Blockchain 2.0

Opportunities and Risks

Patrick Valduriez

The Hype

Harvard
Business
Review

BITCOIN

Bitcoin is the 'mother of all scams' and blockchain is most hyped tech ever, Roubini tells Congress

- One of the few economists who predicted the 2008 financial crisis warns U.S. senators of the pernicious side of cryptocurrencies.
- He also criticized bitcoin's underlying technology, blockchain, calling it the most "over-hyped — and least useful — technology in human history."
- "Crypto is the mother or father of all scams and bubbles," Roubini told the U.S. Senate Committee on Banking, Housing and Community Affairs at a hearing Thursday.

Kate Rooney | @Kr00ney

Published 5:21 PM ET Thu, 11 Oct 2018 | Updated 3:09 PM ET Fri, 12 Oct 2018

CNBC

Bitcoin

- Bitcoin: A Peer-to-Peer Electronic Cash System
  - Satoshi Nakamoto (pseudo), Oct. 31, 2008 (Halloween)
  - Cryptocurrency and payment system
  - Blockchain is the infrastructure
- Since then
  - Many blockchains: Etherum in 2013, Ripple in 2014, etc.
  - Increasing use for high-risk investment
    - Initial Coin Offerings
  - But also in fraudulent or illegal activities !
    - Scam, purchase on the dark web, money laundering, tax evasion, …
  - Warnings from market authorities and beginning of regulation (China, South Korea, Japan, EU, …)

The Currency of Tomorrow?

- Pros
  - Low transaction fee (set by the sender to speed up processing)
  - Fewer risks for merchants (no fraudulent chargebacks)
  - Security and control (protection from identity theft)
  - Trust through the blockchain
- Cons
  - Unstable: no backing by a state or fed bank (unlike $ and €)
  - Unrelated to real economy, e.g. GPD: fosters speculation
  - High volatility, e.g. between 6K and 7K$ in 3 hours
  - Small user base: 20 million bitcoin wallets
    - Versus billions of users of e-payment systems like AliPay and Paypal
- The Crypto Bubble (2017)*
  - Bitcoin price increased from $1k to 10K, then peaked almost at $20K in December 2017 to collapse 4 months later to below $6k (down 70% from the peak), and close to $6k since then

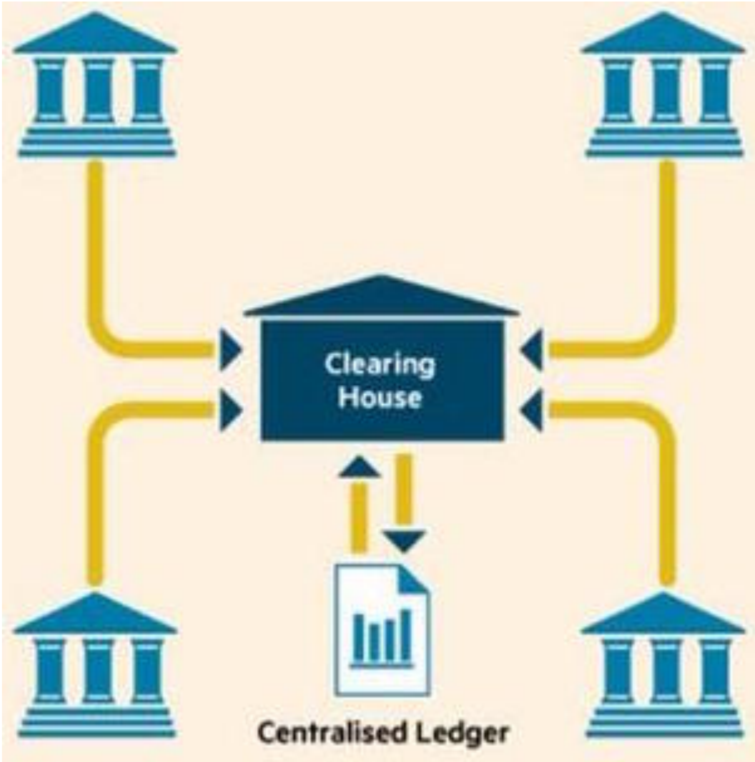
* Testimony for the Hearing of the US Senate Committee on Banking, Housing and Community Affairs On "Exploring the Cryptocurrency and Blockchain Ecosystem". Nouriel Roubini (NYU), october 2018.

Outline

- Trust with blockchain
- Consensus protocols
- How the blockchain works
- Blockchain 2.0
- Use cases
- Opportunities and risks
- Issues

Trust in a Modern Economy

- Context
  - How to exchange assets safely between two parties?
- Centralized ledger
  - An account book that records all transactions
  - Controlled by a trusted central authority
    - E.g. a clearing house

