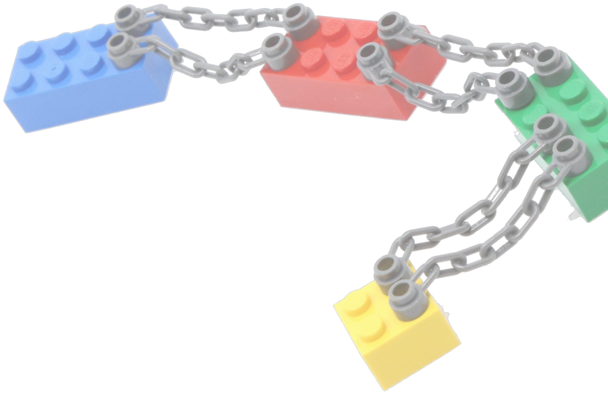
Designing Layer-2 based solutions to tackle the blockchain trilemma

Supervisors
Prof. Laura Ricci
Dr. Paolo Mori

PhD candidate
Andrea Lisi

Committee
Prof. Anna Bernasconi
Prof. Andrea Vitaletti

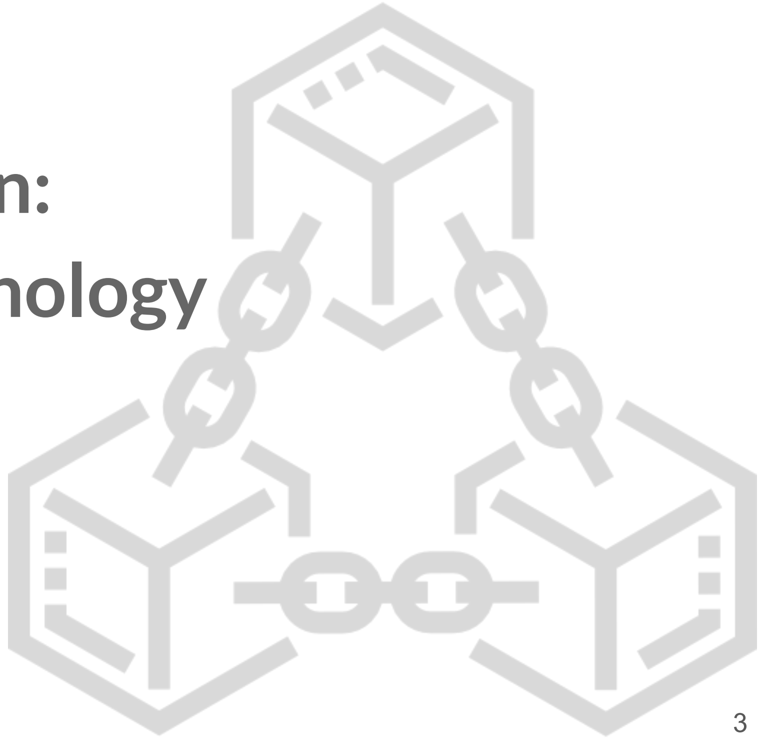
Università degli studi di Pisa
A.A. 2020 / 2021



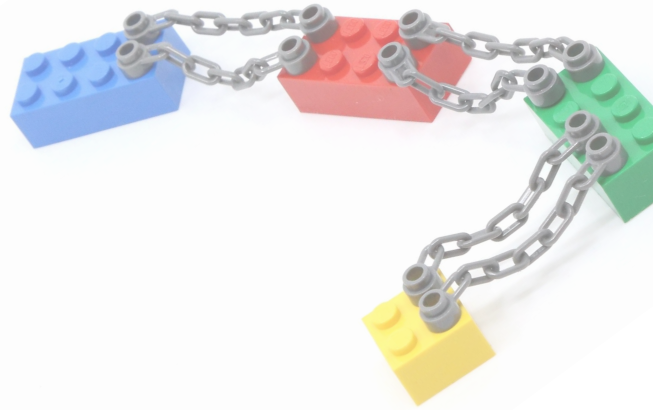
Roadmap

1. Introduction: Blockchain technology
2. State of the Art: Layer-2 technology
3. Proposal
4. Conclusions

Introduction:
Blockchain technology

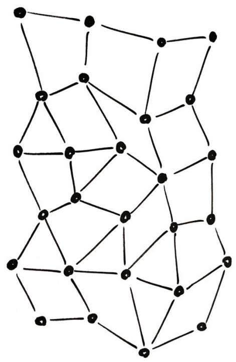


Blockchain technology



A data structure **replicated** by all the peers of a P2P network

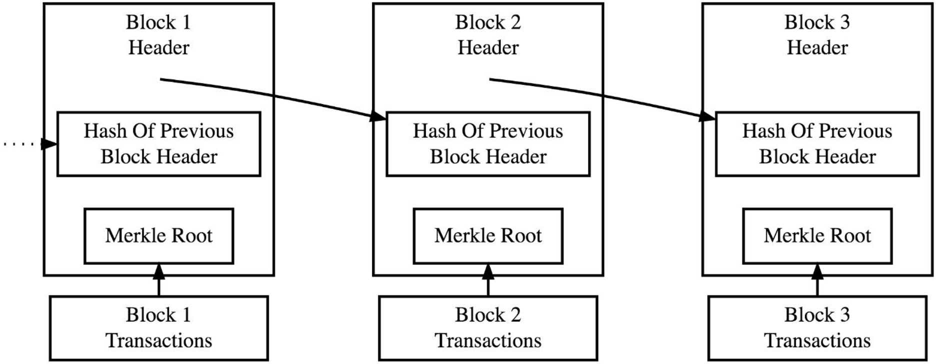
- The peers store the same data
- The peers agree on upgrades
  - This redundancy guarantees **immutability**
- Typically, freedom of connection
  - This guarantees **transparency**



Blockchain technology

The blockchain is composed by

- A list of blocks hash-linked together
- Each block is composed by
    - An Header
    - A body



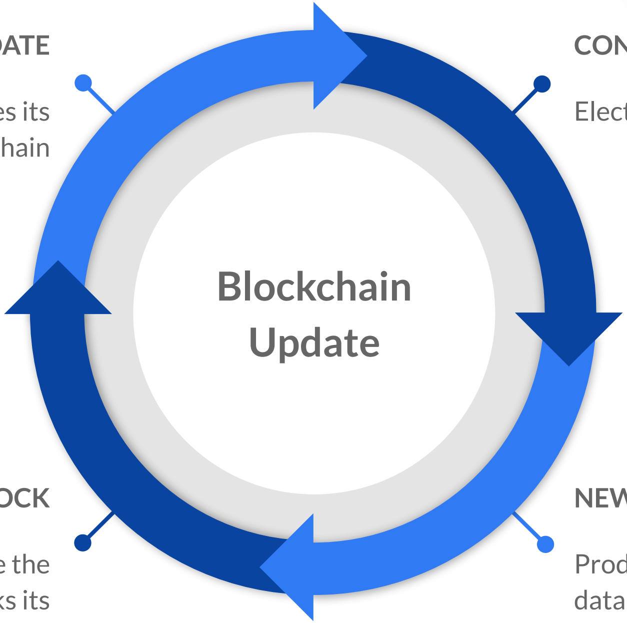
Blockchain technology

UPDATE

Each peer updates its local blockchain

CONSENSUS

Elect a block proposer

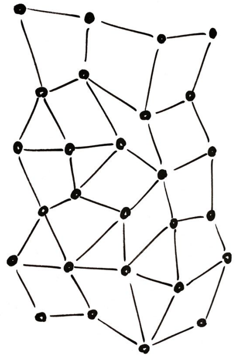


RECEIVE THE BLOCK

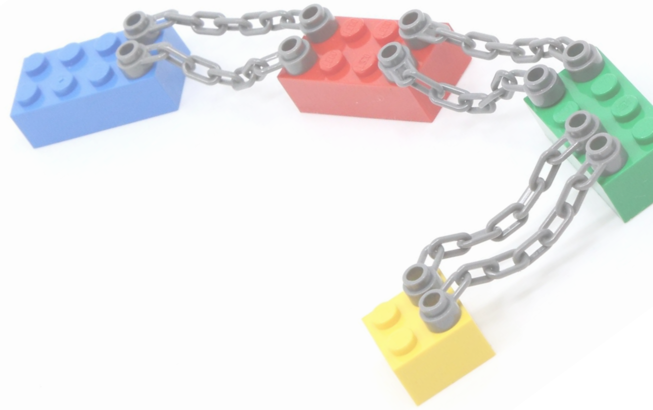
Each peer receive the block, and checks its correctness

NEW BLOCK

Produce a new block of data, and broadcast it



Bitcoin and Ethereum



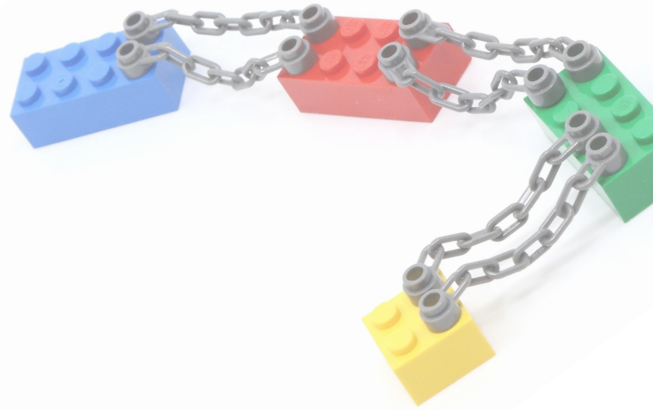
(1st Gen) Bitcoin implements the first decentralized digital currency, known as cryptocurrency, called bitcoin (BTC)

- The block body is composed by monetary **transactions**

(2nd Gen) Ethereum implements a platform for the execution of Decentralized Applications

- Execute Turing-complete programs called **smart contracts** on the Ethereum Virtual Machine

Architectural variations



Writer: A node that contributes to update the blockchain
Reader: A node that can read the blockchain and send transactions

If there are **NO** restrictions on the

- Writers => **Permissionless** / Readers => **Public**

Real world scenarios however might have different requirements

- Limit the peers to trusted ones (**Permissioned**)
- The state is not fully available (**Private**)

