Introduction à la technologie Blockchain

Introduction à la technologie Blockchain

- Introduction
  1. Quick history of Blockchains
  2. Blockchain terminologies
  3. Distinction between databases and blockchain ledgers
  4. Why and when to use a Blockchain?
- Cryptographic component
  1. Cryptography, hash functions and digital signatures
- Consensus components
  1. Principles and paradigms of distributed systems
  2. Blockchain consensus algorithms

Introduction à la technologie Blockchain

- Blockchain structures
  1. Blockchain structure
  2. Types of blockchain
- Smart contract theory
  1. Smart Contract Theory and architecture
  2. Architectures and decentralized autonomous systems
- Smart contract application
  1. Existing blockchain applications, related structures and architectures
- Research goals
  1. Current research and challenges faced by Blockchains

Introduction à la technologie Blockchain

- **Introduction**
  1. Quick history of Blockchains
  2. Blockchain terminologies
  3. Distinction between databases and blockchain ledgers
  4. Why and when to use a Blockchain?
- Cryptographic component
- Consensus components
- Blockchain structures
- Smart contract theory
- Smart contract application
- Research goals

Introduction

"To understand the power of blockchain systems, and the things they can do, it is important to distinguish between three things that are commonly muddled up, namely the bitcoin currency, the specific blockchain that underpins it and the idea of blockchains in general."

The Trust Machine, THE ECONOMIST, Oct. 31, 2015

Introduction à la technologie Blockchain

- **Introduction**
  1. **Quick history of Blockchains**
  2. Blockchain terminologies
  3. Distinction between databases and blockchain ledgers
  4. Why and when to use a Blockchain?
- Cryptographic component
- Consensus components
- Blockchain structures
- Smart contract theory
- Smart contract application
- Research goals

Quick history of Blockchains

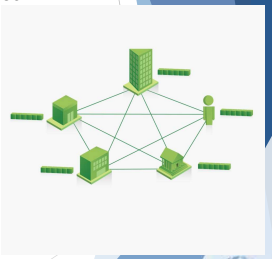
- The idea of a shared ledger emerged a few decades ago
  - Stuart Haber and W. Scott Stornetta made an immutable ledger in 1991
- But the first decentralized implementation appeared with Bitcoin in 2009
  - Created by an unknown person called Satoshi Nakamoto
- Since then, thousands of different cryptocurrencies and blockchains have been made
  - New features (smart contracts, privacy…)
  - Different approaches (Decentralized Autonomous Organizations, or DAO)

Introduction à la technologie Blockchain

- **Introduction**
  1. Quick history of Blockchains
  2. **Blockchain terminologies**
  3. Distinction between databases and blockchain ledgers
  4. Why and when to use a Blockchain?
- Cryptographic component
- Consensus components
- Blockchain structures
- Smart contract theory
- Smart contract application
- Research goals

Blockchain terminologies

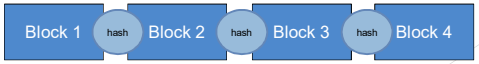
- **Blockchain - What is it?**
  - Aka DLT (Distributed Ledger Technology) - rudimentary shared accounting system
  - Technologically, it is :
    - Distributed database – public ledger (you can insert, select data, but **can't** update or delete data.
    - Distributed computer – execute digital contracts
    - Based on **p2p** (peer-to-peer) technology, cryptology and API