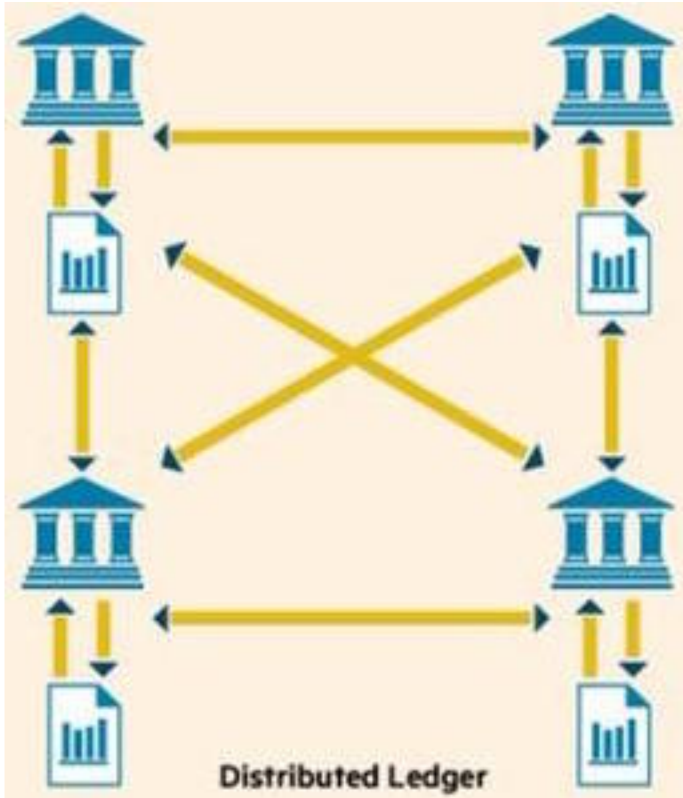


Problems with Central Authority

- Single point of failure
  - And easy target for attackers
- Favors concentration of actors
  - Banks
    - Exploit our money to make big money
  - Web giants (GAFAM) and other intermediaries (Uber, etc.)
    - Exploit our data to make big money

Trust with Blockchain

- A distributed ledger
  - Shared by all participants
    - Replicated
  - Decentralized
  - Append-only
    - No update, no delete
  - Distributed transaction validation
    - Consensus
  - Unfalsiable, verifiable

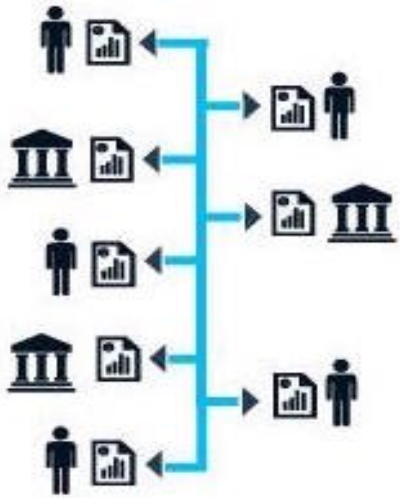


Blockchain Promises

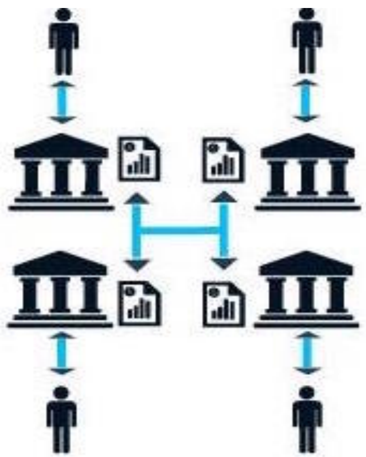
- Increased trust in value exchange
  - Trust the data, not the participants
- No single point of failure
  - Increased security
- Efficient, consistent transactions between participants
  - Faster and cheaper than relying on a long chain of intermediaries, with incompatible systems and rules

Public versus Private Blockchain

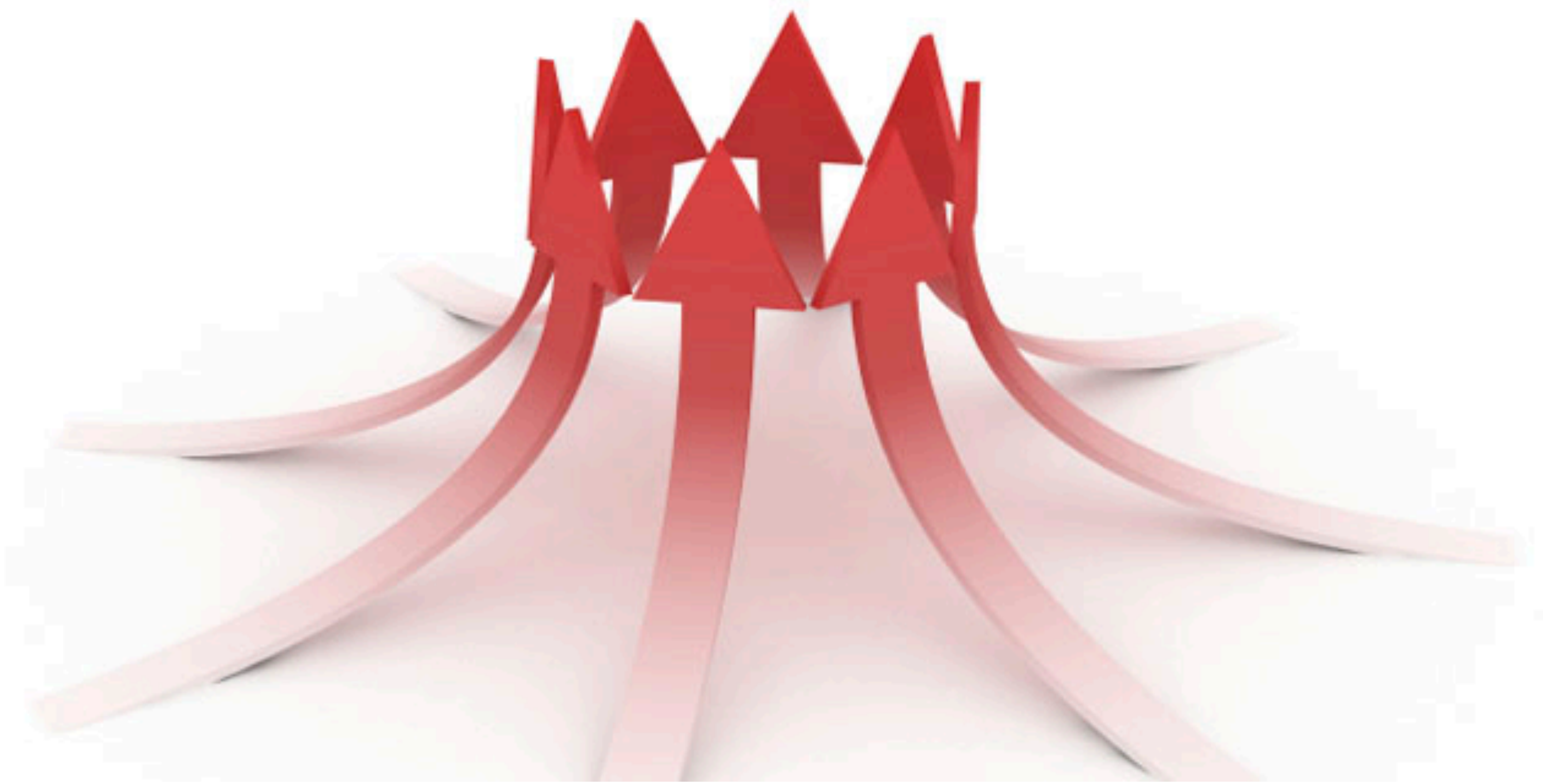
- Public blockchain
  - Open P2P network
    - Participants can join and leave without notification
  - Anonymous, untrusted participants
  - Large-scale distributed ledger



