A block-chain based approach to Resource Sharing in Smart Neighbourhoods

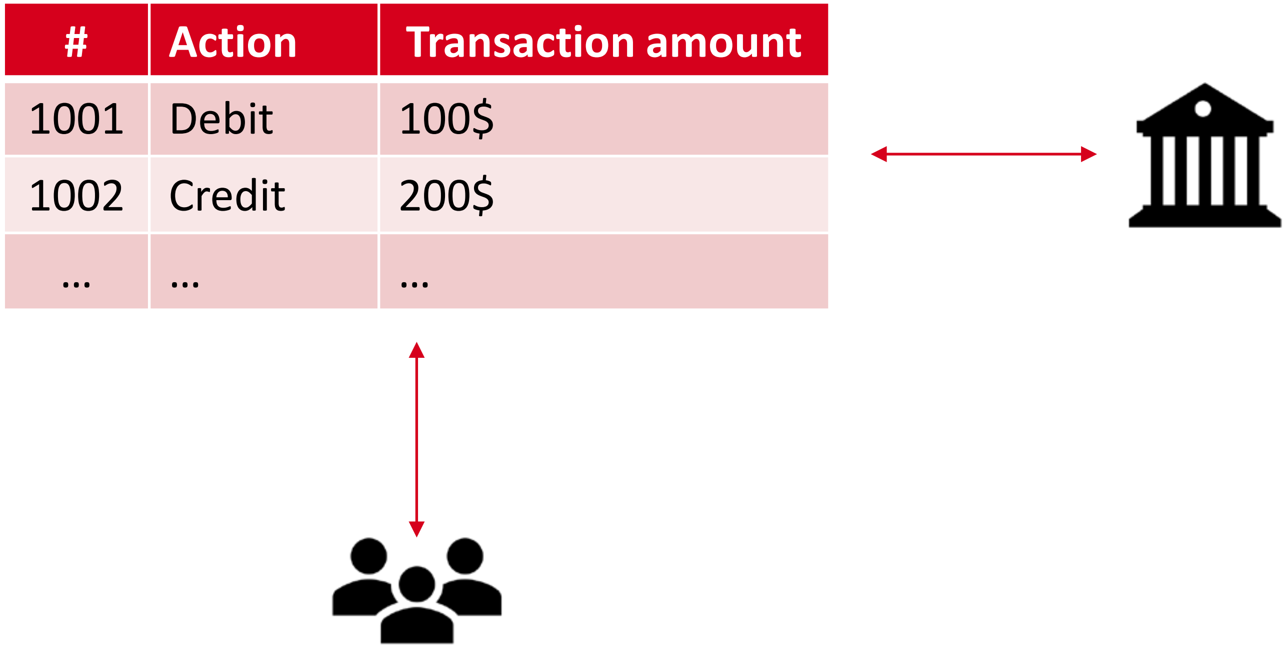
Muni Venkateswarlu Kumaramangalam

31 Jan 2020

UNIVERSITY OF CALGARY

What is a LEDGER?

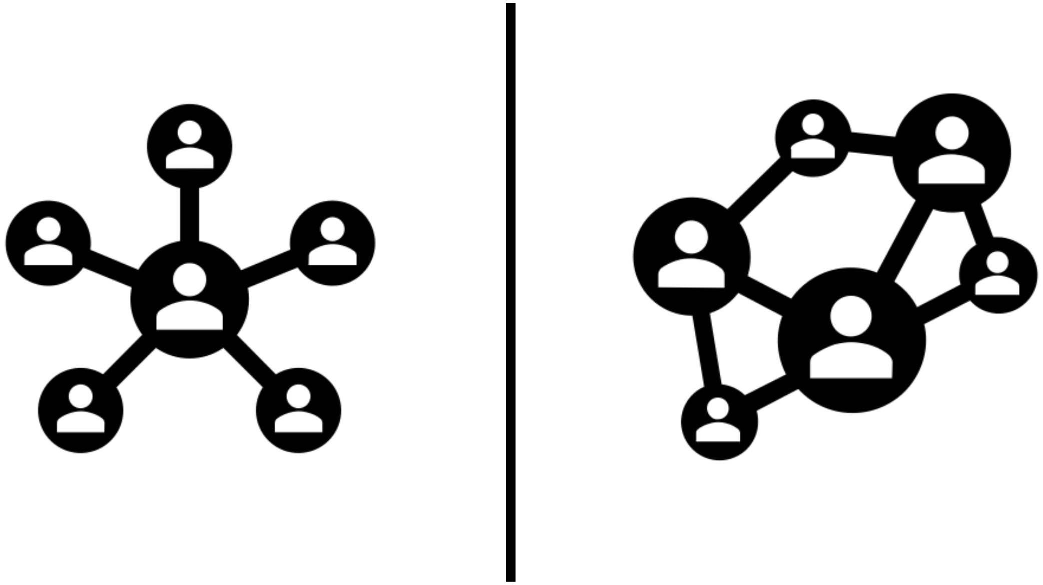
- What is a Ledger?
  - Record of activities
    - Financial, legal, physical or electronic
    - Recent call history
    - Financial transactions
- Centralized ledger
  - Controlled by an authority
  - Single point of failure
    - Accidental or intentional



#	Action	Transaction amount
1001	Debit	100$
1002	Credit	200$
…	…	…

Distributed Ledger

- Shared ledger
  - across a network of multiple sites, geographies or institutions
  - no central administrator or centralized data storage
  - Immutable
- Participants can have their own identical copy of the ledger
  - May have a shard of the ledger
- Driven by cryptography
  - Security, accuracy, privacy



Fill in the answers

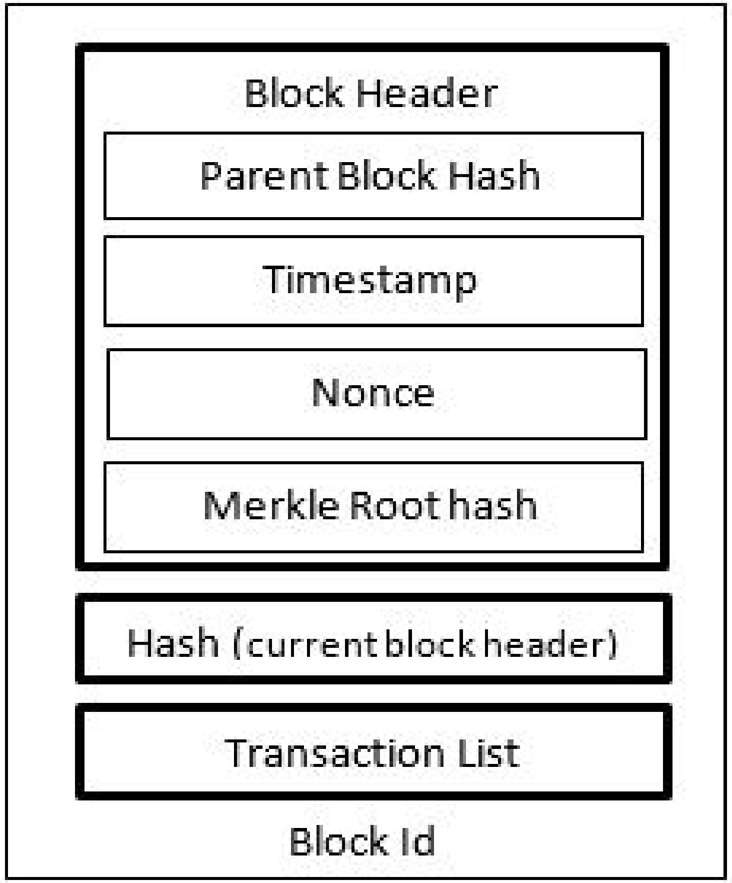
	1 + 2 =	0g5e3
0g5e3	5 x 5 =	5r1t9
5r1t9	A = 100, B = 200, C = A + B =	8s3s9
8s3s9	A + B + C =	3a8r6

Fill in the answers : Distributed Ledger

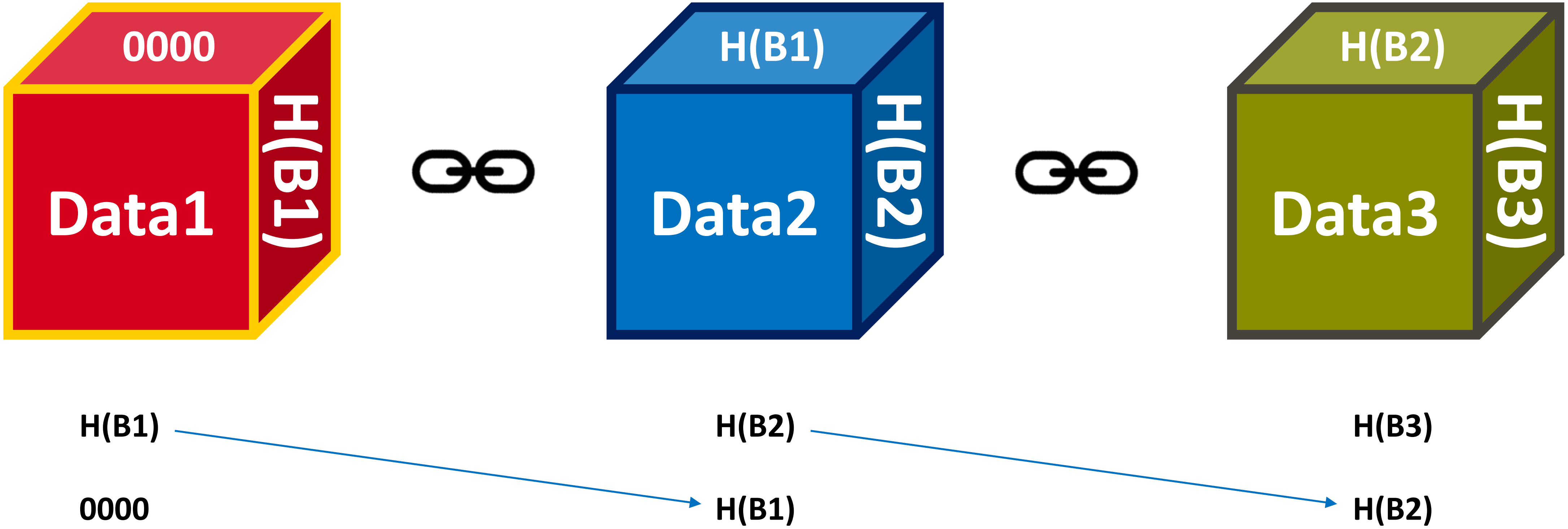
	1 + 2 = 3	0g5e3
0g5e3	5 x 5 = 25	5r1t9
5r1t9	A = 100, B = 200, C = A + B = 300	8s3s9
8s3s9	A + B + C = 600	3a8r6

What is a Blockchain?

- Distributed Ledger Technology (DLT)
  - technological infrastructure and protocols to access, validation, and updating records across multiple entities or locations
- Underlying DLT is blockchain
  - Distributed & P2P network
  - Decentralized trust
  - Immutable
  - Anonymity
- How does a BLOCK look like?
  - Bitcoin block structure