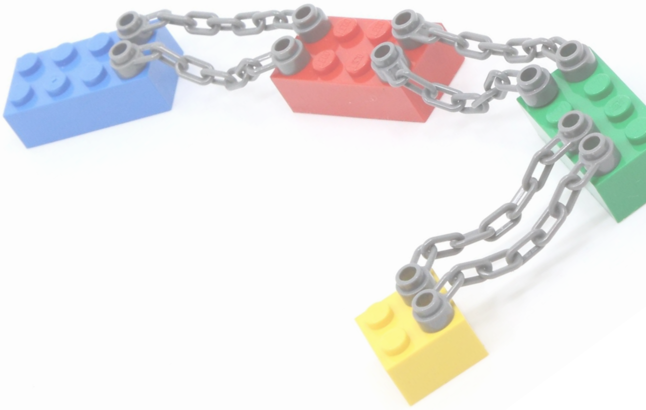
[Wüst]

Taxonomy

Restrictions on Readers

Private Permissionless Not realistic	**Private Permissioned** Required permission to participate Known identities, trusted nodes Non verifiable state Simple consensus, high scalability
Public Permissionless Can participate freely, decentralized (Pseudo) Anonymity Fully verifiable state Complex consensus, low scalability	**Public Permissioned** EOS Required permission to participate Known identities, trusted nodes Fully verifiable state Simple consensus, high scalability

Restrictions on Writers

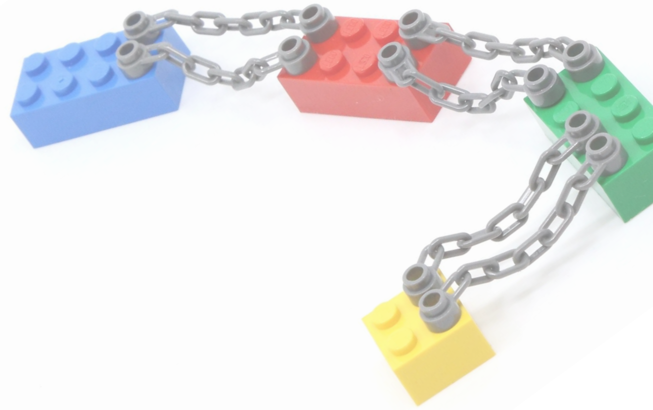


Taxonomy

Restrictions on Readers ↑ / Restrictions on Writers →		
	Private Permissionless Not realistic	**Private Permissioned** Required permission to participate Known identities, trusted nodes Non verifiable state Simple consensus, high scalability
	Public Permissionless Can participate freely, decentralized (Pseudo) Anonymity Fully verifiable state Complex consensus, low scalability	**Public Permissioned** EOS Required permission to participate Known identities, trusted nodes Fully verifiable state Simple consensus, high scalability

Restrictions on Readers

Restrictions on Writers

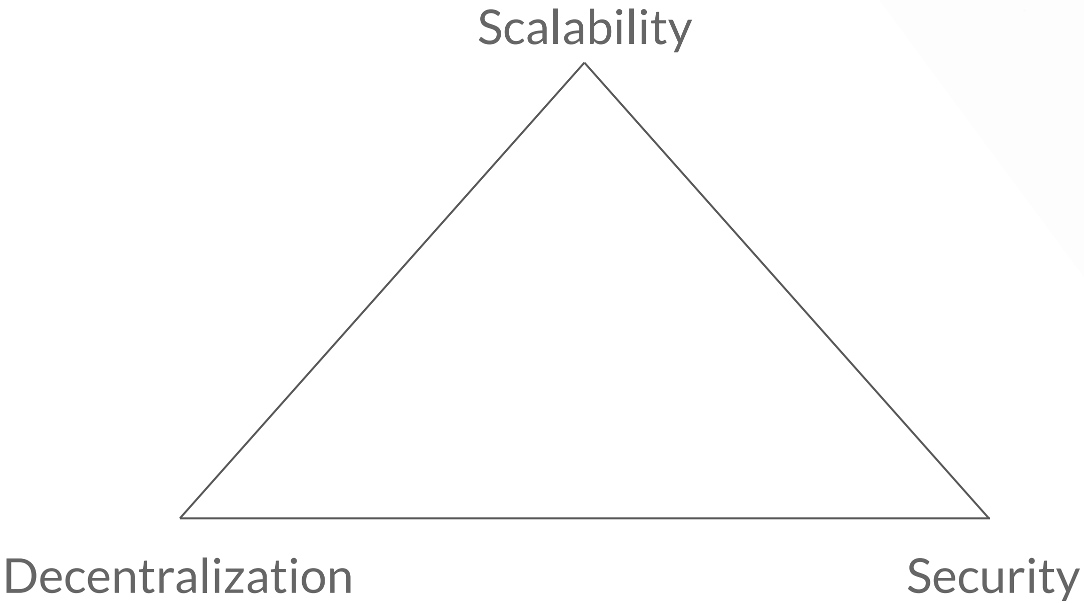


Scalability

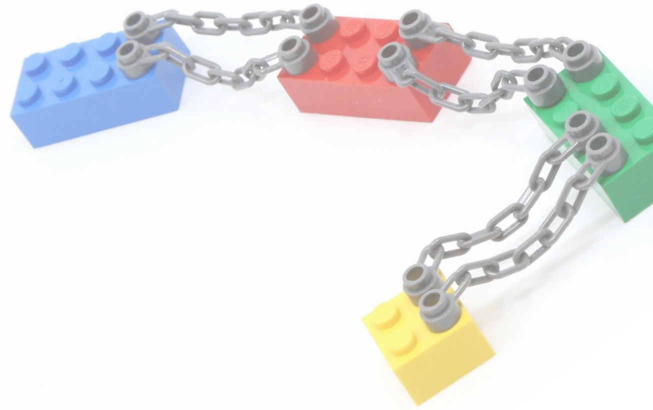
Scalability is typically connected with the permissionless / permissioned axis

- As a consequence, the consensus protocol
- An "open" consensus improves **decentralization**
    - Anybody can participate
- It improves **security**
    - There are strong cryptographic countermeasures
- But it does not **scale**
    - More the participants does not mean more the performance

Scalability



[Trilemma]

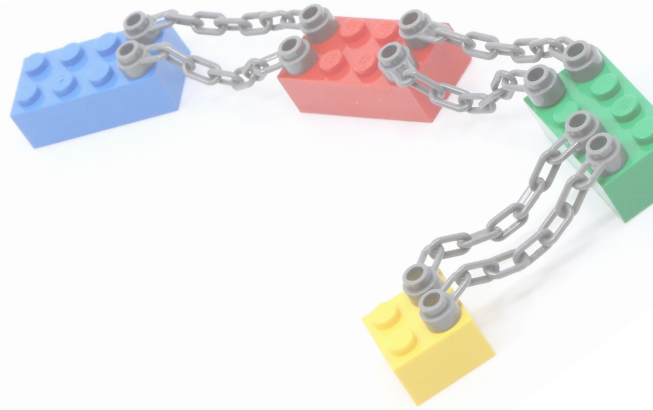


Scalability

			EOS	
Scalability	★½	★½	★★★★½	★★★★½
Decentralization	★★★★★	★★★½	★★½	★
Security	★★★★	★★★★	★★★½	★★½

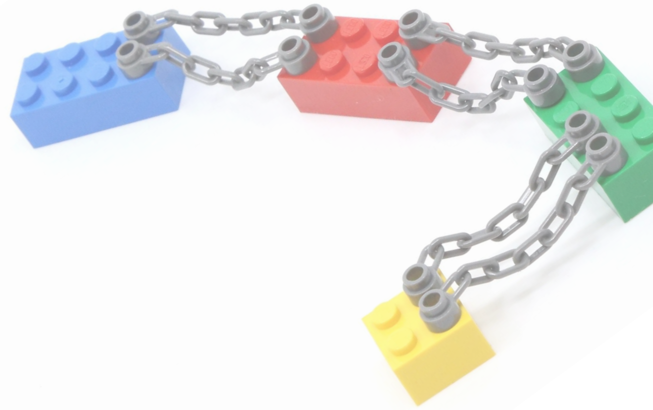
[Garriga]

Privacy



Privacy is typically connected with the public / private axis

- In a public network data have not privacy
  - So GDPR-sensitive data should not be stored
- Such data can be encrypted
  - But verifiability will be lost, that is a key feature of integrating the blockchain in a process

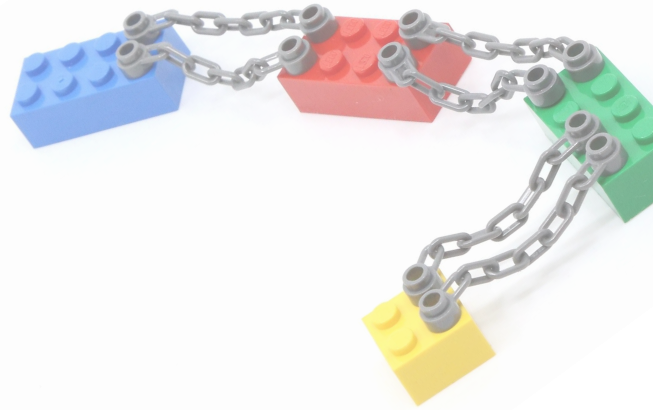


Privacy

			EOS	
Scalability	★½	★½	★★★★½	★★★★½
Decentralization	★★★★★	★★★½	★★½	★
Security	★★★★	★★★★	★★★½	★★½
Privacy	★★	★★	★★	★★★★

[Garriga]

Towards Layer-2



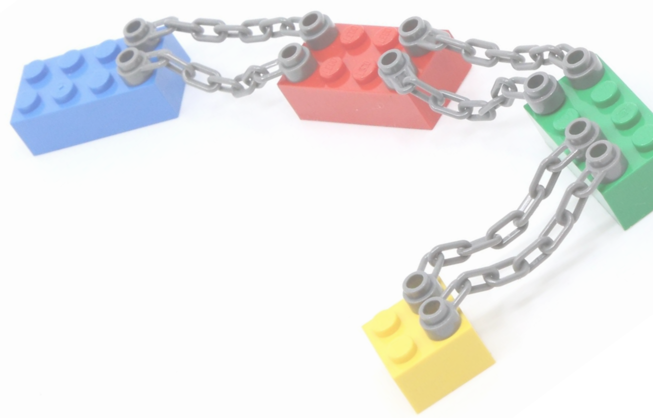
Trade-off within the blockchain trilemma

How can we improve scalability and privacy of public permissionless compositions?