- Private blockchain
  - Closed permissioned network
  - Identified, trusted participants
  - Regulated control
  - Small to medium-scale distributed ledger



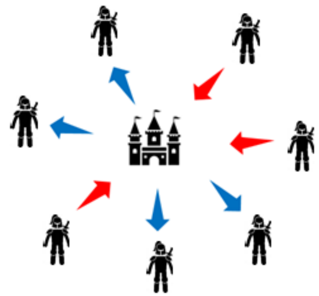
Background on Consensus Protocols



Consensus

- Critical applications
  - Replication, transaction validation, identity verification, etc.
- Major problem of distributed systems
  - How to reach a consensus, i.e. agree on the same value, in the presence of a number of faulty processes?
- Problem statement: given *n* processes and one leader, how to reach:
  - Agreement : all correct processes agree on the same value
  - Validity: if initiator does not fail, all correct processes agree on its value
- Types of failures
  - Crash: the easy case
  - Malicious (also called Byzantine)
    - The process gives different values to different observers
- FLP (Fischer, Lynch, Paterson) impossibility result
  - With only one crash failure, termination is not guaranteed
  - Example: coordinator failure in 2PC

The Byzantine Agreement Problem



- Suppose an army of the Byzantine Empire
  - Generals can only communicate by messengers and must establish a common plan to attack the enemy or retreat
  - A number of these generals may be traitors and vote selectively
    - Example with 5 generals: 2 support the attack and 2 are in favor of retreat; the 5th can send an attack vote to the first two and a retreat vote to the other two and then …
- Problem formulation
  - Find an algorithm (consensus) to ensure that loyal generals can agree on a common battle plan

Paxos Algorithm

- The basis for a family of protocols
  - [Lamport 1999, ACM Turing Award 2013]
  - Used to manage large-scale distributed data
    - Google Spanner & Megastore
    - IBM SAN Volume Controller
    - Microsoft Autopilot Cluster Mgr
    - Ceph (distributed file system)
    - Neo4J (NoSQL graph DBMS)
- Inspired by the functioning of the Parliament of the Paxos Island
  - The Parliament did work, despite the regular absence of legislators and messages loss

Paxos Algorithm

- Principle (simplified)
  - Initialization: a leader is elected by a majority quorum
  - Replication: leader replicates new updates to the majority quorum
  - Leader failure: il the leader fails, a new leader is elected
  - To make progress, at least 1/2 of the participants should be alive
- Limitations
  - Permissioned settings: all participants should be known a priori
    - Not appropriate for public blockchain
  - Tolerates only crash failures
    - Does not deal with malicious nodes
  - Progress is not guaranteed (FLP impossibility)

Practical Byzantine Fault Tolerance (PBFT)

- A three-phase protocol [Castro & Liskov 1999]
  1. Pre-prepare: a leader broadcasts a value to be committed by other nodes
  2. Prepare: the nodes broadcast the values they are about to commit
  3. Commit: confirms the committed value when more than 2/3 of the nodes agree in the previous phase
- Assessment
  - Tolerates Byzantine failures
  - Permissioned settings