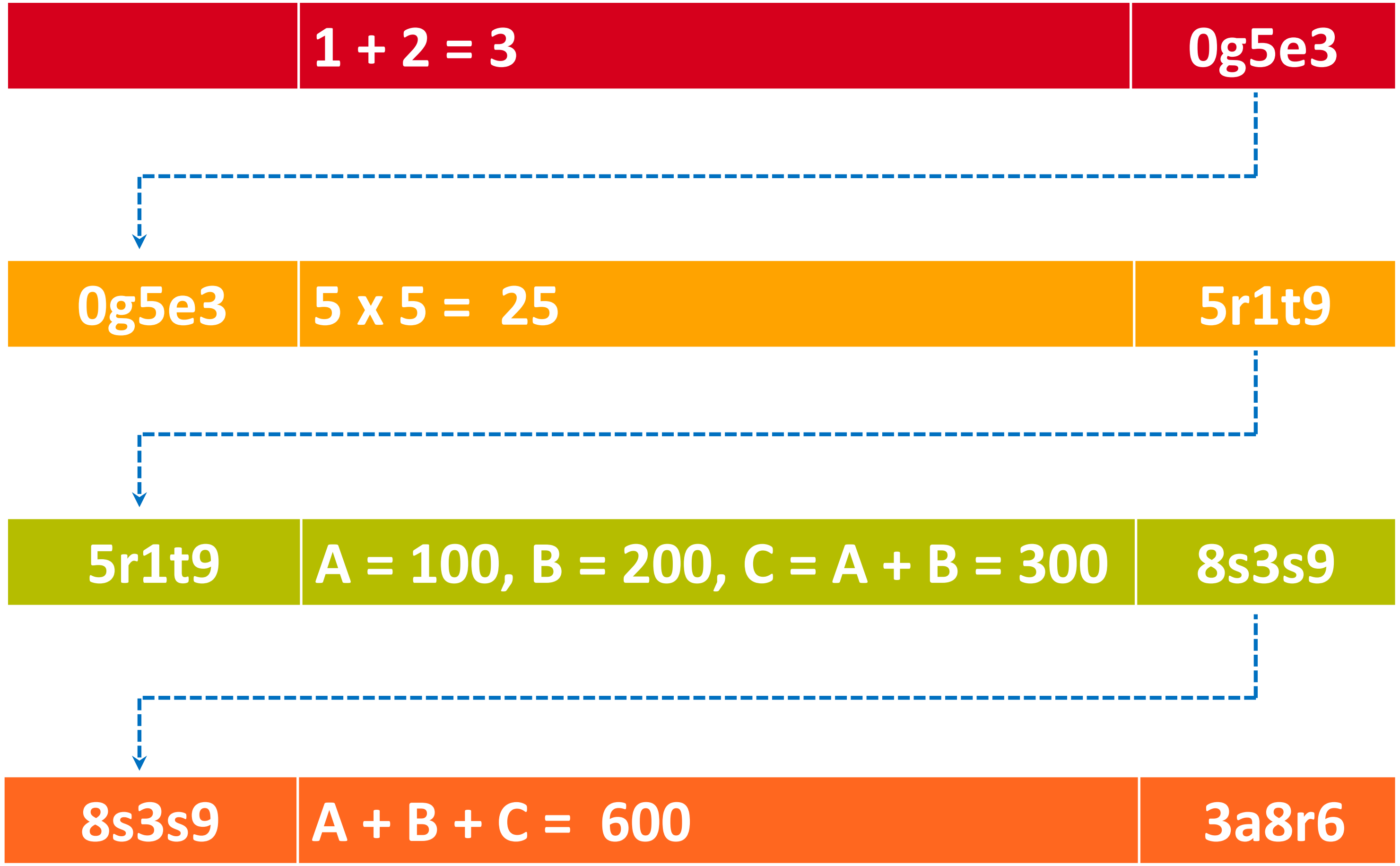
Blockchain



H(B1) H(B2) H(B3)

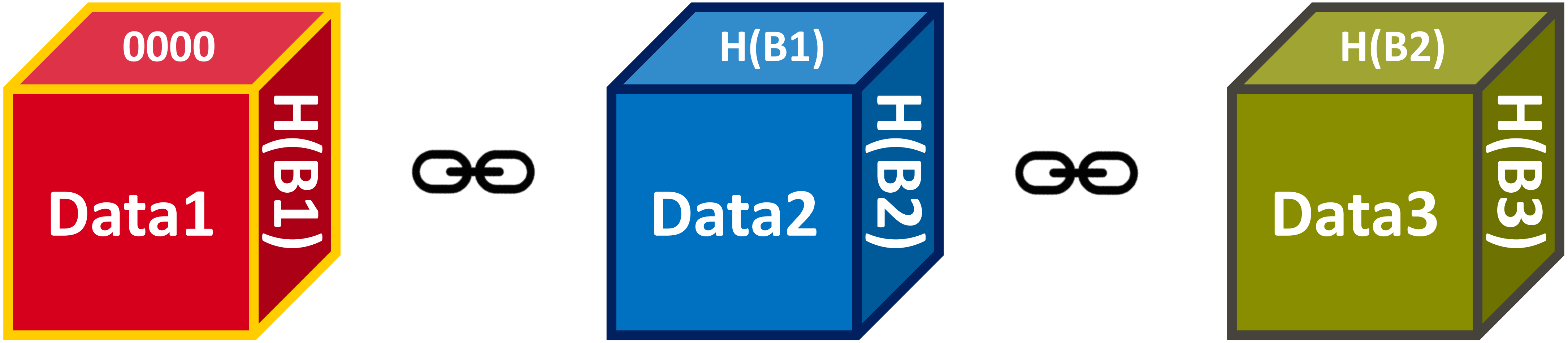
0000 H(B1) H(B2)

Fill in the answers : Blockchain



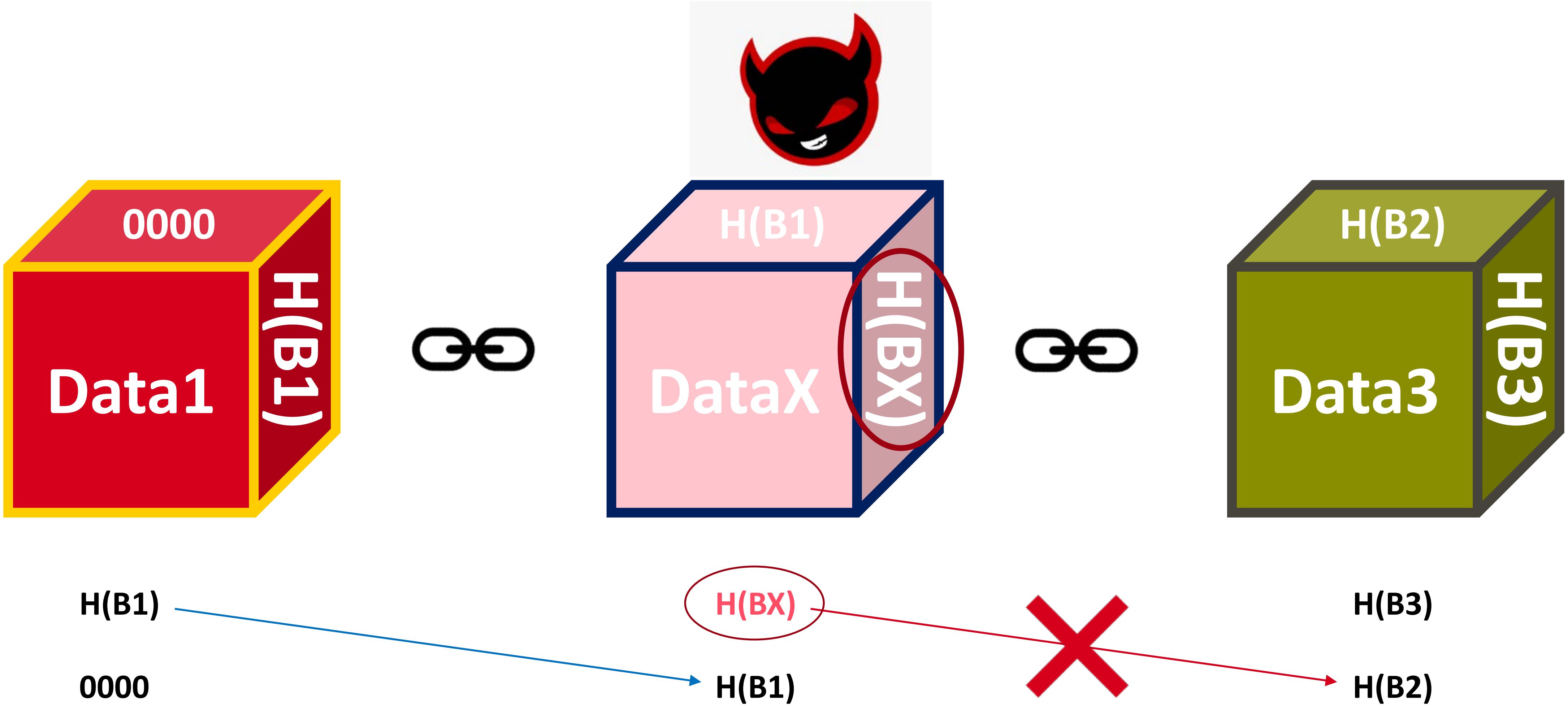
	1 + 2 = 3	0g5e3
0g5e3	5 x 5 = 25	5r1t9
5r1t9	A = 100, B = 200, C = A + B = 300	8s3s9
8s3s9	A + B + C = 600	3a8r6

Blockchain



H(B1)	H(B2)	H(B3)
0000	H(B1)	H(B2)

Blockchain



0000

Data1

H(B1)

H(B1)

DataX

H(BX)

H(B2)

Data3

H(B3)

H(B1)

0000

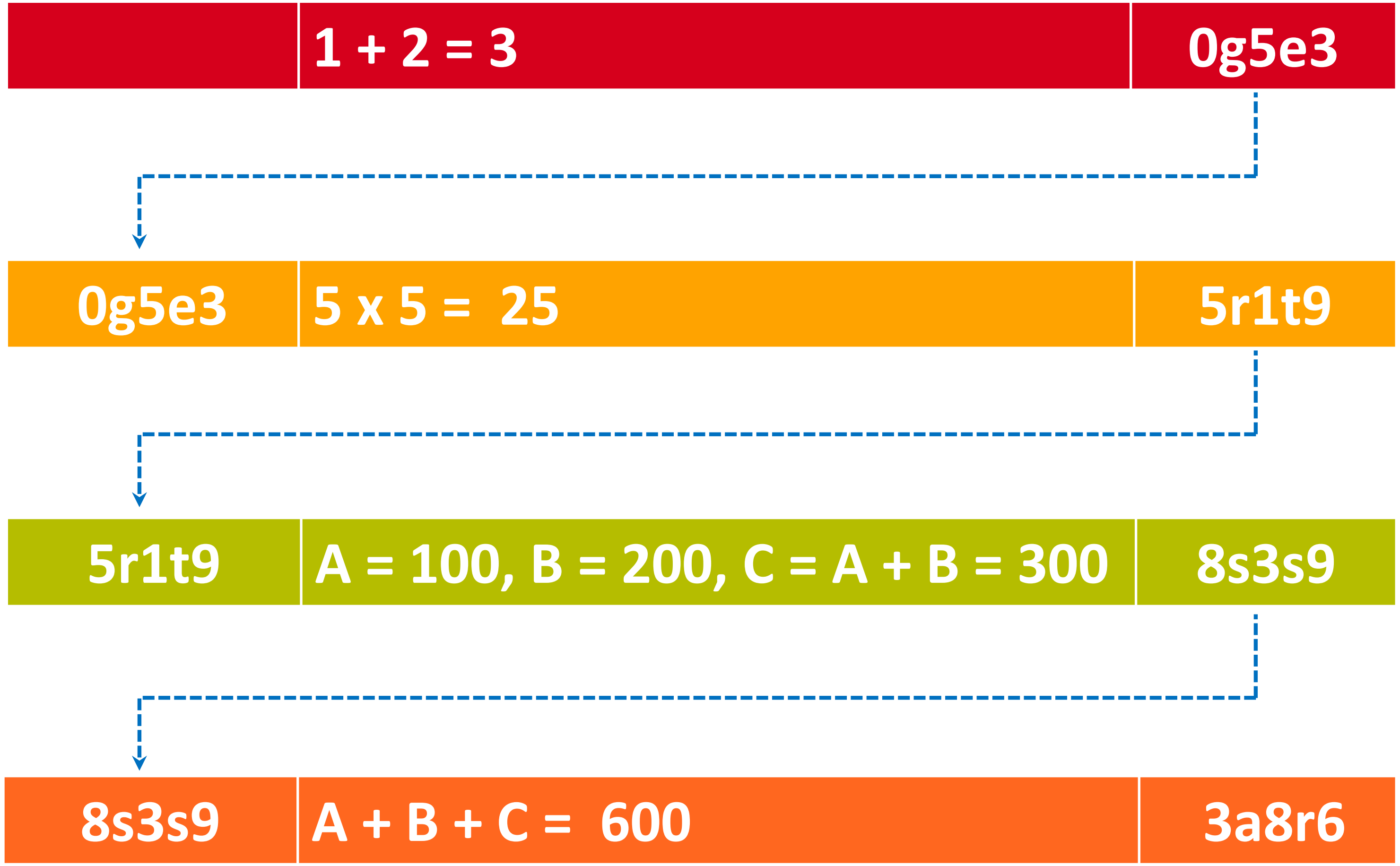
H(BX)

H(B1)

H(B3)

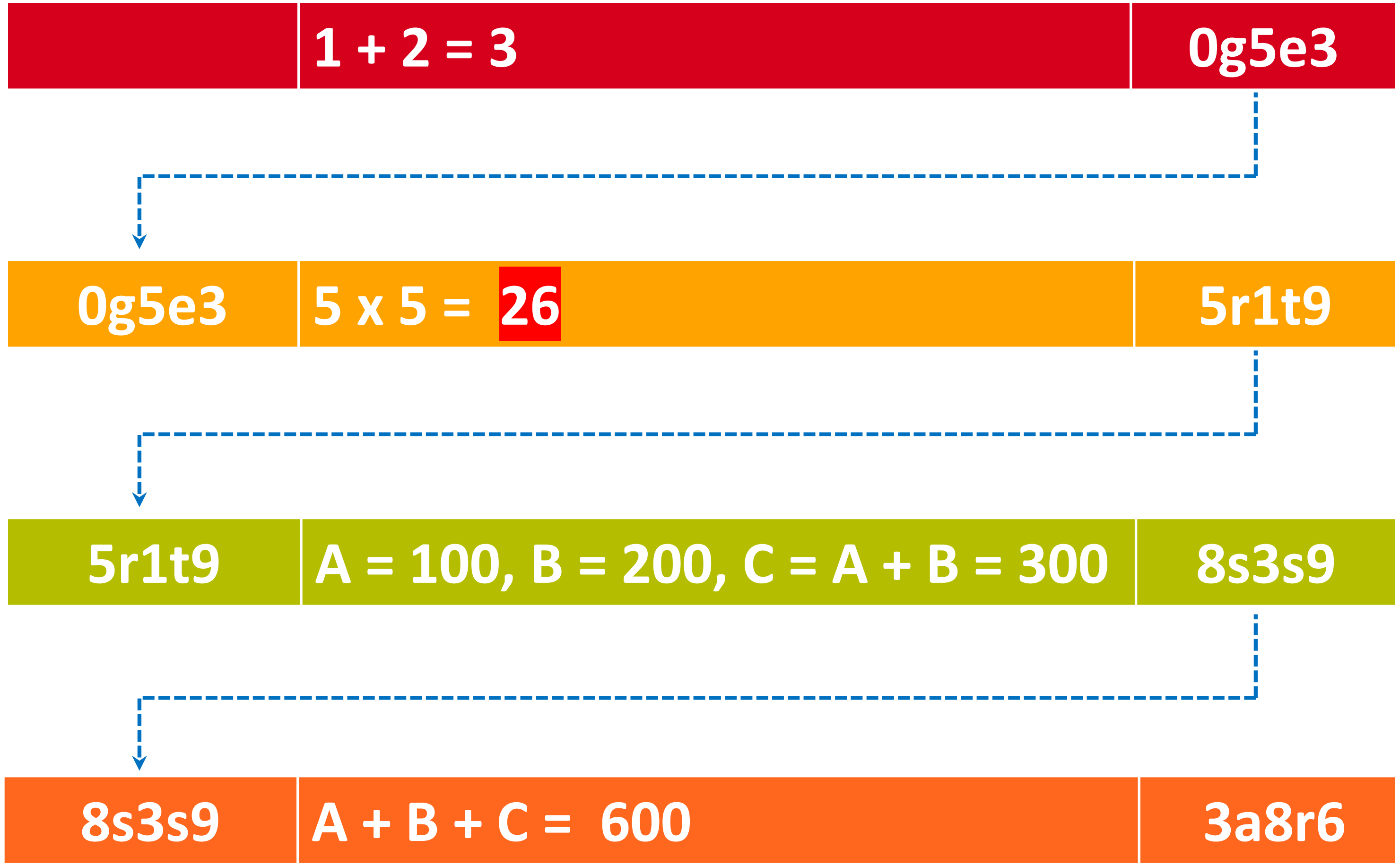
H(B2)

Fill in the answers : Blockchain



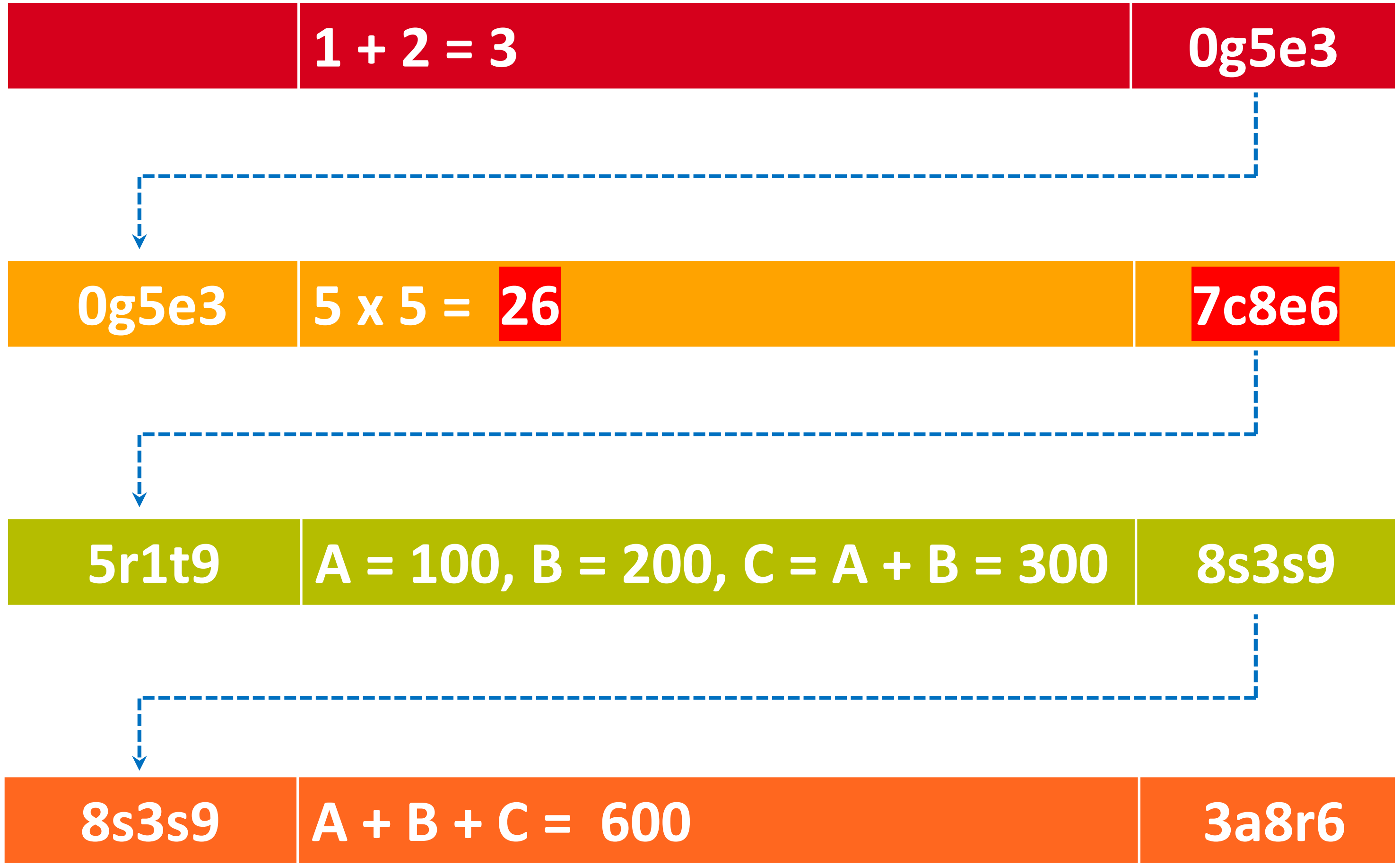
	1 + 2 = 3	0g5e3
0g5e3	5 x 5 = 25	5r1t9
5r1t9	A = 100, B = 200, C = A + B = 300	8s3s9
8s3s9	A + B + C = 600	3a8r6

Fill in the answers : Blockchain



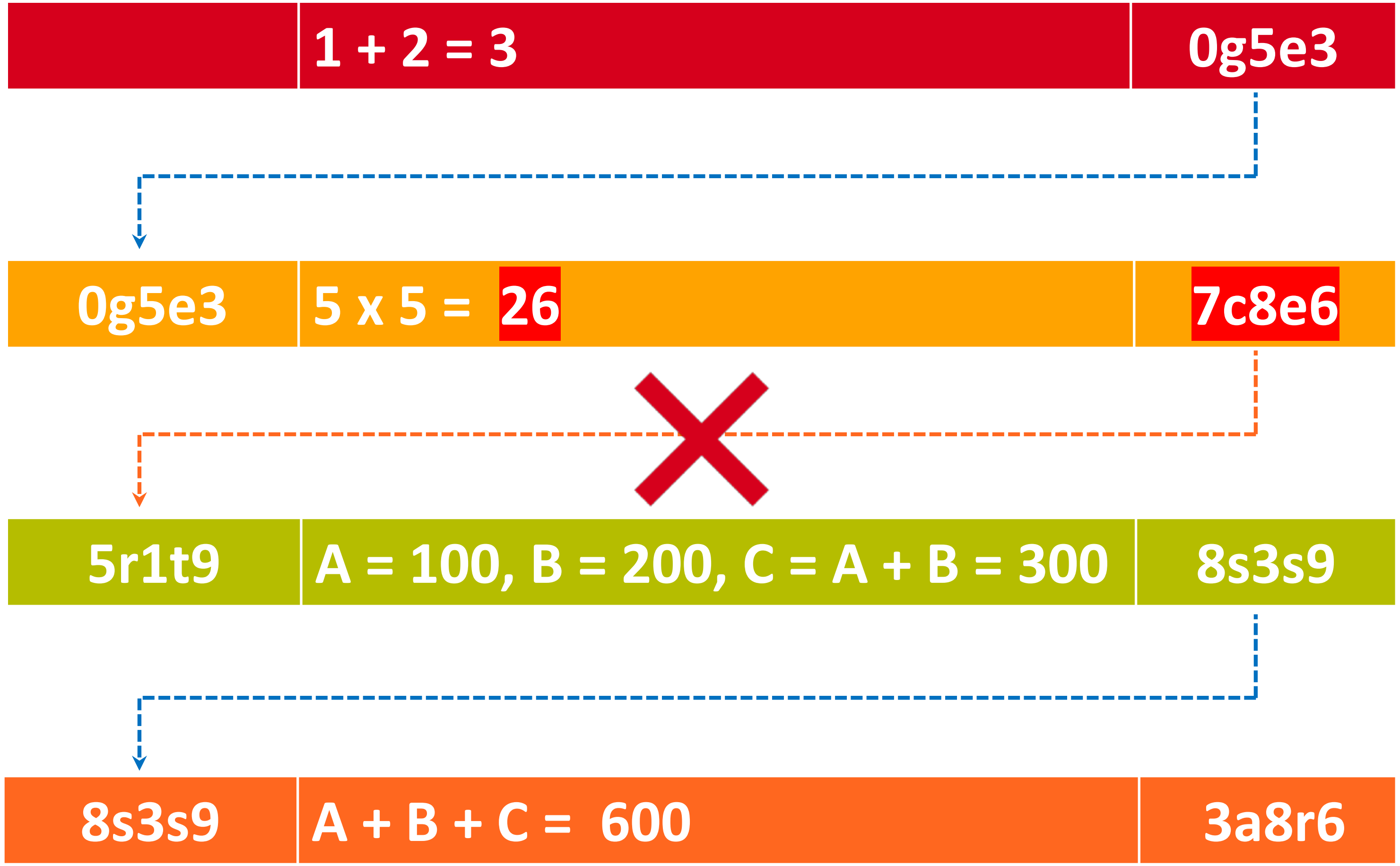
	1 + 2 = 3	0g5e3
0g5e3	5 x 5 = 26	5r1t9
5r1t9	A = 100, B = 200, C = A + B = 300	8s3s9
8s3s9	A + B + C = 600	3a8r6

Fill in the answers : Blockchain



	1 + 2 = 3	0g5e3
0g5e3	5 x 5 = 26	7c8e6
5r1t9	A = 100, B = 200, C = A + B = 300	8s3s9
8s3s9	A + B + C = 600	3a8r6

