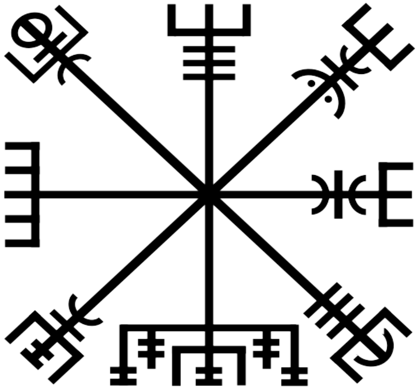
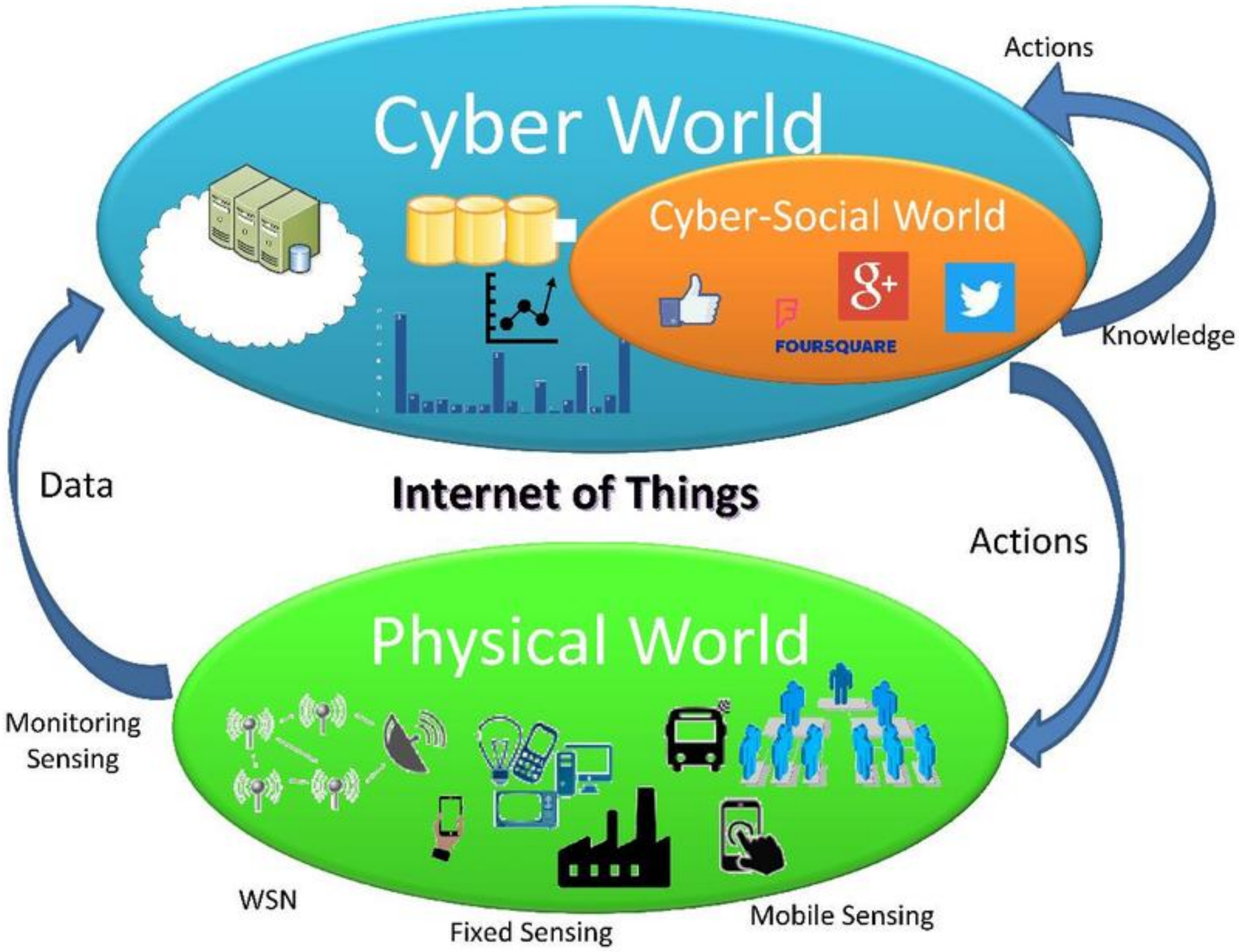
Vegvisir: A Blockchain for IoT



Robbert van Renesse
Hakim Weatherspoon
Stephen Wicker
Danny Adams
Gloire Burambiza
Xinwen Wang

Cornell University



source: Cyber–Physical–Social Frameworks for Urban Big Data Systems: A Survey

A Blockchain for IoT?

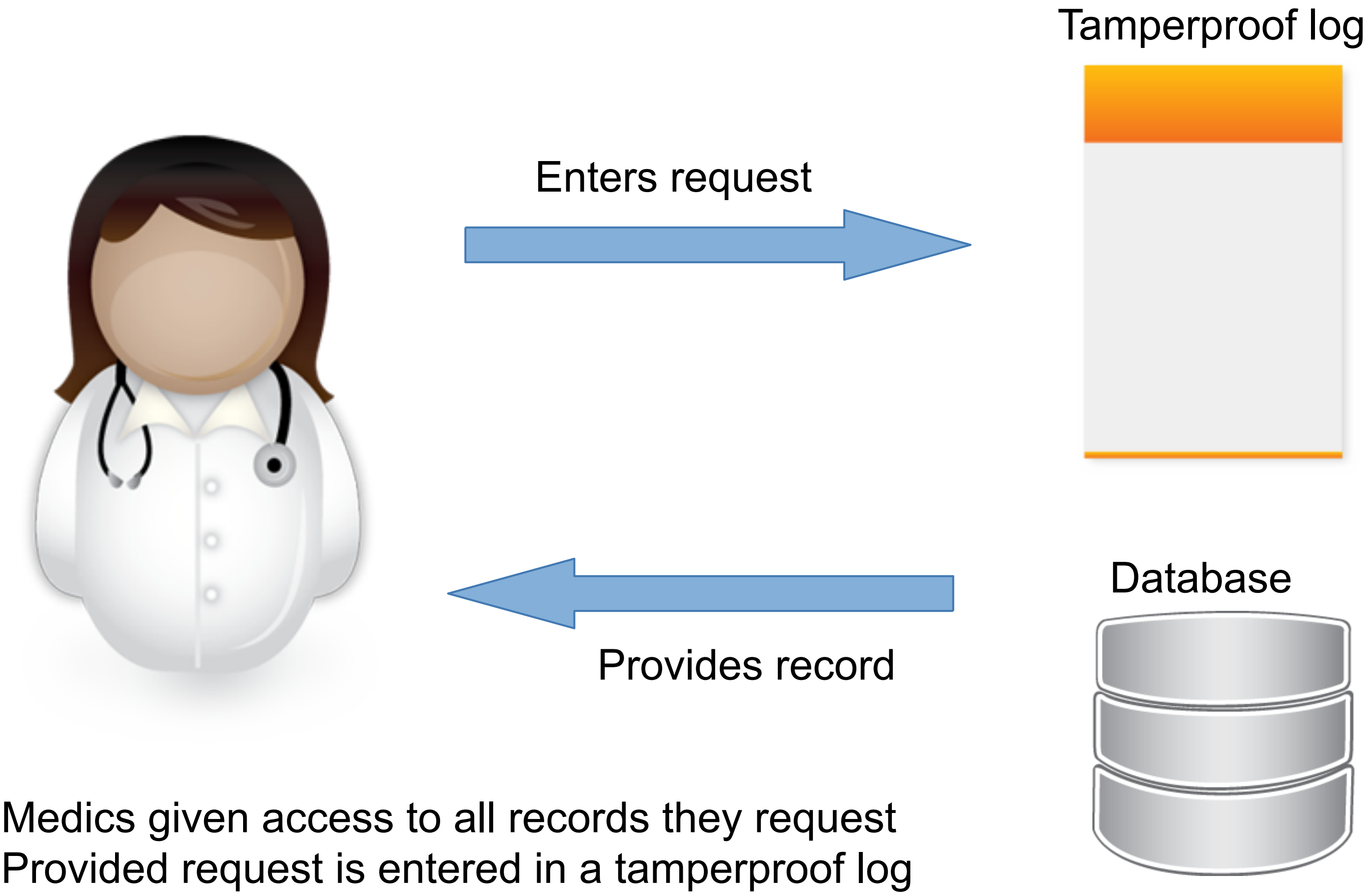
Connect the physical and cyber worlds

- IoT Asset management:
  - what devices are there, how are they being used?
- Programmable IoT
  - smart contracts executed upon certain conditions
- Supply Chain Management:
  - End to end monitoring, auditing
  - Digital Agriculture: farm to table
- Emergency Response:
  - Accountable access to critical information

Prompt and privacy aware access to medical records

Problem: loss of communication with central server

Accountability over access control



Tamperproof log

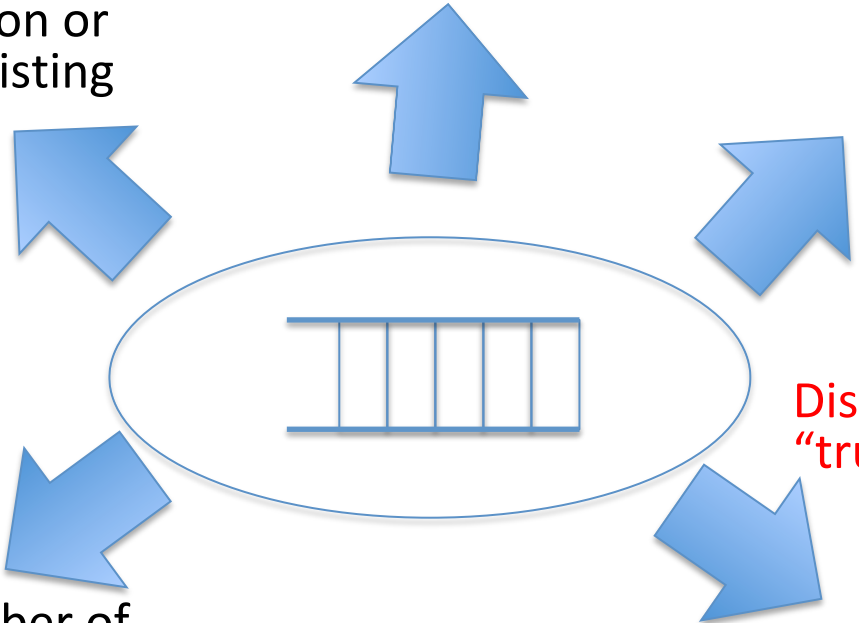
Enters request

Provides record

Database

- Medics given access to all records they request
- Provided request is entered in a tamperproof log
- After emergency is over, logs are reviewed

Challenges



Integrity

- Append-only
- No modification or deletion of existing records

Interoperability

- One size fits all
- Inter-chain Transactions

Scrutinizable

- Protocol and implementation must be easy to understand
- Ideally formally verified

Scalability

- Large number of devices, large amount of data, efficiency

Distributed Trust, or “trustless”

- Not under a single administrative domain
- Yet should be impersonation-resistant

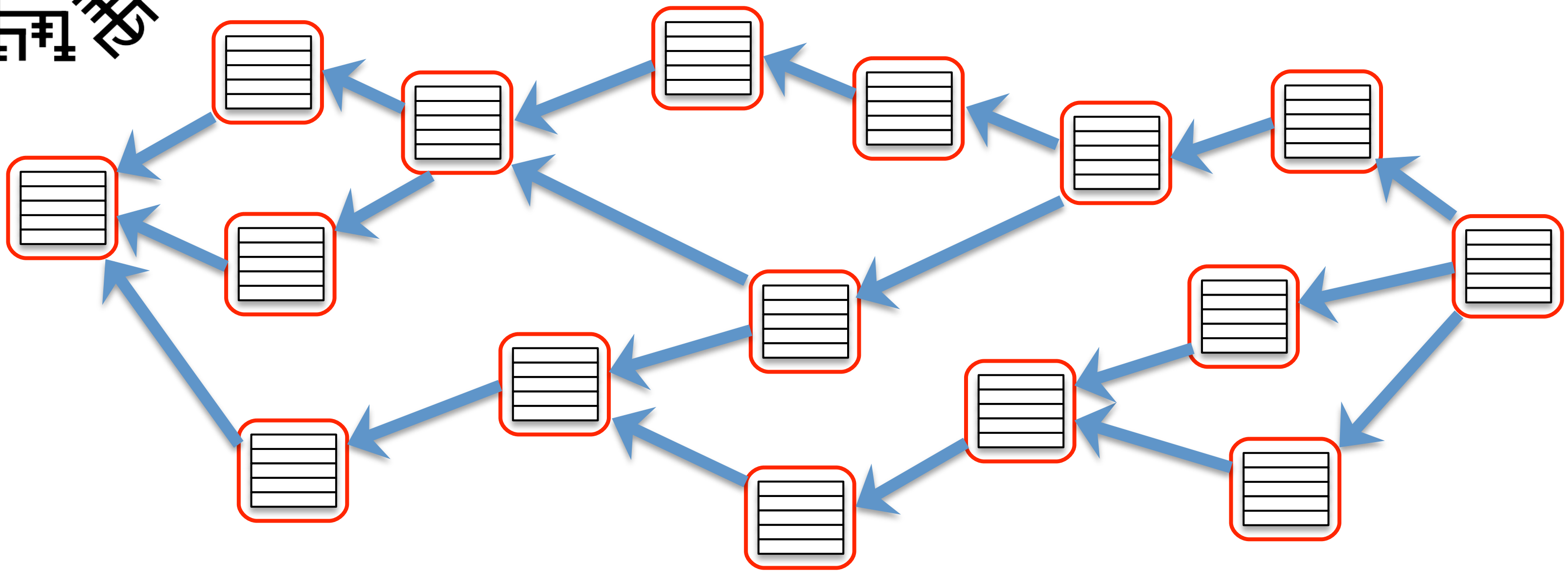
Bitcoin-style blockchains not an option

- Are computationally expensive – and thus battery-draining
- Require high network connectivity
  - Miners typically want to broadcast new blocks asap
  - Protocol can recover from temporary network partitions, but leads to blocks being discarded and work wasted, as well as security issues
- Lack of decentralization harms security

“Permissioned” blockchains can dispense with Proof-of-Work

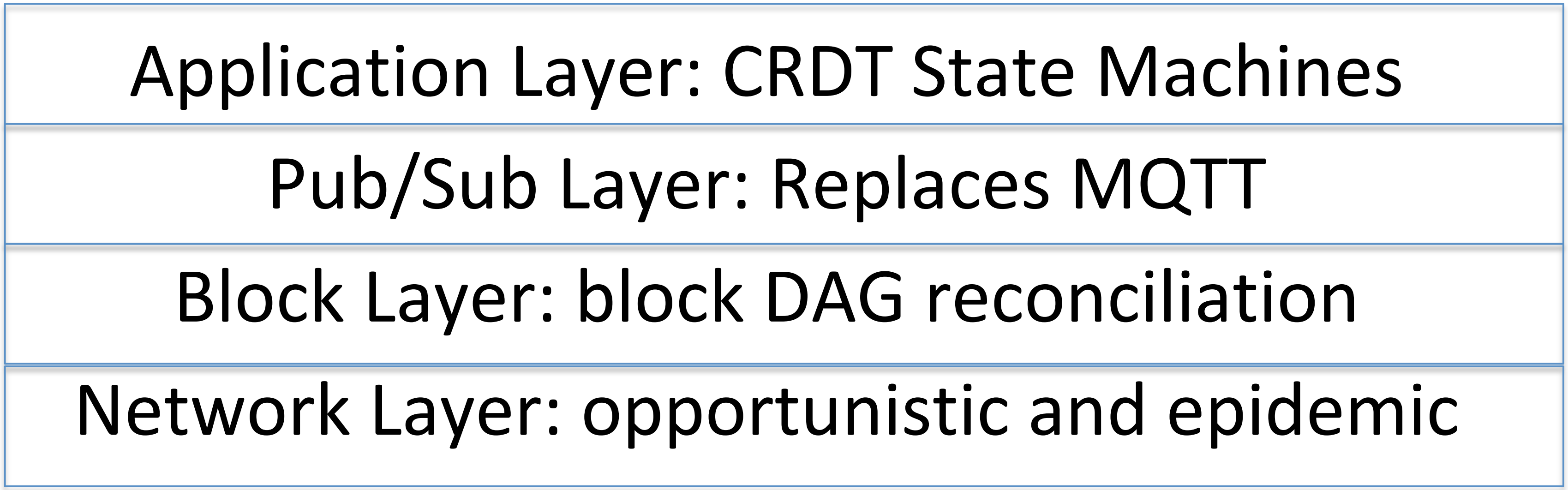
- Blockchain doubles as a PKI
- Owner’s self-signed certificate in genesis block
- Additional users added/removed by placing certificates/revocations on blockchain
- But system-wide consensus is not an option either
  - Requires network connectivity
  - Does not scale

Vegvisir: tolerate branches



- Leads to DAG structure instead of linear blockchain
  - not for throughput, but for disconnected operation
- Not good for cryptocurrencies…
  - but misbehavior is detectable
- Still maintains full causal history of events

Vegvisir Layers



Application Layer: CRDT State Machines

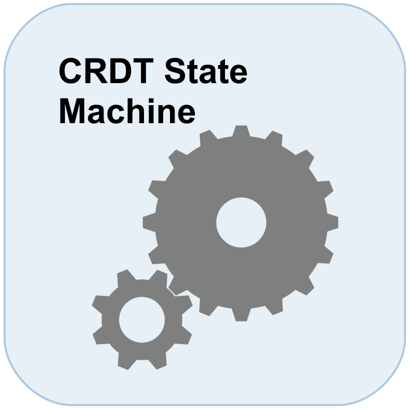
Pub/Sub Layer: Replaces MQTT

Block Layer: block DAG reconciliation

Network Layer: opportunistic and epidemic

- Optional: TEE support

Application Layer



CRDT State Machine

- Challenge 1: consistency
  - – Solution: CRDTs
- *CRDT State Machines* receive the same *transactions* in the same *partial* order
- We designed CRDTs that take advantage of p.o.
  - – nP+ set: a set of prioritized elements
  - – Under concurrent update highest priority wins
  - – 2P set is a special case
    - no dependencies; delete is high priority

Application Layer, cont’d

- Challenge 2: tamperproof
  - Solution: Proof-of-Witness
  - PoW for tx is also PoW for all dependent txs
  - Each app specifies set of “safe sets” of devices
  - How to find independent witnesses?

No more than k malicious devices in any neighborhood

Pub/Sub Layer

- Distributes transactions between devices
- Challenge: Byzantine devices can

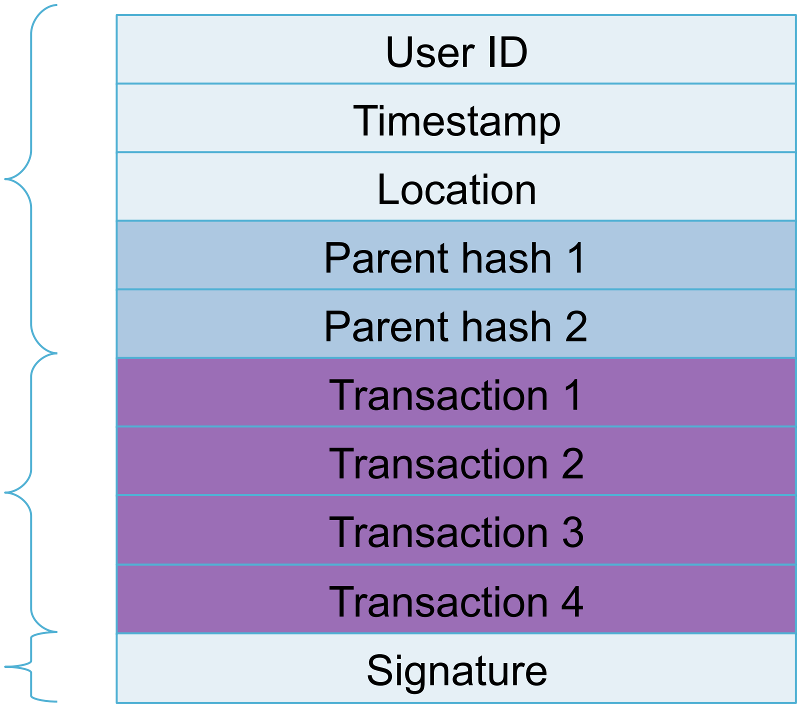
  - submit bogus transactions
    - does not hurt consistency but may hurt utility
  - DDoS with many transactions
- Solutions: membership and rate control
  - membership itself is an nP+ set of devices

Membership CRDT

- Two operations
  - add-membership(device)
  - revoke-membership(device)
- Proof-of-misbehavior implicitly revokes membership
- Only members can add transactions
  - and must sign them

Block Layer

- Aggregates transactions



Header

User ID

Timestamp

Location

Parent hash 1

Parent hash 2

Body

Transaction 1

Transaction 2

Transaction 3

Transaction 4

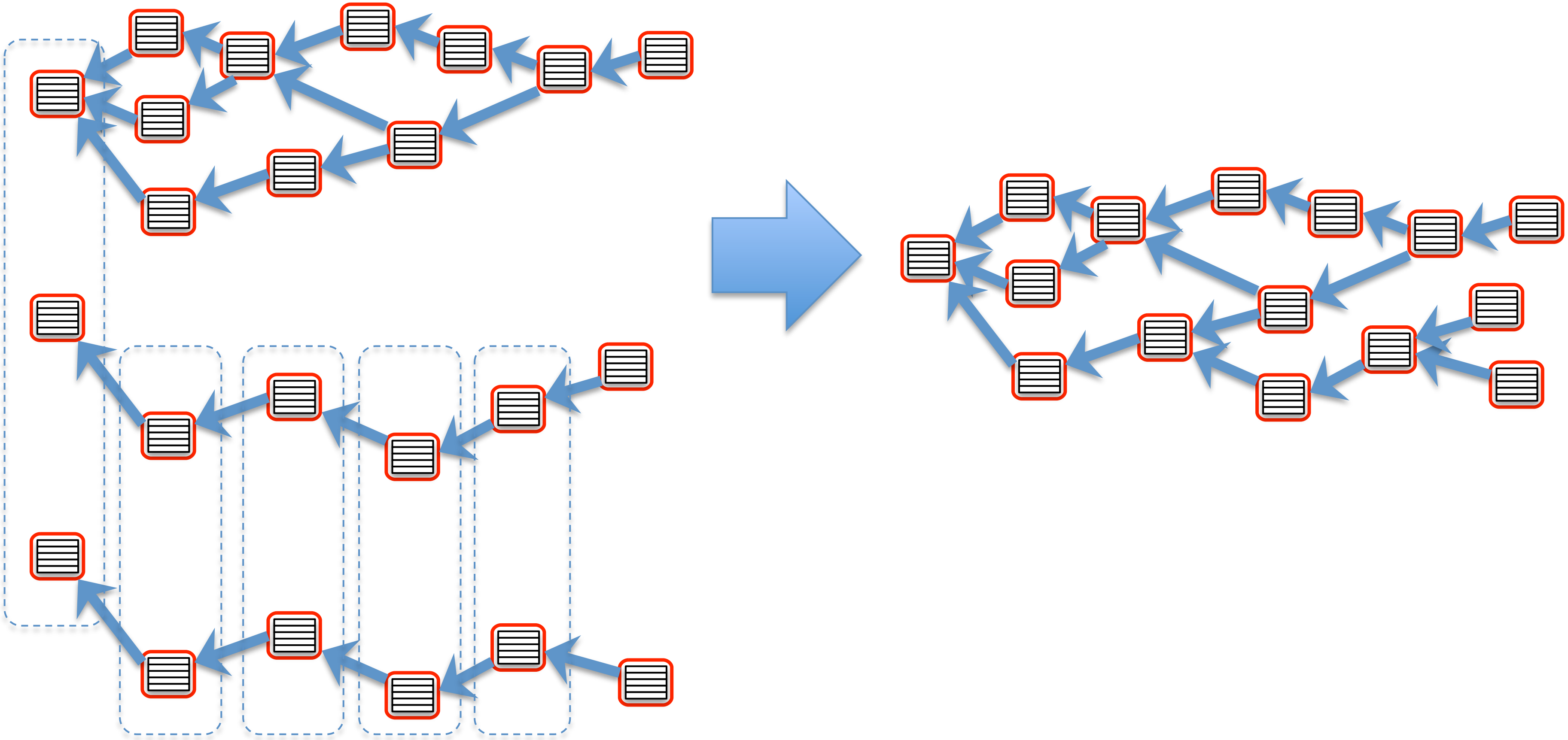
Signature

Signature

Blocks are certificates

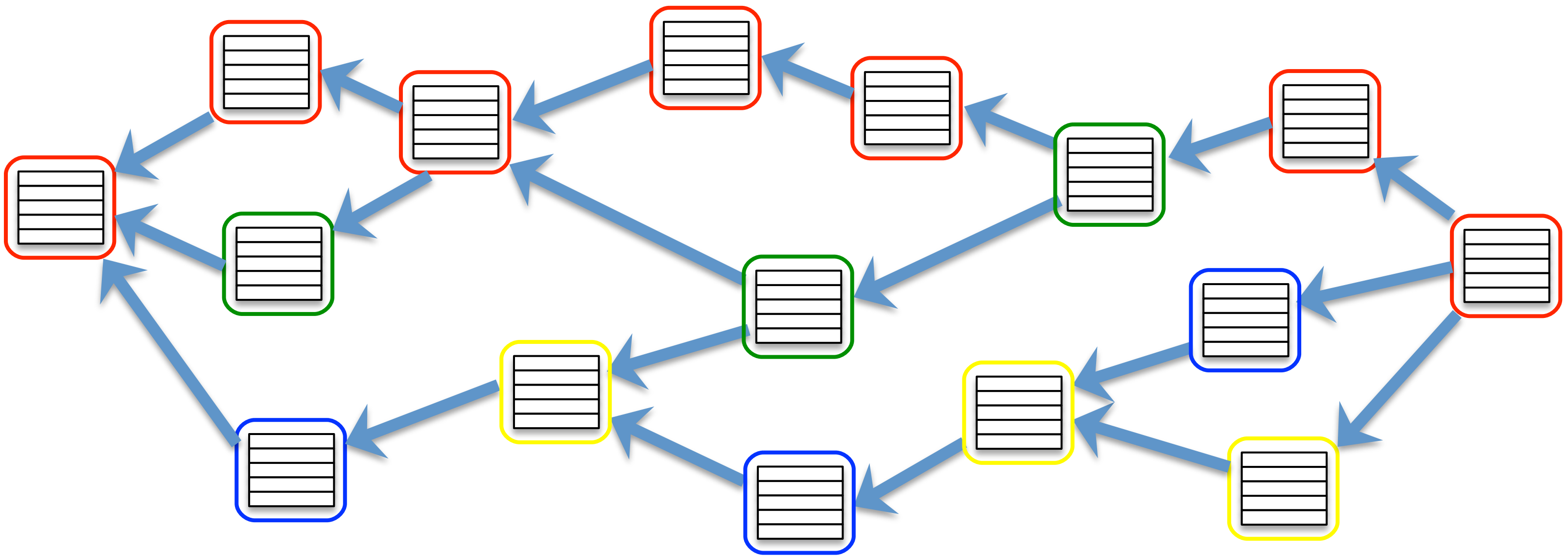
Block Layer

- Challenge 1: Efficient Reconciliation



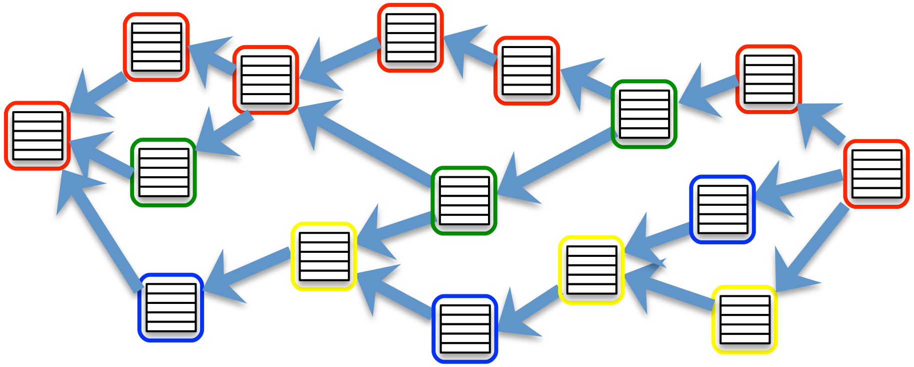
Block Layer, cont'd

- Challenge 1: Efficient Reconciliation
- Solution:
  - simplifying assumption: two tx from the same (honest) device are always dependent