How the Blockchain Works

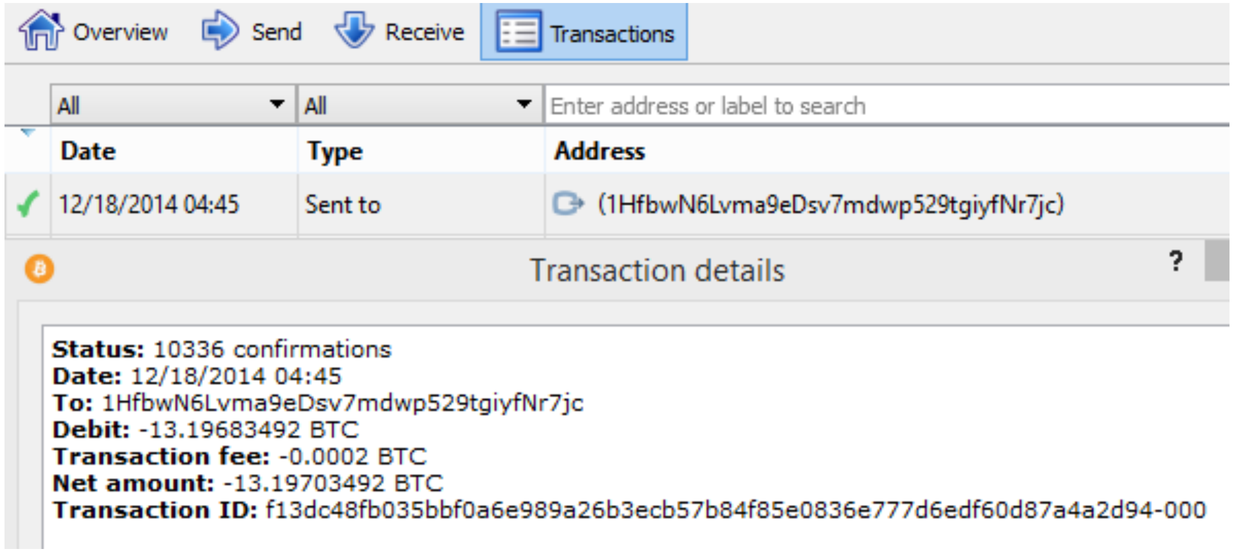
Blockchain Concepts

- Blockchain
  - An *immutable* distributed database, i.e. a log of blocks, which are linked and replicated on *full nodes*
- A block
  - Digital container for transactions, contracts, property titles, etc.
  - Transactions are secured using public key encryption
- The code of each new block is built on that of the preceding block
  - Guarantees that it cannot be changed or tampered
- The blockchain is viewed by all participants
  - Enables validating the entries in the blocks
  - Privacy: users are pseudonomyzed

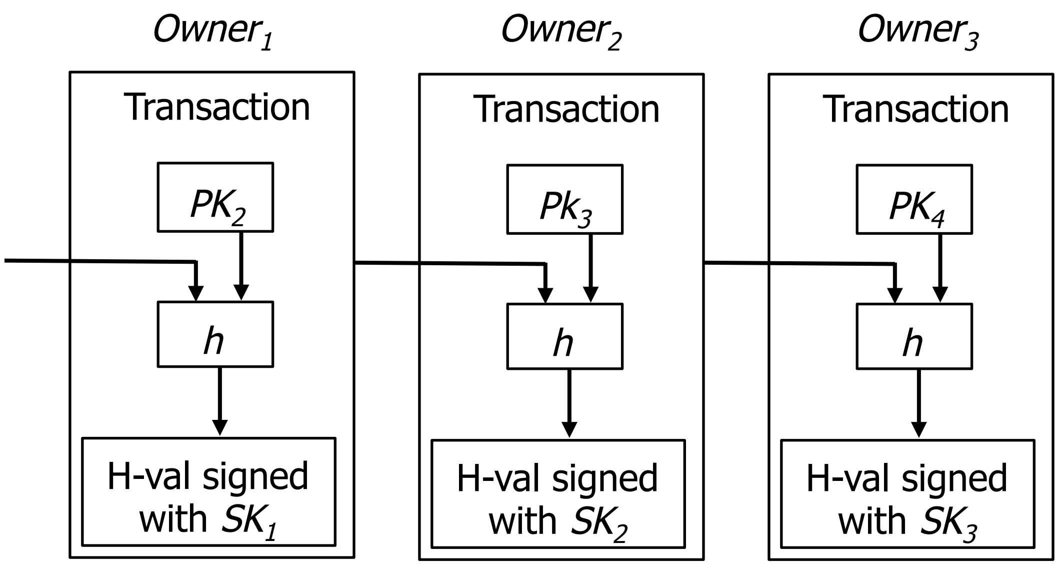
Blockchain Protocol (Nakamoto 2008)

0. Initialization (of a *full node*)
   - Synchronization with the network to obtain the blockchain (185 GB on Q3, 2018)
1. Two users agree on a transaction
   - Information exchange: wallet addresses, public keys, …
2. Grouping with other transactions in a block and validation of the block (and of the transactions)
   - Consensus using "mining"
3. Addition of the validated block in the blockchain and replication in the P2P network
4. Transaction confirmation

Transaction

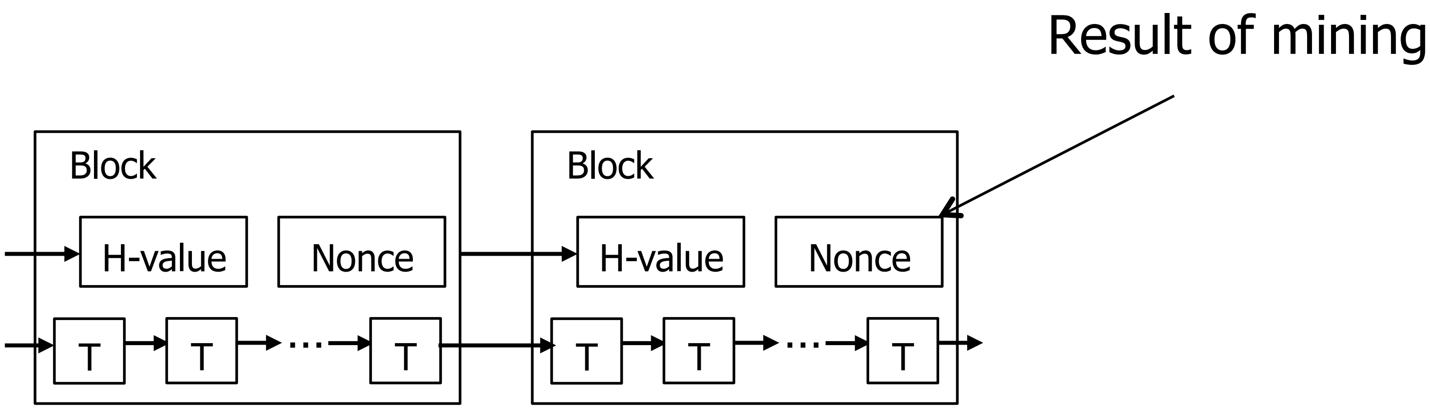


- The coin owner signs the transaction by
  1. Creating a hash value of
     - The previous transaction
     - And the public key (PK) of the next owner
  2. Signing it with its secret key (SK)