This is where **Layer-2 technology** comes in

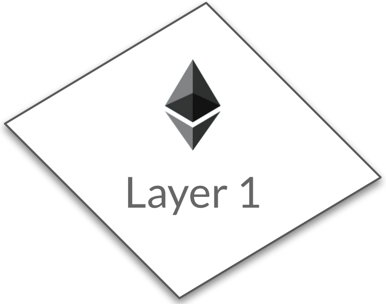
Layer-2 technology

Layer-2 technology (1)

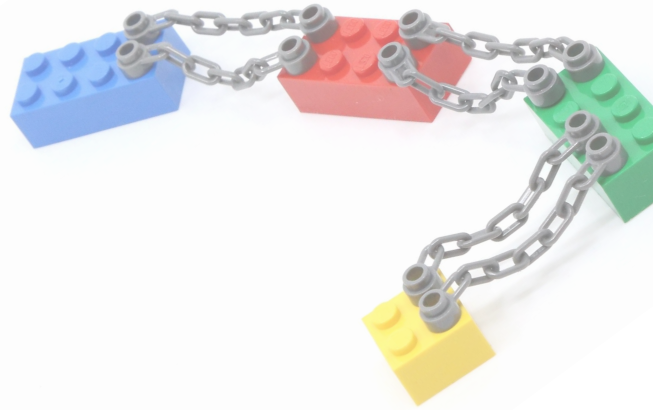


Any change to the blockchain protocol is labeled as **Layer-1**

- Block size, consensus algorithm…[Zhou]
- Any change is an important point of debate within the community
- And the impact on the scalability is relatively small
- Unless the overall architecture and protocol is drastically different

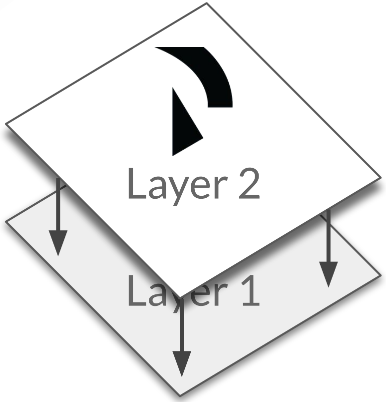


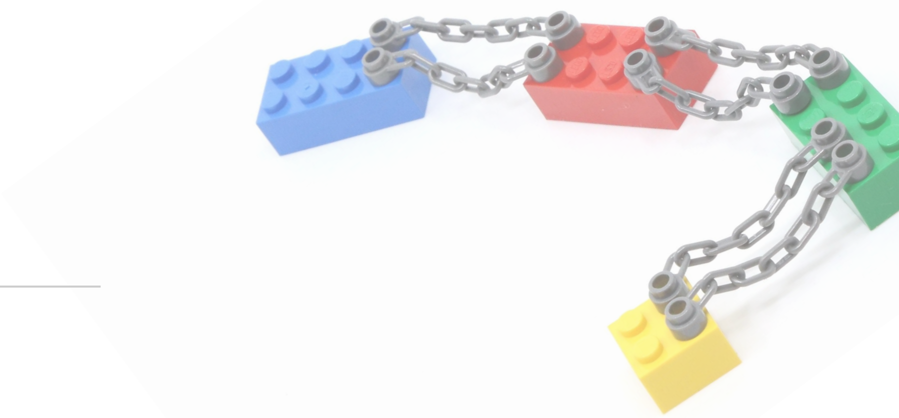
Layer-2 technology (2)



Layer-2 are those technologies improving a blockchain, e.g. scalability or privacy, with off-chain operations <u>bound to the blockchain</u>

- Operations are not subjected to consensus latencies
- Are not recorded
- But the link with the blockchain ensures auditability

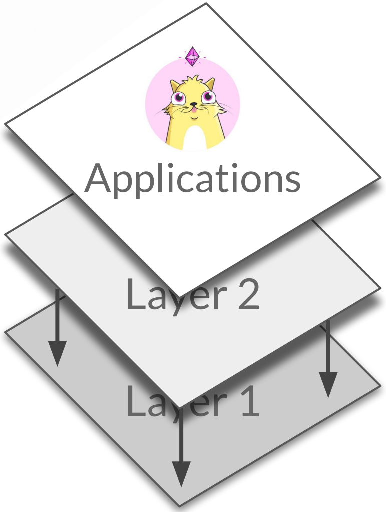




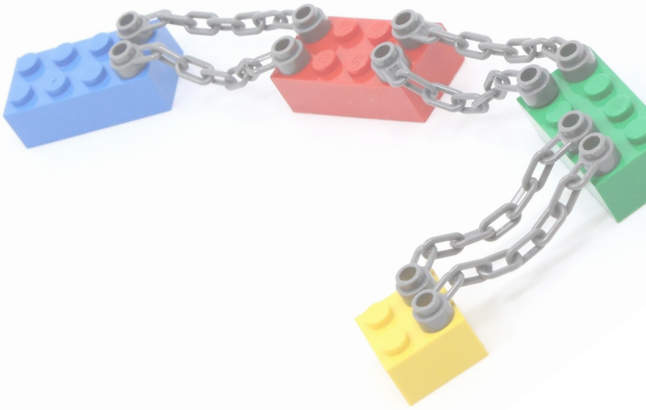
Layer-2 technology (3)

Layer-2 are those technologies improving a blockchain, e.g. scalability or privacy, with off-chain operations <u>bound to the blockchain</u>

- Operations are not subjected to consensus latencies
- Are not recorded
- But the link with the blockchain ensures auditability

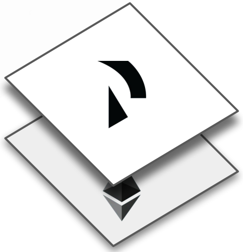


Layer-2 technology



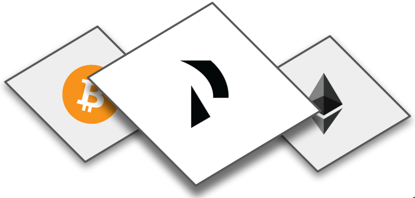
There are 3 Layer-2 high level models:

- State Channel
- Sidechains
- Off-chain computation and/or storage

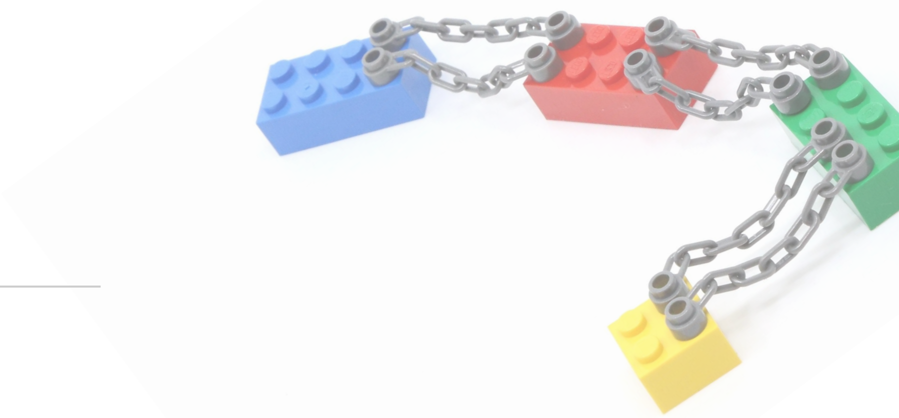


All of these models can

- Work on top of a blockchain protocol, **Single-chain**
- Connect more blockchains, **Multi-chain**

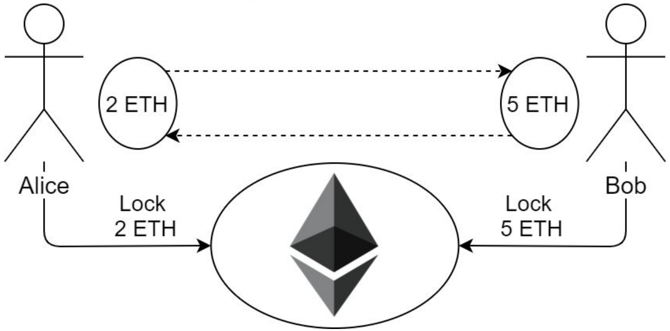


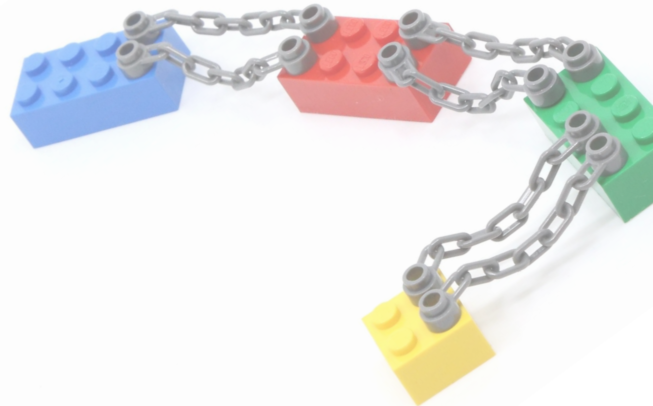
State channels



A state channel is a virtual connection between two peers

- The peers create a channel to share a state
  - The operation is **bound** with a transaction
- Each peer can update the state
  - The update happens off-chain
- Either peer can close the connection
  - **Bound** with a transaction

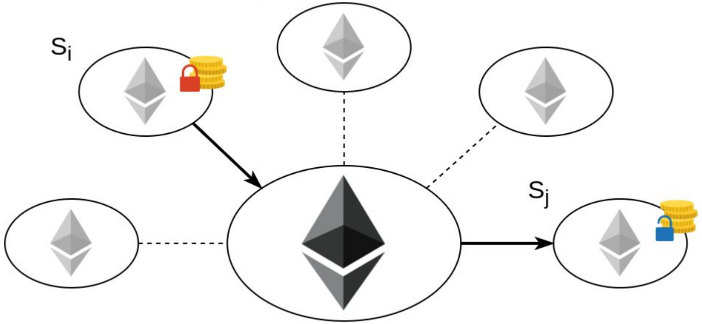




Sidechains

In a sidechain model the state is split among different chains

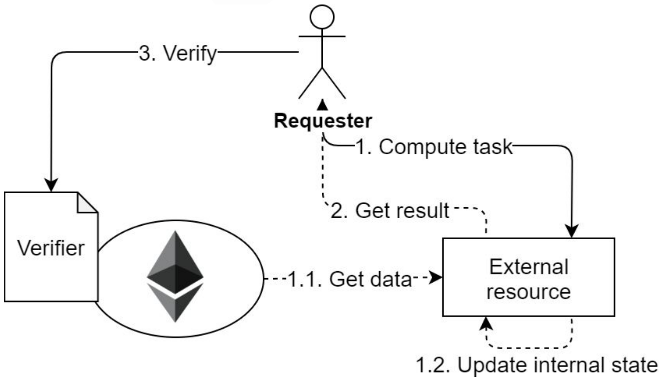
- Each sidechain S_i store a portion of a state (e.g. a subset of transactions)
- Its block headers are **bound** to a Main Chain
  - A Main Chain for many sidechains
- Hierarchy of validators
- Asset may go from S_i to S_j
  - Locked in S_i, unlocked in S_j



