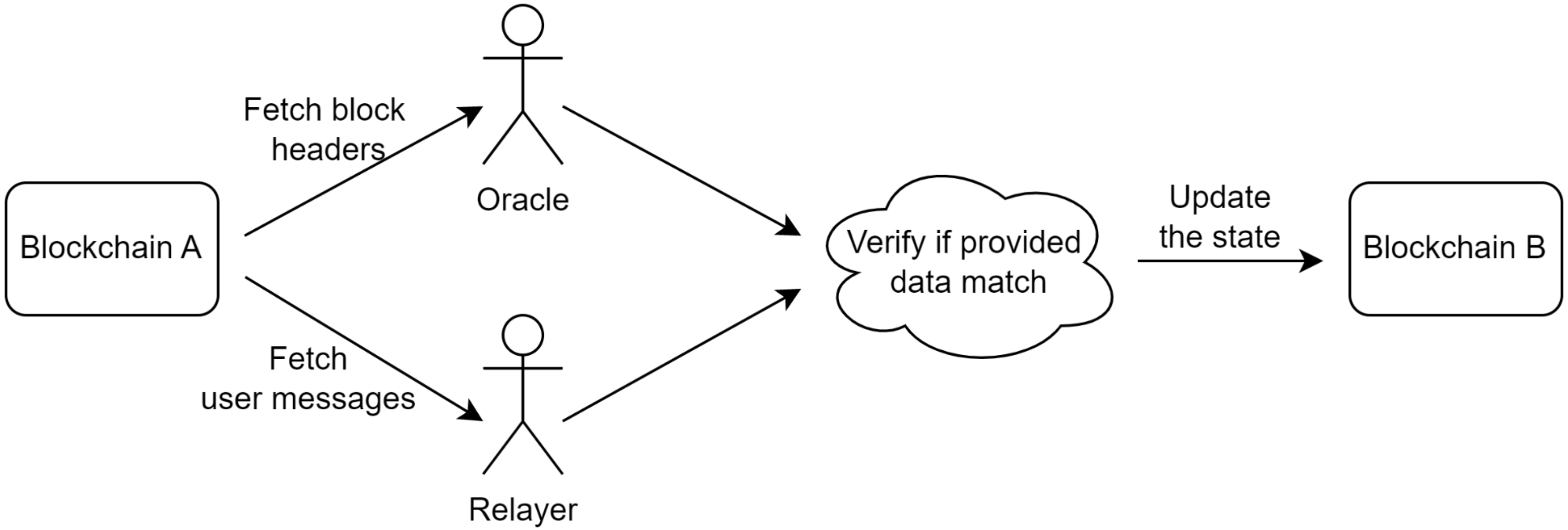
Oracle-network bridges

- Middleman: a set of validators or an oracle network
- Verification module: verifies signatures

❌ **Trustless**
✅ **Extensible**
✅ **Generalizable**

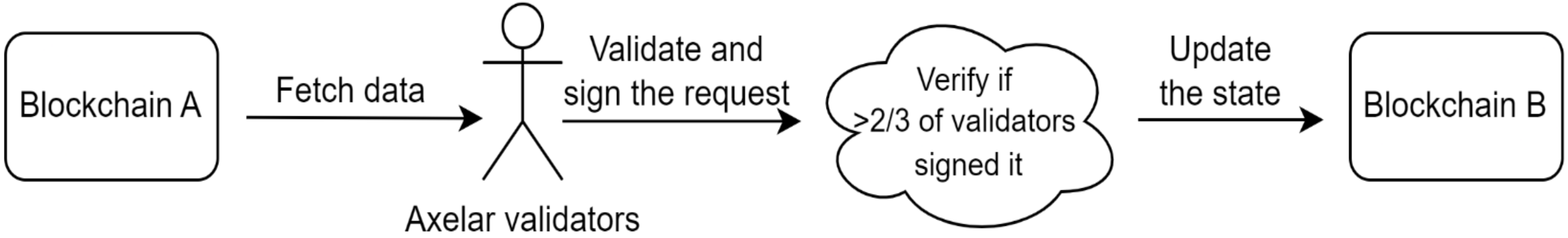
Layer 0

- An Oracle like ChainLink verifies the block headers
- Nodes called Relayers provide proof of inclusion of a message
- **Security assumption:** Relayer and Oracle assumed not to collude



