Blockchain / Distributed Ledger

- Expected properties
- Hashing functions, hasing (Merkle) trees
- Blockchain basics
- Blockchain demo

Expected properties

- Safety

What does it?

- Store data
  - Numbers, text, … simple, structured
- Perform operations
  - Write, evaluate expression, execute instruction
- Per blocs: acts (state change)
  - Sync (block time) or
  - Async (event driven)
- Data + operations/instructions = code
- *Code is Law* (Lessig)→ *Law is code* (o contracts)
  - Smart contracts (Ethereum):
    - Procedures, with its own "independent" life

Which properties

- Descentralized registry:
  - Replicated, immutable, transparent, eventually consistent …
- **Transparent**
- **Irreversible**
- **Inexorable**
- "Disintermediation":
  - Known or not (identified or anonymous) participants can interact directly and transparently without any trusted intermediary

How can it be done

- **Registry** r*eplicated, immutable-irreversible, transparent, eventually consistent, inexorable …*
- Data is stored and processed:
  - In the "cloud": Bitcoin, Ethereum *(cost per operation and data: gas)*
  - In our servers:
    a group (sufficient number replicas) of entities *(cost per infraestructure: risk)*
  - Execution (transactions): multiple, consistent (compute new state: mining)
- *Avoid corruption, ensure preservation*

How this is done

- Software:
  - Ethereum (geth)
  - Hyperledger (Fabric)
- Code: smart contracts
  - Programming language: solidity, go, …
  - Standarization: templates, models for basic contracts, verified and interoperable: ERC

For instance, ourselves ...

- Our own unit of exchange (ERC20)
- Inventory of devices we have (ERC721)
  - Devices (token, title), owners / users (Id),
  - Inventory data (oracle)
- Decentralized Internet access: MeshDapp
  - Pre-payment (tokens) → connectivity (consumption)
  - Distribution between devices and their owners
- Circular economy traceability: eReuse
  - Manufacture, use, second hand, repair, recycling
  - Token-Id device, token 'deposit pay for recycling

More ...

- Demo: http://blockchain.mit.edu/how-blockchain-works/