Block Layer, cont’d

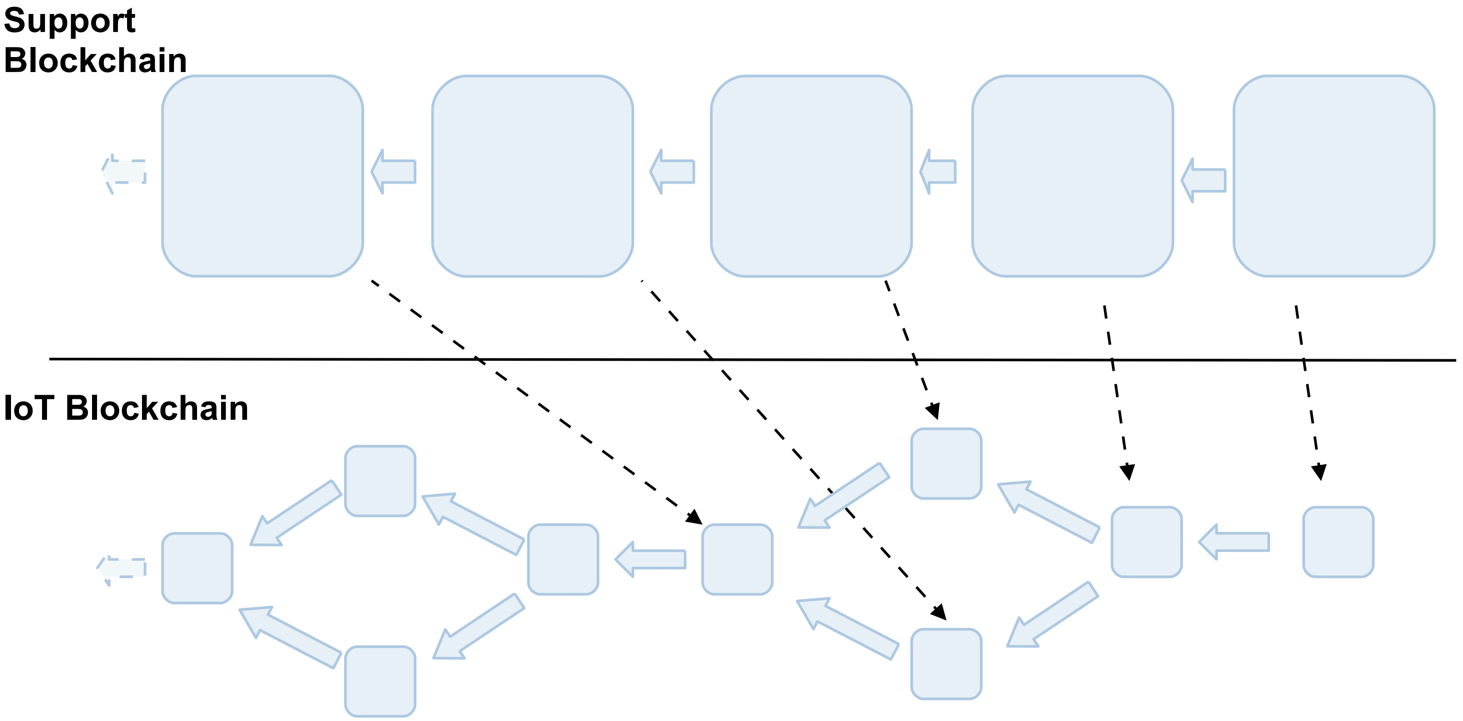
- Challenge 1: Efficient Reconciliation
  - Solution: Hash Enhanced Vector Timestamp
    - #blocks + hash for each device
    - same #blocks + different hash = Proof-of-Misbehavior



🟥	6
🟩	3
🟨	3
🟦	3

Block Layer, cont’d

- Challenge 2: Offloading Storage
  - Solution: Use a “support blockchain”
  - Allows regular peers to discard old blocks
  - Design invariant: block availability monotonically increasing



Network Layer

- Challenge: no reliable network infrastructure
  - Solution: “opportunistic networking”
    - reconcile when in range
    - reconcile randomly when connected to infrastructure
      - i.e., gossip, using membership CRDT
    - device changes periodically between “advertise” and “discovery” modes at random
    - also switches Wifi between infrastructure and p2p modes

ARM TrustZone

- ARM TrustZone “secure worlds” can help:
  - Who is a good witness?
    - secure access to device location and time
  - Check PoW and provide access to secured data
  - Secure sensor values
    - secure retrieval of sensor values

Conclusion

- Vegvisir is a DAG-based blockchain to allow for partitioned operations
    - *not for higher throughput per se*
- Replaces Proof-of-Work with *“Proof-of-Witness”*
- CRDTs enable consistently evolving views
- Prototype available for Android devices