Axelar

- A validator set of 50 validators run a full consensus protocol


Blockchain A
Fetch data
Axelar validators
Validate and sign the request
Verify if >2/3 of validators signed it
Update the state
Blockchain B

Light-client bridges

- Light clients are blockchain nodes that can verify the confirmation of certain transactions without running a full node
- Middleman: trustless relayers
- Verification module: on-chain light clients

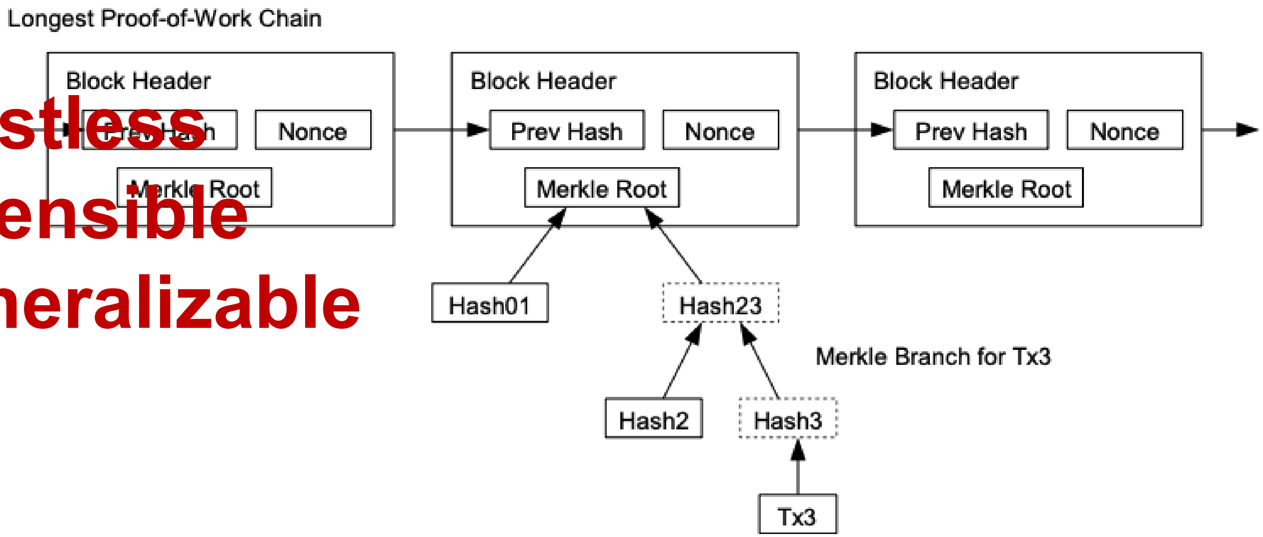
✅ Trustless

❌ Extensible

✅ Generalizable

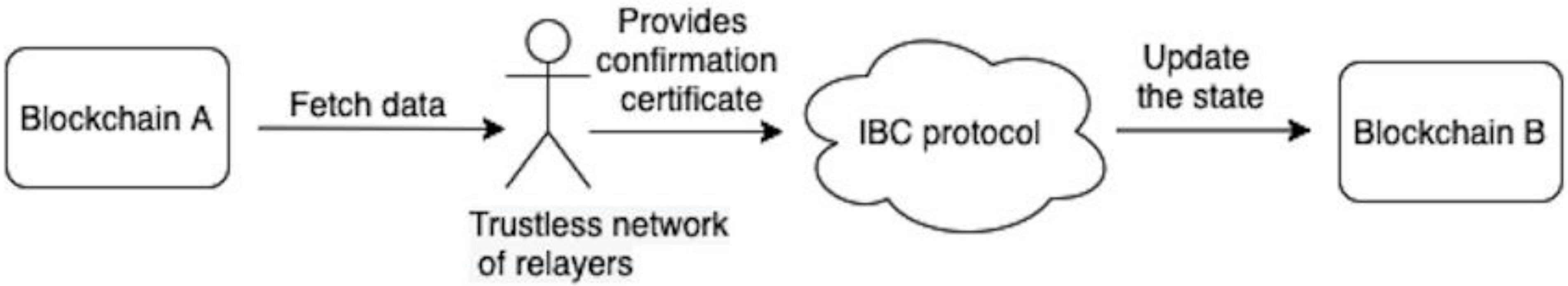
8. Simplified Payment Verification

It is possible to verify payments without running a full network node. A user only needs to keep a copy of the block headers of the longest proof-of-work chain, which he can get by querying network nodes until he's convinced he