Blockchain terminologies

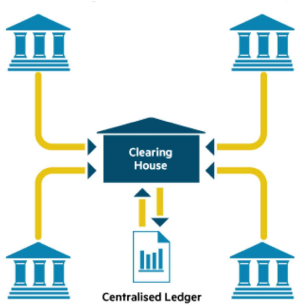
- **Blockchain - What is it?**

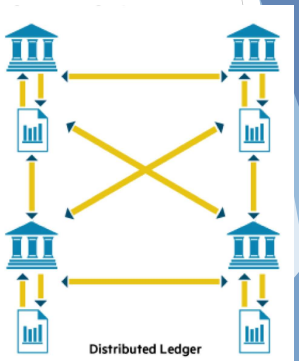
  In fact, the blockchain is more than a technology, it
  - Usually contains financial transactions
  - Is replicated across a number of systems in almost real-time
  - Uses cryptography and digital signatures to prove identity, authenticity and enforce read/write access rights
  - Can be written by certain participants
  - Can be read by participants, often a wider audience
  - Has mechanisms to make it hard to change historical records, or at least make it easy to detect when someone is trying to do so



Blockchain terminologies

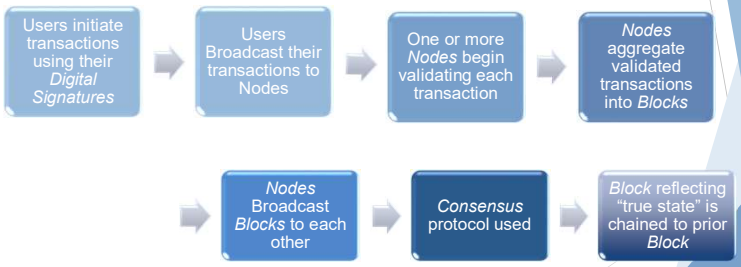
- **Distributed ledger - What is it?**





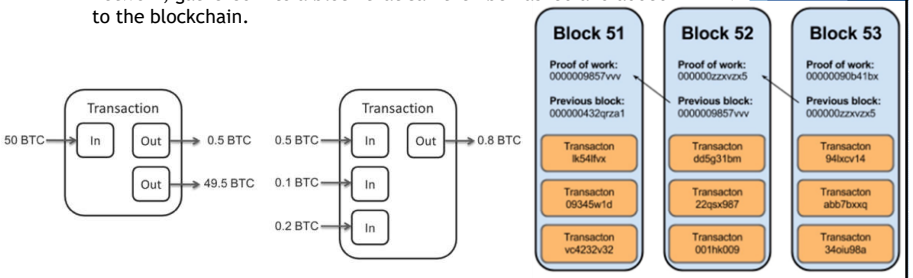
Blockchain terminologies

- **Distributed ledger - How it works?**



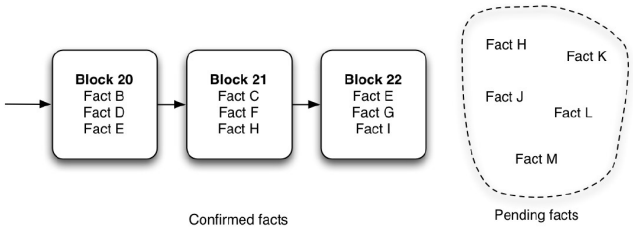
Blockchain terminologies

- **Transaction & blocks**
  - A transaction block is a collection of transactions on the bitcoin network, gathered into a block that can then be hashed and added to the blockchain.



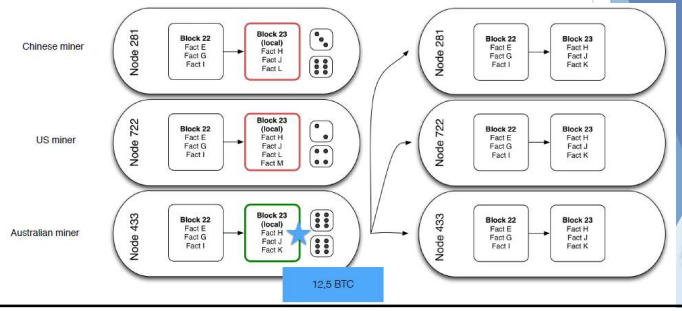
Blockchain terminologies

- **Mining**
  - The process by which transactions are verified and added to a blockchain.