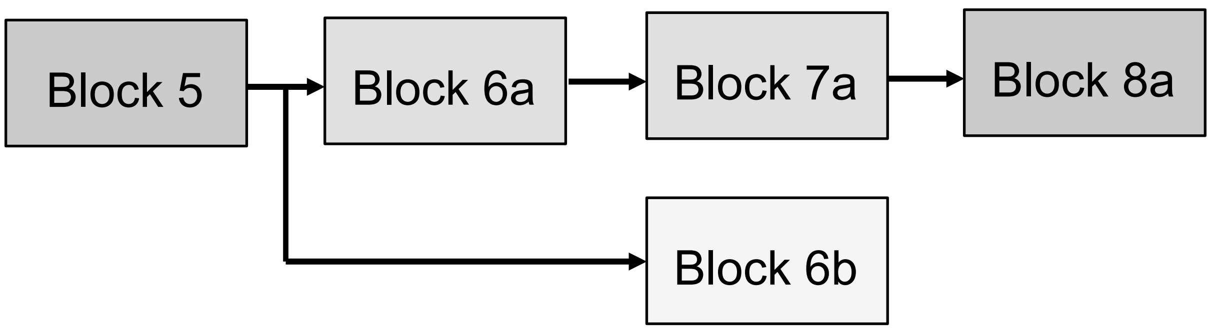
Block Management

- Transactions are placed into blocks, validated (by checking inputs/outputs, etc.) and linked by their addresses
  - Size of a bitcoin block = 1 Megabyte



Validation by the Network

- Each block is validated by network nodes, the *miners*, by a consensus protocol (see next)
- Problem: accidental fork
  - As different blocks are validated in parallel, one node can see several candidate chains at any time
  - Solution: longest chain rule



Transactions in a validated block are provisionally validated; confirmation must be awaited

Intentional Fork

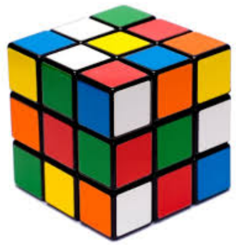
- Main reasons
  - To add new features to the blockchain (protocol changes) => new software
  - To reverse the effects of hacking or catastrophic bugs
- Soft versus hard fork
  - Soft fork: backward compatible
    - The old software recognizes blocks created with new rules as valid
    - Makes it easy for attackers
  - Hard fork
    - The old software recognizes blocks created with new rules as invalid
    - Example: the battle between (new) Ethereum and Ethereum Classic
      - In 2016, after an attack against the Decentralized Autonomous Organization (DAO), a complex smart contract for venture capital, the blockchain forked but without momentum
      - Battle is more philosophical and ethical than technical

Consensus Protocol: *mining*

- Why not Paxos?
  - Remember: participants are unknown
- To validate a block, miner nodes compete (as in a lottery) to produce a *nonce* (number used once)
  - One of the first competing solutions is selected, e.g. the one that includes the largest number of transactions
  - The winner miner is paid, e.g. 12.5 bitcoins today (originally 50)
  - This increases the money supply
- Mining is designed to be difficult
  - The more mining power the network has, the harder it is to compute the nonce
  - This allows controlling the injection of new blocks ("inflation") in the system, on avg. 1 block every 10mn
  - Advantages powerful nodes

Mining Difficulty : Proof of Work (PoW)

- PoW
  - A piece of data that is difficult to calculate but easy to verify
  - First proposed to prevent DoS attacks



- Hashcash PoW
  - Computed by each miner to produce the nonce
- Goal: produce a value v such that $h(v)<T$ where
  - h is a hash function (SHA-256)
  - T is a target value which is shared by all nodes and reflects the size of the network
  - v is a 256-bit number starting with n zero bits
    - Low probability of success : $1/2^n$

The 51% Attack

- Also called Goldfinger attack
  - Enables the attacker to invalidate valid transactions and double spend funds
- How
  - By holding more than 50% of the total computing power for mining
    - Miners coalition
  - It then becomes possible to modify a received chain (e.g. by removing a transaction) and produce a longer chain that will be selected by the majority
- Solution: monitoring by the community
  - In January 2014, Ghash.io reached 42%, then dropped to 9% after the Bitcoin community alert