Fill in the answers : Blockchain



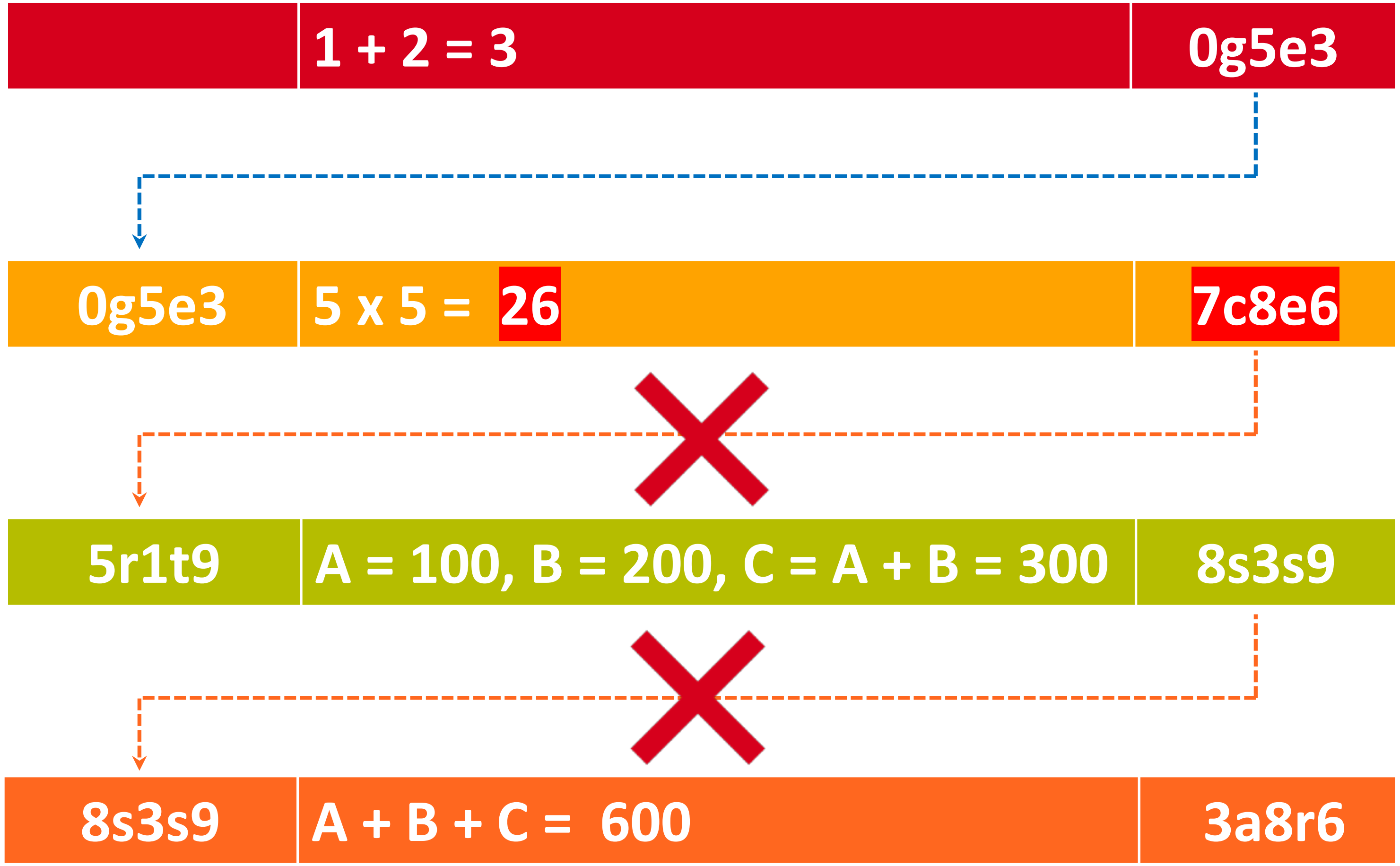
1 + 2 = 3 | 0g5e3

0g5e3 | 5 x 5 = 26 | 7c8e6

5r1t9 | A = 100, B = 200, C = A + B = 300 | 8s3s9

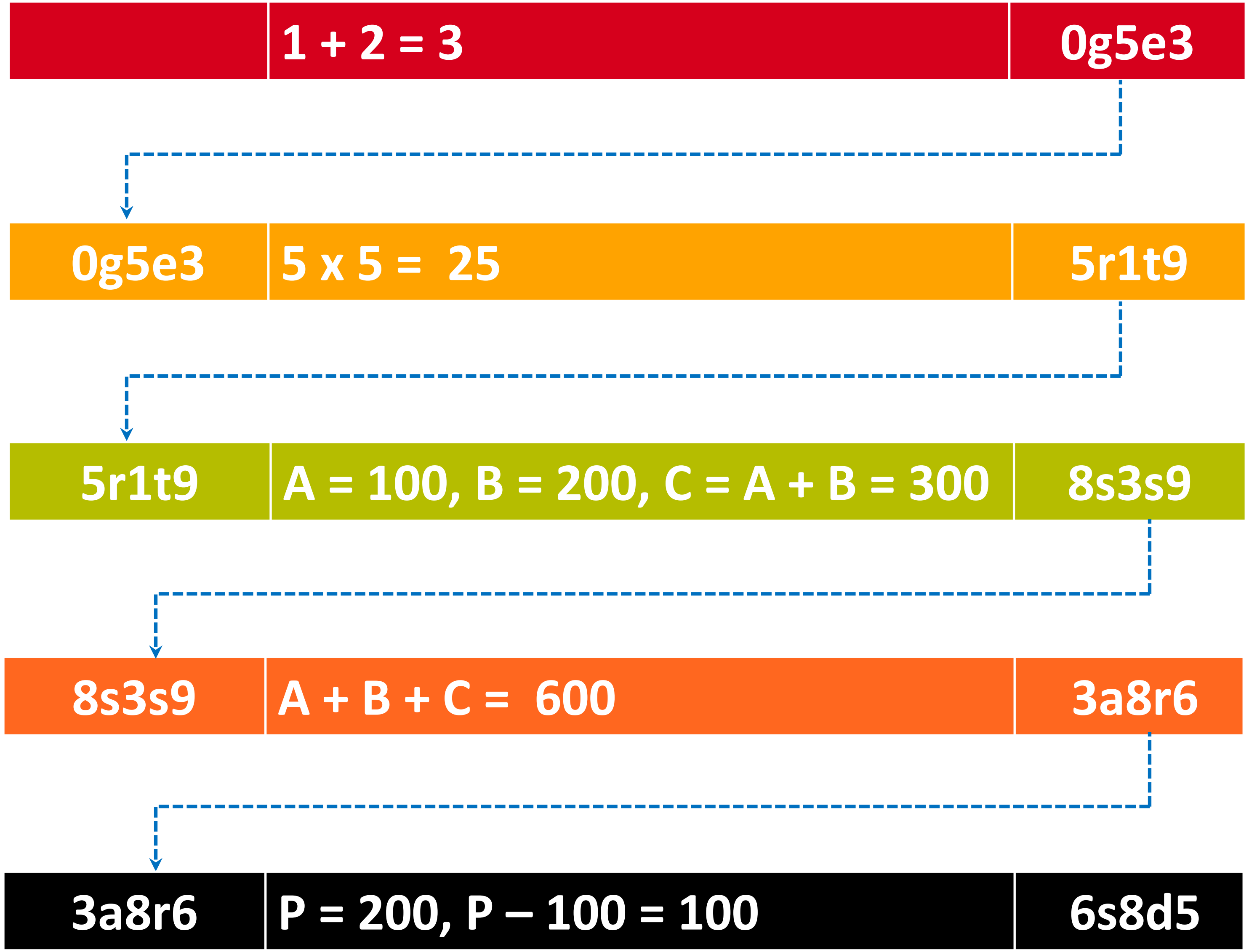
8s3s9 | A + B + C = 600 | 3a8r6

Fill in the answers : Blockchain



	1 + 2 = 3	0g5e3
0g5e3	5 x 5 = 26	7c8e6
5r1t9	A = 100, B = 200, C = A + B = 300	8s3s9
8s3s9	A + B + C = 600	3a8r6

Adding a new block

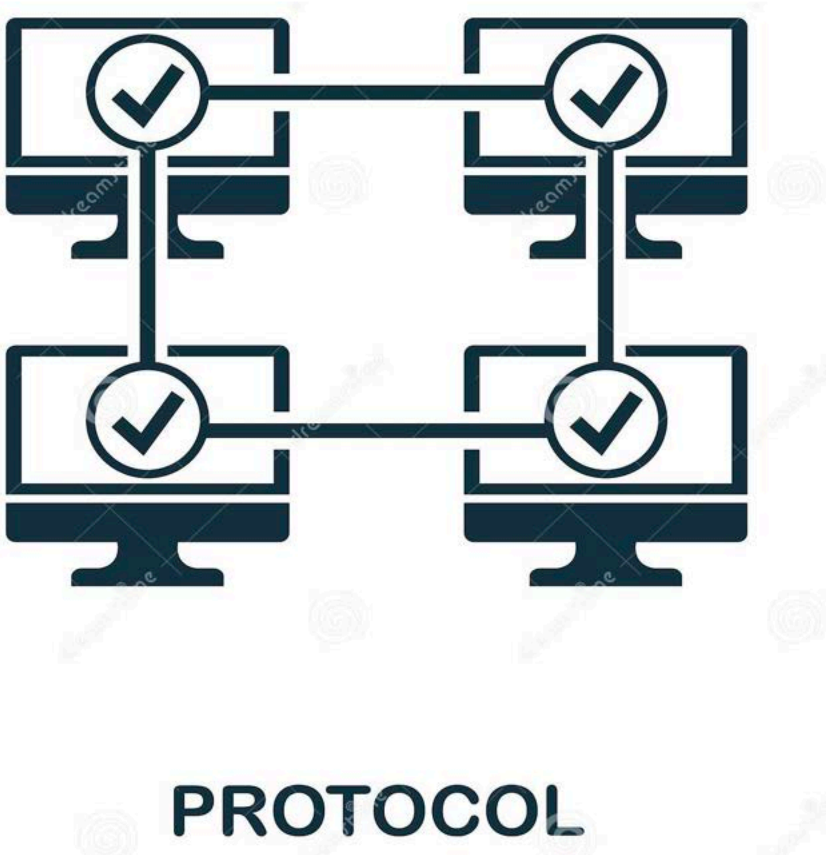


	1 + 2 = 3	0g5e3
0g5e3	5 x 5 = 25	5r1t9
5r1t9	A = 100, B = 200, C = A + B = 300	8s3s9
8s3s9	A + B + C = 600	3a8r6
3a8r6	P = 200, P – 100 = 100	6s8d5

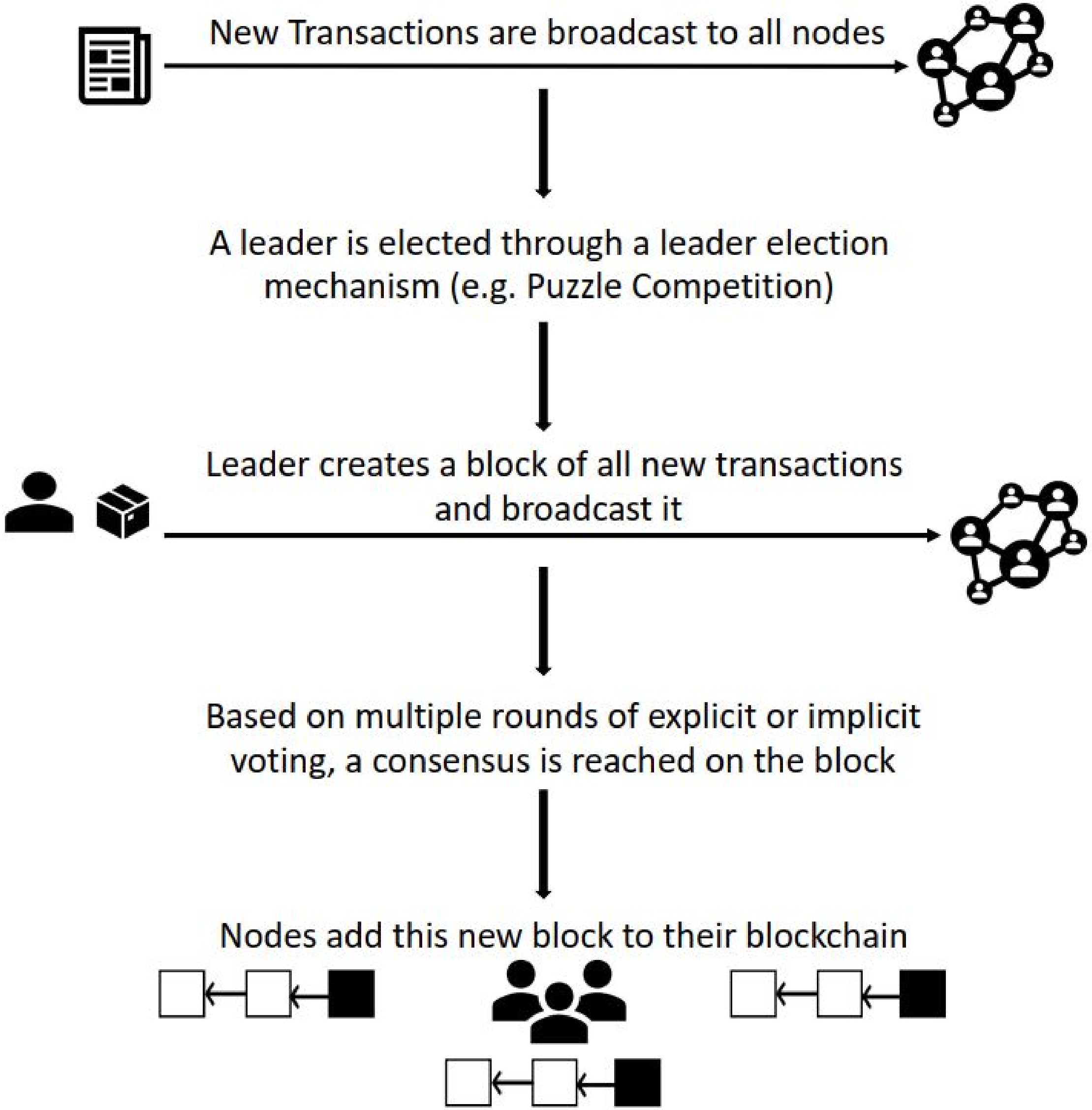
Consensus

- To add a new block to the blockchain, all participating nodes must come to a common agreement (called ***consensus***)

- Major Consensus models:
  - Proof of Work (PoW)
  - Proof of Stake (PoS)
  - Round Robin
  - RAFT
  - Practical byzantine fault tolerance (PBFT)



Consensus Protocol



New Transactions are broadcast to all nodes

A leader is elected through a leader election mechanism (e.g. Puzzle Competition)

Leader creates a block of all new transactions and broadcast it

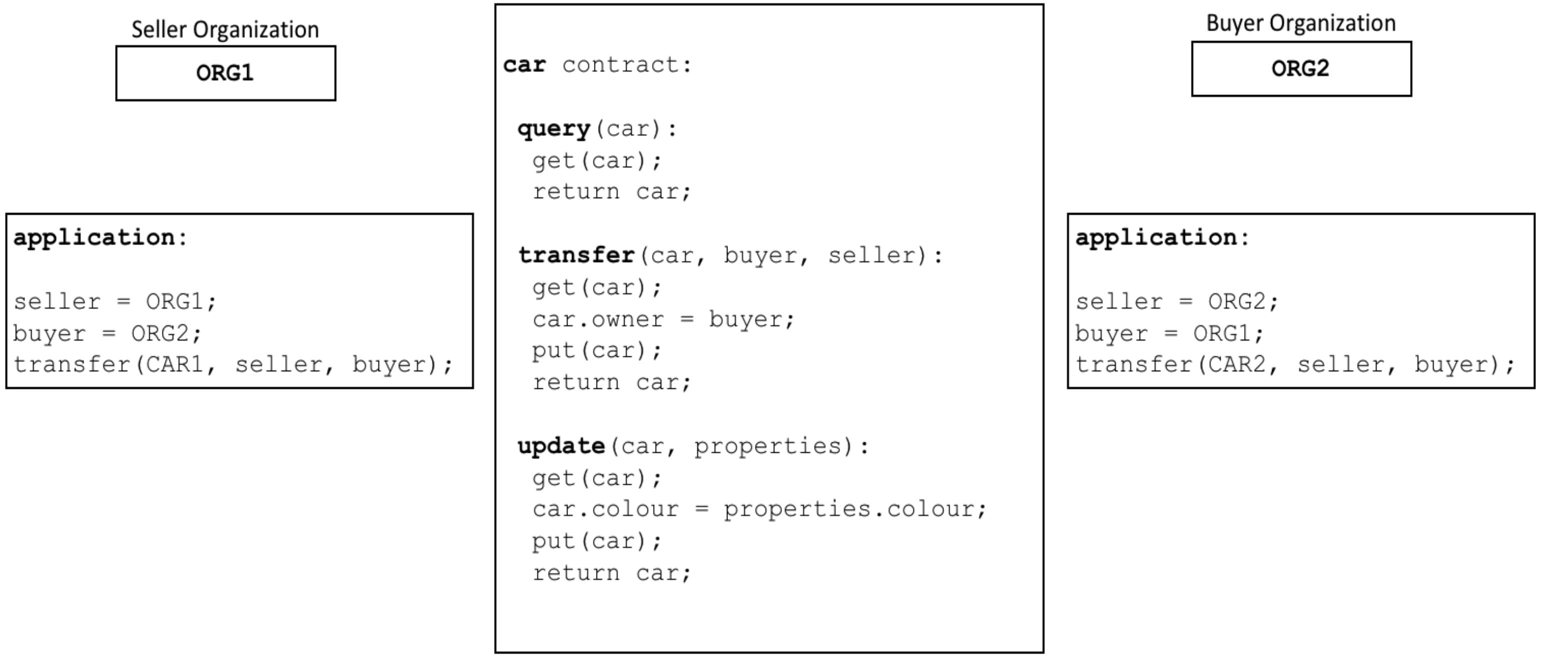
Based on multiple rounds of explicit or implicit voting, a consensus is reached on the block

Nodes add this new block to their blockchain

Picture source: M. Raikwar, D. Gligoroski, K. Kralevska, “SoK of Used Cryptography in Blockchain”

Smart Contracts

- Executable code stored in a blockchain
- Distributed execution
- Verify and enforce negotiations
- Third-party
- Transparent



Seller Organization

ORG1

```
application:

seller = ORG1;
buyer = ORG2;
transfer(CAR1, seller, buyer);
```

```
car contract:

 query(car):
  get(car);
  return car;

 transfer(car, buyer, seller):
  get(car);
  car.owner = buyer;
  put(car);
  return car;

 update(car, properties):
  get(car);
  car.colour = properties.colour;
  put(car);
  return car;
```

Buyer Organization

ORG2

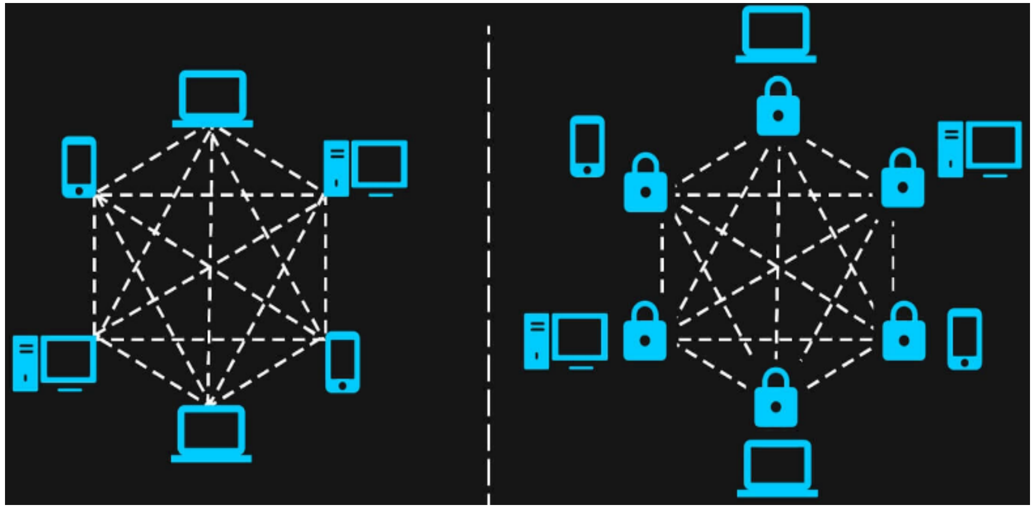
```
application:

seller = ORG2;
buyer = ORG1;
transfer(CAR2, seller, buyer);
```

Types of Blockchain

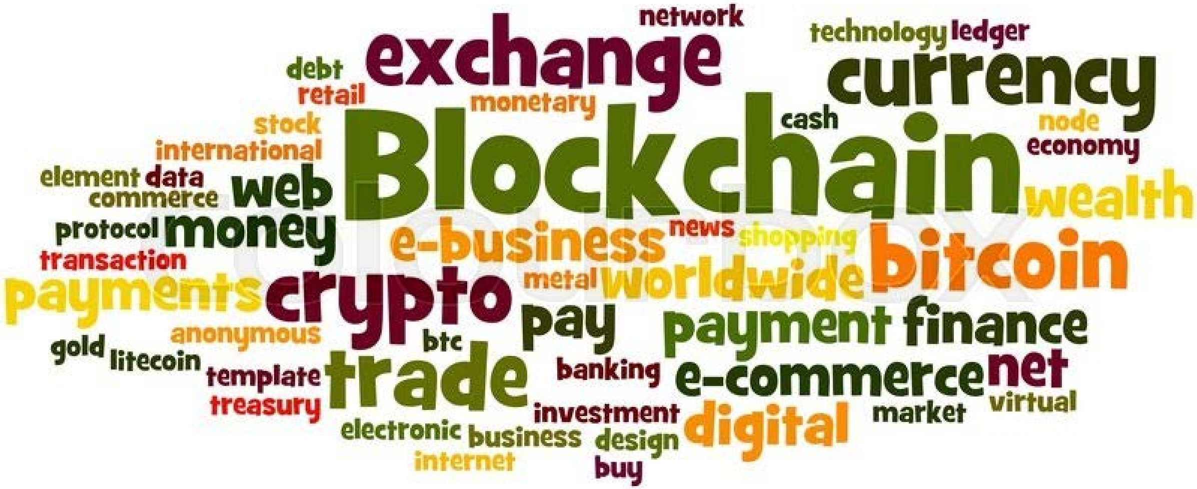
- Permission-less blockchain (Public)
  - Any one can be a participant
  - E.g., Bitcoin, Ethereum

- Permissioned blockchain (Private)
  - Only invited can become a participant
  - Maintains an access control layer
  - E.g., Hyperledger Fabric, Corda

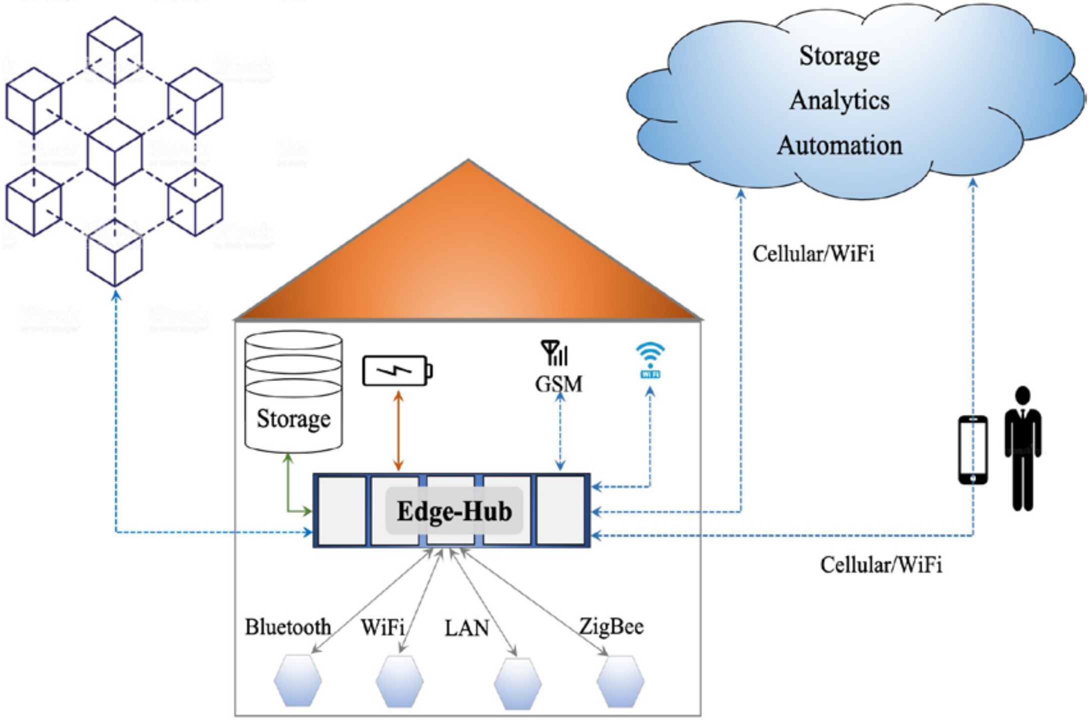


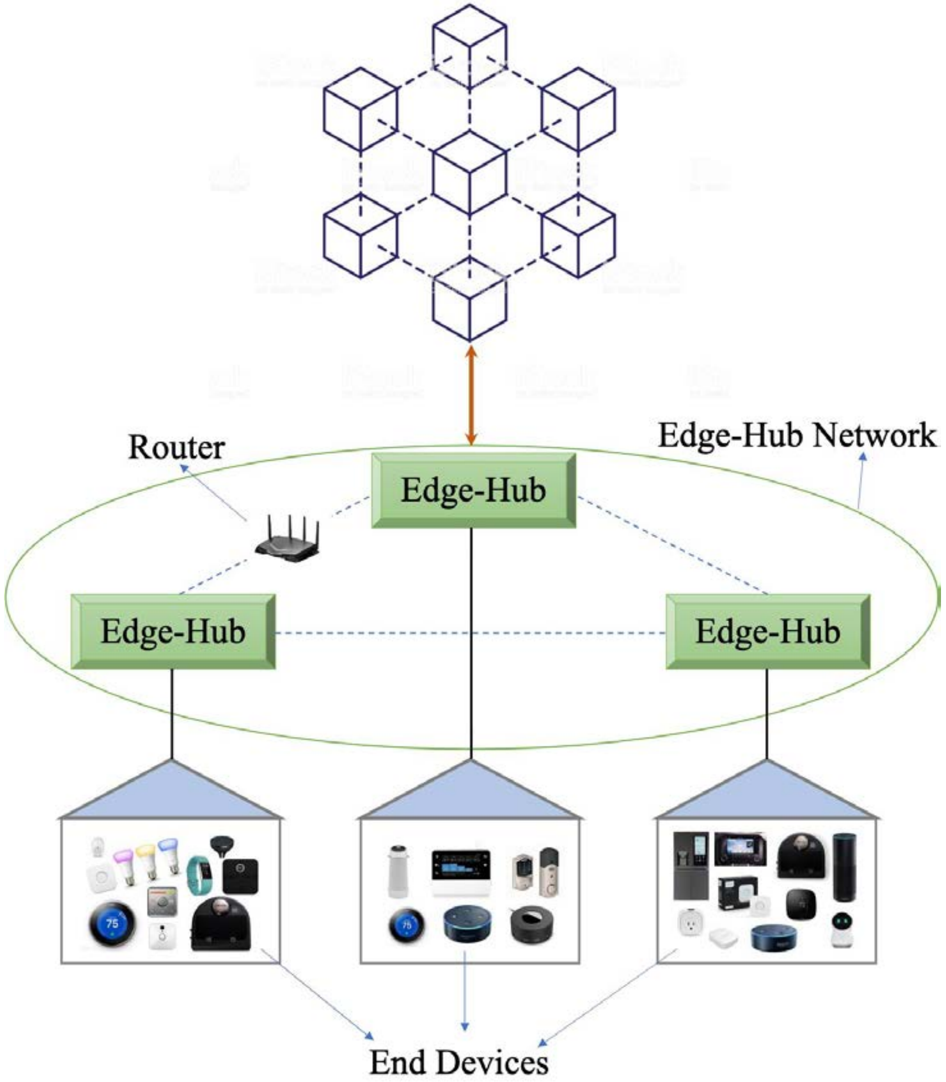
Blockchain Applications

- Major industries using blockchain:
  - Banking/Finance
  - Real state
  - Insurance
  - Healthcare
  - Legal system
- We focus on resource sharing in smart neighborhood