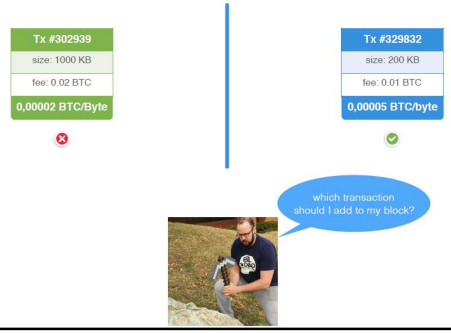
Blockchain terminologies

- **Mining**
  - This process of solving cryptographic problems using computing hardware also triggers the release of cryptocurrencies



Blockchain terminologies

- **Mining**
  - Miners on the network select transactions from pools and form them into a ‘block’.

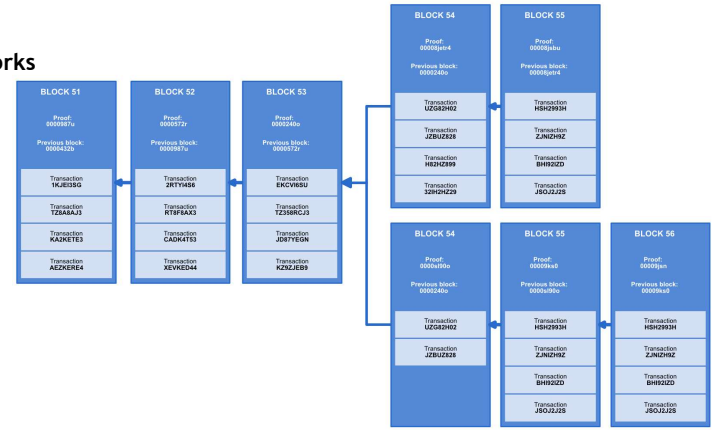


Blockchain terminologies

- **Forks**
  - The creation of an ongoing alternative version of the blockchain, by creating two blocks simultaneously on different parts of the network. This creates two parallel blockchains, where one of the two is the winning blockchain.
  - When does it happens?
    - Block found at the same time
    - Software incompatibility
    - "We don't agree" split

Blockchain terminologies

- **Forks**



Blockchain terminologies

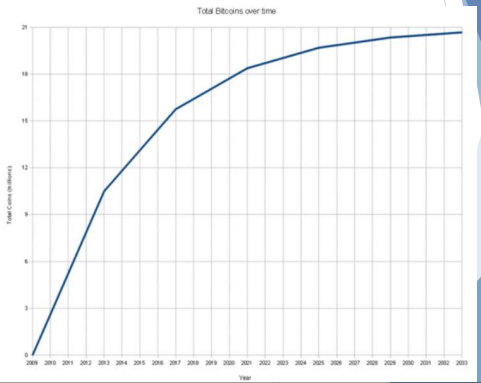
- **Bitcoin**
  - Crypto currency, first asset based on Blockchain
  - Used for drug/weapons e-commerce, ransom ware
  - Used for remittance, speculation, store of value

"What is needed is an electronic payment system based on cryptographic proof instead of trust, allowing any two willing parties to transact directly with each other without the need for a trusted third party."