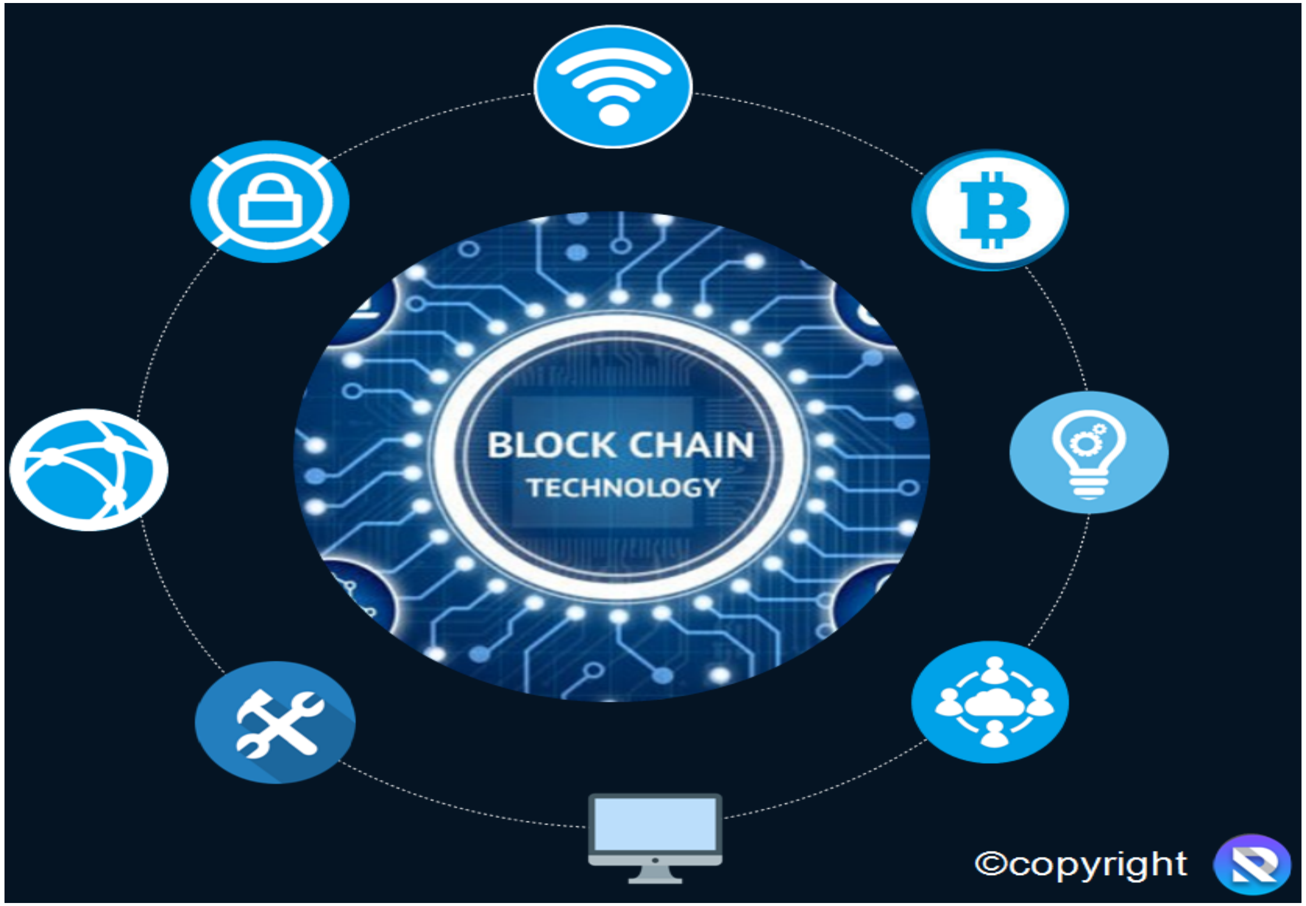
Transaction Confirmation

- A provisionally validated transaction in a candidate block ensures that it has been verified and is viable
- Each new block accepted in the chain after the validation of the transaction is considered as a confirmation
  - A transaction is considered mature after 6 confirmations (1 hour on average)
  - New bitcoins (mining products) are only valid after 120 confirmations, to avoid the 51% attack