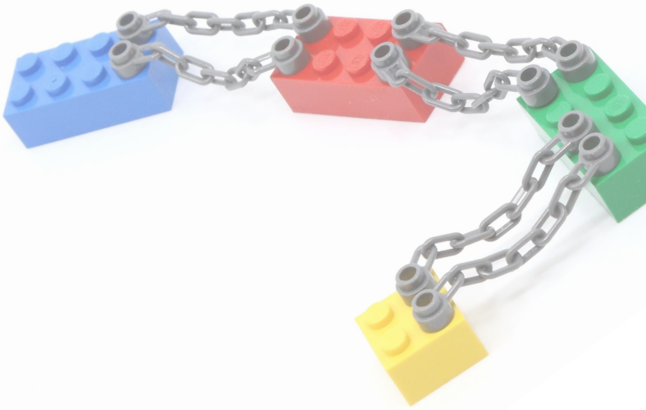
Off-chain computation

The off-chain computation model is based on relief the smart contracts from heavy computation

- A computation is performed “outside”
- A smart contract has to verify the outcome
  - Or a “proof”
- The verification has to be cheaper
  - **Binds** the outcome on the blockchain
- Similar approach for the storage

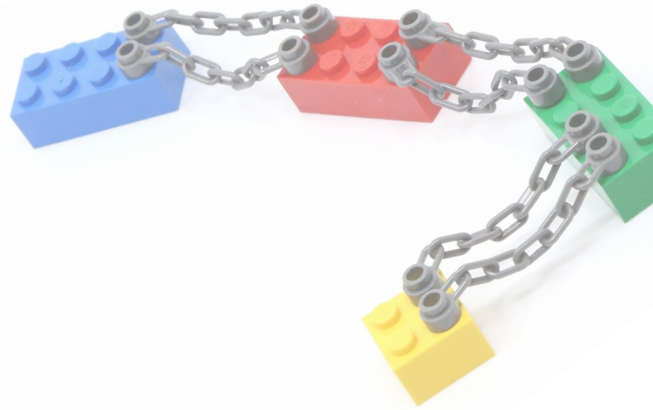


Lightning Network



The Lightning Network[Poon] is a layer-2 solution for Bitcoin based on state channels

- Known as payment channels
- Two users, A and B, lock funds in the channel
- Exchange funds within the channel
  - An exchange is known as **commitment**
  - No fees, fast time, no public record
- Either user can close the channel at any time



Lightning Network

Commitments

S0

Comm-1

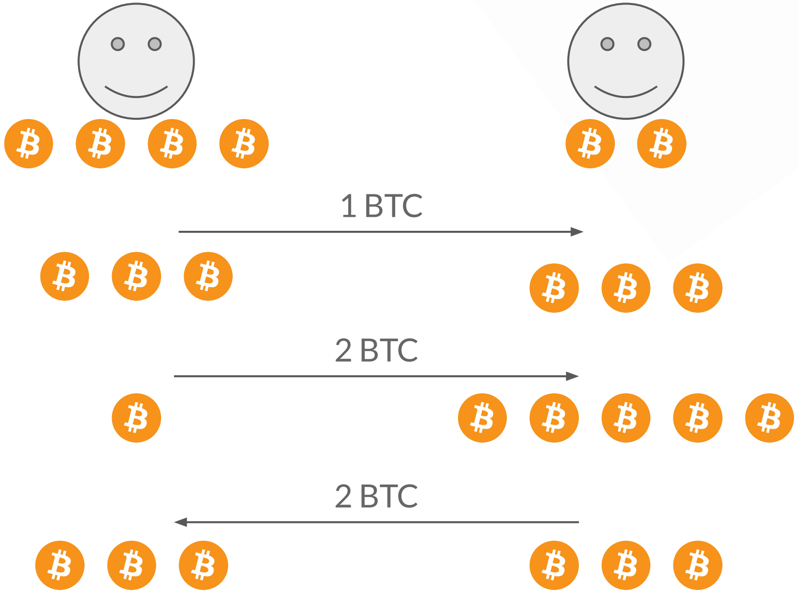
S1

Comm-2

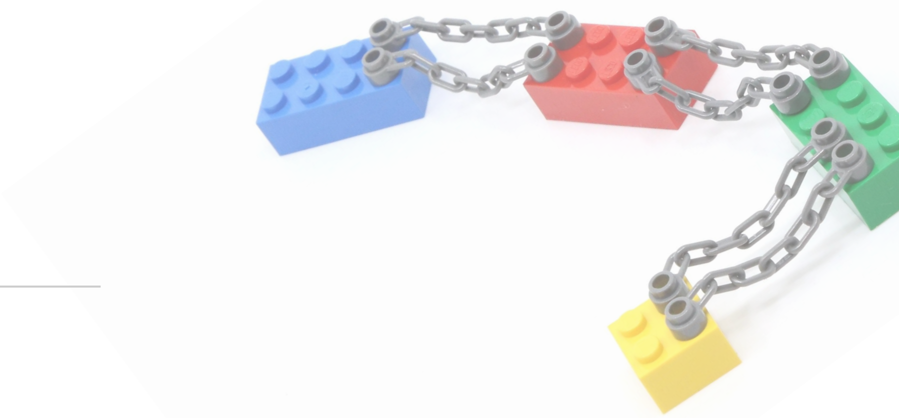
S2

Comm-3

S3

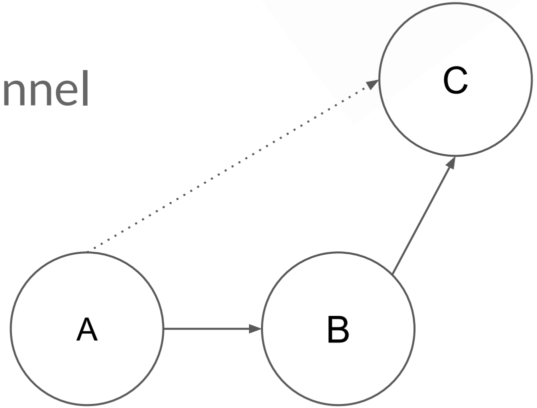


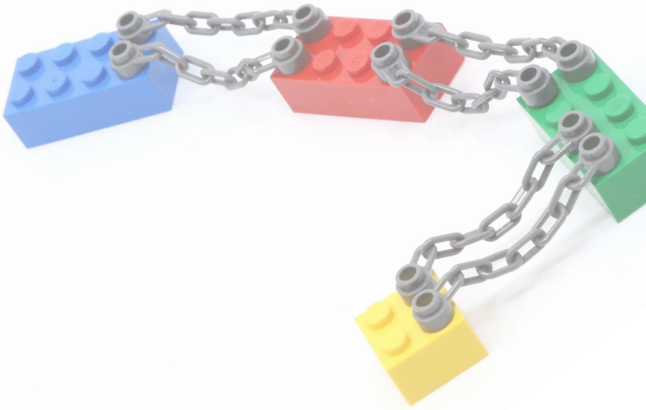
Lightning Network



If A wants to send BTC to C

- A and C need to open a new channel
    - A may not have enough funds
        - A needs to close another channel
- B and C are connected
    - A can exploit that connection

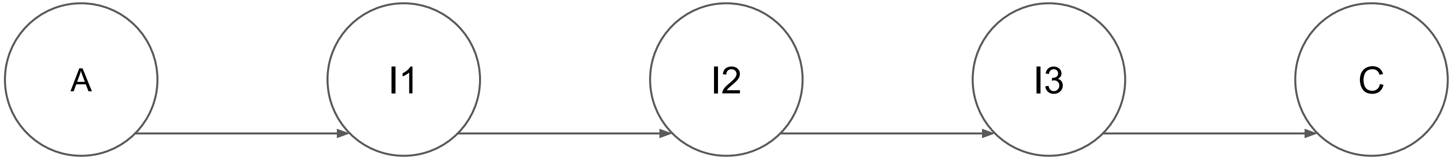




Lightning Network

A needs to find a path to C

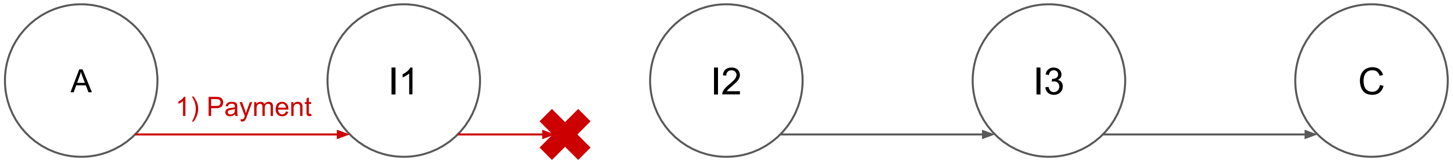
- Assuming the path A, I1, I2, I3, C

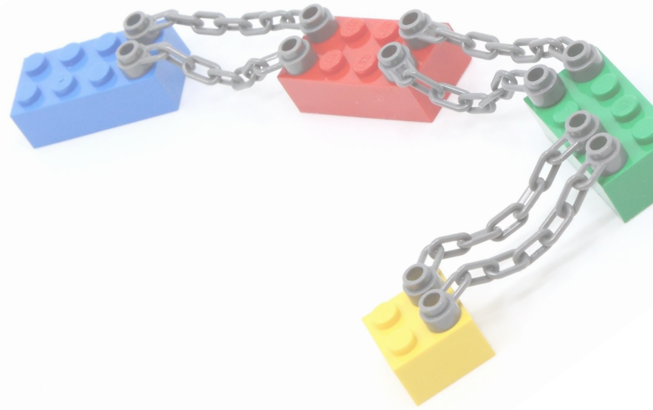


Lightning Network

A needs to find a path to C

- Assuming the path A, I1, I2, I3, C
- If A pays I1
  - I1 may not forward the payment