Satoshi Nakamoto – October 31st, 2008

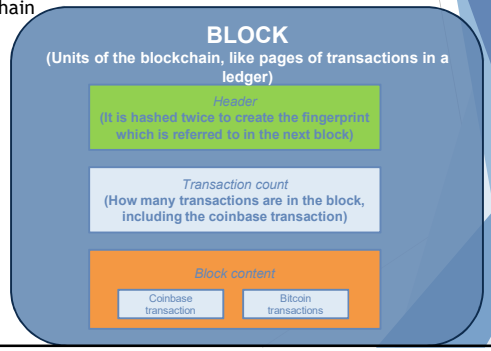
Blockchain terminologies

- **Bitcoin**
  - Monetary creation



Blockchain terminologies

- **Bitcoin**
  - Inside Bitcoin's Blockchain



Blockchain terminologies

- **Bitcoin**
  - Inside Bitcoin's Blockchain

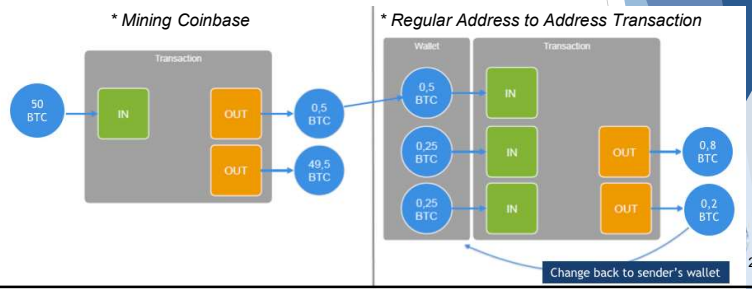
- *Block Header* : includes Technical data, Previous block hash, Merkle Root, Timestamp, Difficulty target, Nonce. Here is an example:

Height	448909
Block time	2017-01-19 09:32:58
Trades sum	5,340.87080329 BTC
Nb txs	1637
Difficulty	336,899,932,795.81
Fee	0.41239309 BTC
Hash	000000000000000000dbc2853f4939baad1f09d086fa68a0105d79378bf7629
Version	127
Confirmations	1
Merkle root	a4772eff88cbe645bba832d31730f0b42ea4d8d05d02ea62be533316bd3fb197
Prev block hash	000000000000000015278f089845eaa41753e61a0f97c54b364325ca74a8275
Size	947.32 kB
Coin days destroyed	2,913.95

Blockchain terminologies

- **Bitcoin**
  - Inside Bitcoin's Blockchain

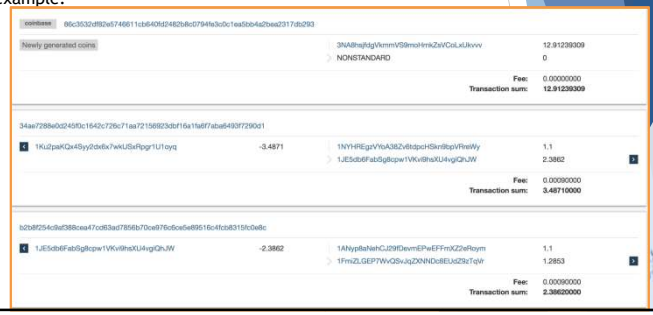
- *Block content* : Transaction Flow



Blockchain terminologies

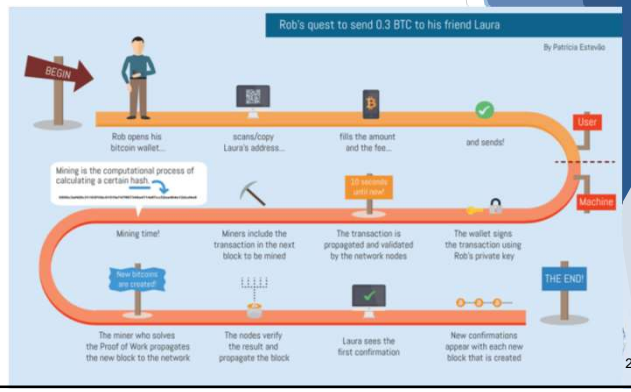
- **Bitcoin**
  - Inside Bitcoin's Blockchain

- Block Transaction example:



Blockchain terminologies

- **Bitcoin**
  - How the money transfer works



Blockchain terminologies

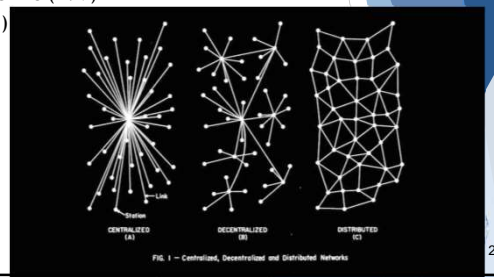
- **Ethereum**
  - Proposed in late 2013 by Vitalik Buterin (cryptocurrency researcher and programmer)
  - Online crowdsale during summer 2014
  - Bitcoin on steroids!