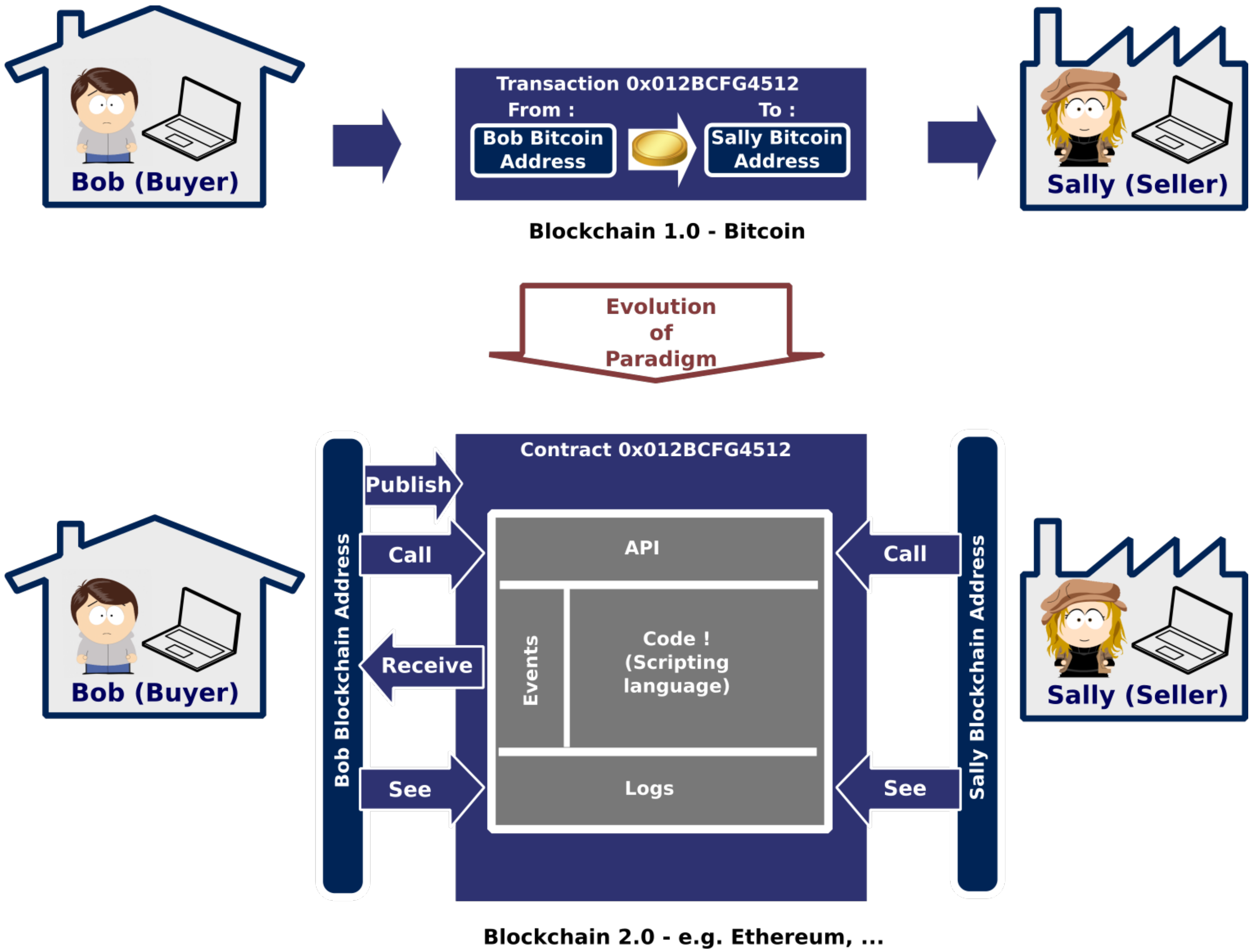
Public Blockchain Limitations

- Complexity and low scalability
  - Difficult evolution of operating rules
  - Increasing chain size
  - Low number of transactions per second (TPS)
    - 5-7 TPS for Bitcoin versus 25K TPS for VISA
  - Unpredictable duration of transactions, from minutes to days
- Cost
  - High energy consumption
  - Favors concentration of miners
- Users are pseudonymized, not anonymized
  - Making a transaction with a user reveals all its other transactions
- Lack of control and regulation
  - Hard for states to watch and tax transactions

Blockchain 2.0



Evolution of Paradigm



Evolution of Paradigm

- Beyond Bitcoin and other cryptocurrencies
  - Recording and exchange of assets without powerful intermediaries
  - Example: smart contracts
- Positioning in the internet
  - TCP/IP: the communication protocol
  - Blockchain: the value-exchange protocol?

Blockchain 2.0 Technology

- Programmable blockchain, e.g. Etherum
  - Allows application developpers to build APIs on the Blockchain protocol
    - APIs to allocate digital resources (bandwidth, storage, etc.) to the connected devices, e.g. FileCoin
    - Micropayment APIs tailored to the type of transaction (e.g. tipping a blog versus tipping a car share driver)
- Private blockchain
  - Efficient transaction validation since participants are trusted
    - No need to produce a PoW
  - Efficient management, e.g. in the cloud

Blockchain 2.0 Development

- Support from all major industry players
  - Finance services: Mastercard, VISA, …
  - Audit firms: EY, KPMG, PwC, Deloitte
  - Consulting firms: Accenture, Capgemini,
  - Web giants: Amazon, Google
  - Software suppliers: IBM, Oracle, Microsoft, SAP
  - Technology platform companies: Cisco, Fujitsu, IBM, Intel, NEC, Red Hat, VMware
- New blockchain ISVs
  - Blockchain, ConsenSys, Digital Asset, R3, Onchain