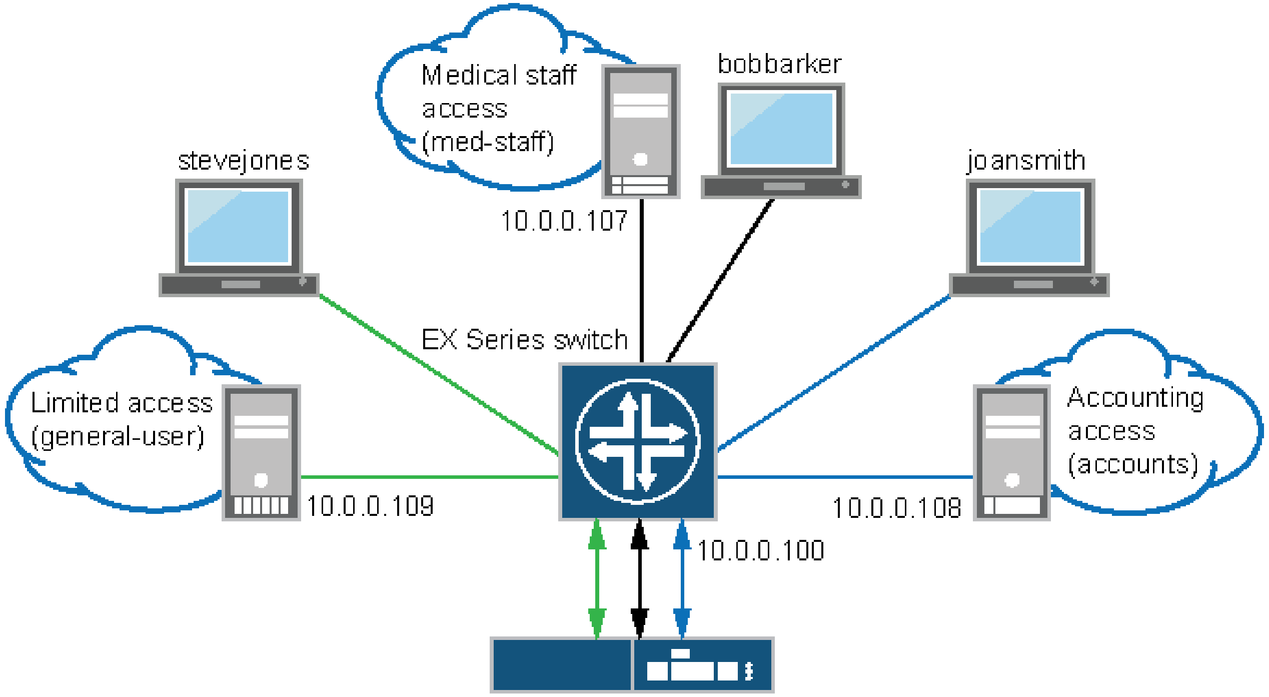


Smart Home & Neighborhood





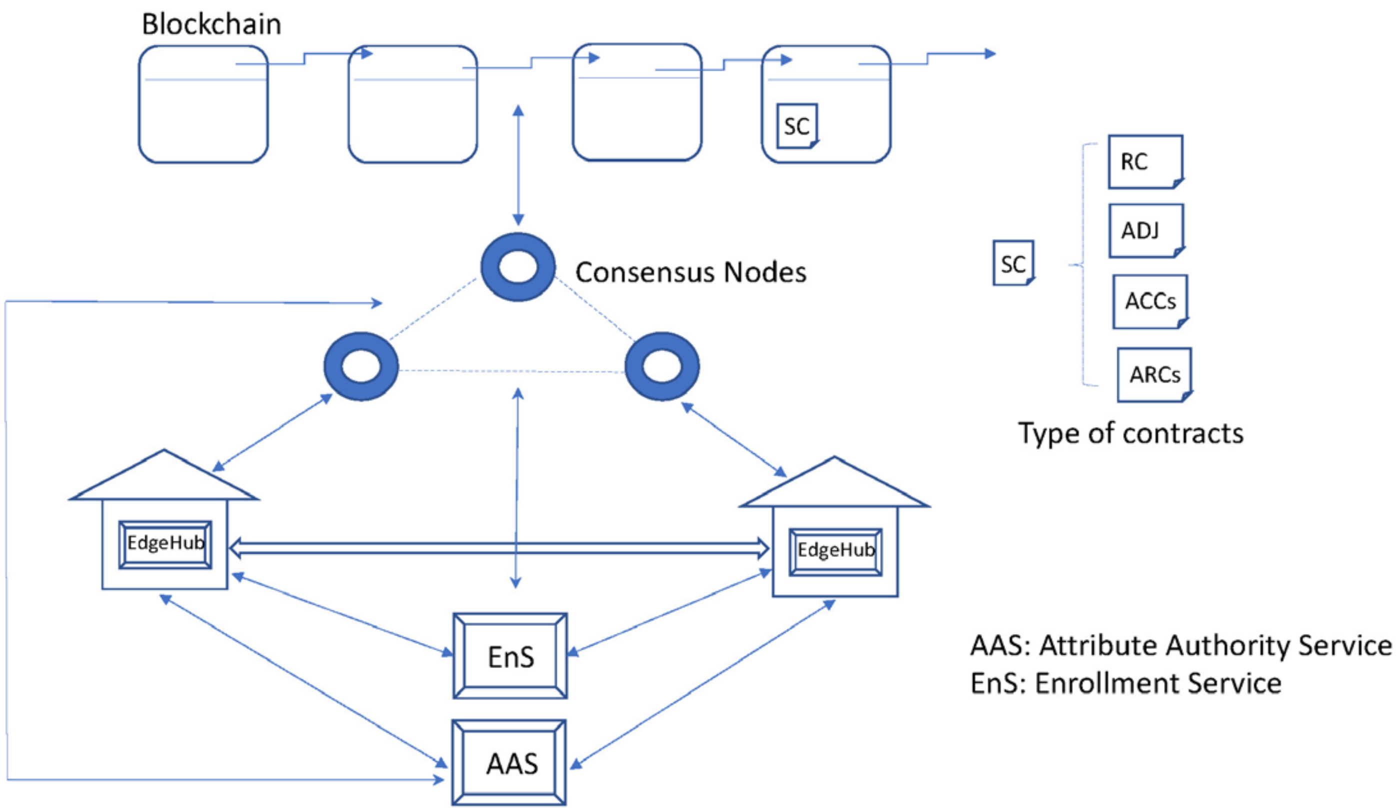
Traditional Resource Sharing



Medical staff access (med-staff)
bobbarker
stevejones
joansmith
10.0.0.107
EX Series switch
Limited access (general-user)
Accounting access (accounts)
10.0.0.109
10.0.0.108
10.0.0.100

- Issues in a centralized system
  - TA must be trusted
  - TA learns all interactions
  - High burden with conflicting tasks
  - Single point of failure
- **Goal**
  - Simulating TA based on DL system

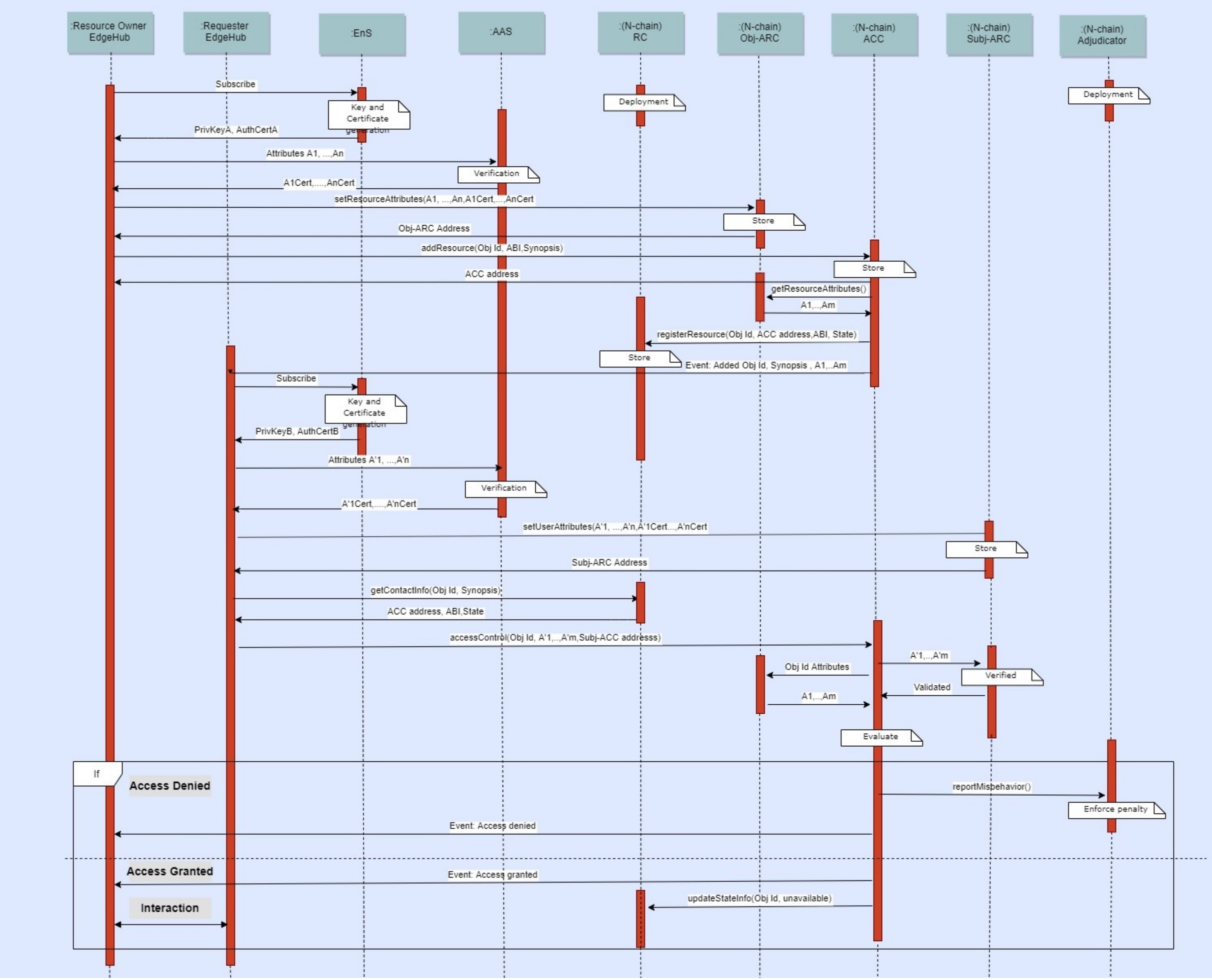
System Model



AAS: Attribute Authority Service
EnS: Enrollment Service

- Infrastructure
  - Enrollment service (EnS)
  - Attribute Authority Service (AAS)
  - Consensus nodes
  - Smart neighborhood
  - Attribute based access control
  - Smart contracts
    - Register Contract (RC)
    - Adjudicator Contract (ADJ)
    - Access Control Contract (ACC)
    - Attribute Repository Contract (ARC)

Securing access using N-Chain



UNIVERSITY OF CALGARY

Goals and Assumptions

- Security goals
  - Access to resources will be provided only to requests that satisfy the access policy of the resource, and
    - Outsiders should not be able to send a request to access the resources,
    - The requesters who are cheating should be detected, and
    - The requester who has the required attributes end up with access granted
  - Transactions do not leak more information compared to what is publicly available on the blockchain
    - Privacy (future work)