"A blockchain is a magic computer that anyone can upload programs to and leave the programs to self-execute, where the current and all previous states of every program are always publicly visible, and which carries a very strong cryptoeconomically secured guarantee that programs running on the chain will continue to execute in exactly the way that the blockchain protocol specifies."

Vitalik Buterin

Blockchain terminologies

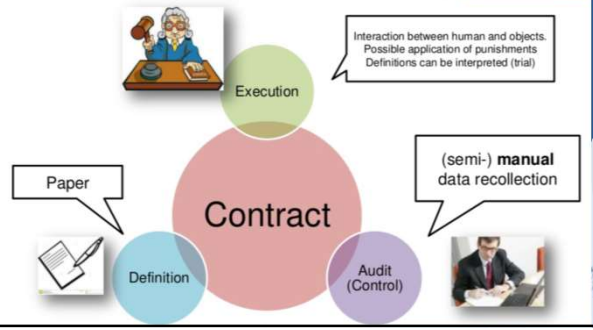
- **Ethereum**
  - Decentralised app platform (dapps)
  - Transaction & smart-contracts ledger
  - Based on the Ethereum Virtual Machine (EVM)
  - Cryptocurrency called ether ($ETH)



Blockchain terminologies

- **Ethereum**
  - *Smart Contract*

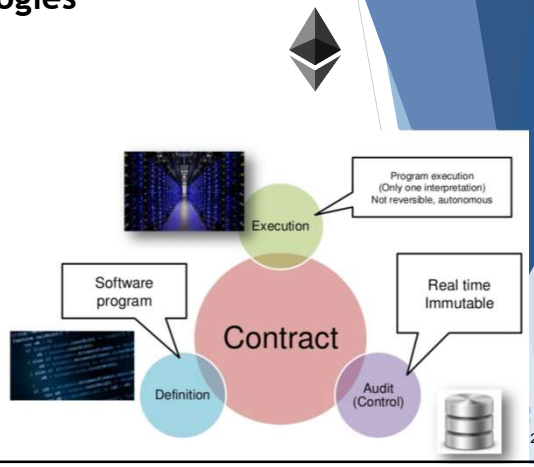
How a "Traditional" contract works:



Blockchain terminologies

- **Ethereum**
  - *Smart Contract*

How a "*Smart Contract*" contract works:



29

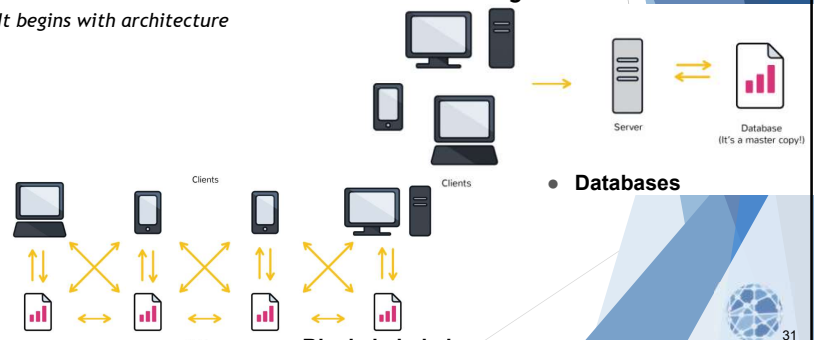
Introduction à la technologie Blockchain

- **Introduction**
  1. Quick history of Blockchains
  2. Blockchain terminologies
  3. **Distinction between databases and blockchain ledgers**
  4. Why and when to use a Blockchain?
- Cryptographic component
- