Decentralized Systems

Security in an untrusted environment

Michael L Perry
qedcode.com
@michaellperry

pluralsight
hardcore dev and IT training

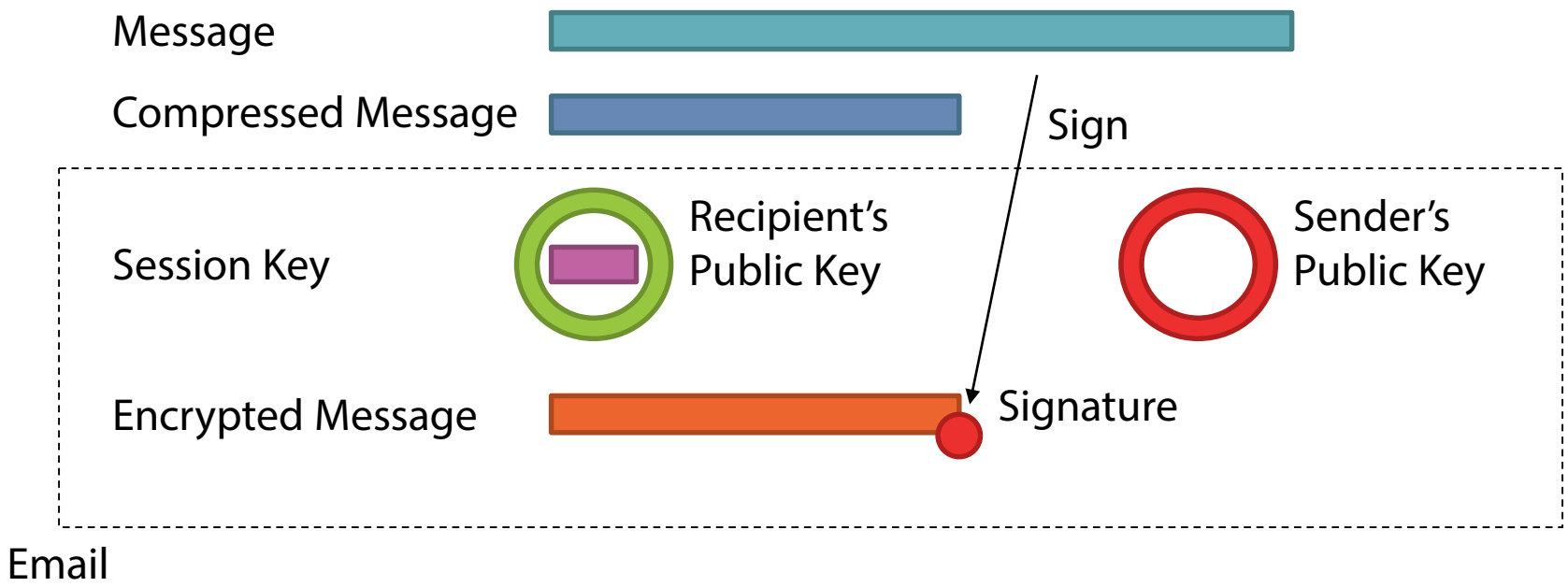

Trust me!

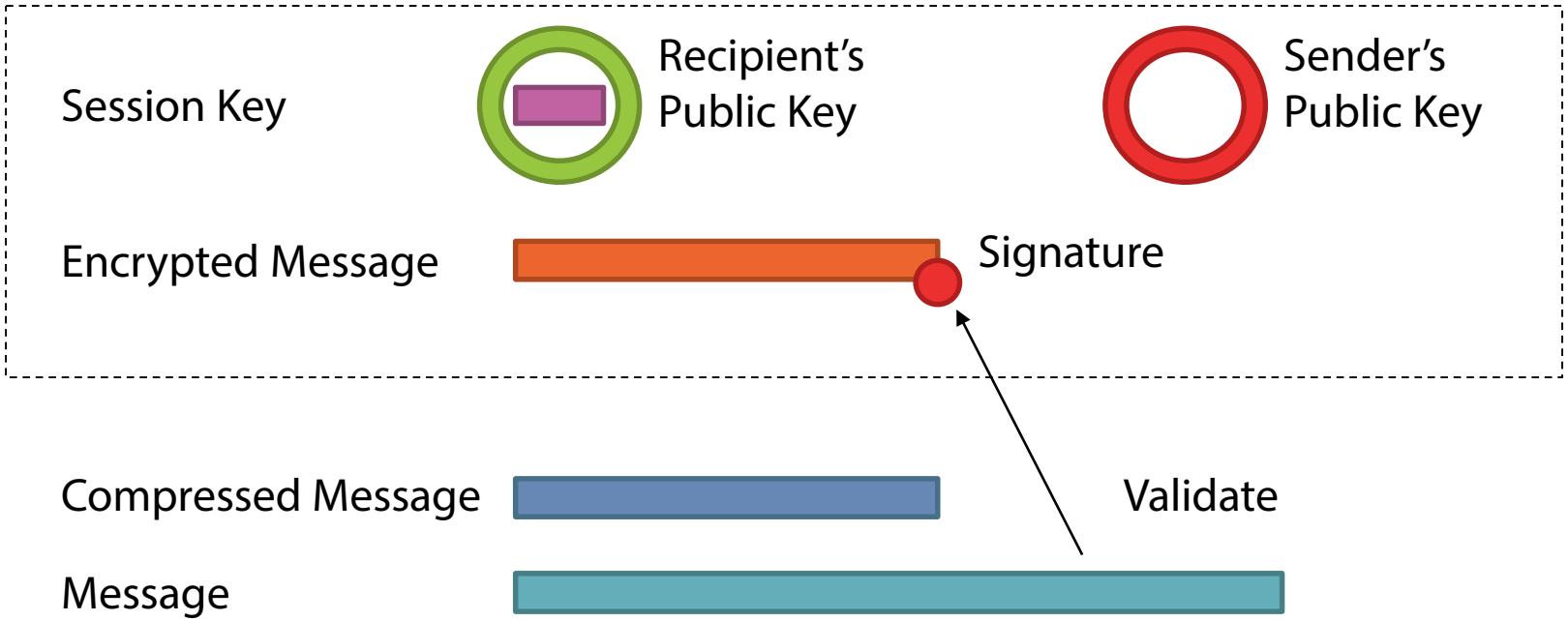
PGP (Pretty Good Privacy)

Phil Zimmerman

1991

Secure email exchange


Message
Compressed Message
Sign
Session Key
Recipient's
Public Key
Sender's
Public Key
Encrypted Message
Signature
Email


Session Key
Recipient's Public Key
Sender's Public Key
Encrypted Message
Signature
Compressed Message
Validate
Message