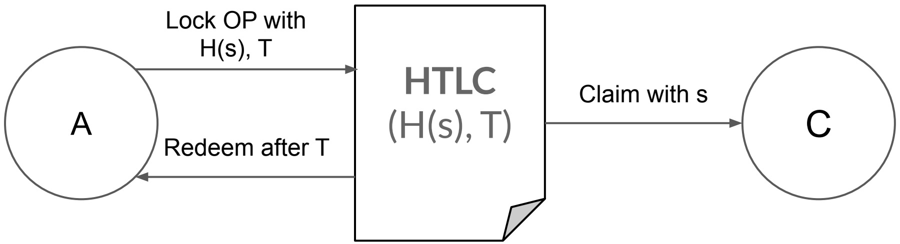




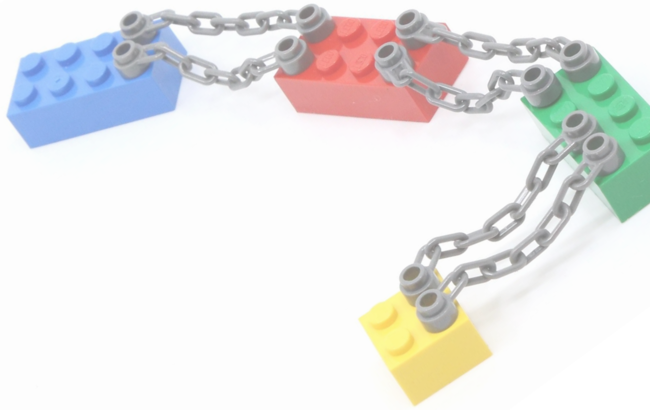
Lightning Network

Hash Time Lock Contracts (HTLC), contracts that exploit

- **Hashlock:** lock an operation unless the preimage of a hash value is revealed
- **Timelock:** lock an operation within a deadline



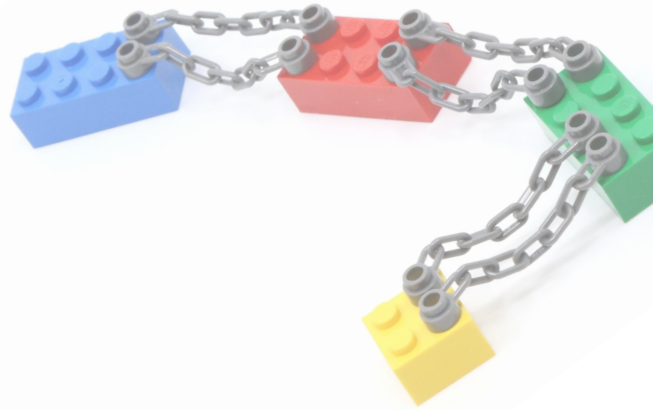
Lightning Network

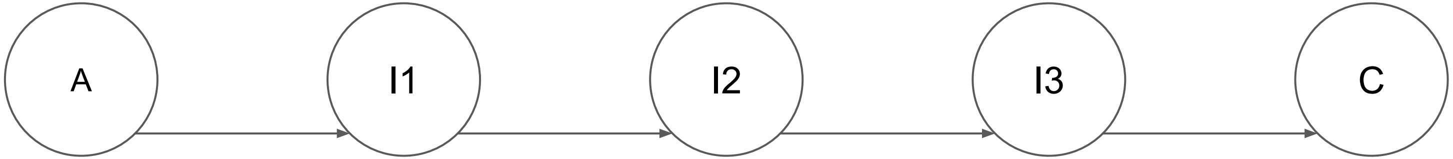


A needs to find a path to C

- Assuming the path A, I1, I2, I3, C
    a. Generation of secret and hashlock
    b. Hashlock communication
    c. HTLC setup route
    d. Secret revelation and payment route

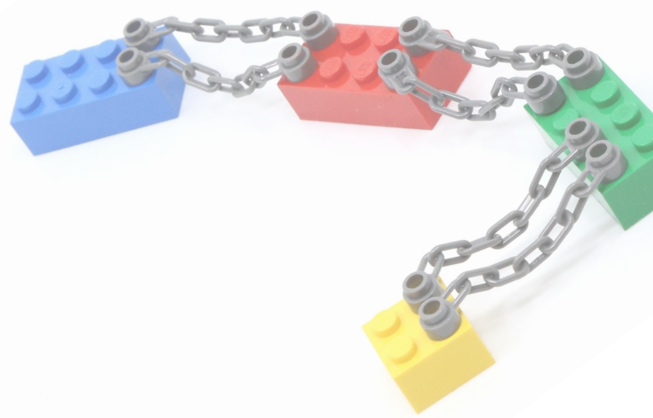
Lightning Network

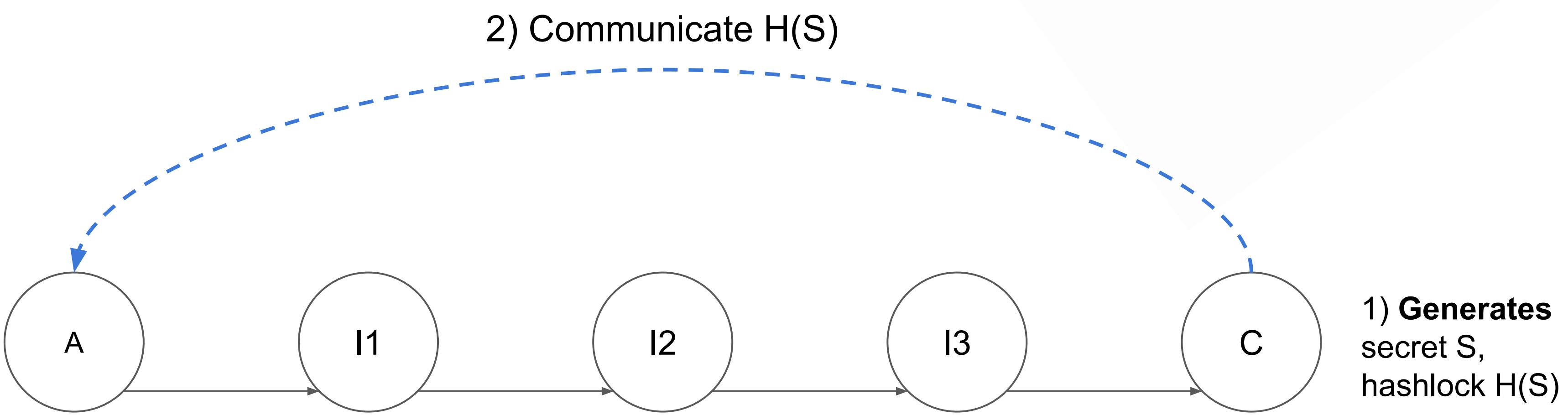




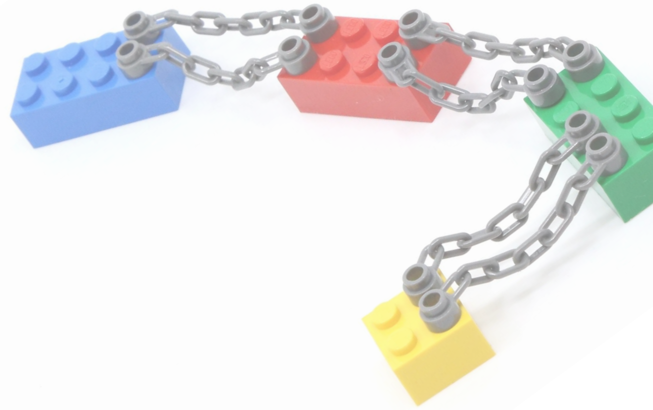
1) **Generate** secret S, hashlock H(S)

