NANYANG TECHNOLOGICAL UNIVERSITY SINGAPORE

Bitcoin

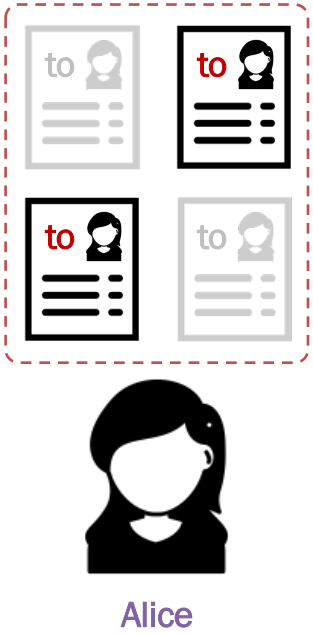
# Mining and Proof-of-Work

Dr Sourav SEN GUPTA
Lecturer, SCSE, NTU

Bitcoin Network

# Node Types and Roles

# Recall




Input [0] : UTXO in own Wallet [index]

Signature Script to “unlock” UTXO

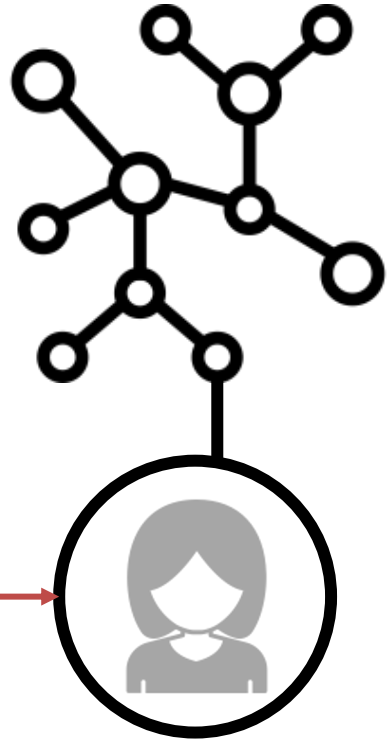
Input [1] : UTXO in own Wallet [index]

Signature Script to “unlock” UTXO

Output [0] : Script to “lock” UTXO

Output [1] : Script to “lock” UTXO

Metadata for this new Transaction



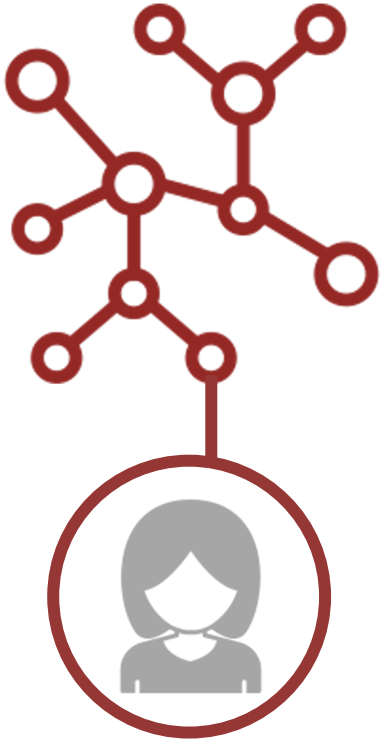
# The Bitcoin Network

Peer-to-Peer Network Architecture

overlayed on the regular Internet

- The nodes connect over a mesh network
- The network adheres to a “flat” topology
- There is no server or centralized service
- There is no hierarchy within the network

Built on the philosophy of decentralized control.



# “Extended” Bitcoin Network [1]

Protocols in the real-life Bitcoin system

- P2P protocol for the nodes maintaining Bitcoin
- Stratum protocol for mining and mobile wallets
- Pool Mining protocols for Bitcoin mining pools

Extended Bitcoin network includes

- Core nodes running the P2P bitcoin protocol
- Gateway or routing servers connecting others
- Stratum servers connecting stratum miners
- Other nodes “extending” system components

[1] reading : Chapter 6 of the book “Mastering Bitcoin”

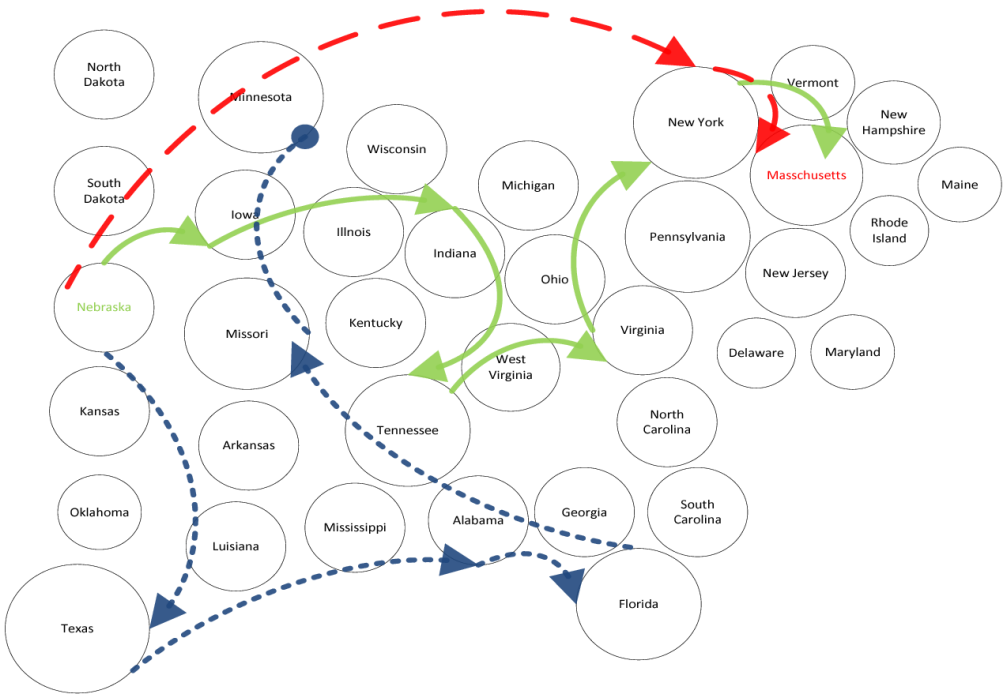
# Recall : Peer-to-Peer Systems

Each node maintains an ad-hoc list of neighbors (discovered at the startup phase of a node).

Communication in P2P networks

- Flooding to all Neighbors
- Random Walk on Network

Bitcoin adopts “Propagation” to some/all “Neighbors” of a node.