has the longest chain, and obtain the Merkle branch linking the transaction to the block it's timestamped in. He can't check the transaction for himself, but by linking it to a place in the chain, he can see that a network node has accepted it, and blocks added after it further confirm the network has accepted it.



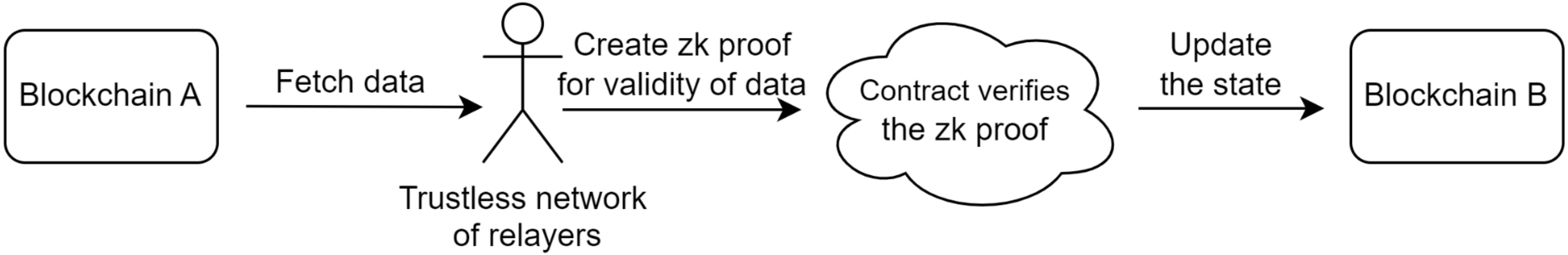
Cosmos IBC

- The confirmation certificate is a set of valid signatures by the source blockchain nodes
- IBC protocol is a part of Cosmos consensus mechanism
- The validators of destination chain verify the validity of data


Blockchain A
Fetch data
Trustless network of relayers
Provides confirmation certificate
IBC protocol
Update the state
Blockchain B

zkBridge

- On-chain smart contract verifies the data
- Verification cost reduced by using zero-knowledge proofs