PGP

Communicate through email

with confidentiality

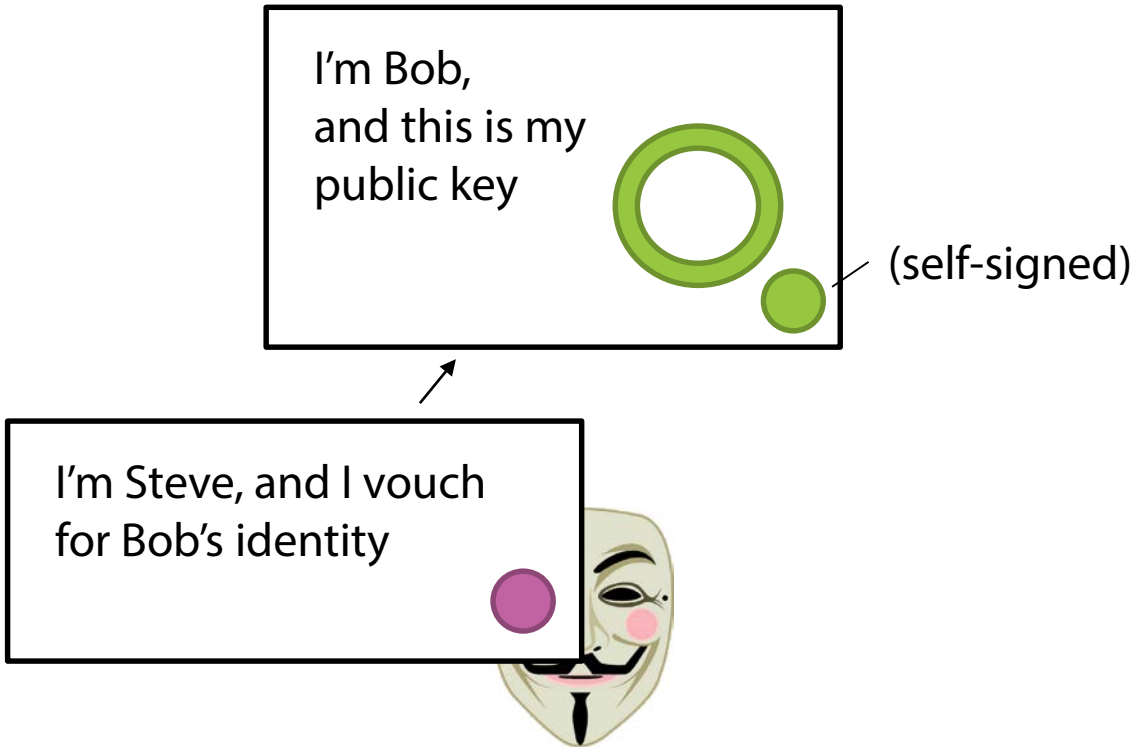
and authenticity

Need to Know Public Key

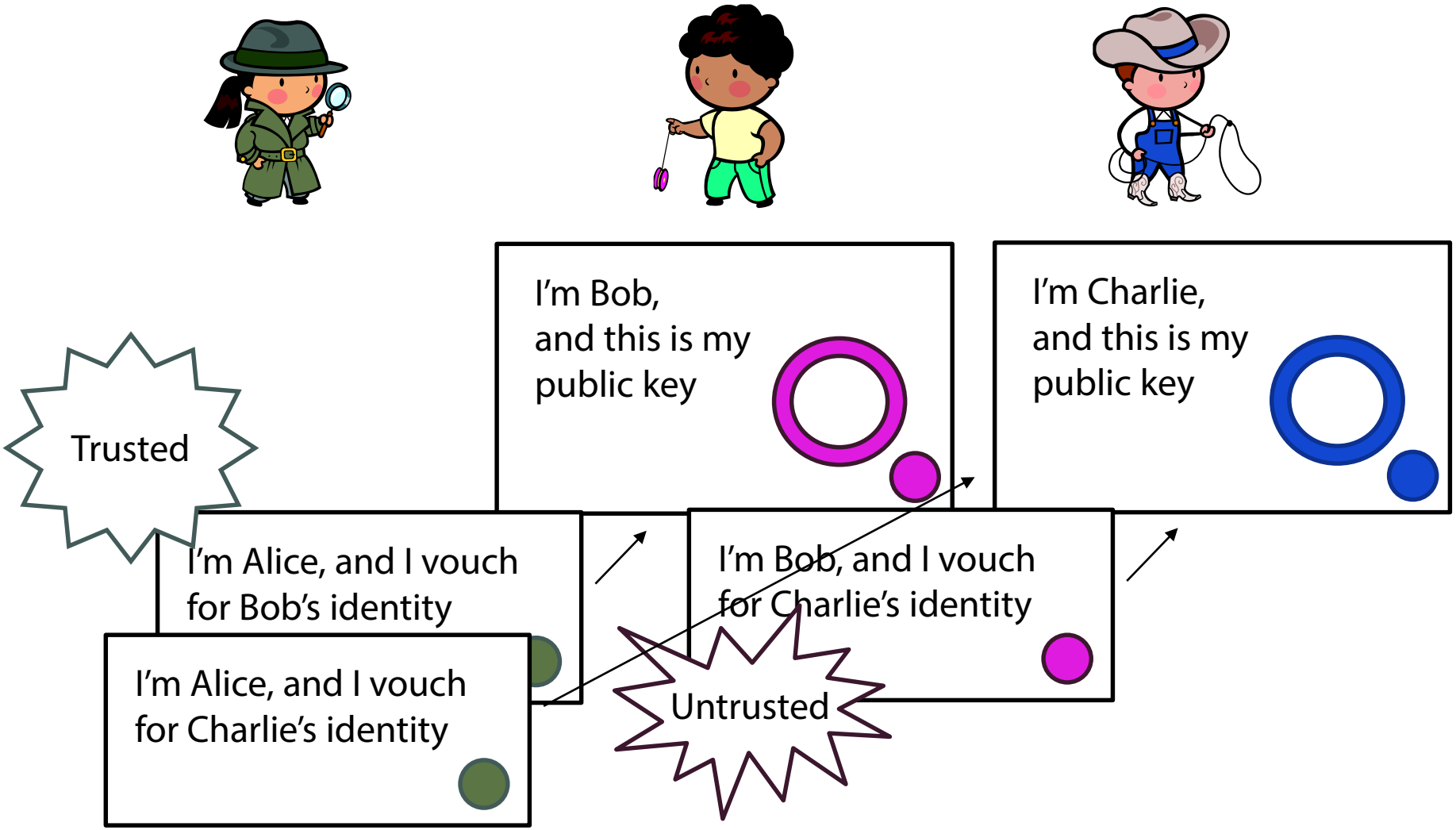
- **Direct**
  - USB thumb drive
  - Not always possible
- **Indirect**
  - Electronic means
  - Is it really their public key?

Ask Them Questions

- **Only they would know the answer**
  - You need to know the answer, too! (Shared secret)
- **Man in the middle attack**
  - Email is intercepted
  - Attacker sends their own public key
  - Shuttle questions and answers


I'm Bob,
and this is my
public key
(self-signed)
I'm Steve, and I vouch
for Bob's identity

Web of Trust


I'm Bob,
and this is my
public key
I'm Charlie,
and this is my
public key
Trusted
I'm Alice, and I vouch
for Bob's identity
I'm Bob, and I vouch
for Charlie's identity
I'm Alice, and I vouch
for Charlie's identity
Untrusted

Comparison with TLS

PGP	TLS
Public Keys	X.509 Certificates
Signed by Individuals	Signed by a CA
Asymmetric Cryptography	Asymmetric Cryptography
Web of Trust	Chain of Trust

Hashcash

Adam Back

Prevent Spam

Spam

- **Cheap to send email**
- **Need to send a lot**
- **Apply a cost, and it no longer works**
- **Cost multiplies for unsolicited email**
  - Cost in CPU cycles

michael@qedcode.com 20140419 10746251943

4751566e9379079fbc99b21ccfef5570b8e00902

Proof of Work

michael@qedcode.com 20140419 19646101417

00000f7ae7cb7e787acb429bd844895e044f5657

Amount of Work

$$2^{20} = 1{,}048{,}576$$

times the number of recipients

Proof of Work

- Not widely used for spam prevention
- Central to cryptocurrencies, like

Bitcoin

Bitcoin

Satoshi Nakamoto

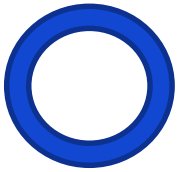
2009

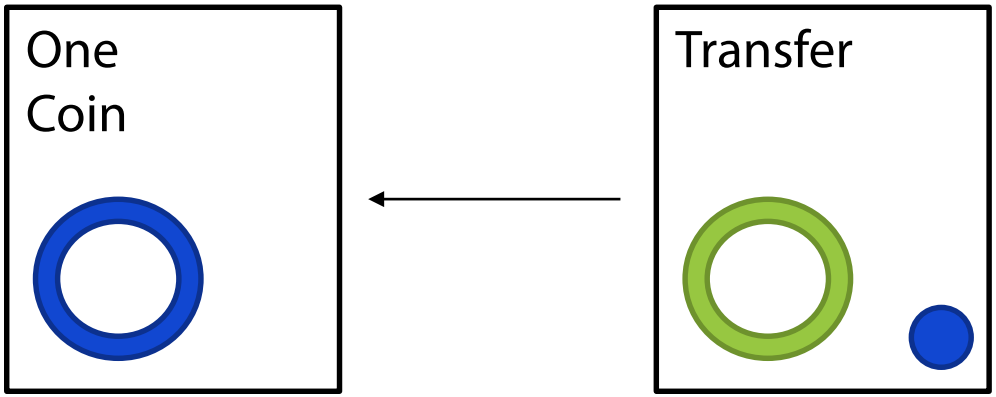
Distributed Secure Money Exchange

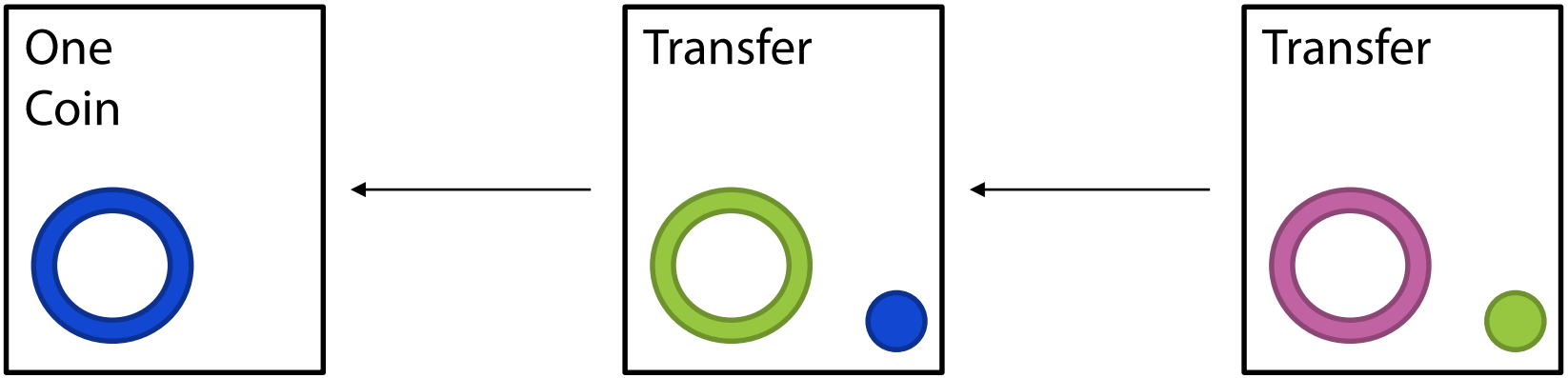
Double-Spending Problem

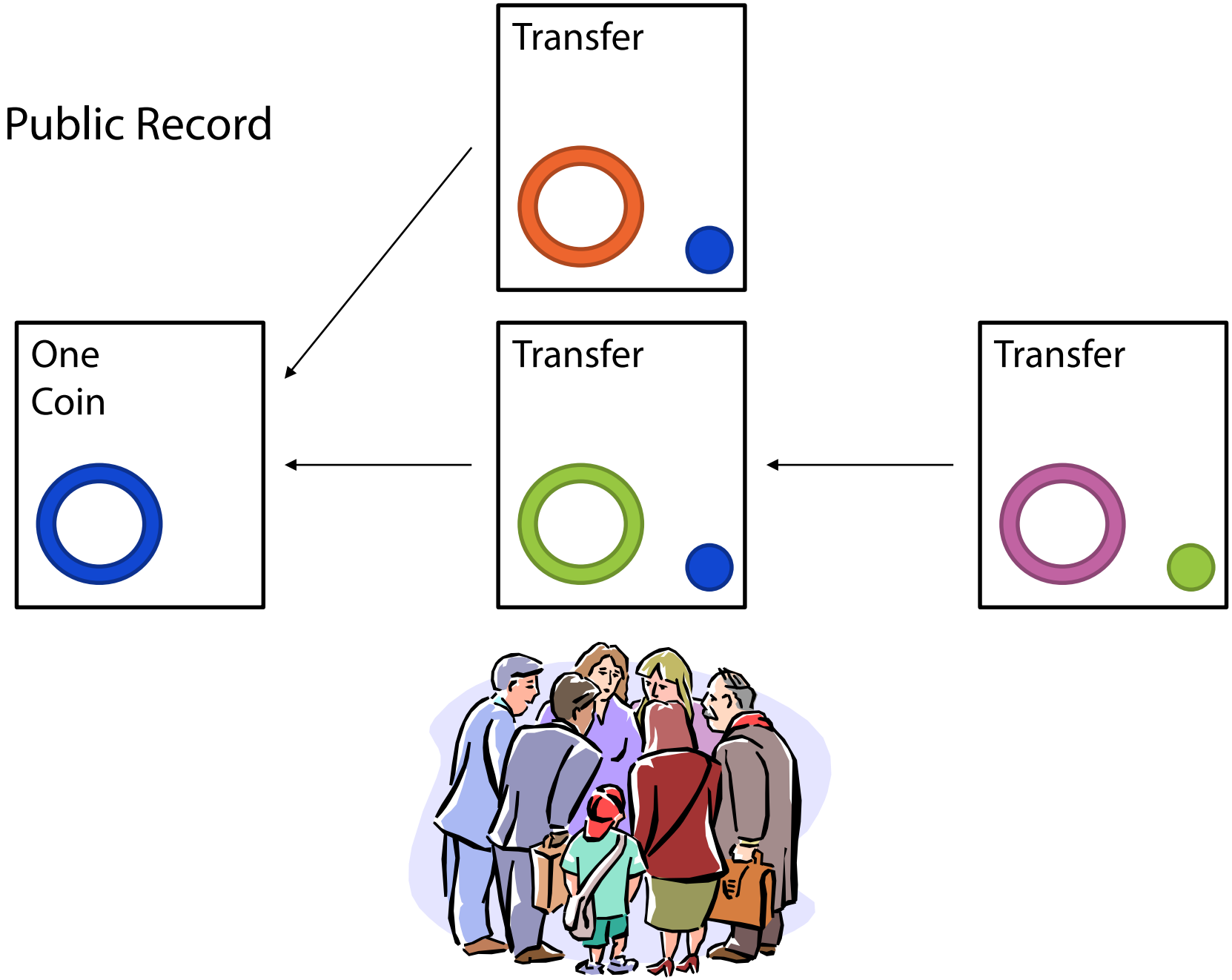
- **Spend your money and keep it too**
- **Central Authority**
  - Bank
- **Bitcoin is Decentralized**