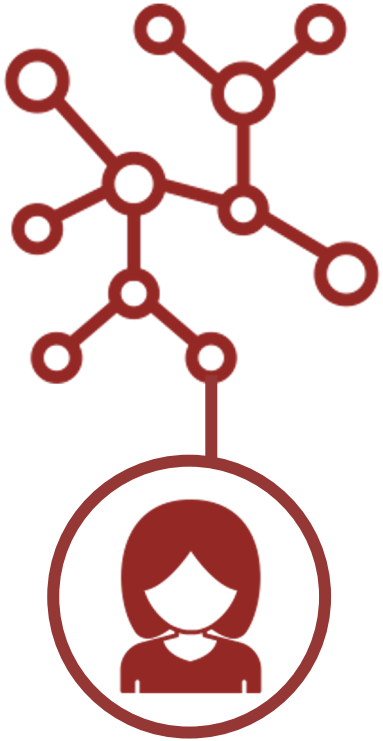
# Bitcoin Nodes

**Nodes may choose from 4 core functions**

- Networking : Routing and Propagating
- Database : Storing the Full Blockchain
- Mining : Maintaining Bitcoin Blockchain
- Wallet : Services for Bitcoin “Accounts”

Every Bitcoin node validates and propagates Transactions and Blocks in the Bitcoin network.

Extended network nodes may choose otherwise.



# Nodes in the Bitcoin Network

## Reference Client or Bitcoin Core

Network routing + Full Blockchain Database + Mining + Bitcoin Wallet

## Full Blockchain Node

Network routing + Full Blockchain

## Solo Miner Node

Network routing + Full Blockchain + Mining

## User Wallets

Network routing + Full Blockchain + Bitcoin Wallet

# Nodes in “Extended” Bitcoin Network

## Lightweight (SPV) Wallets

Network routing + Bitcoin Wallet + Simplified Payment Verification

## Pool Protocol Servers

Gateway servers connecting Pools or Stratum nodes

## Mining Nodes

Mining without Full Blockchain + Pool/Stratum protocol

## Lightweight Stratum Wallet

Wallet + Stratum Network (without Full Blockchain)

# Communicate

Find a bitcoin node who can validate a transaction before propagating it to its neighboring nodes.

Alice

Input [0] : UTXO in own Wallet [index]

Signature Script to “unlock” UTXO

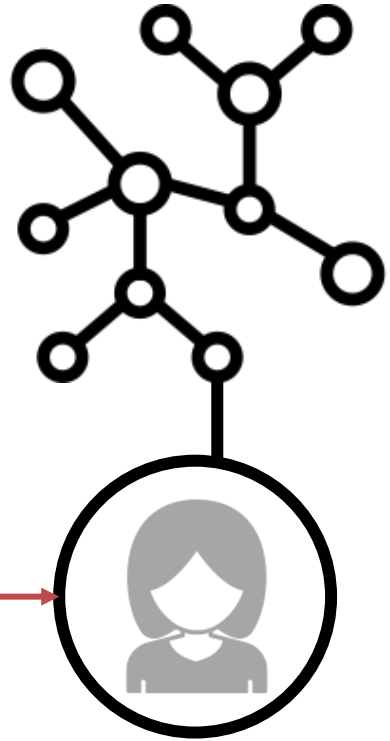
Input [1] : UTXO in own Wallet [index]

Signature Script to “unlock” UTXO

Output [0] : Script to “lock” UTXO

Output [1] : Script to “lock” UTXO

Metadata for this new Transaction



Bitcoin Consensus

# Mining and Proof of Work

# Bitcoin Consensus [2]

Depends on a few processes occurring independently across the P2P network.

- **Transaction Validation** : Verification of each transaction by every Full Node
- **Transaction Aggregation** : Recording transactions into blocks by Mining Nodes
- **Proof-of-Work** : Proof of demonstrated computation (mining) by Mining Nodes
- **Block Validation** : Verification of new blocks by every node with the Blockchain
- **Chain Selection** : Selection of “longest” chain with highest total Proof-of-Work

Every node can validate transactions or blocks propagated in bitcoin network.

However, every node can’t produce the required proof-of-work to mine a block.

[2] reading : Chapter 8 of the book “Mastering Bitcoin”

# Recall

required

Bitcoin Script Exec

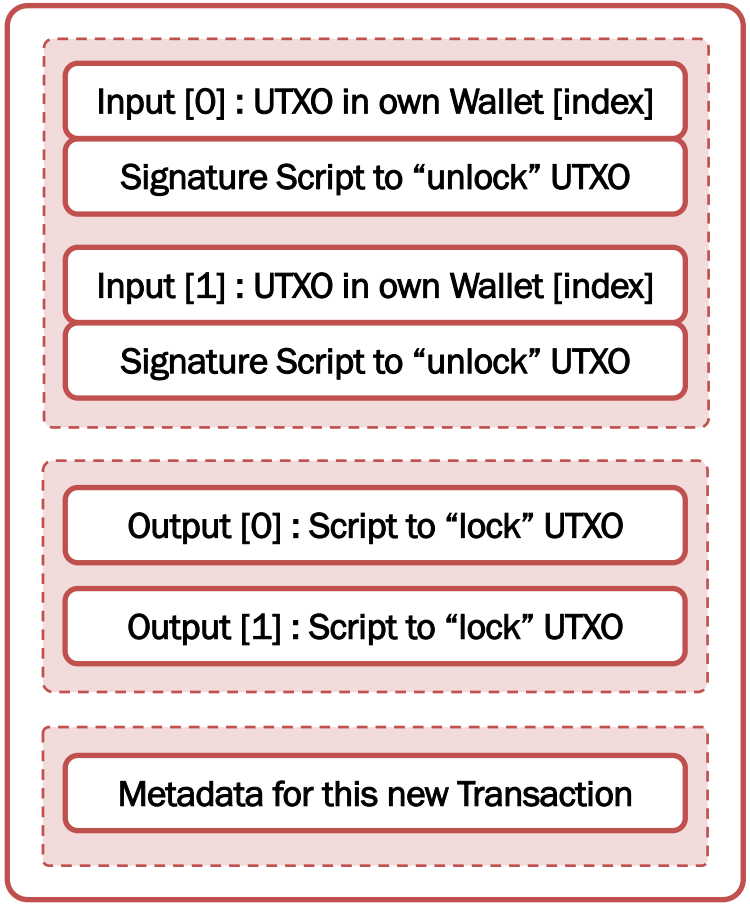
Recipient Public Key

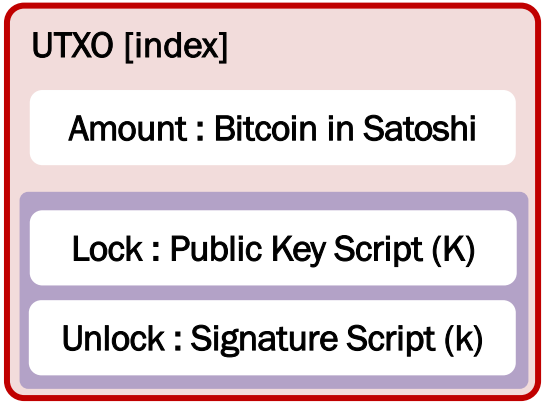
iteration

Validate all Inputs

track

Fee = Input – Output




# Transaction Validation

UTXO [index]