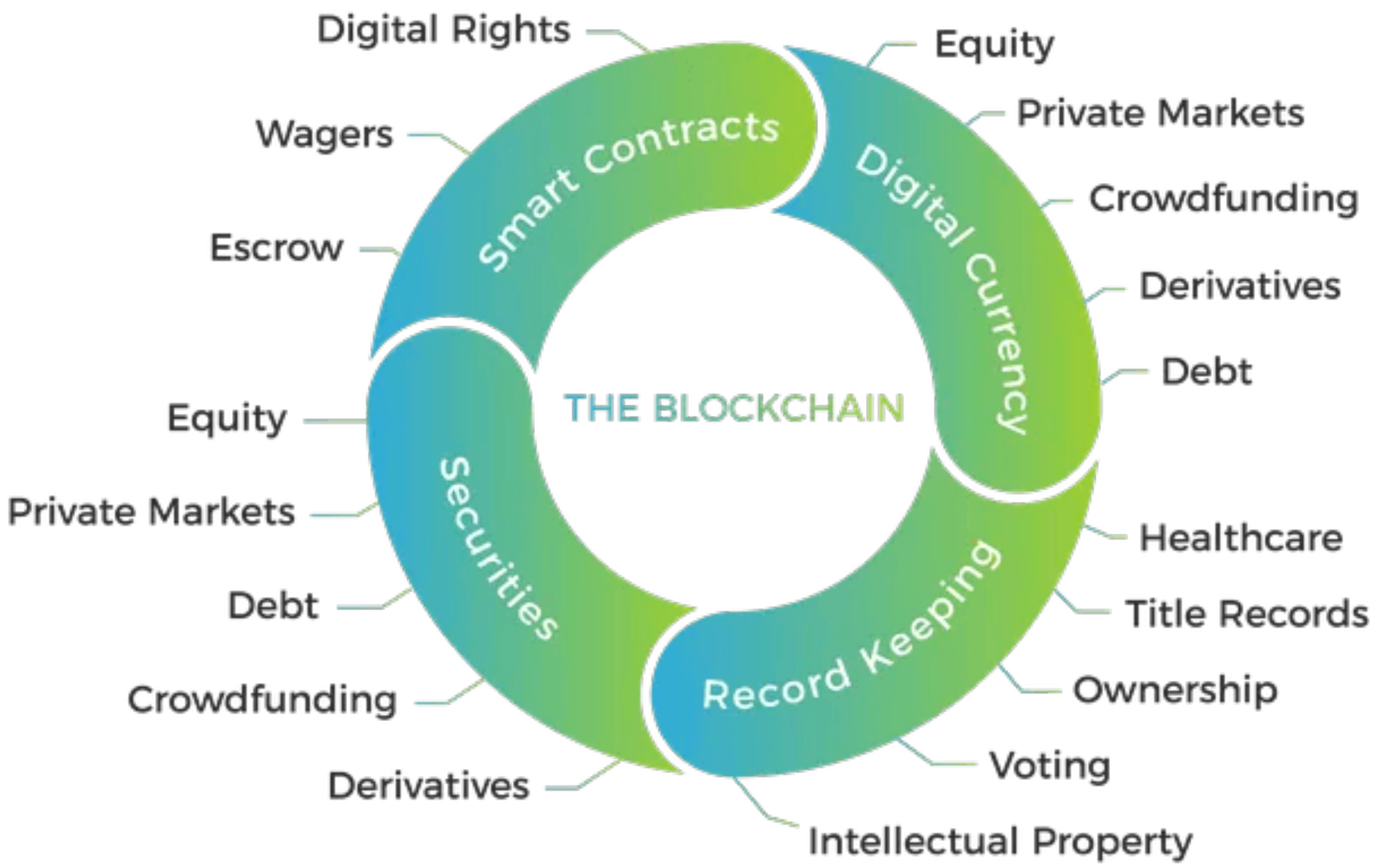
Smart Contracts

- "Code is law", Lawrence Lessig, Harvard Law School
- Smart contract (Nick Szabo, 1993)
  - Self-executing contract, with code that embeds the terms and conditions of a contract
  - Early application: digital rights management schemes
- Deployment in the blockchain 2.0 (e.g. Etherum)
  - Participants can be unknown to each other
  - Contracts can be with many third parties, e.g. IoT devices, at low cost
- Challenges
  - Bug-free code, which requires code certification
  - Compliance with mandatory regulation, which requires collaboration between programmers and lawyers

Hyperledger Project (Linux Foundation)

- Started in 2015 (IBM, Intel, Cisco, …)
- Open source blockchains and related tools
- Major frameworks
  - Hyperledger Fabric (IBM, digital Asset): a permissioned blockchain infrastructure
    - Smart contracts, configurable consensus (PBFT, …) and membership services
  - Sawtooth (Intel): a new consensus "Proof of Elapsed Time" that builds on trusted execution environments
  - Hyperledger Iroha (Soramitsu): based on Hyperledger Fabric, with a focus on mobile applications

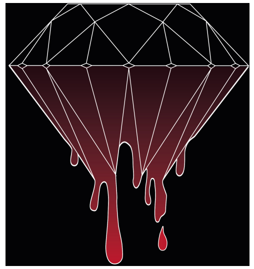
Blockchain Use Cases



Blockchain 2.0 Apps

- Critical characteristics of the applications
  - Asset and value are exchanged (transactions)
  - Multiple participants, unknown to each other
  - Trust is critical
- Top use cases
  - Financial services, micropayments
  - Digital rights using smart contracts
  - Digital identity
  - Supply chain management
  - Internet of Things (IoT)
- POCs in many industries
  - Publishing, retail, music, healthcare, rental, real estate, government, energy, agriculture, etc.

Diamond Supply Chain Management



- Problems
  - How to trace diamonds, in an era of “blood diamonds”?
  - Complex and multi-tiered diamond and jewelry supply chain
- Objective of TrustChain
  - Provide trusted products with documented authenticity, guaranteeing quality and environmental responsibility
- Solution (IBM Hyperledger)
  - A permissioned blockchain that establishes a single shared view of information without compromising detail, privacy, or confidentiality

Opportunities and Risks

Opportunities

- Disruptive technology
  - For recording transactions and verifying records
  - The ability to program applications and business logic in the blockchain opens up many possibilities for developers
    - E.g. smart contracts

- Disruptive power
  - The goal of cypherpunk activists
  - It may establish a sense of democracy and equality for individuals and small businesses in countries with non-transparent, unsecure jurisdictions

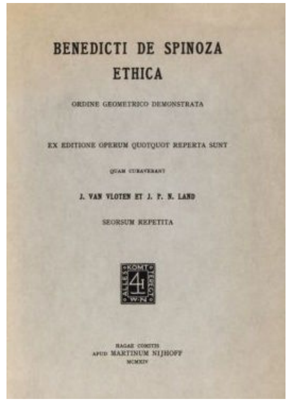
Risks

- Market disruption
  - Massive disintermediation of the current system, replacing all procedures that deal with transactions with a system where participants trade directly
- Public blockchain
  - Consumer protection: significant volatility of Bitcoin and other cryptocurrencies (no government backup)
  - Increasing use for fraudulent or illegal activities
  - Security concerns: if a private key is lost or stolen, an individual has no recourse
  - Lack of control and regulation, and hard for states to agree on what to do

Research Issues

- Scalability of the public blockchain
  - Alternatives to PoW : proof of stake, proof of hold, proof of use, proof of stake/time, …
  - New generation blockchains, e.g. Bitcoin-NG [Usenix 2016]
- Smart contracts
  - Code certification and verification
- Blockchain interoperability
  - Blockchain Interoperability Alliance (BIA), to promote cross-blockchain transactions
- Blockchain and big data
  - Blockchain-generated data analysis, e.g. fraud prevention based on real-time transactions
  - Blockchain-based DBMS, e.g. BigchainDB

Ethical Issues



- Blockchain can have strong (good or bad) impact on the world
  - People, society, economy, environment, …
  - Remember: the public blockchain is great for crooks and criminals
- This raises ethical issues that we cannot simply ignore
  - See the recent panel: A Debate on Data and Algorithmic Ethics (VLDB 2018)

About Trust Again