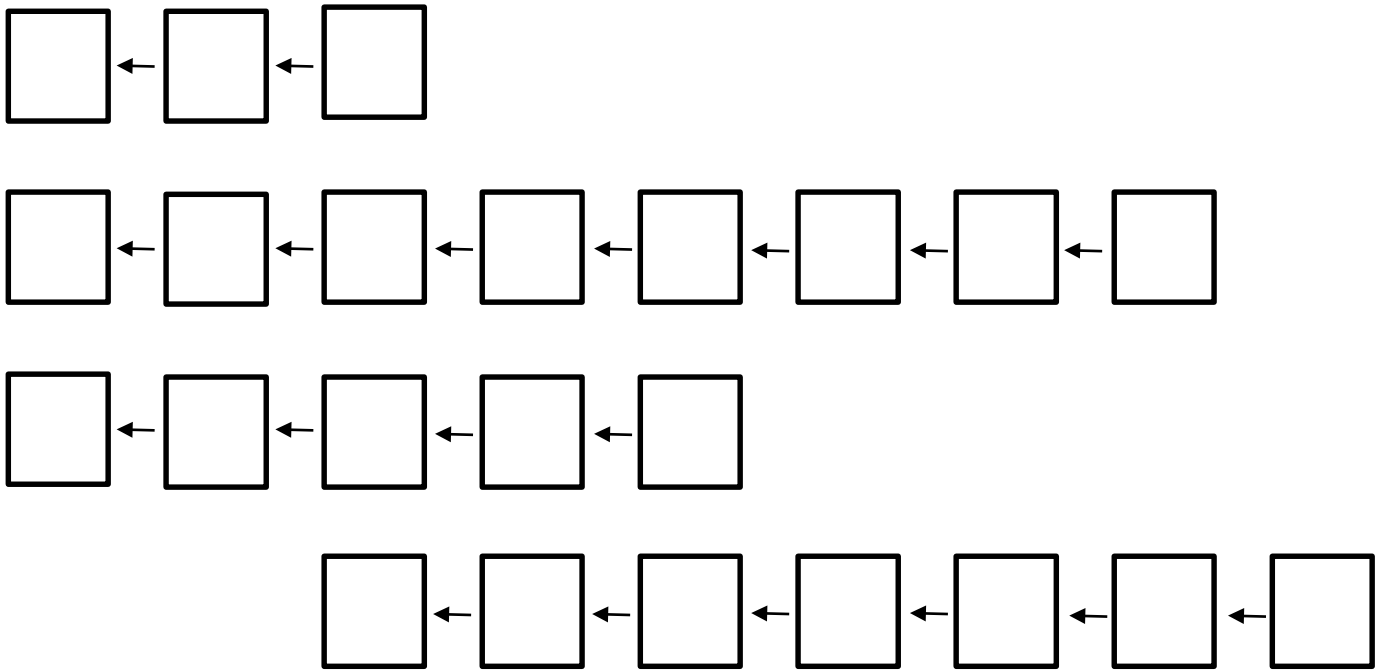
One
Coin

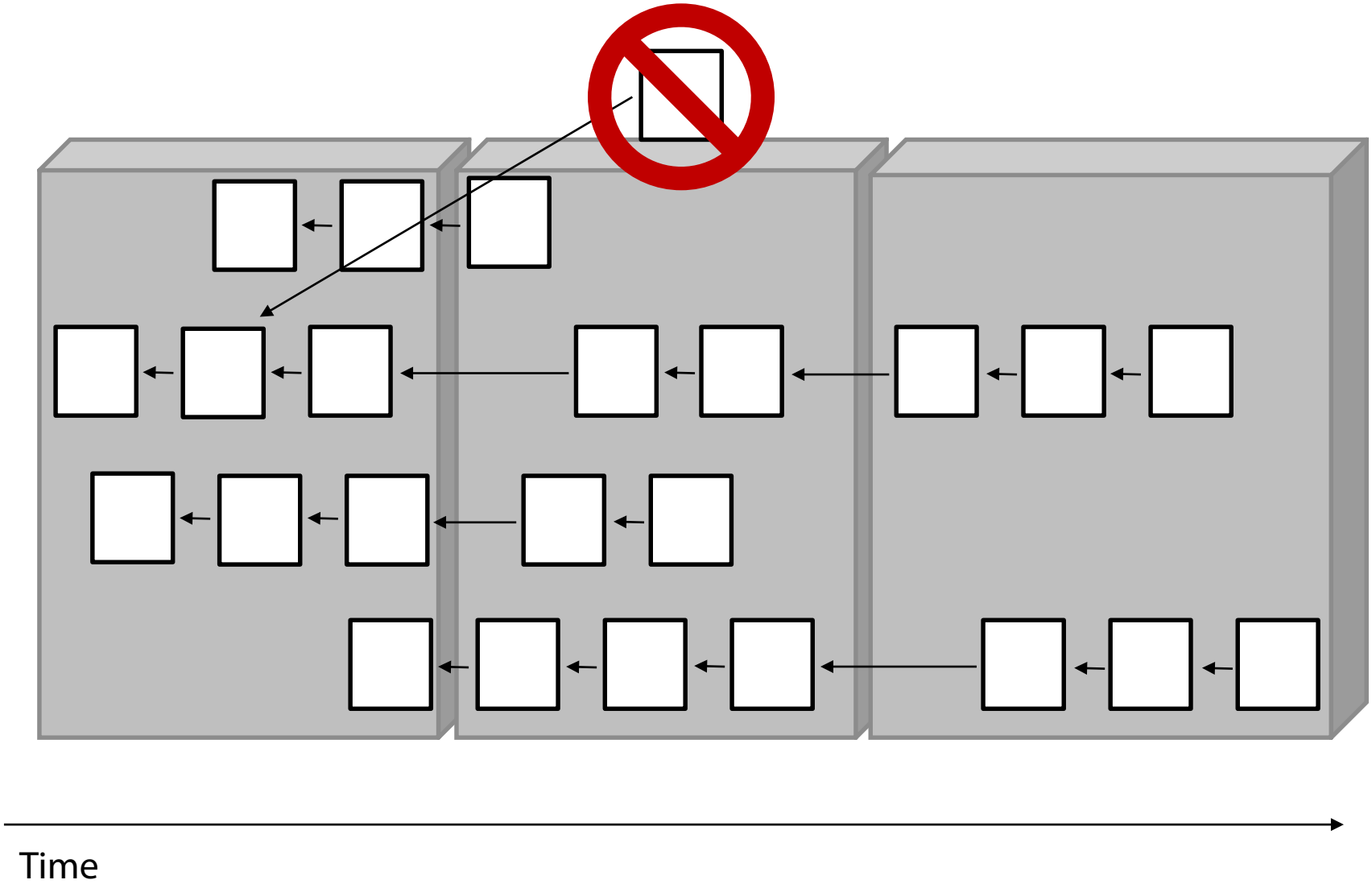


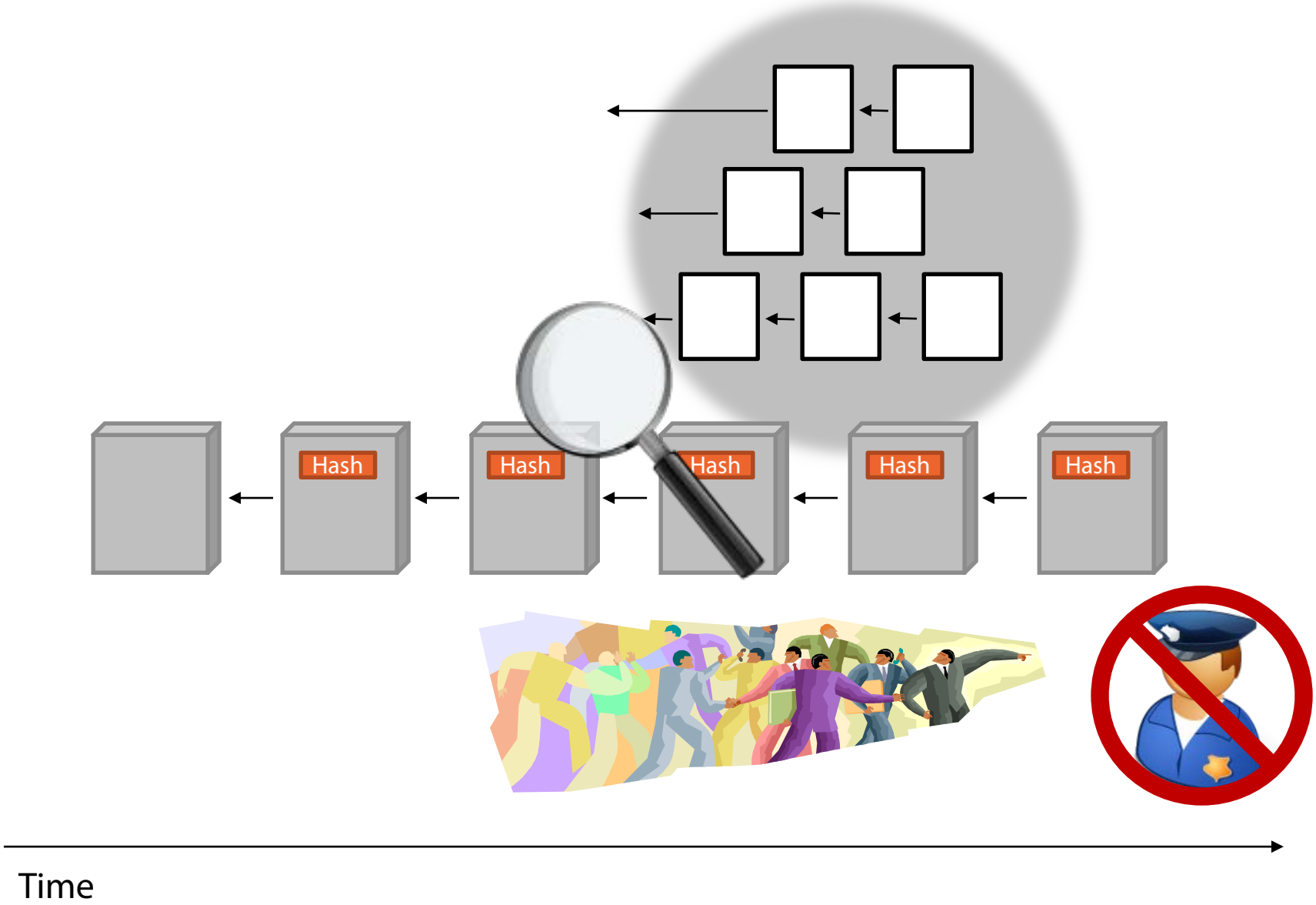

One
Coin
Transfer


One
Coin
Transfer
Transfer


Public Record
Transfer
One
Coin
Transfer
Transfer

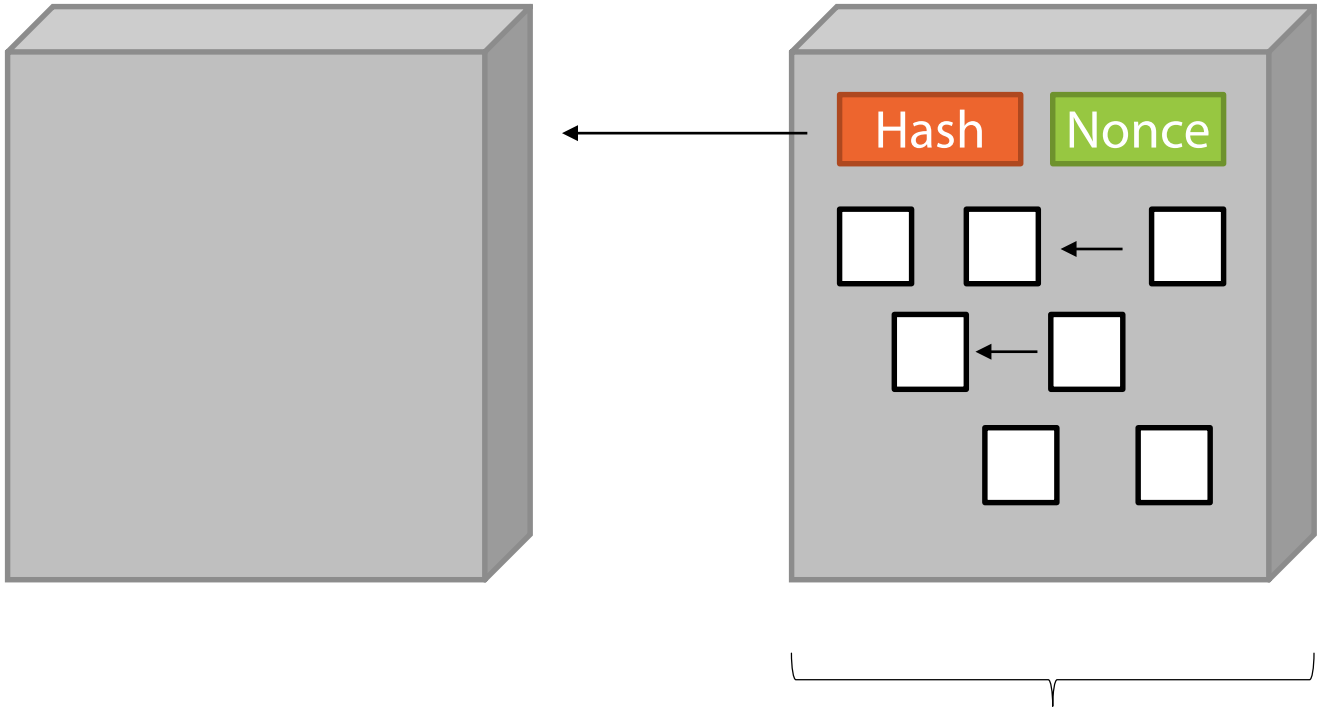



Time


Hash
Hash
Hash
Hash
Hash
Time

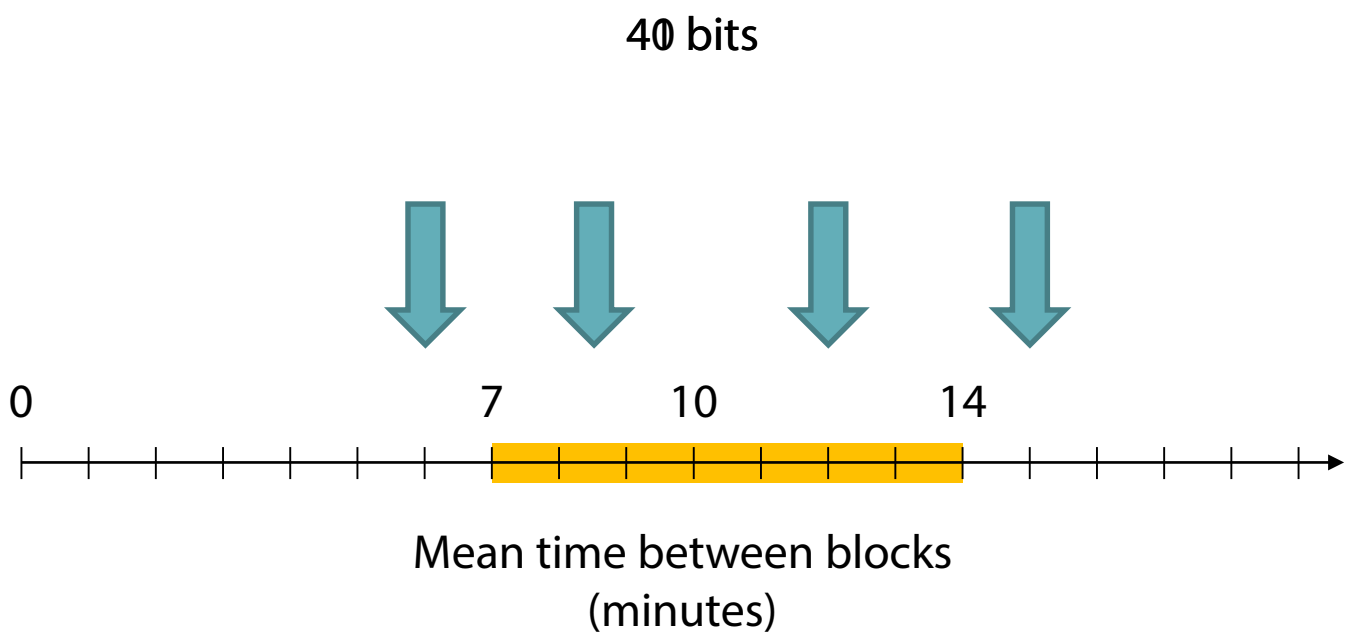
Convergence

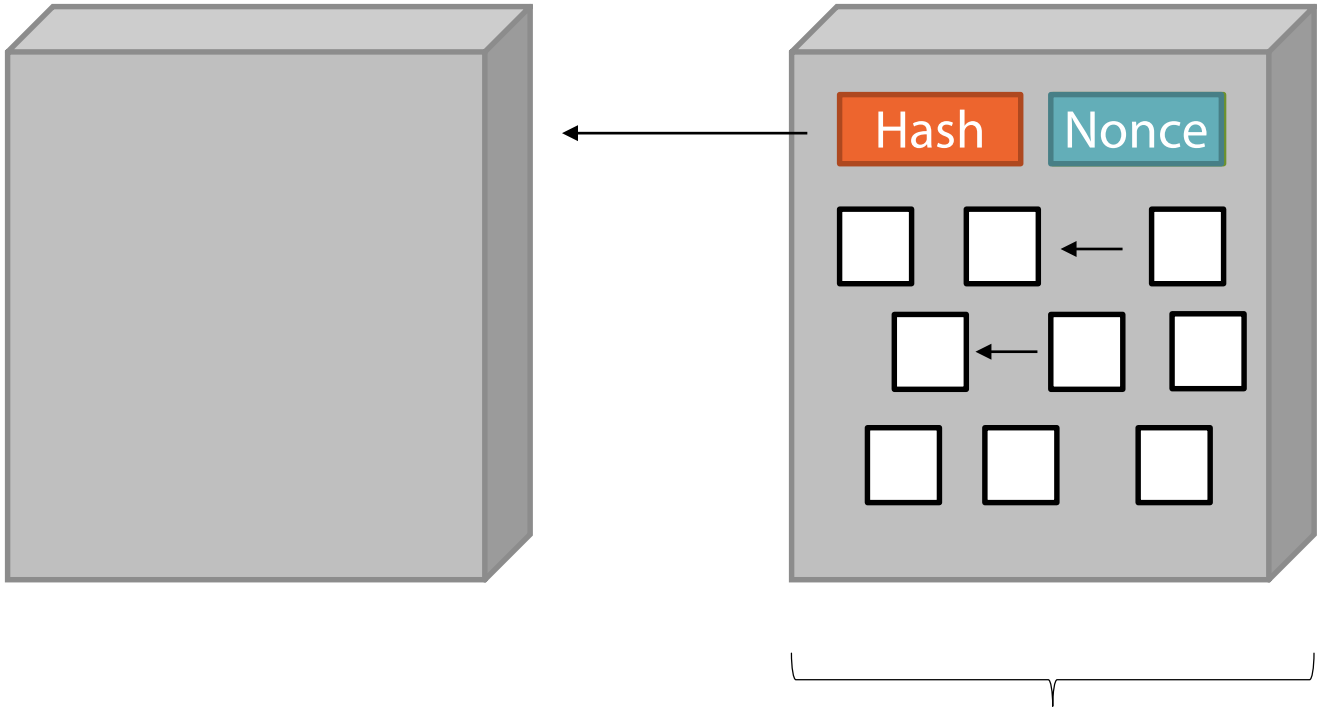
- **Impose a cost**
  - Proof of work



0000000000ed38a36779bd1ce5450f57df0e48c4

Adjusting the Proof of Work





2900413da0ed38a36779bd1ce5450f57df0e48c4

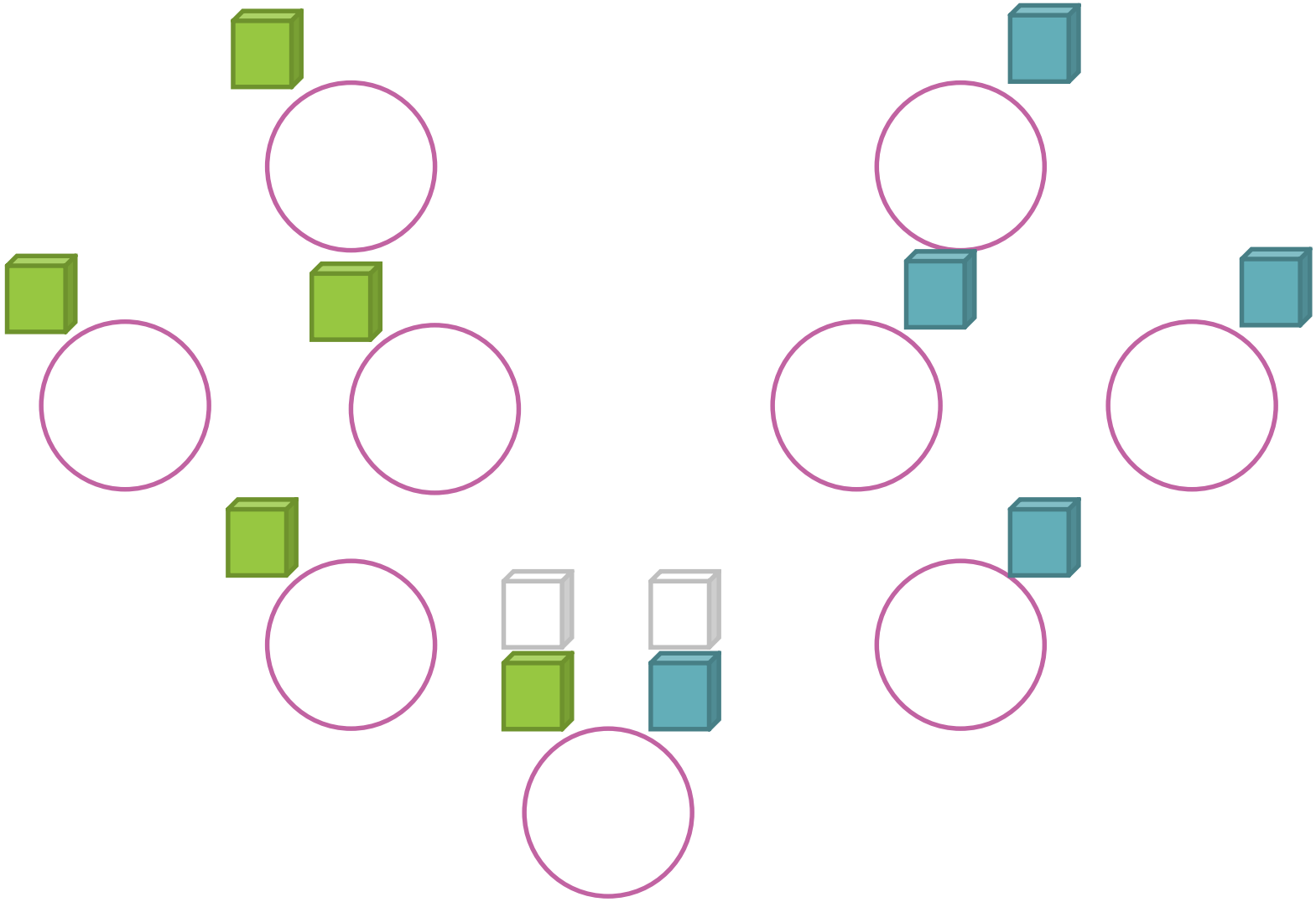
Validate a Block

- **The hash begins with enough leading zeroes**
- **Compare the hash of the previous block**
- **Look for double spending**
  - Set of transactions not signed over
  - If present, the entire block is invalid

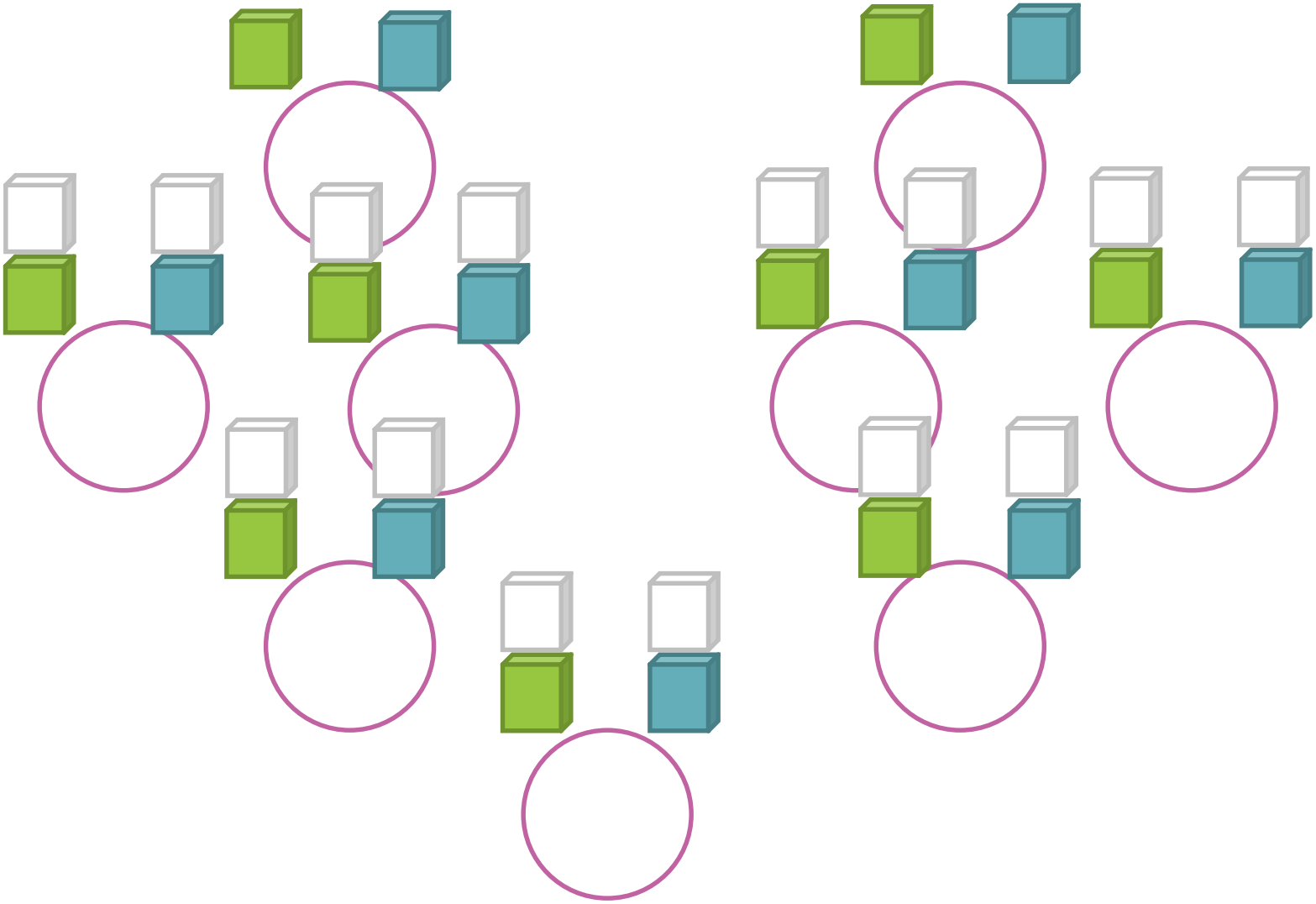
Process

- **Receive new transactions**
  - Forward to neighbors
  - Add to candidate block
- **Receive new blocks**
  - Validate
  - Forward to neighbors
  - Use as the basis for the next block

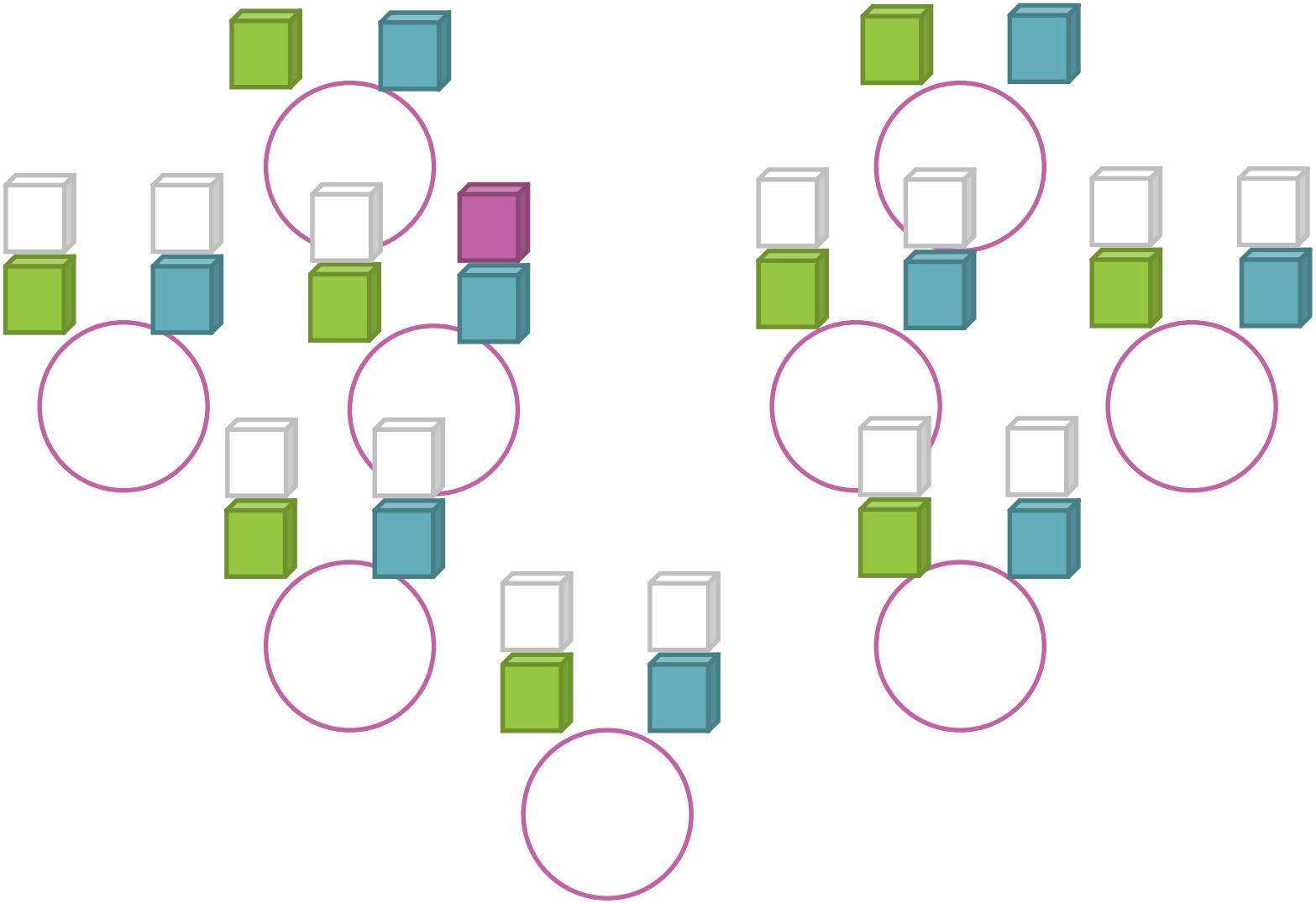
Conflict Resolution



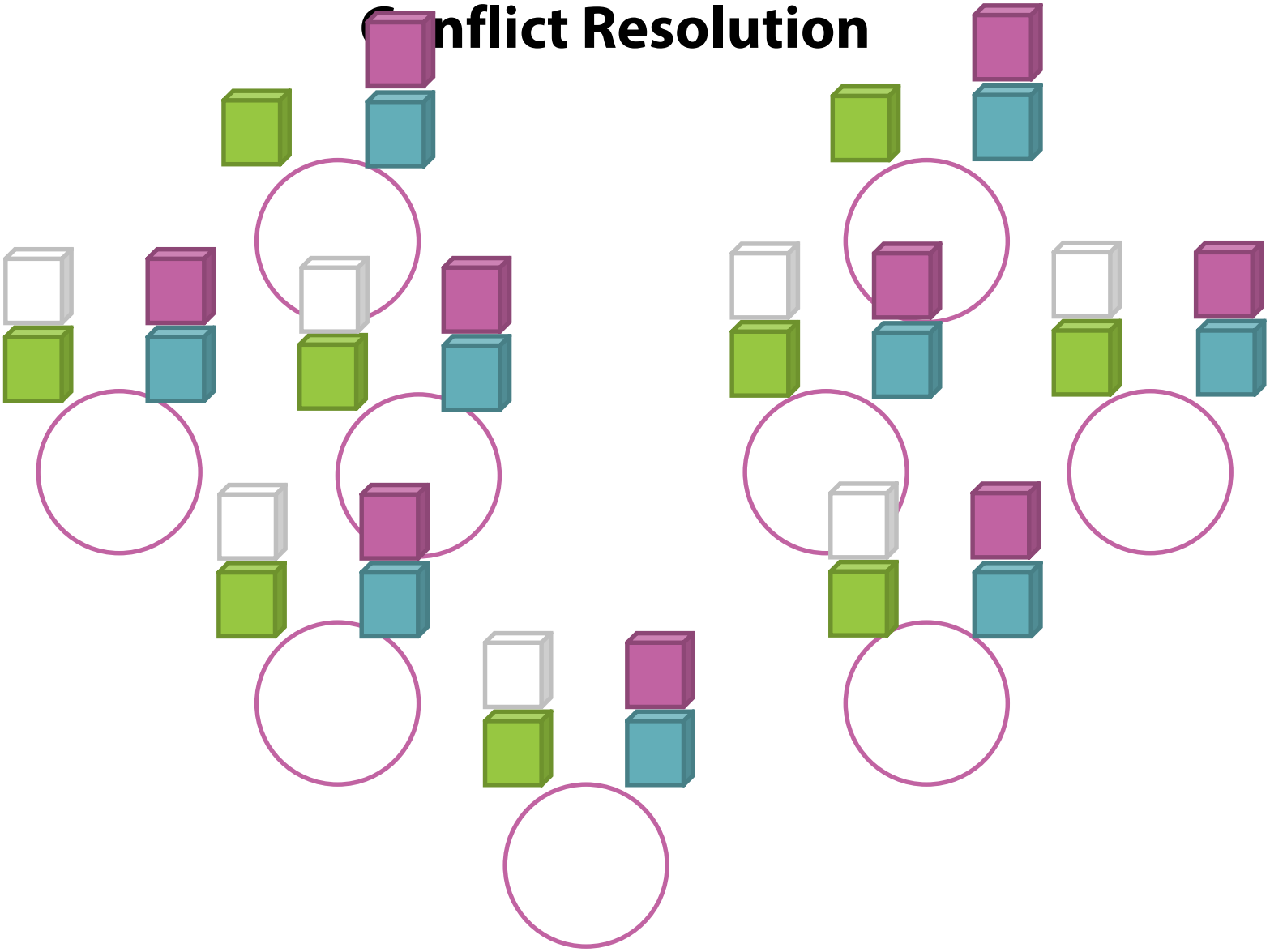
Conflict Resolution



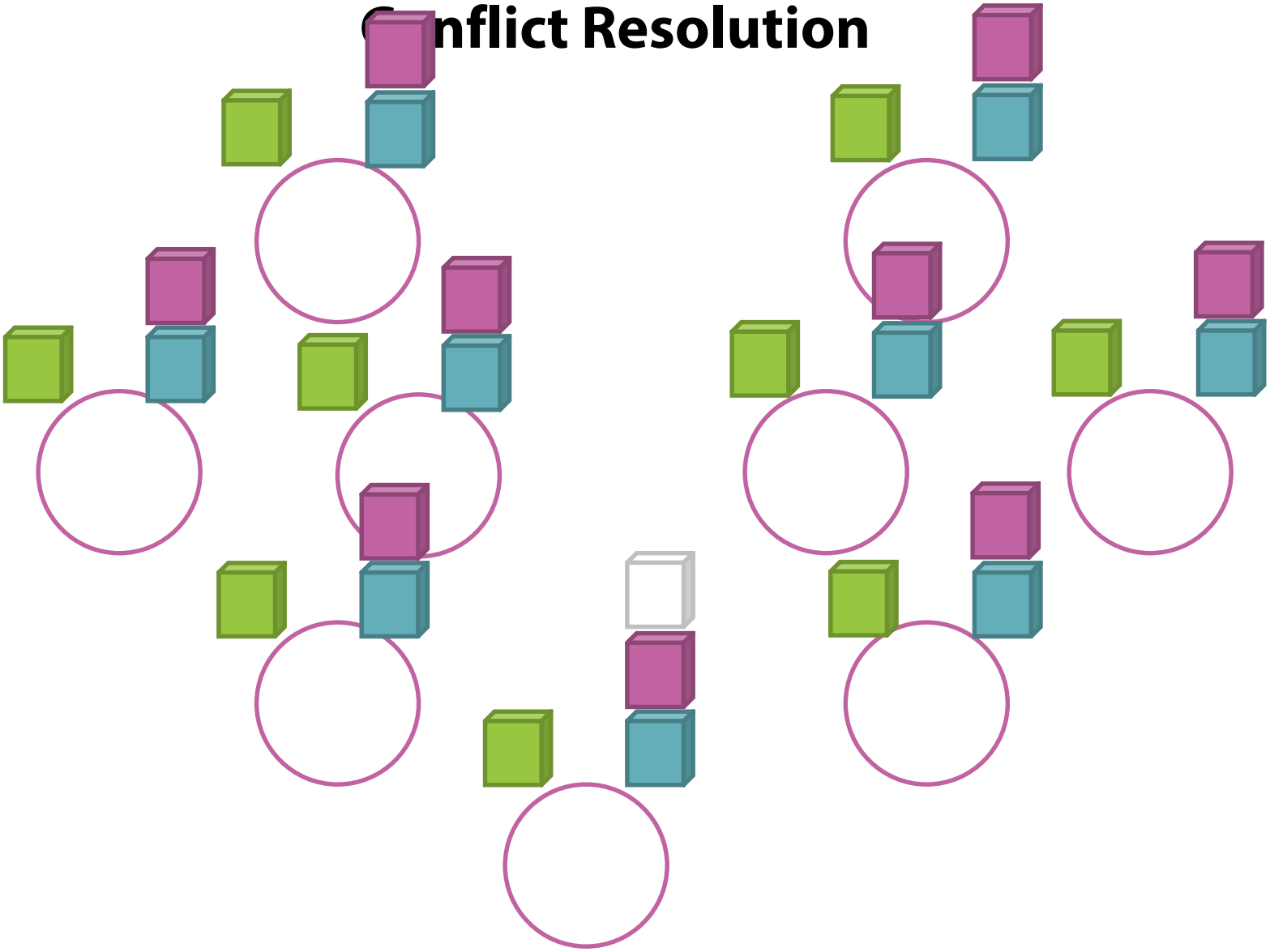
Conflict Resolution



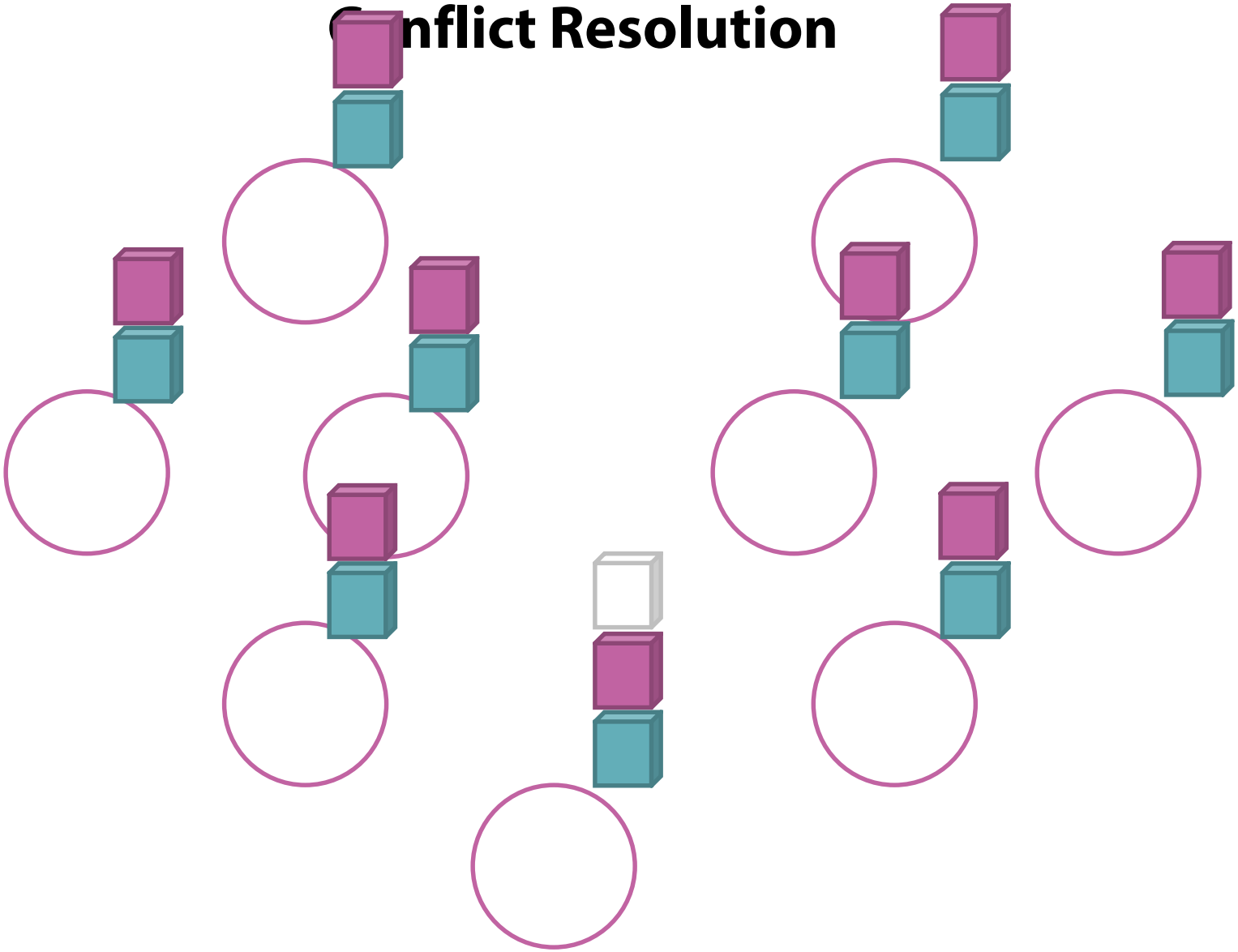
Conflict Resolution



Conflict Resolution



Conflict Resolution

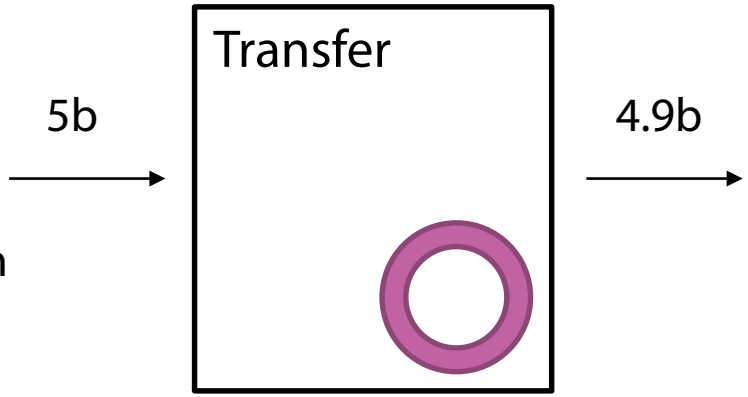


Race

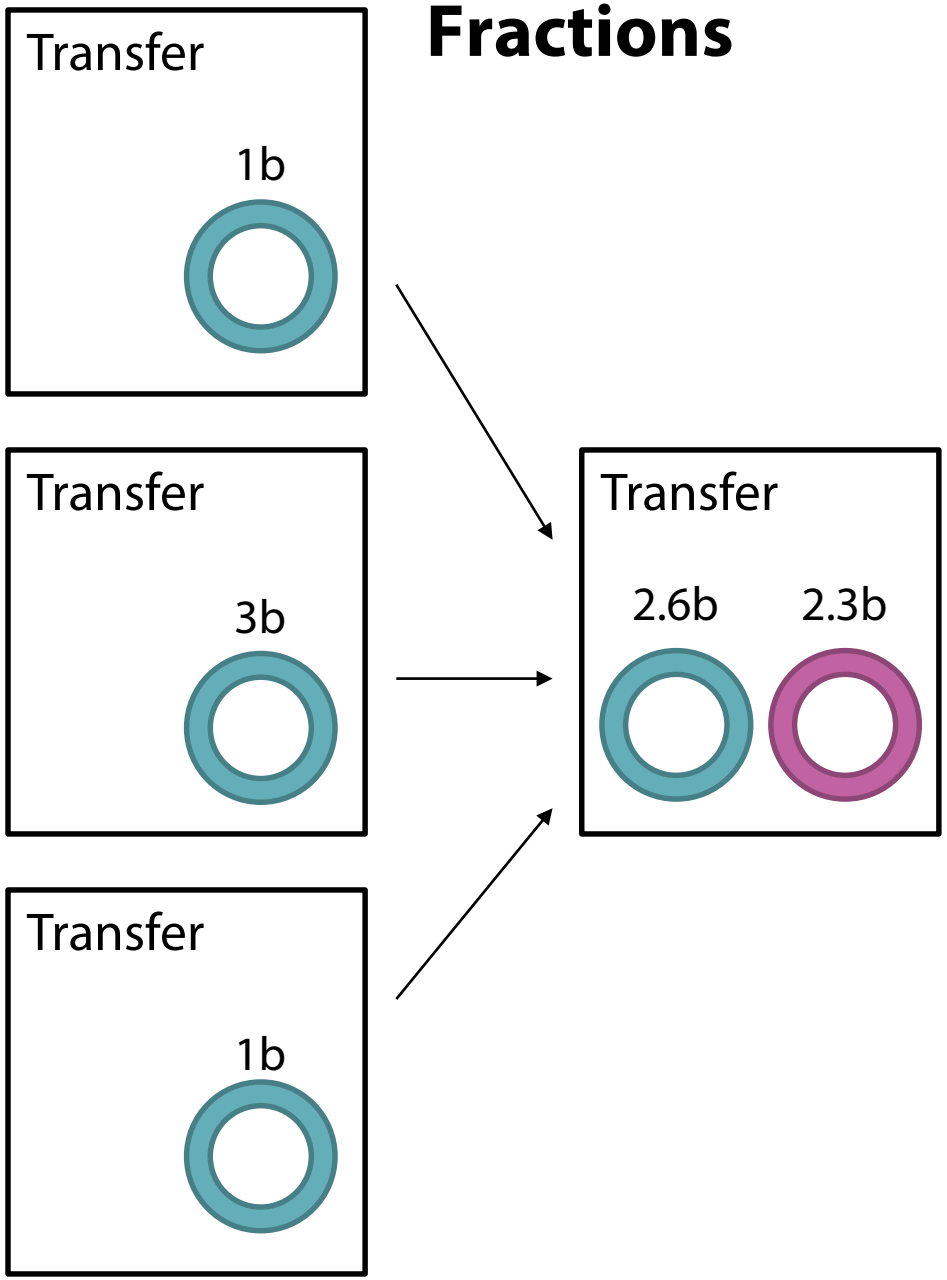
- **A block on the other chain is found**
- **Proof of work ensures**
  - Collisions are rare
  - Forks diverge

Incentives

- **First transaction creates a new bit coin**
  - Bit coin awarded to the node
  - Earn bitcoin
- **Transaction fees**
  - Fees awarded to the node
  - Encourages nodes to include transaction
- **Mining**



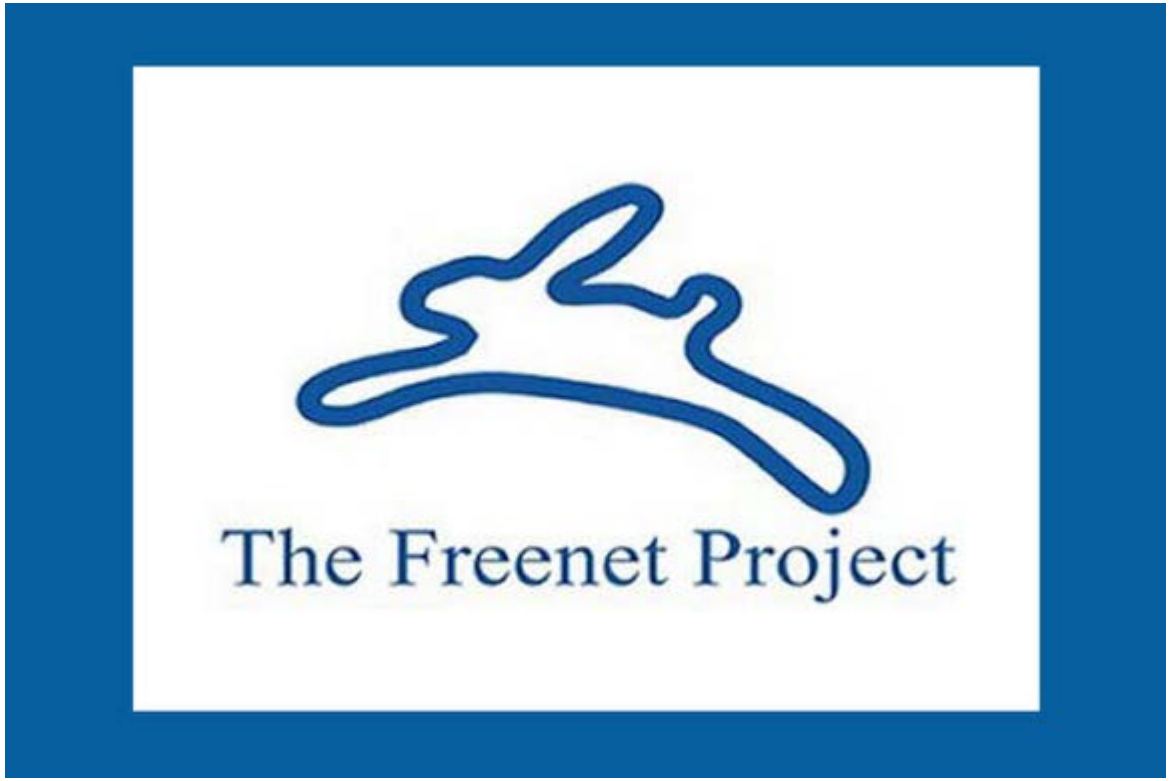
Fractions



Distributed Data Storage

- **Advantages**
  - Scalability
  - Redundancy
- **Cost**
  - Control

Freenet



Fight censorship

Provide plausible deniability

Peer-to-peer

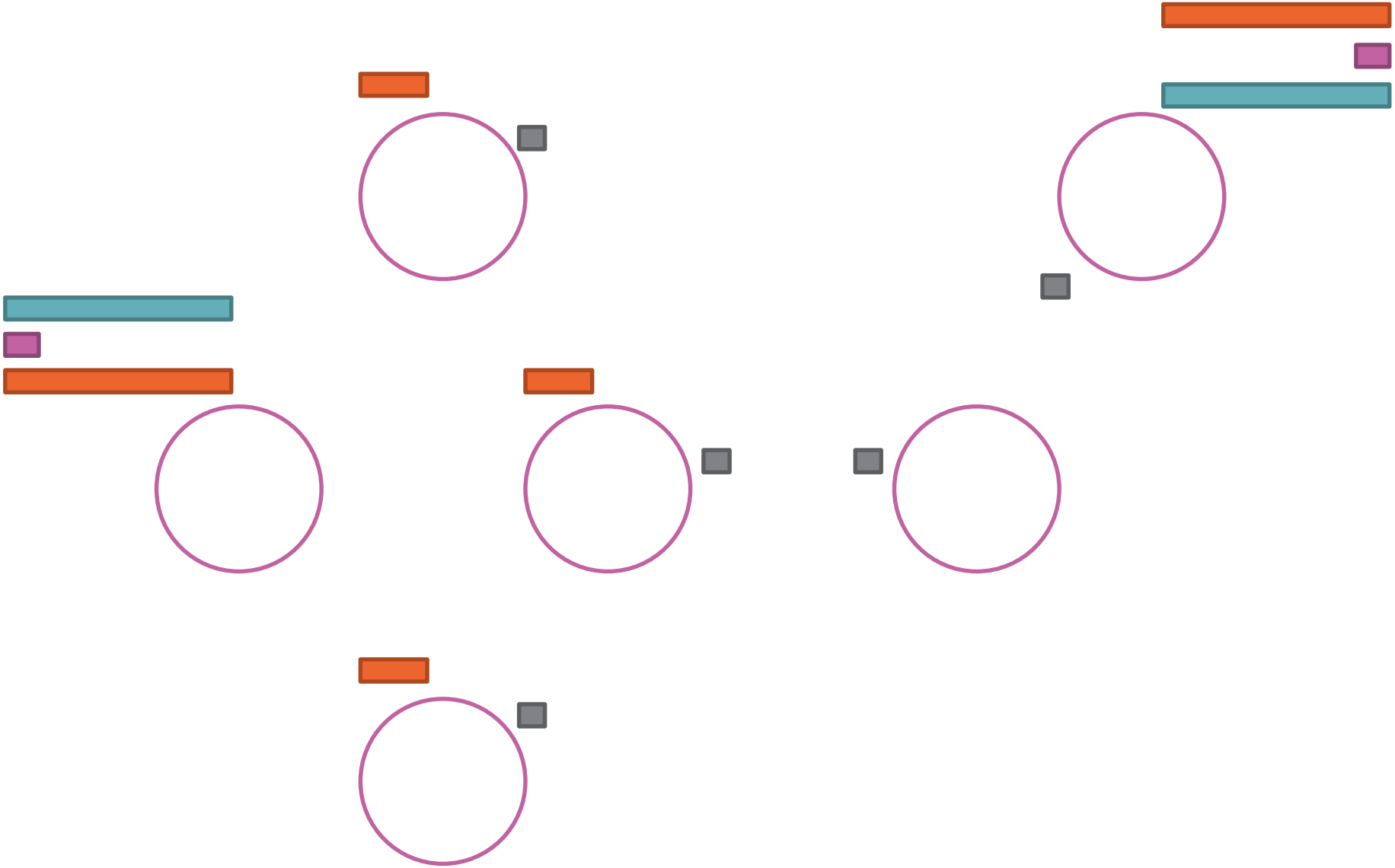
2000

Identify a File

- **By hash**
  - SHA-256
- **Advantages**
  - Recipient can validate that it hasn't been altered
  - Multiple versions have different identities
  - Cannot modify a file

Content Hash Key

CHK@	SVbD9...X5Brs,	bA7qL...6bbNQ,	AAEA--8
	file hash	symmetric key	algorithms



Nodes Cannot Compute Hash

- **Segment of file**
  - File is encrypted
  - Given hash and segment
- **Possible to forge on write**
  - Validity checked on read
- **Documents are immutable**
  - Publish hash and symmetric key
  - After published, cannot be updated

Have to Share Symmetric Keys

- **Cannot ensure confidentiality**
- **Can ensure authenticity**
  - Signature appended to document

Signed Subspace Key

SSK@ GB3wu...HK35w, c63Ez07...duXDs, ABAQAAEA /site-1

public key hash symmetric key algorithms

Cloud Storage

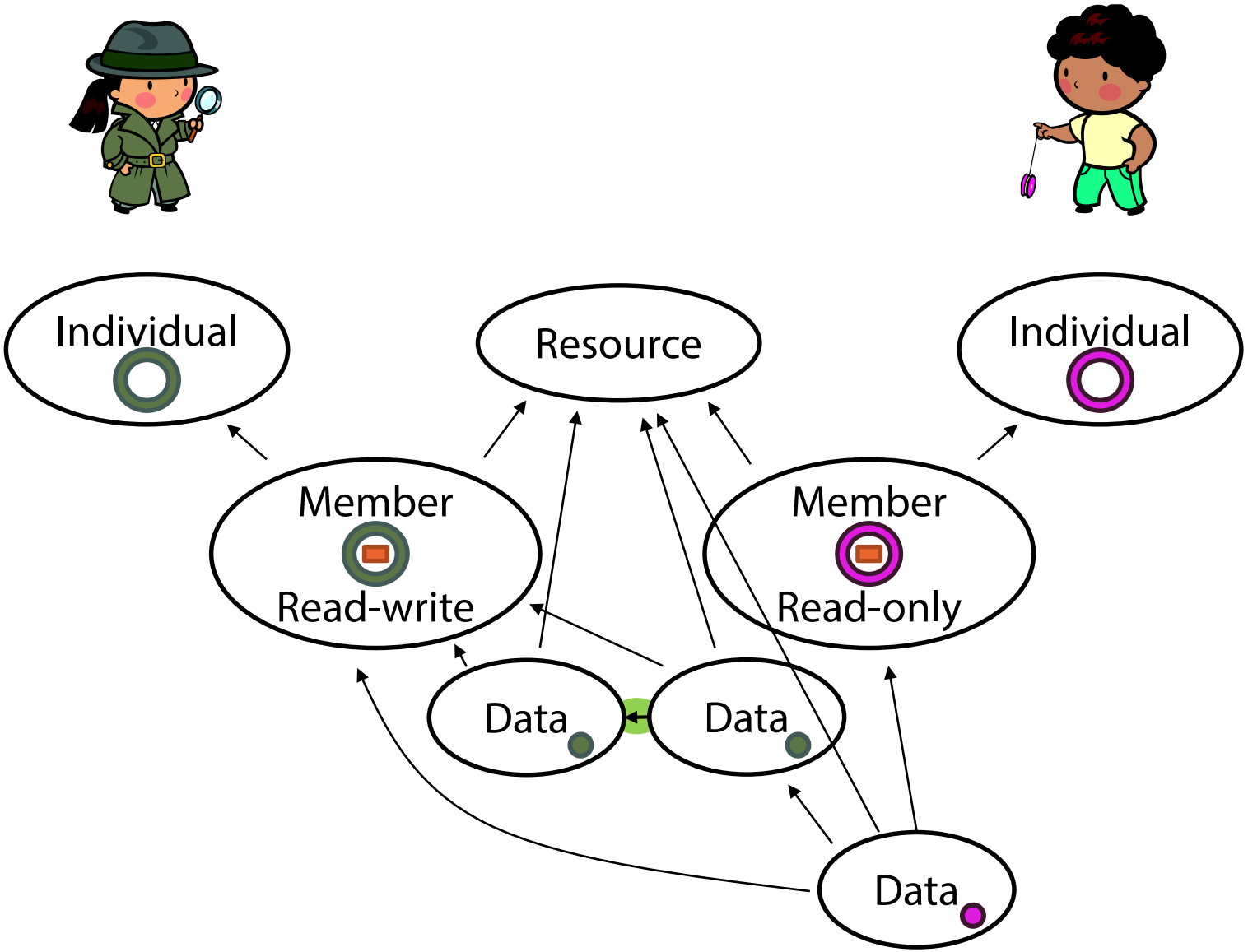
- Export laws
- Cloud servers are untrusted
- Ensure that data is secure
- Enterprise security using cryptography alone

CORRESPONDENCE

Open source
Distributed
Mobile back end as a service

Access to Resources

- **Read-only**
- **Read-write**
- **Asymmetric cryptography for identity**
- **Symmetric cryptography for confidentiality**
- **How to protect writes?**
  - Authentication provider
  - Trust relationship


Individual
Resource
Individual
Member
Read-write
Member
Read-only
Data
Data
Data

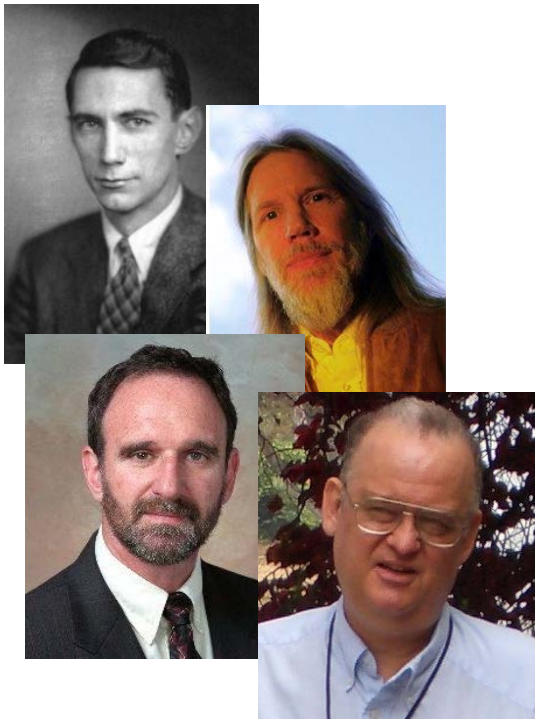
Untrusted Network

- **Clients ensure authorization**
  - Not a function of the server
- **Encrypt data at rest**
  - Protected against unauthorized access
- **Can outsource to cloud**
  - Even without trust

Assurances in Decentralized Systems

- **PGP**
  - Web of trust
  - Public key cryptography
  - Exchange symmetric keys
- **Hashcash**
  - Proof of work
- **Bitcoin**
  - Compete to create blocks
  - Public history of all transactions
  - Sign transactions to spend money
- **Freenet**
  - Identify documents via hash
  - Symmetric cryptography for plausible deniability
- **Correspondence**
  - Read and read write access
  - No central authorization

Cryptography



RSA

AES

SHA

Mathematicians

Algorithms

Tools