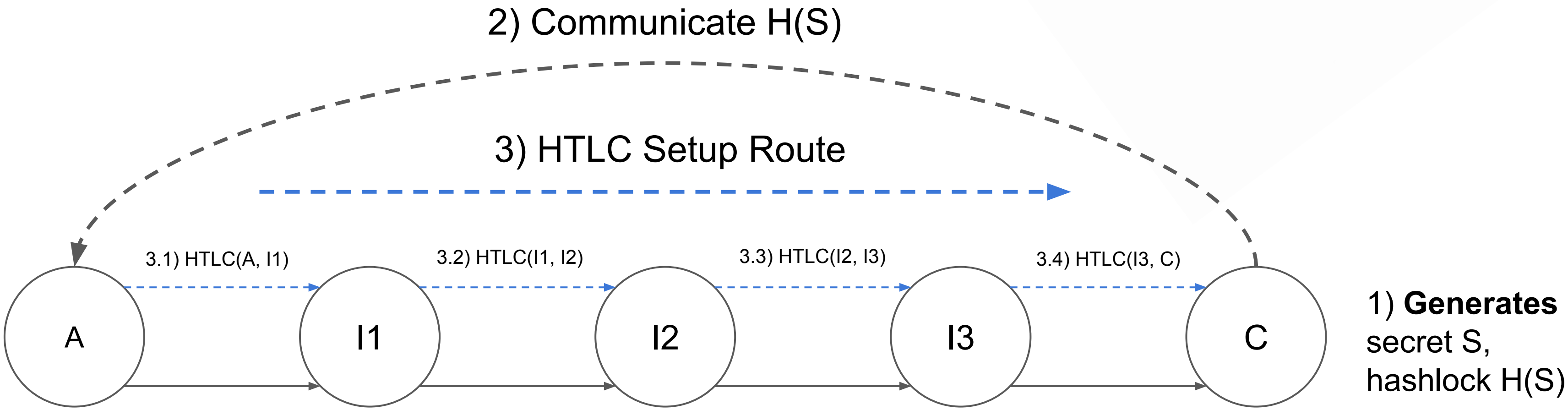
Lightning Network



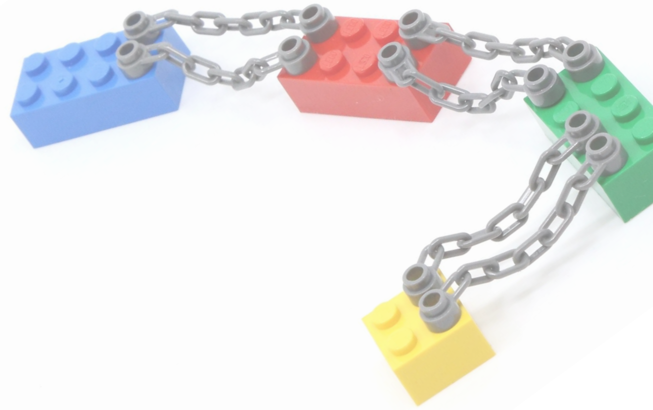


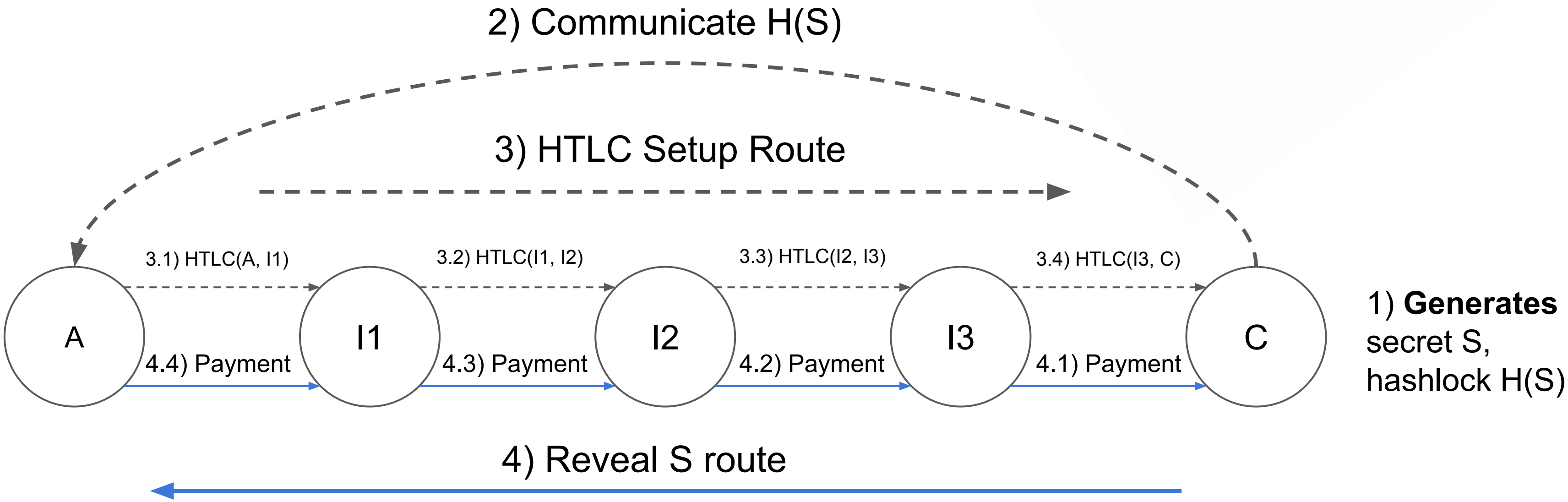
Lightning Network

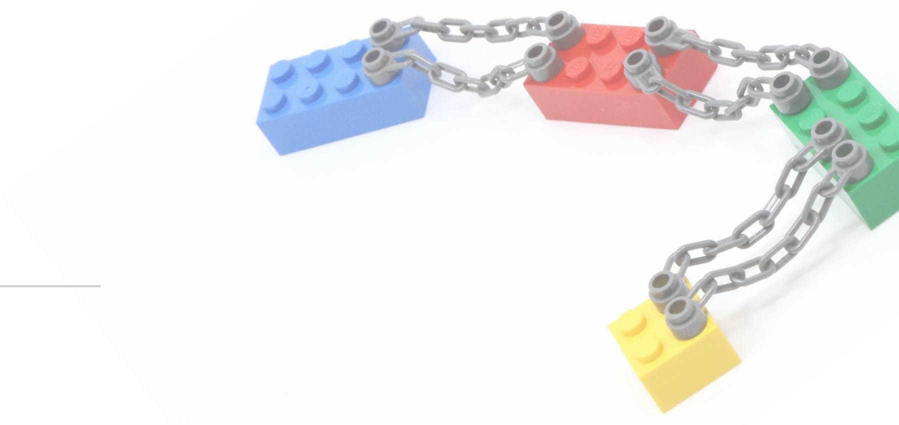




Lightning Network







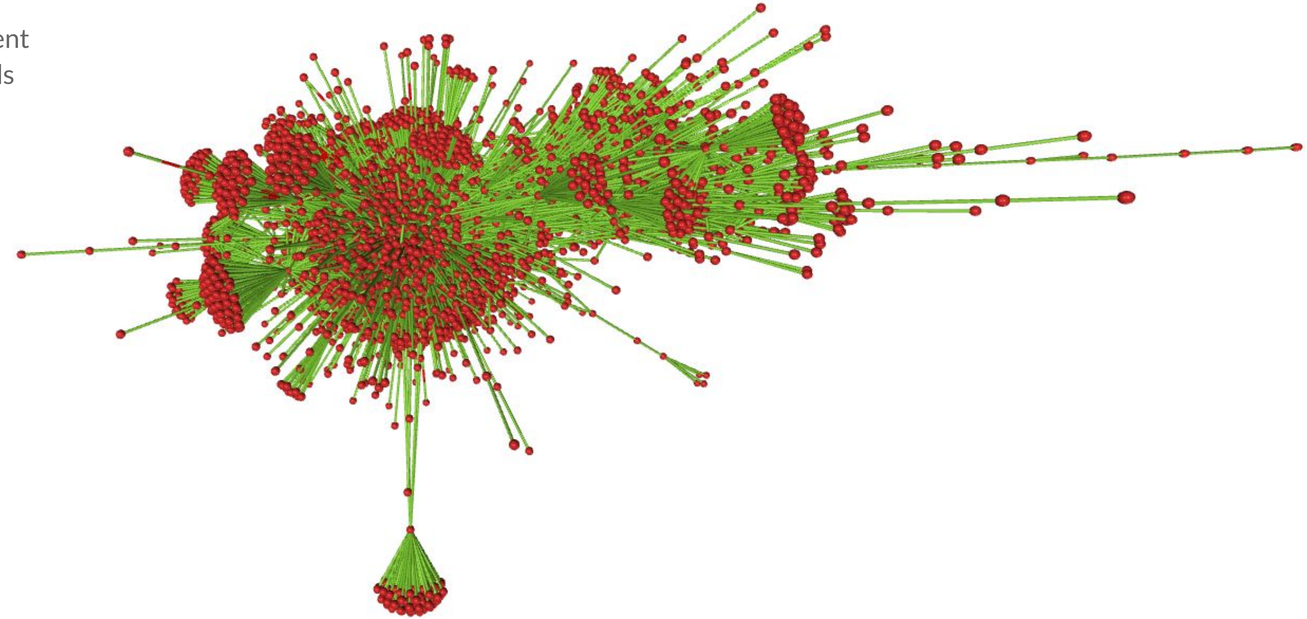
Lightning Network

Snapshot September 2020

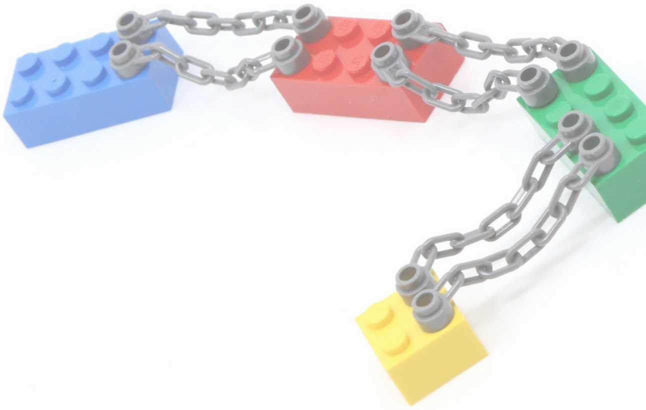
14K nodes, 37K channels
Biggest connected component
90% of the network channels

Node

Channel



Proposal



Proposal

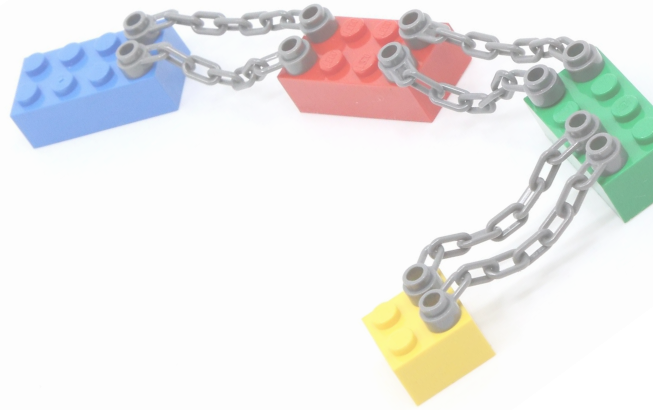
Idea

- Tackle the blockchain trilemma and privacy issues with Layer-2 technologies

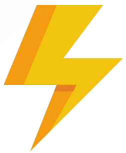
Goal

- Analyze the Layer-2 technology, and develop a framework for their adoption for blockchain applications

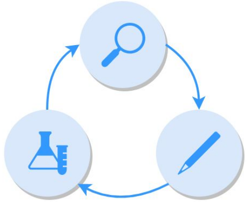
Proposal



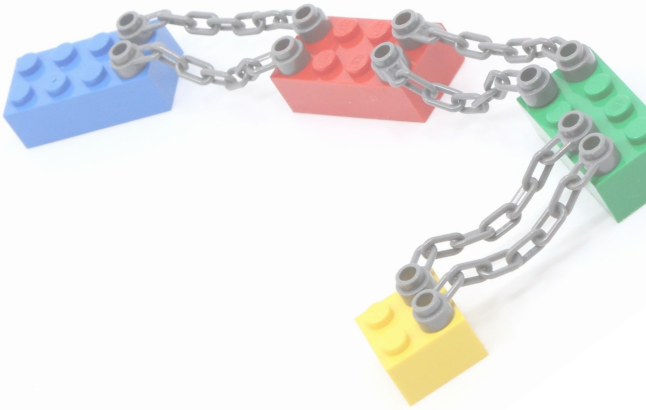
Two research directions



- Analysis of the Bitcoin Lightning Network
  - Graph analysis, security issues, routing protocol



- Development of a methodological framework for the adoption of Layer-2 models in applications
  - Use case analysis, prototyping, generalization



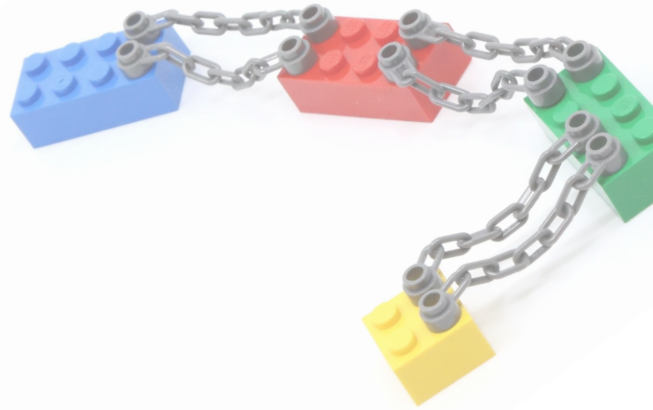
Proposal

Two research directions

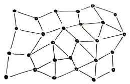
- Analysis of the Bitcoin Lightning Network
  - Graph analysis, security issues, routing protocol