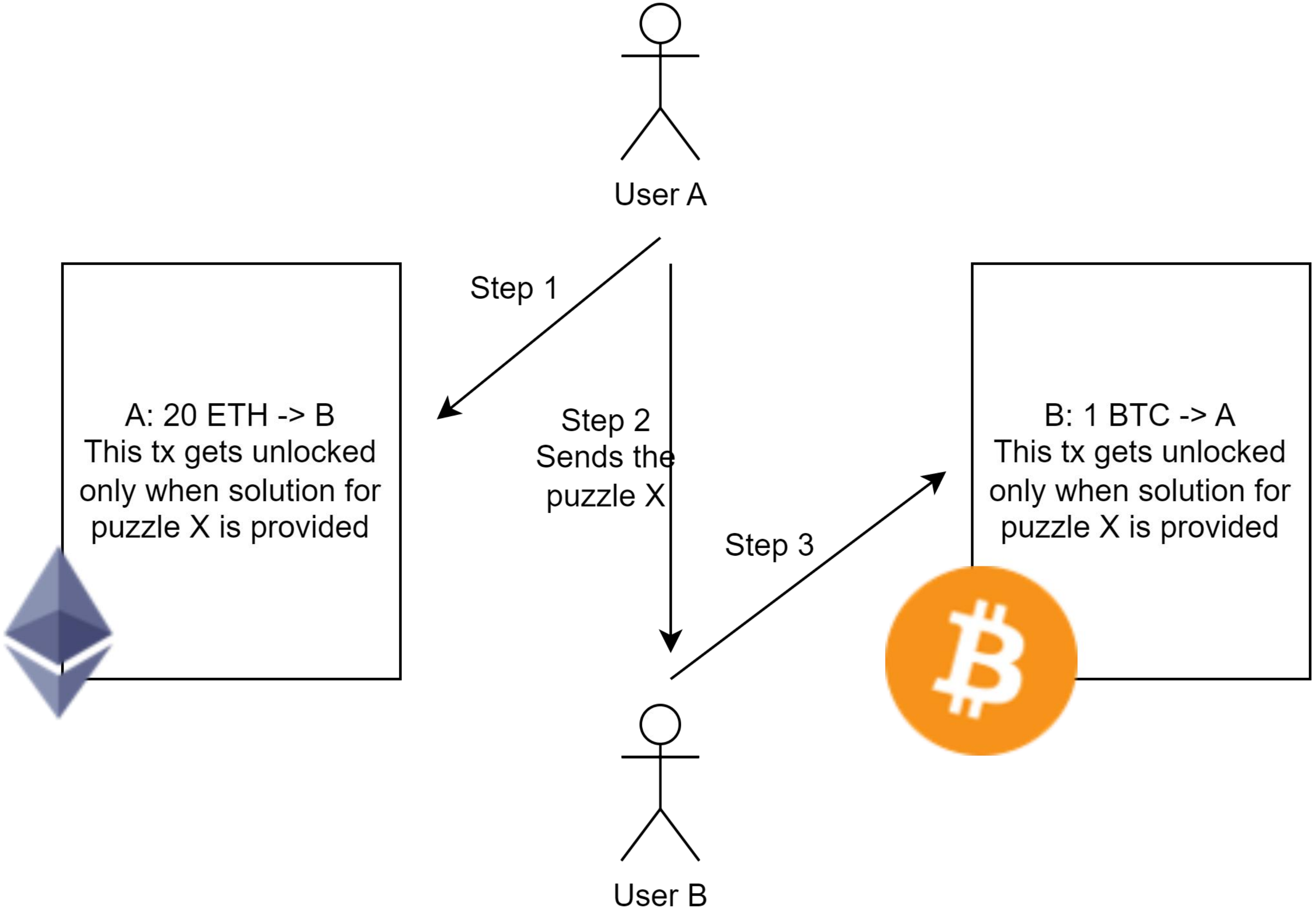

Blockchain A
Fetch data
Trustless network of relayers
Create zk proof for validity of data
Contract verifies the zk proof
Update the state
Blockchain B

Interoperability not via bridges

Hashed timelock contract (HTLC)

- HTLC stores a pair (x, t), where x is a hash puzzle and t is a timeout
- If the contract receives the matching secret s such that x = H(s), before time t has elapsed, then the transaction will be executed.
- If the contract does not receive the matching secret s before time t has elapsed, then the transaction will be aborted.

Cross-chain atomic swap



User A

Step 1

A: 20 ETH -> B
This tx gets unlocked only when solution for puzzle X is provided

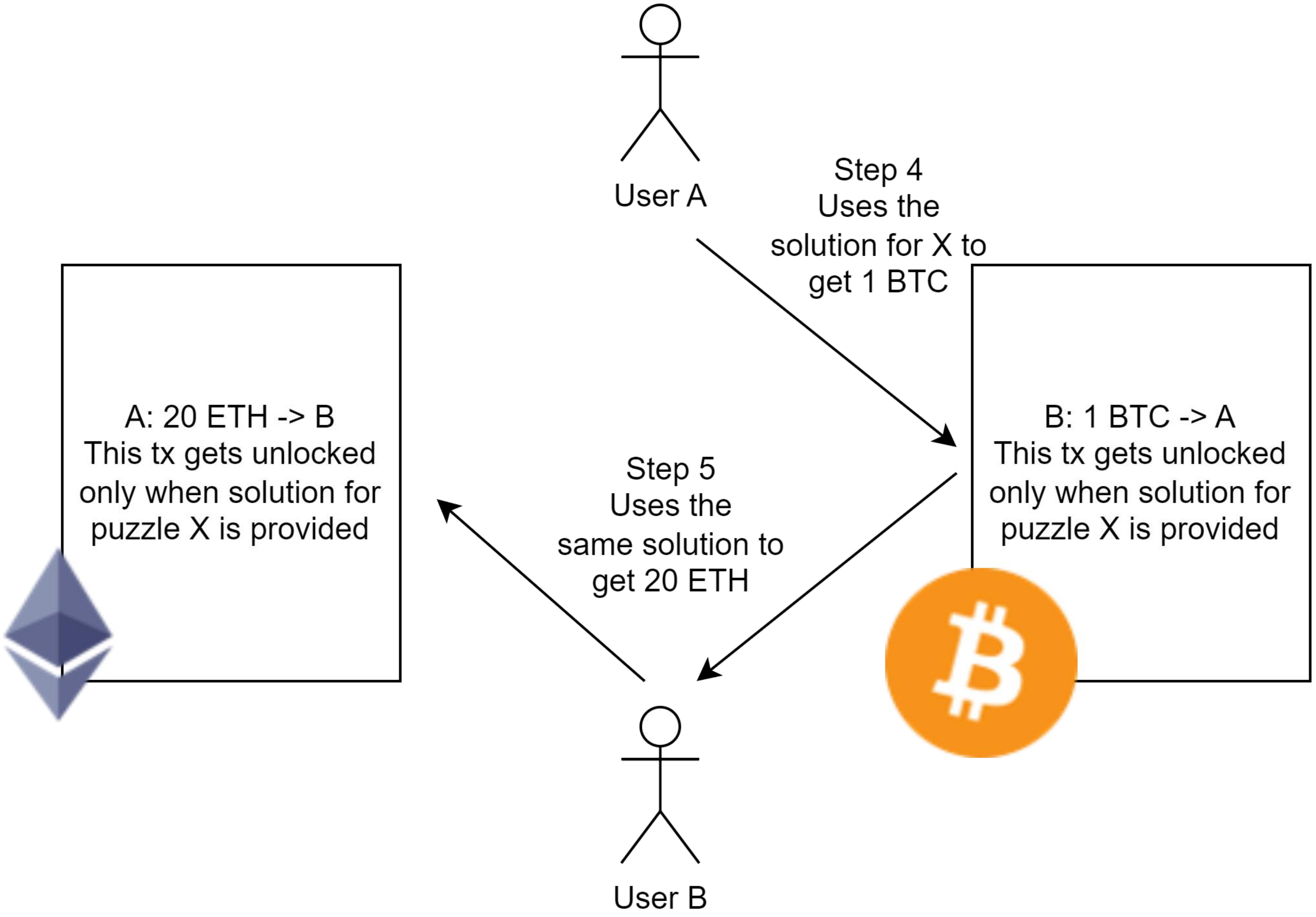
Step 2
Sends the puzzle X

Step 3

B: 1 BTC -> A
This tx gets unlocked only when solution for puzzle X is provided

User B

Commit



User A

Step 4
Uses the
solution for X to
get 1 BTC

A: 20 ETH -> B
This tx gets unlocked
only when solution for
puzzle X is provided

B: 1 BTC -> A
This tx gets unlocked
only when solution for
puzzle X is provided

Step 5
Uses the
same solution to
get 20 ETH

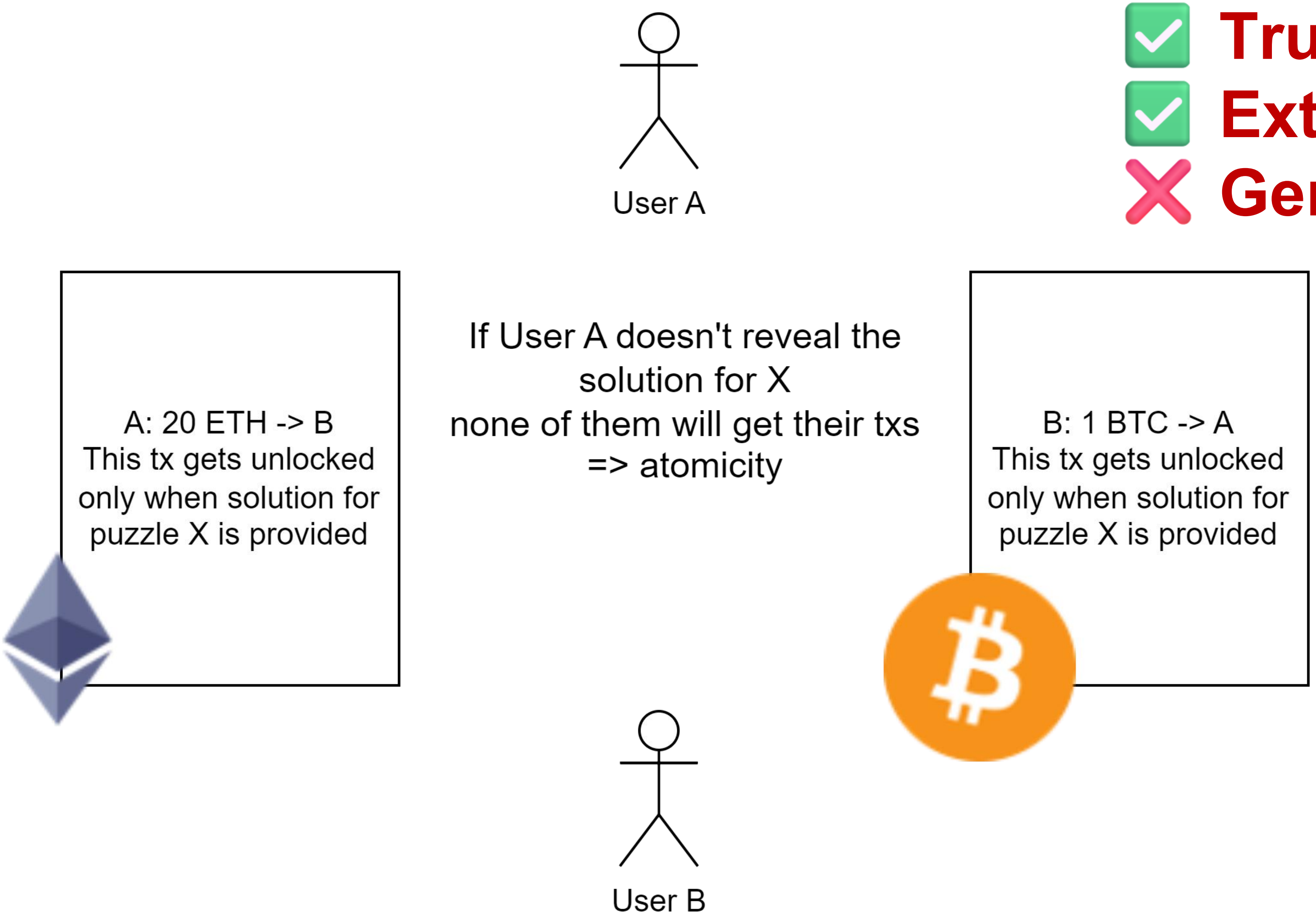
User B

Abort

✅ **Trustless**

✅ **Extensible**

❌ **Generalizable**



A: 20 ETH -> B
This tx gets unlocked only when solution for puzzle X is provided

If User A doesn't reveal the solution for X
none of them will get their txs
=> atomicity

B: 1 BTC -> A
This tx gets unlocked only when solution for puzzle X is provided

Comparison

	Trustlessness	Extensibility	Generalizability	Middleman assumption
WBTC	No	Yes	No	- Trusted middleman
Atomic swap	Yes	Yes	No	- No middleman
Layer 0	No	Yes	Yes	- Relayer and Oracle should not collude - One honest relayer
Axelar	No	Yes	Yes	- Full consensus
Cosmos IBC	Yes	No	Yes	- One honest relayer
zkBridge	Yes	No	Yes	- One honest relayer

LECTURE ENDS