Amount : Bitcoin in Satoshi

Lock : Public Key Script (K)

Unlock : Signature Script (k)

Nodes in the bitcoin network generally maintain

- UTXO Pool for all unspent transaction outputs
- Transaction Pool for unconfirmed transactions
- Orphan Pool for transactions without a parent

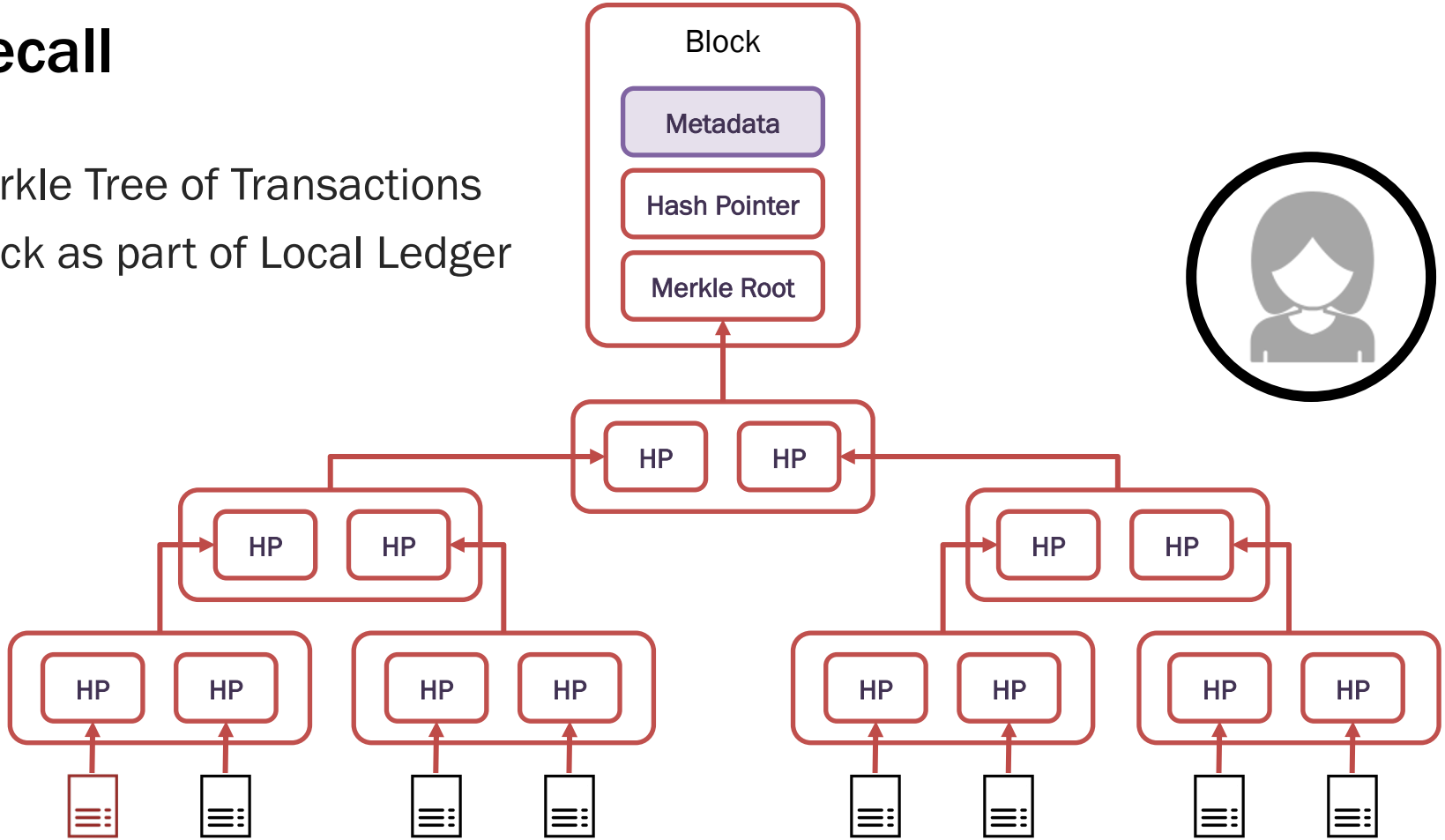
Each transaction is verified against a long “checklist” of syntax, input, output, fee, confirmations, signatures etc., before storing it in the Transaction Pool, in some “order”.

**Mining Nodes create Blocks out of their Transaction Pool.**

# Recall

Merkle Tree of Transactions

Block as part of Local Ledger



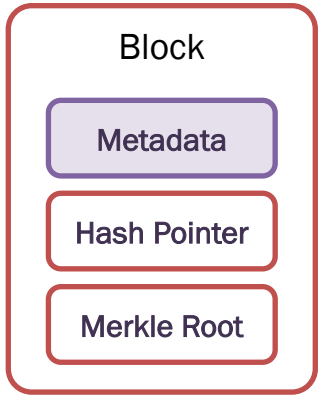
# Transaction Aggregation

**Mining Nodes create Blocks out of their Transaction Pool** according to certain order based on “transaction priority”.

Priority = Sum (Value of input * Input Age) / Tx Size

- Old UTXOs have higher priority to be spent as input
- High valued UTXOs have higher priority to be spent

High priority transactions have reserved space in blocks. Beyond that, a node may prioritize high fee transactions. Transactions not included in the block remain in the pool.




# Recall : Coinbase Transaction

Transaction with no Input, meant to Mine new Bitcoin into the ecosystem.

| | | | | |
|---|---|---|---|---|
| Hash | 8785cfd428b67e6f4419db7fb4529eb1216fd936476ceed888f77... | | | 2020-08-23 19:23 |
| | COINBASE (Newly Generated Coins) | → | 1MvYASoHjqynMaMnP7SBmenyEWiLsTqoU6 | 6.39861453 BTC |
| | | | OP_RETURN | 0.00000000 BTC |
| | | | OP_RETURN | 0.00000000 BTC |

Mining Reward is calculated as per block height.
50 BTC per block, halved every 210,000 blocks.

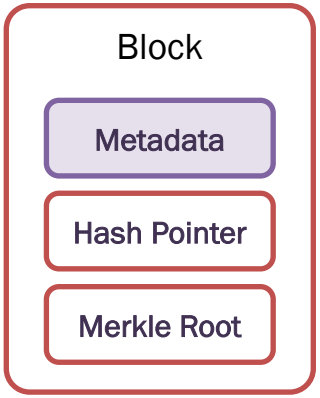
Transaction Fee = Sum (Inputs) – Sum (Outputs)

ref : https://www.blockchain.com/btc/tx/8785cfd428b67e6f4419db7fb4529eb1216fd936476ceed888f77daaec63fd64

# Block Header

Structure of Block Header (“metadata”) in a Block.

| Size | Field | Description |
|---|---|---|
| 4 bytes | Version | A version number to track software/protocol upgrades |
| 32 bytes | Previous Block Hash | A reference to the hash of the previous (parent) block in the chain |
| 32 bytes | Merkle Root | A hash of the root of the Merkle-Tree of this block’s transactions |
| 4 bytes | Timestamp | The approximate creation time of this block (seconds from Unix Epoch) |
| 4 bytes | Difficulty Target | The Proof-of-Work algorithm difficulty target for this block |
| 4 bytes | Nonce | A counter used for the Proof-of-Work algorithm |




Hash in context of Bitcoin means **SHA256( SHA256( ) )**.

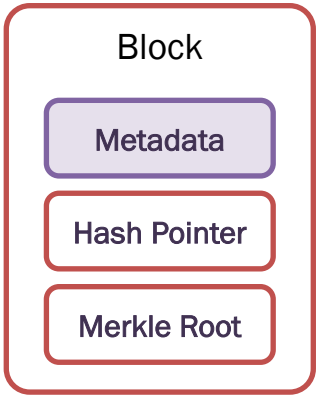
Difficulty and Nonce pertains to **Proof-of-Work** in mining.

ref : Chapter 8 of the book “Mastering Bitcoin”

# Reusable Proof-of-Work (RPoW) [2]

**Mining Nodes need to solve the following puzzle to Mine.**

- Choose random nonce in the Block Header (metadata).
- Hash the block and check if Hash(Block) < target value.
- If so, broadcast the block with that specific nonce value.
- If not, change the value of nonce in header to try again.




Food for thought …

Is it possible to predict which nonce may solve the puzzle?

Is it possible to use solution for one block to solve another?

[2] reading : Chapter 8 of the book “Mastering Bitcoin”

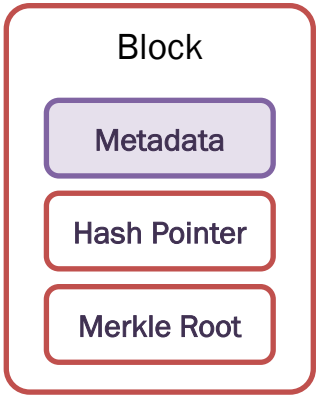
# Difficulty in Proof-of-Work

## Block 644984

Hash 00000000000000000000906cc2122074a
dc6c7a5dda178e050626c4102e2ef685

Nonce 654,975,388

Difficulty 16,947,802,333,946.61




Successfully mining a block requires **multiple trials**.
However, verifying a correct Nonce is **constant time**.

Difficulty is re-adjusted every **2016 blocks**, so that the expected time to mine a block is **10 minutes**.

ref : https://www.blockchain.com/btc/block/644984

Bitcoin Consensus

# Consistency and Forks

# Bitcoin Consensus

Depends on a few processes occurring independently across the P2P network.

- **Transaction Validation** : Verification of each transaction by every Full Node
- **Transaction Aggregation** : Recording transactions into blocks by Mining Nodes
- **Proof-of-Work** : Proof of demonstrated computation (mining) by Mining Nodes
- **Block Validation** : Verification of new blocks by every node with the Blockchain
- **Chain Selection** : Selection of “longest” chain with highest total Proof-of-Work

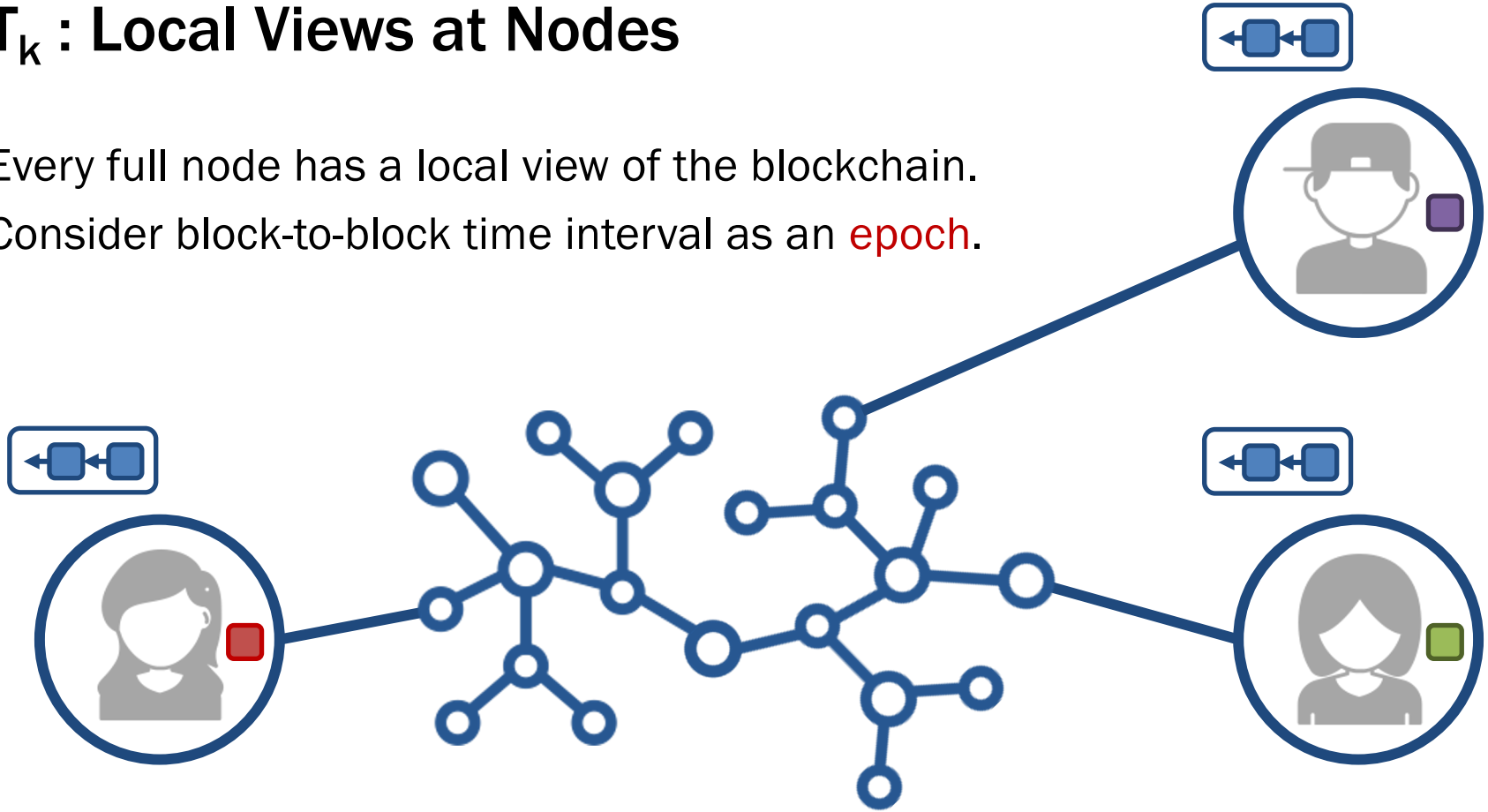
Block Creation and Mining is a local process to generate valid Proof-of-Work.
Block Validation is also a local process to verify the Block and Proof-of-Work.

**Chain Selection considers a global view of Consensus in the bitcoin network.**

# $T_k$ : Local Views at Nodes

Every full node has a local view of the blockchain.

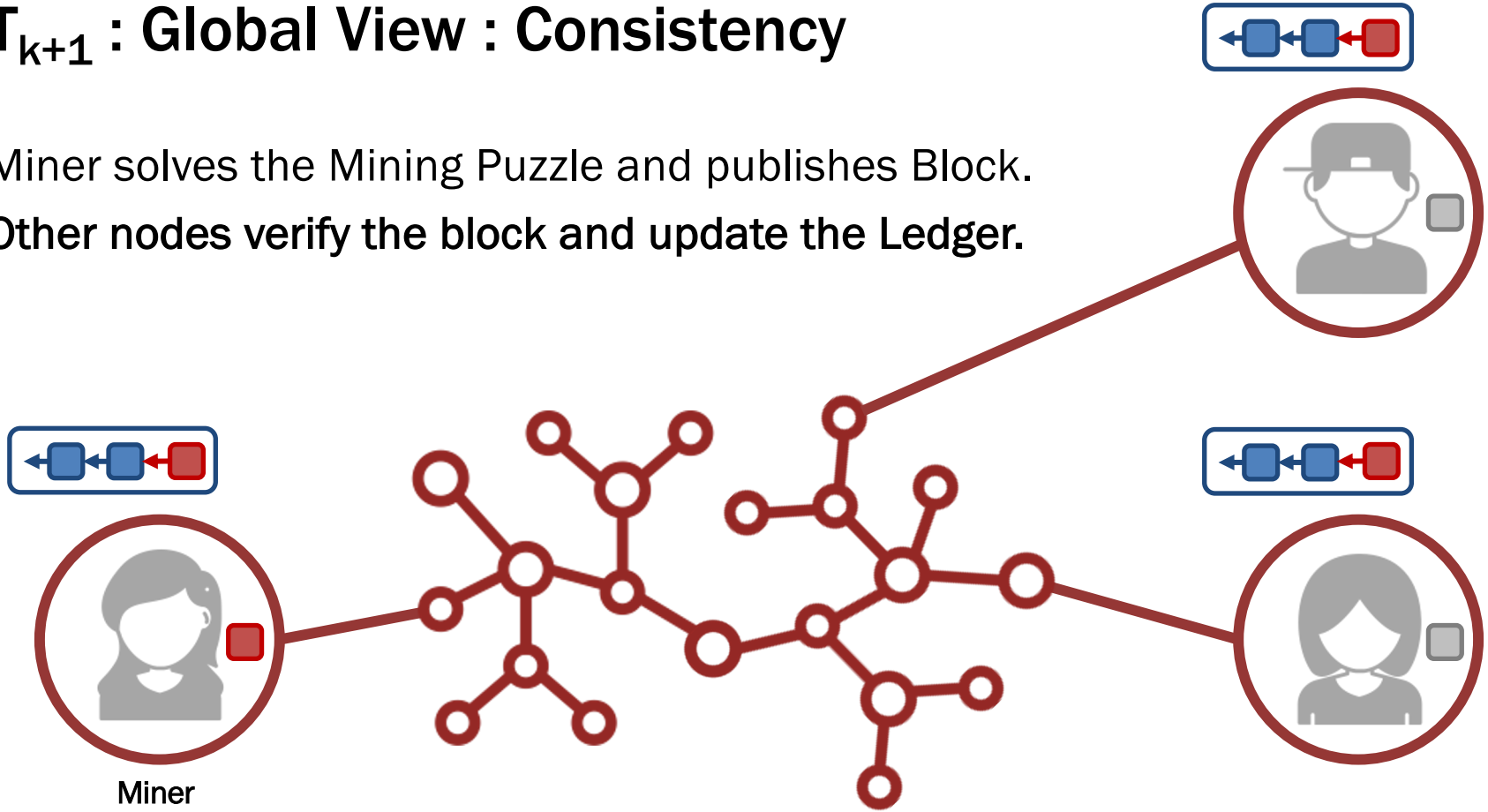
Consider block-to-block time interval as an epoch.



# $T_{k+1}$ : Global View : Consistency

Miner solves the Mining Puzzle and publishes Block.

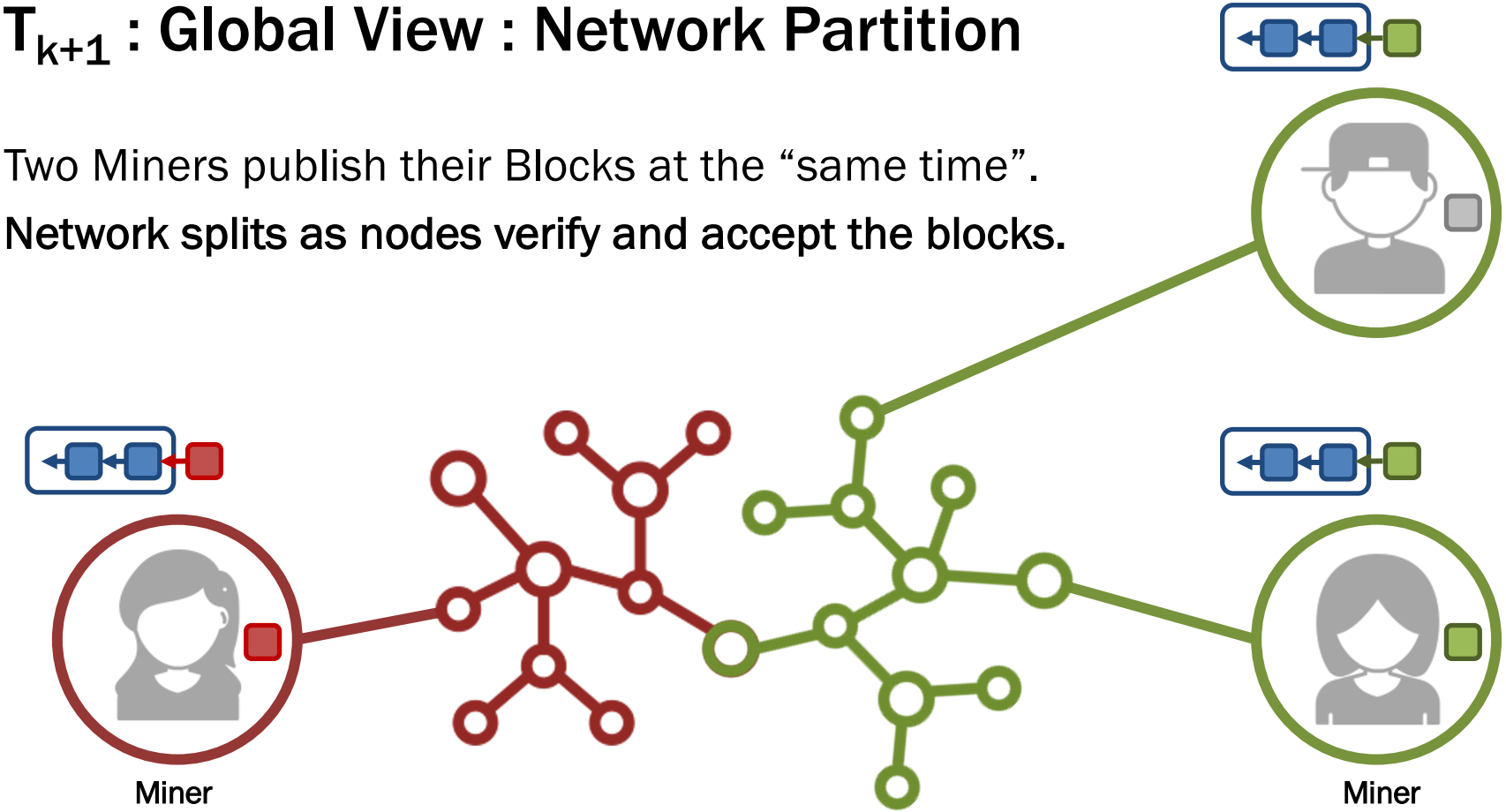
**Other nodes verify the block and update the Ledger.**



# $T_{k+1}$ : Global View : Network Partition

Two Miners publish their Blocks at the “same time”.

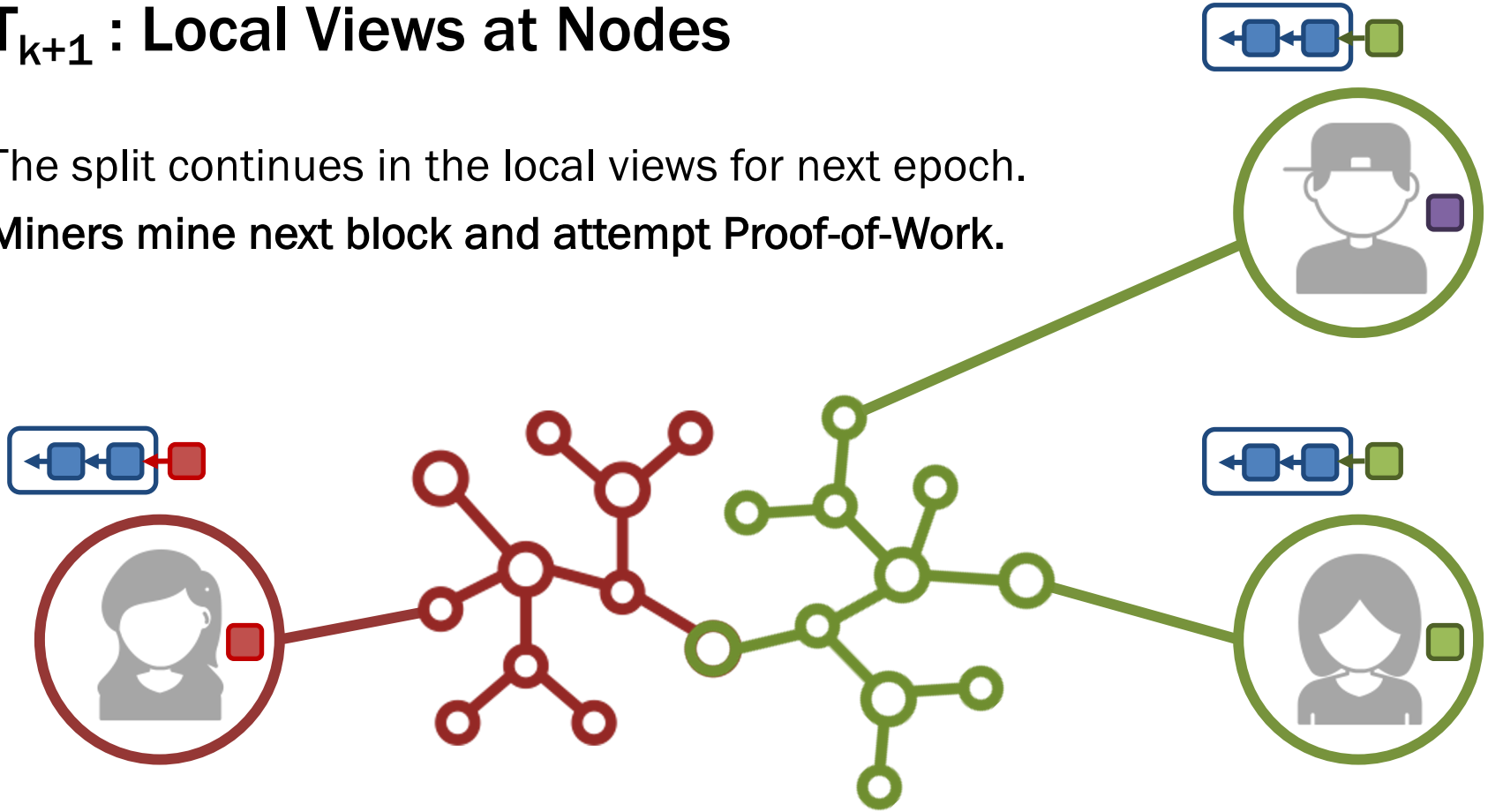
**Network splits as nodes verify and accept the blocks.**



# $T_{k+1}$ : Local Views at Nodes

The split continues in the local views for next epoch.

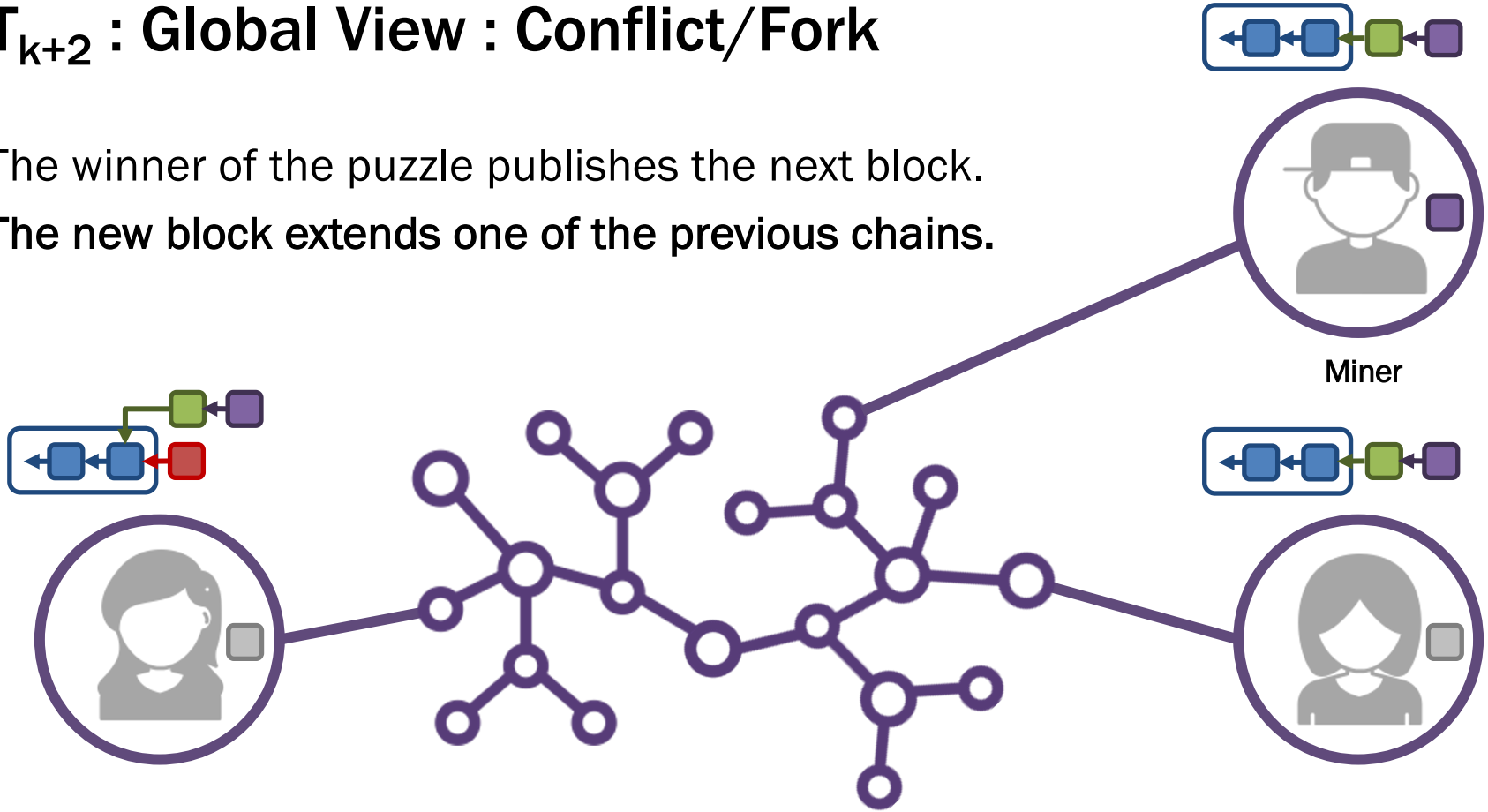
**Miners mine next block and attempt Proof-of-Work.**



# $T_{k+2}$ : Global View : Conflict/Fork

The winner of the puzzle publishes the next block.

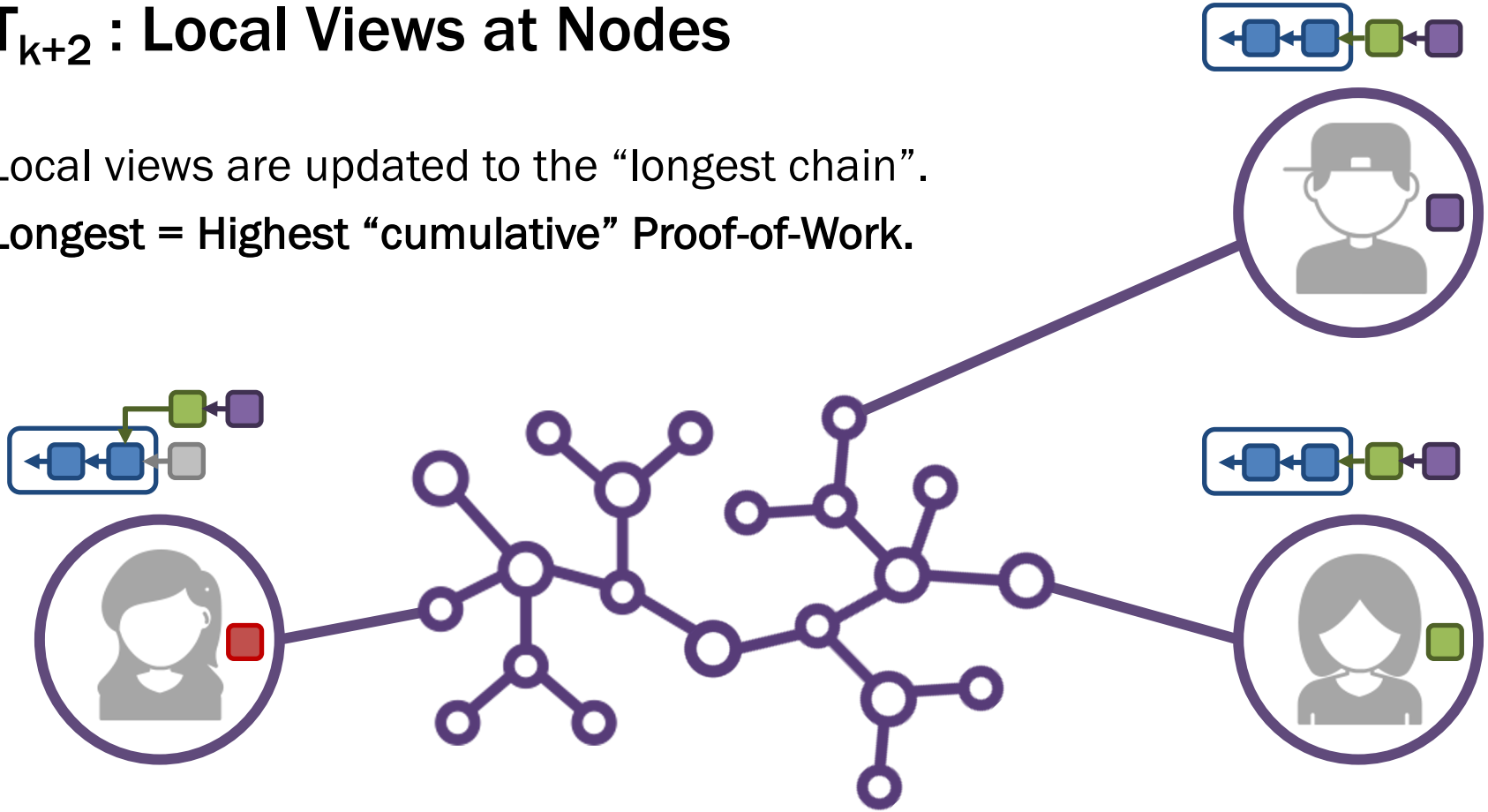
**The new block extends one of the previous chains.**



# $T_{k+2}$ : Local Views at Nodes

Local views are updated to the “longest chain”.

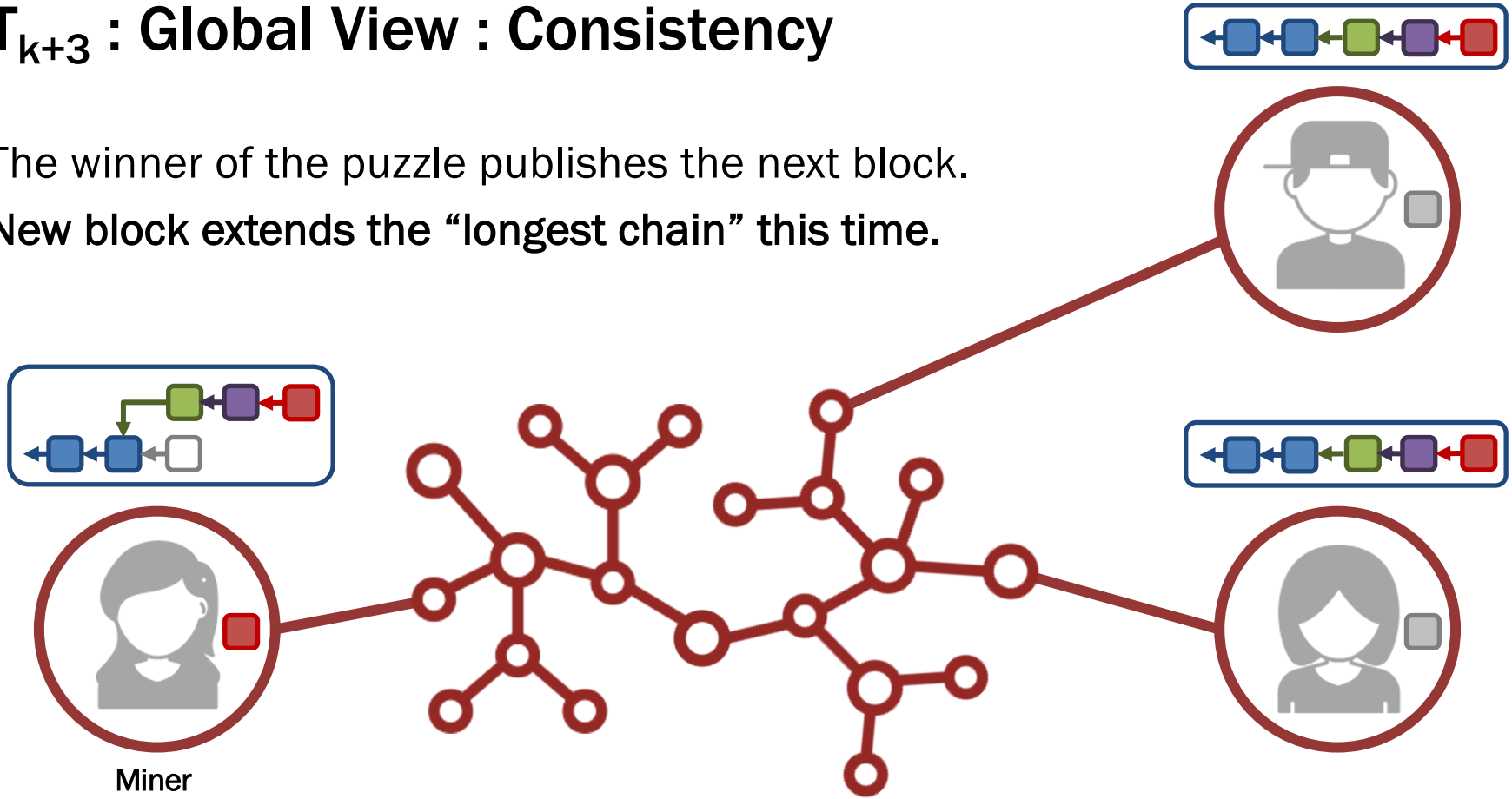
**Longest = Highest “cumulative” Proof-of-Work.**



# $T_{k+3}$ : Global View : Consistency

The winner of the puzzle publishes the next block.

**New block extends the “longest chain” this time.**



# Eventual Consensus

Bitcoin consensus designed by Nakamoto considers several issues.

- **Lack of Identity** : No long-term fixed or authentic identity for any bitcoin node
- **Identity Duplication** : Nodes may assume any number of identities in bitcoin
- **Leader vs Winner** : Leader “election” is impossible but “winning” is possible
- **Proof of Winning** : Bound to computational power rather than identity voting
- **Distributed Voting** : Implicit “voting” for blocks by extending respective chain

The core motivation for honesty is built on “incentive design” for mining.

**Therefore, consensus in bitcoin works better in practice than in theory!**