- Development of a methodological framework for the adoption of Layer-2 models in applications
  - Use case analysis, prototyping, generalization

Analysis



Work in progress Done TO DO



1 2 3 4

Crawling the Lightning Network

Download a snapshot of the network

Get an historical trace of snapshots

Analyze the network

Node degree distribution, topological properties

Temporal behavior and churn

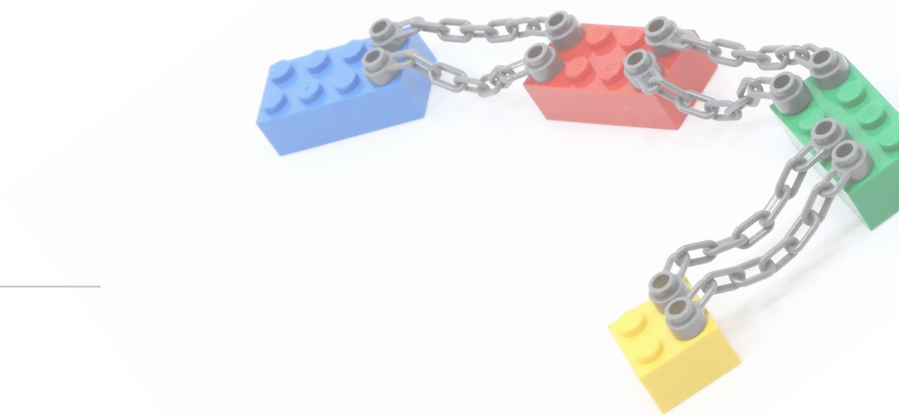
Evaluate the security issues

Presence of payment hubs[Martinazzi], empty channels[Rohrer], privacy data disclosure[Joancomarti]

Design a routing protocol

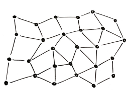
Constraints close to a Flow optimization problem, plus properties to preserve[Gudgeon]

Analysis

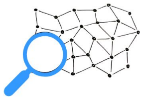


Work in progress — Done — TO DO

1



2



3

4

Crawling the Lightning Network

Download a snapshot of the network

Get an historical trace of snapshots

Analyze the network

Node degree distribution, topological properties

Temporal behavior and churn

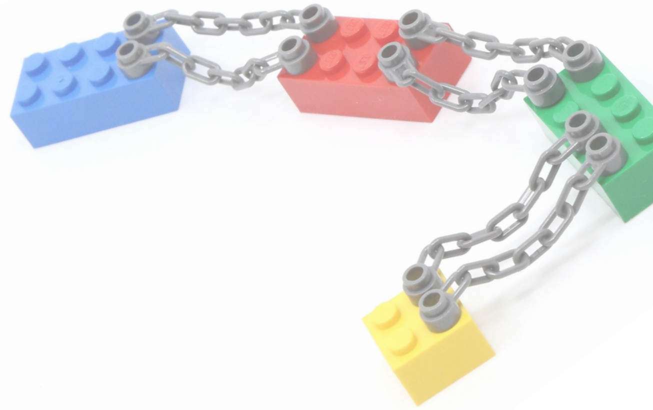
Evaluate the security issues

Presence of payment hubs[Martinazzi], empty channels[Rohrer], privacy data disclosure[Joancomarti]

Design a routing protocol

Constraints close to a Flow optimization problem, plus properties to preserve[Gudgeon]

Analysis



Work in progress | Done | TO DO

1

2

3

4

Crawling the Lightning Network

Download a snapshot of the network

Get an historical trace of snapshots

Analyze the network

Node degree distribution, topological properties

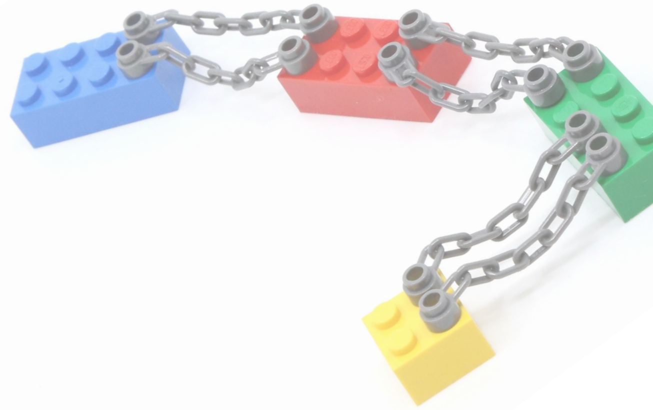
Temporal behavior and churn

Evaluate the security issues

Presence of payment hubs[Martinazzi], empty channels[Rohrer], privacy data disclosure[Joancomarti]

Design a routing protocol

Constraints close to a Flow optimization problem, plus properties to preserve[Gudgeon]

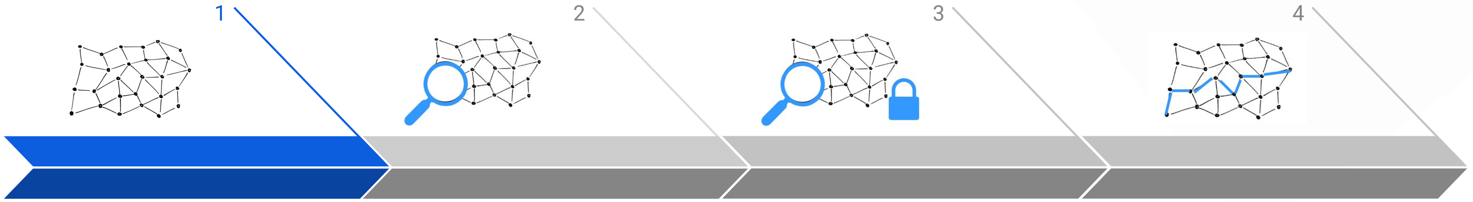


Analysis

Work in progress

Done

TO DO



Crawling the Lightning Network

Download a snapshot of the network

Get an historical trace of snapshots

Analyze the network

Node degree distribution, topological properties

Temporal behavior and churn

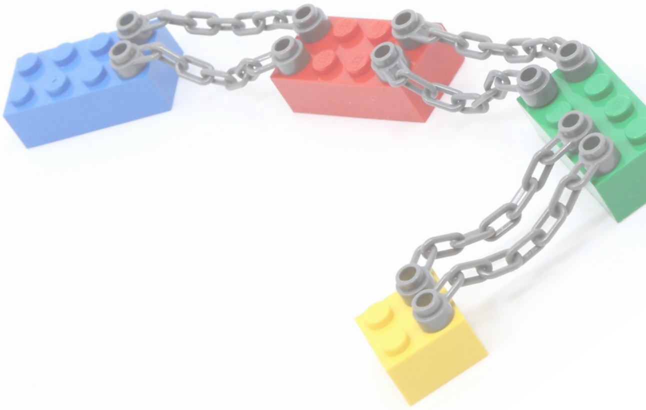
Evaluate the security issues

Presence of payment hubs[Martinazzi], empty channels[Rohrer], privacy data disclosure[Joancomarti]

Design a routing protocol

Constraints close to a Flow optimization problem, plus properties to preserve[Gudgeon]

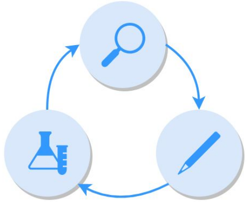
Proposal



Two research directions

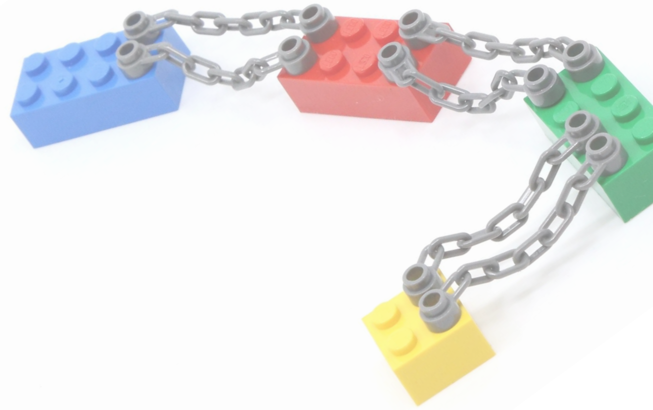
- Analysis of the Bitcoin Lightning Network
  - Graph analysis, security issues, routing protocol

- Development of a methodological framework for the adoption of Layer-2 models in applications
  - Use case analysis, prototyping, generalization

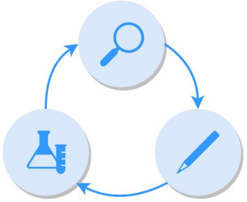


Development

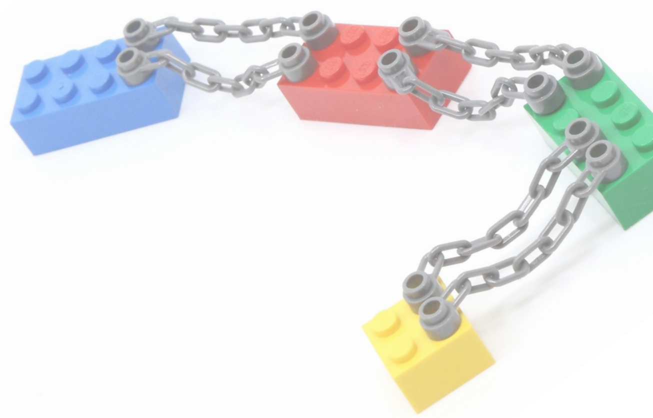
Work in progress | Done | TO DO



1. **Analyze requirements of test use cases**
   - Rating and Recommender Systems (RS)[Ricci]
   - Responsible Disclosure (RD)[Lagutin]
2. **Design a Layer-2 prototype**
   - Identify the model(s) best satisfying the requirements
   - Evaluate scalability and privacy
3. **Define a methodology**
   - Map requirements and constraints to Layer-2 models
4. **Design a general framework**
   - Output the best two-layered architecture for an input use case
   - Understand gains and losses in terms of scalability and privacy



Development



Work in progress — Done — TO DO

1

Analyze requirements of test use cases

Rating and Recommender Systems (RS)[Ricci]

Responsible Disclosure (RD)[Lagutin]

2

Design a Layer-2 prototype

Identify the model(s) best satisfying the requirements

Evaluate scalability and privacy

3

Define a methodology

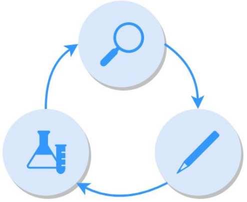
Map requirements and constraints to Layer-2 models