- Trust assumptions
  - Edge-Hubs are tamper-proof
  - Smart home user maybe malicious
  - C-Nodes and other authorities are honest but curious

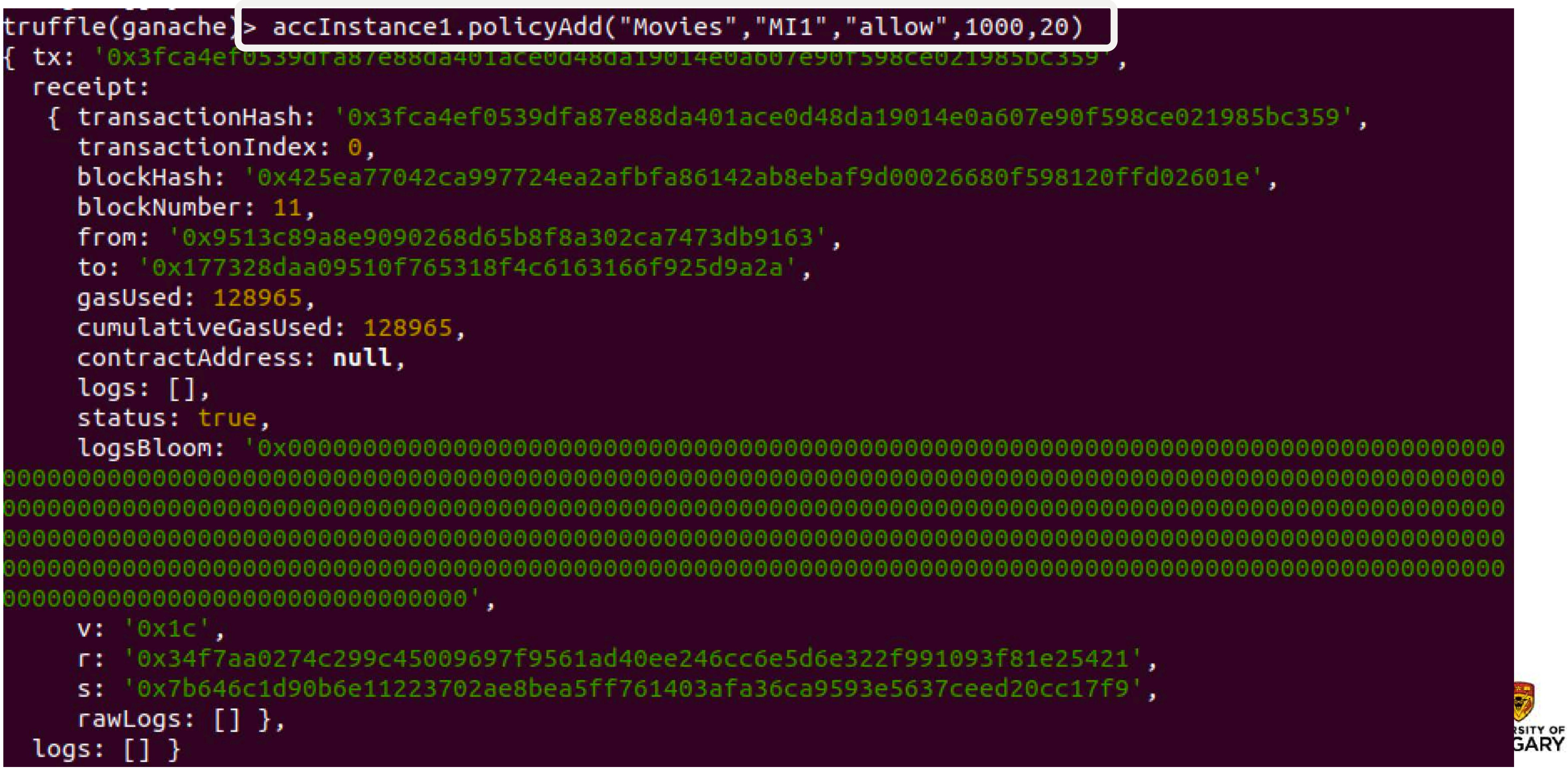
- What is stored (N-chain)
  - Resource and user information
  - Access control policies
  - Authenticated supplementary info
  - Misbehavior handling and penalties

Proof of concept implementation

- Truffle
  - Development environment
  - Testing framework – EVM & JVM
- Ganache
  - Personal blockchain for **Ethereum** and **Corda** development
  - Deploy contracts
  - Develop applications
  - Run tests and understand contract functionalities
  - Runs as a desktop application and as a command-line tool

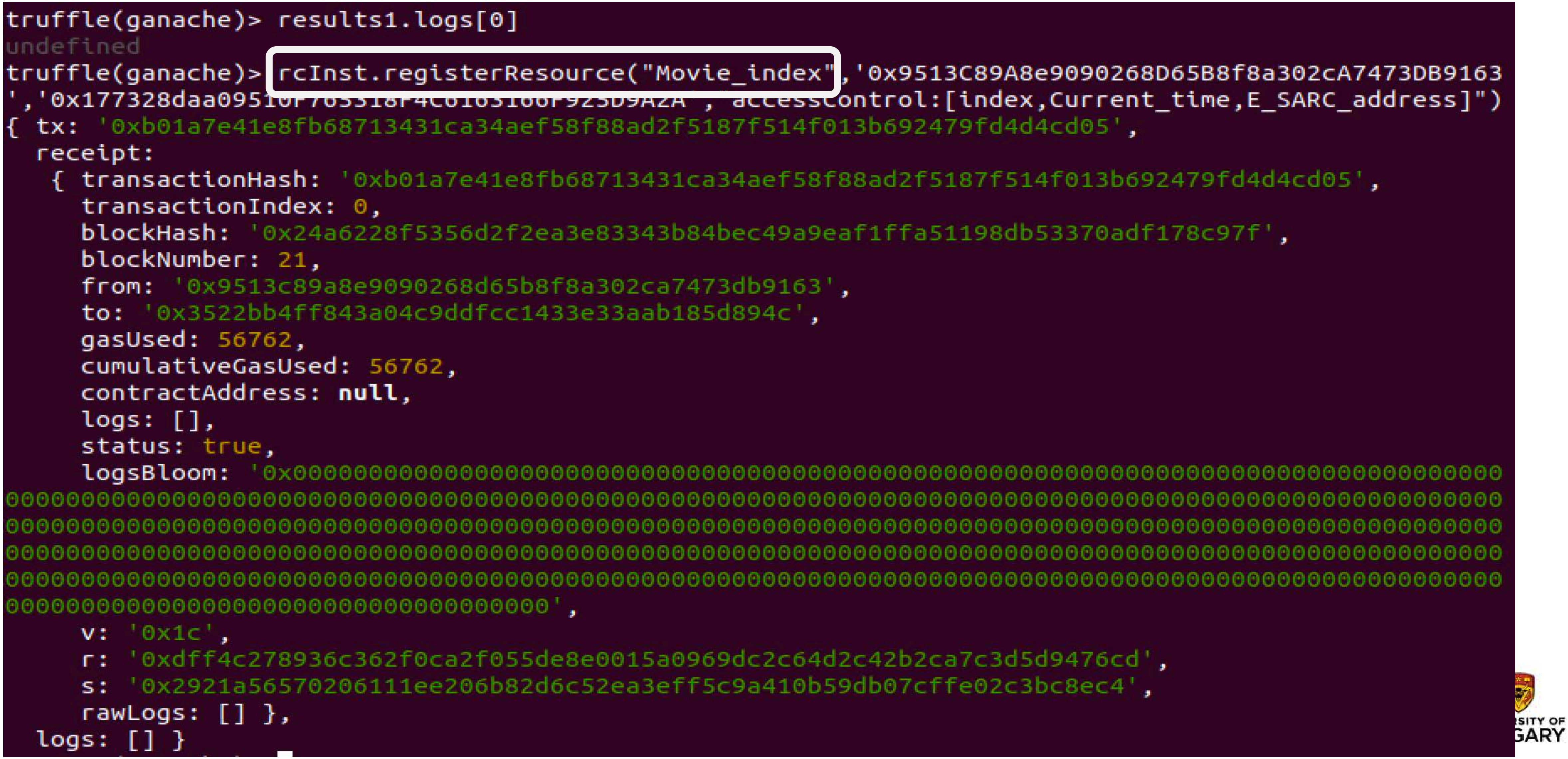
- Setup
  - Simulated a neighborhood of 5 smart homes each equipped with an EdgeHub
  - Register using EnS (RC)
  - AAS certifies the resource and EdgeHub attributes
  - Access control policies (ACC)
  - Misbehavior handling and penalties (ADJ)

Adding a policy



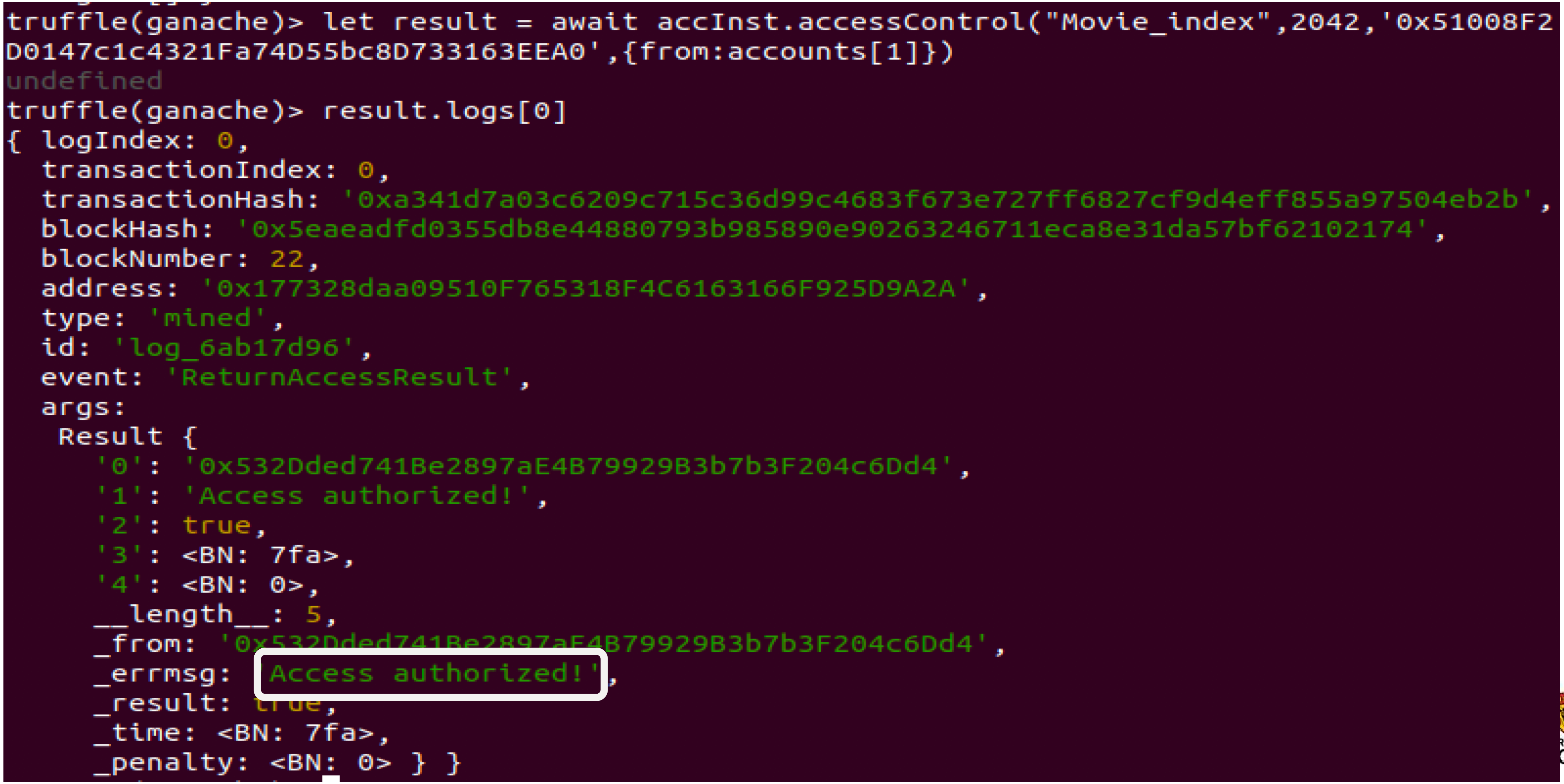
```
truffle(ganache)> accInstance1.policyAdd("Movies","MI1","allow",1000,20)
{ tx: '0x3fca4ef0539dfa87e88da401ace0d48da19014e0a607e90f598ce021985bc359',
  receipt:
   { transactionHash: '0x3fca4ef0539dfa87e88da401ace0d48da19014e0a607e90f598ce021985bc359',
     transactionIndex: 0,
     blockHash: '0x425ea77042ca997724ea2afbfa86142ab8ebaf9d00026680f598120ffd02601e',
     blockNumber: 11,
     from: '0x9513c89a8e9090268d65b8f8a302ca7473db9163',
     to: '0x177328daa09510f765318f4c6163166f925d9a2a',
     gasUsed: 128965,
     cumulativeGasUsed: 128965,
     contractAddress: null,
     logs: [],
     status: true,
     logsBloom: '0x000000000000000000000000000000000000000000000000000000000000000000000000000000000
0000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000
0000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000
0000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000
0000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000
0000000000000000000000000000000',
     v: '0x1c',
     r: '0x34f7aa0274c299c45009697f9561ad40ee246cc6e5d6e322f991093f81e25421',
     s: '0x7b646c1d90b6e11223702ae8bea5ff761403afa36ca9593e5637ceed20cc17f9',
     rawLogs: [] },
  logs: [] }
```

Registering a resource



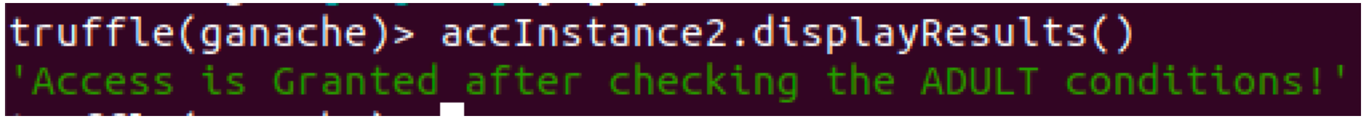
```
truffle(ganache)> results1.logs[0]
undefined
truffle(ganache)> rcInst.registerResource("Movie_index",'0x9513C89A8e9090268D65B8f8a302cA7473DB9163
','0x177328daa09510F763318F4C6163166F923D9A2A',"accessControl:[index,Current_time,E_SARC_address]")
{ tx: '0xb01a7e41e8fb68713431ca34aef58f88ad2f5187f514f013b692479fd4d4cd05',
  receipt:
   { transactionHash: '0xb01a7e41e8fb68713431ca34aef58f88ad2f5187f514f013b692479fd4d4cd05',
     transactionIndex: 0,
     blockHash: '0x24a6228f5356d2f2ea3e83343b84bec49a9eaf1ffa51198db53370adf178c97f',
     blockNumber: 21,
     from: '0x9513c89a8e9090268d65b8f8a302ca7473db9163',
     to: '0x3522bb4ff843a04c9ddfcc1433e33aab185d894c',
     gasUsed: 56762,
     cumulativeGasUsed: 56762,
     contractAddress: null,
     logs: [],
     status: true,
     logsBloom: '0x00000000000000000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000',
     v: '0x1c',
     r: '0xdff4c278936c362f0ca2f055de8e0015a0969dc2c64d2c42b2ca7c3d5d9476cd',
     s: '0x2921a56570206111ee206b82d6c52ea3eff5c9a410b59db07cffe02c3bc8ec4',
     rawLogs: [] },
  logs: [] }
```

Access Control Results

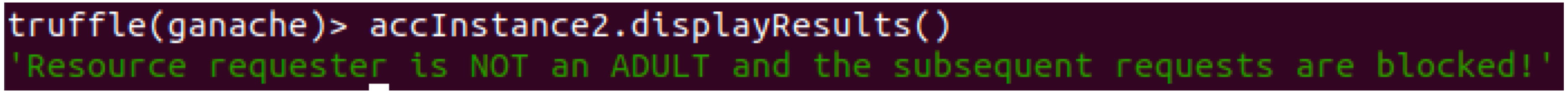


```
truffle(ganache)> let result = await accInst.accessControl("Movie_index",2042,'0x51008F2
D0147c1c4321Fa74D55bc8D733163EEA0',{from:accounts[1]})
undefined
truffle(ganache)> result.logs[0]
{ logIndex: 0,
  transactionIndex: 0,
  transactionHash: '0xa341d7a03c6209c715c36d99c4683f673e727ff6827cf9d4eff855a97504eb2b',
  blockHash: '0x5eaeadfd0355db8e44880793b985890e90263246711eca8e31da57bf62102174',
  blockNumber: 22,
  address: '0x177328daa09510F765318F4C6163166F925D9A2A',
  type: 'mined',
  id: 'log_6ab17d96',
  event: 'ReturnAccessResult',
  args:
   Result {
     '0': '0x532Dded741Be2897aE4B79929B3b7b3F204c6Dd4',
     '1': 'Access authorized!',
     '2': true,
     '3': <BN: 7fa>,
     '4': <BN: 0>,
     __length__: 5,
     _from: '0x532Dded741Be2897aE4B79929B3b7b3F204c6Dd4',
     _errmsg: 'Access authorized!',
     _result: true,
     _time: <BN: 7fa>,
     _penalty: <BN: 0> } }
```

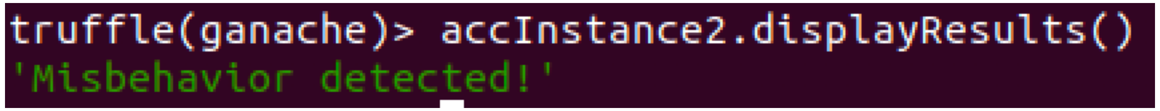
Results



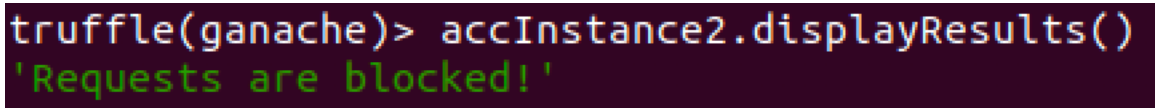
```
truffle(ganache)> accInstance2.displayResults()
'Access is Granted after checking the ADULT conditions!'
```



```
truffle(ganache)> accInstance2.displayResults()
'Resource requester is NOT an ADULT and the subsequent requests are blocked!'
```



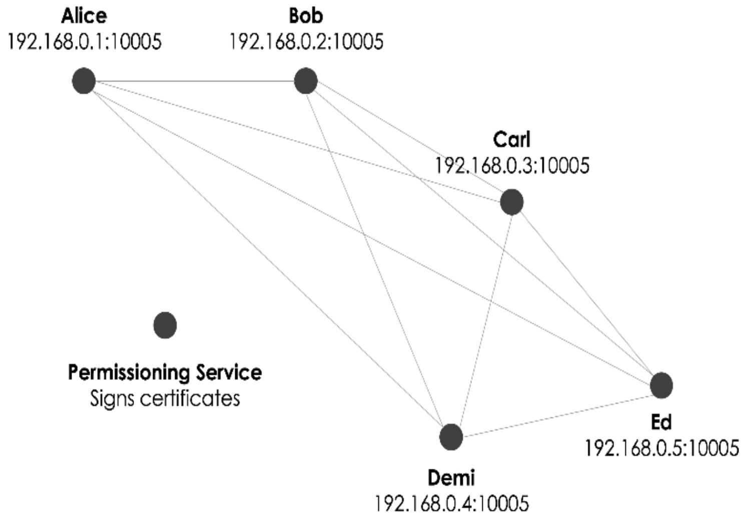
```
truffle(ganache)> accInstance2.displayResults()
'Misbehavior detected!'
```

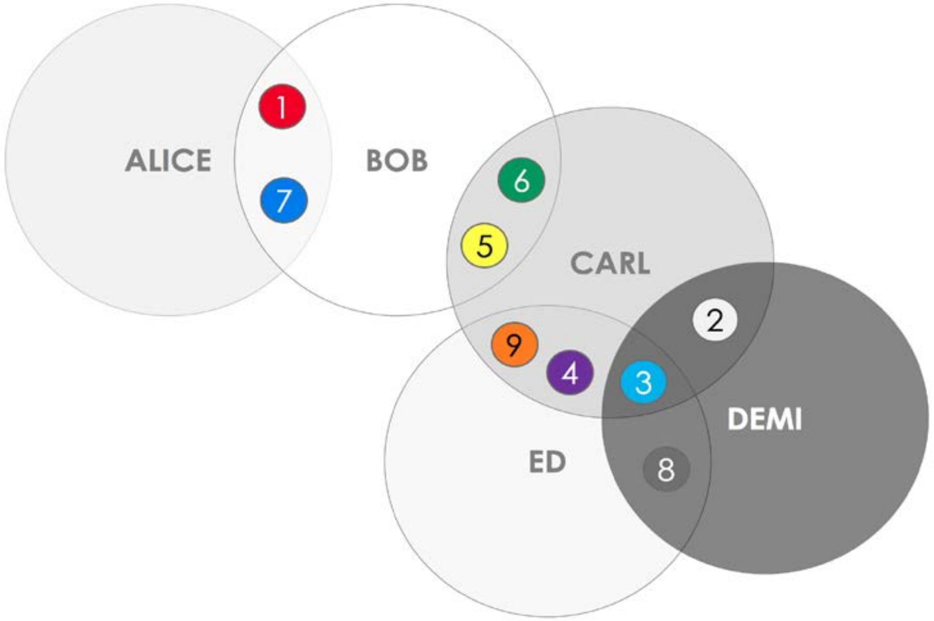


```
truffle(ganache)> accInstance2.displayResults()
'Requests are blocked!'
```

Corda

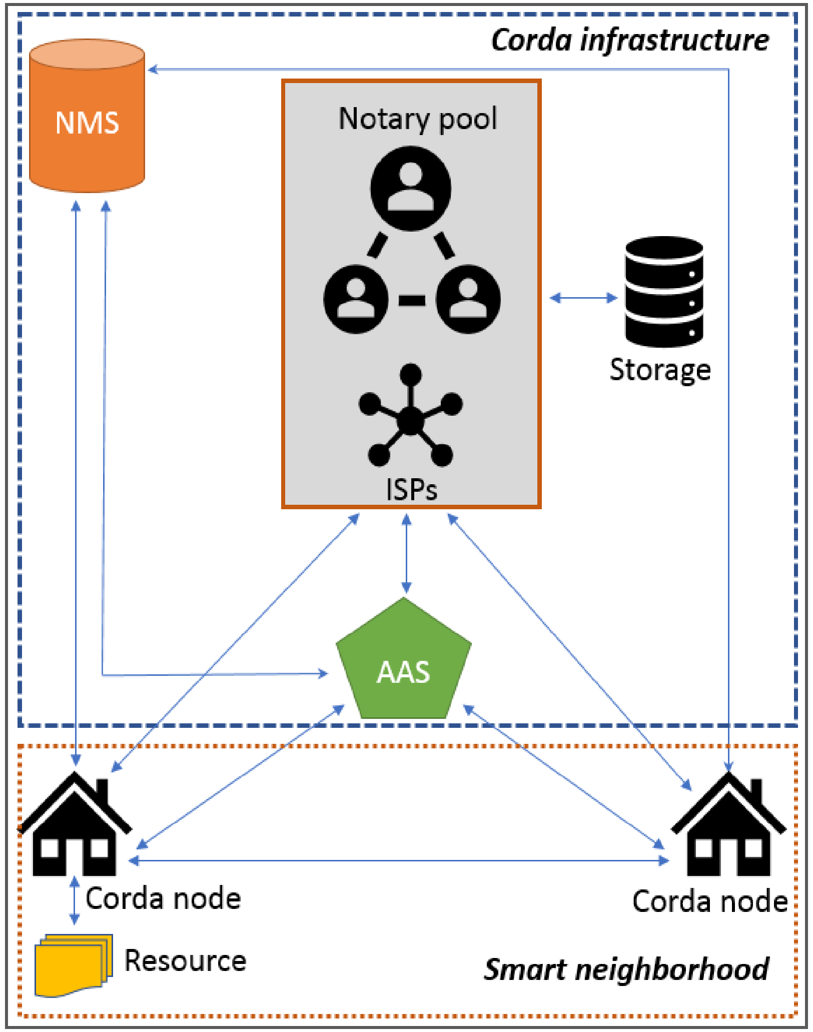
- Corda network model
  - Distributed Ledger Technology
  - No party will have everything
  - Not completely trustless
  - Not fully decentralized
  - Permissioned and p2p
  - Communication is TLS encrypted
  - Notary pool
    - Validity consensus
    - Uniqueness consensus
  - Network map service
  - Local vault





Proof of Concept Implementation using Corda

- Infrastructure entities
    - ISPs and Notary pool
        - Validity consensus
        - Uniqueness consensus
    - Event handling
    - Storage service
    - Network map service (NMS)
    - Attribute authorization service (AAS)
    - Smart contracts
        - ARC
        - ACC
        - ADJ


Corda infrastructure
NMS
Notary pool
Storage
ISPs
AAS
Corda node
Corda node
Resource
Smart neighborhood