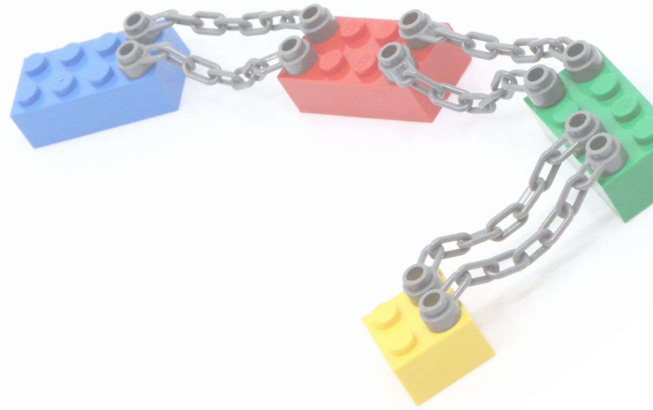
4

Design a general framework

Output the best two-layered architecture for an input use case

Understand gains and losses in terms of scalability and privacy





Development

Work in progress   Done   TO DO

1

Analyze requirements of test use cases

Rating and Recommender Systems (RS)[Ricci]

Responsible Disclosure (RD)[Lagutin]

2

Design a Layer-2 prototype

Identify the model(s) best satisfying the requirements

Evaluate scalability and privacy

3

Define a methodology

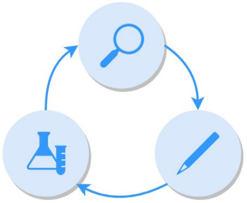
Map requirements and constraints to Layer-2 models

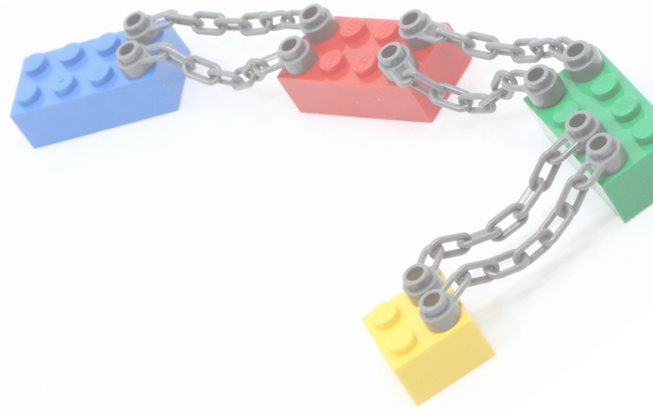
4

Design a general framework

Output the best two-layered architecture for an input use case

Understand gains and losses in terms of scalability and privacy





Development

Work in progress

Done

TO DO

1

Analyze requirements of test use cases

Rating and Recommender Systems (RS)[Ricci]

Responsible Disclosure (RD)[Lagutin]

2

Design a Layer-2 prototype

Identify the model(s) best satisfying the requirements

Evaluate scalability and privacy

3

Define a methodology

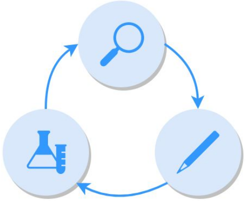
Map requirements and constraints to Layer-2 models

4

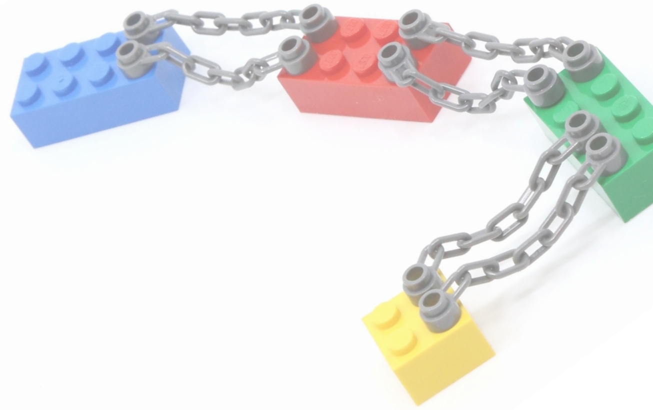
Design a general framework

Output the best two-layered architecture for an input use case

Understand gains and losses in terms of scalability and privacy



Conclusions



Conclusions

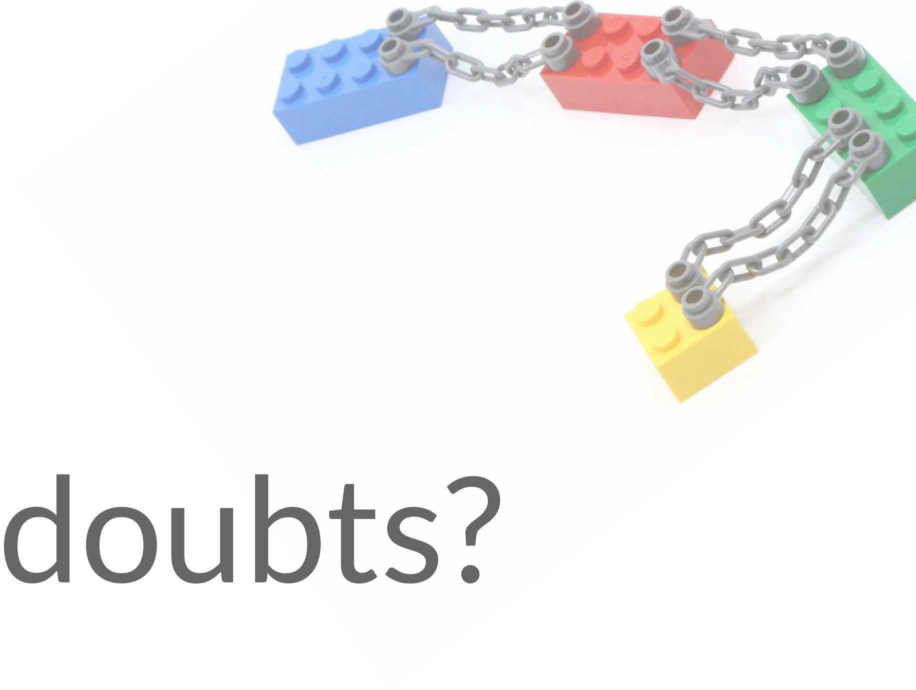
Blockchain-based applications suffer of the trilemma and privacy issues

- Layer-2 technologies try to solve them working on top of existing Layer-1 architectures

Goal

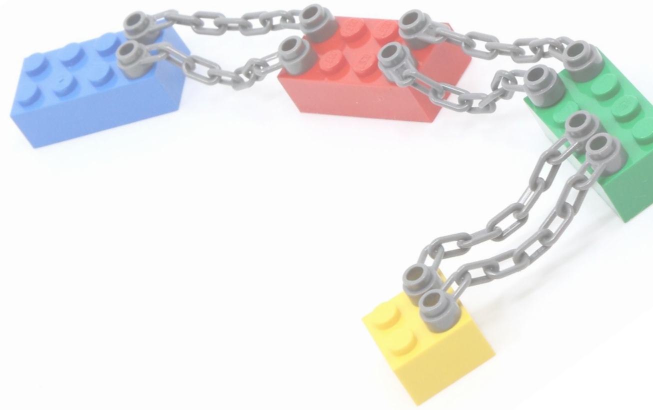
Study these technologies, and find a standard way to apply them to use-cases of research and industrial interest

Thank you!



Any doubts?

References



[Wüst] **Do you Need a Blockchain?**, Karl Wüst et al

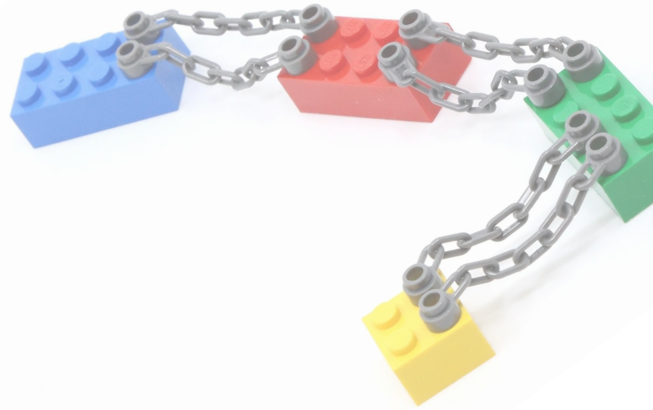
[Garriga] **Blockchain and cryptocurrencies: A classification and comparison of architecture drivers**, Martin Garriga et al

[Trilemma] **Sharding FAQ**, https://eth.wiki/sharding/Sharding-FAQs

[Gudgeon] **SoK: Layer-Two Blockchain Protocols,** Lewis Gudgeon et al

[Zhou] **Solutions to scalability of blockchain: A survey,** Zhou, Qiheng et al

[Poon] **The bitcoin lightning network: Scalable off-chain instant payments**, Joseph Poon et al



References

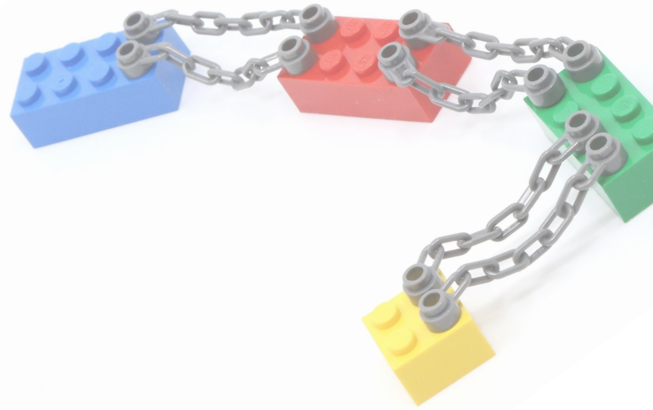
[Martinazzi] **The evolving topology of the Lightning Network: Centralization, efficiency, robustness, synchronization, and anonymity,** Stefano Martinazzi et al

[Joancomarti] **On the difficulty of hiding the balance of lightning network channels**, Jordi Herrera-Joancomarti et al

[Ricci] **Introduction to recommender systems handbook**, Francesco Ricci

[Rohrer] **Discharged Payment Channels: Quantifying the Lightning Network's Resilience to Topology-Based Attacks,** Elias Rohrer

[Lagutin] **Leveraging Interledger Technologies in IoT Security Risk Management,** Dmitrij Lagutin et al



Publications

[Lisi_GECON] **A smart contract based recommender system,** Lisi Andrea et al

[Lisi_FGCS] **Rewarding reviews with tokens: an Ethereum-based approach**, Lisi Andrea et al (SUBMITTED)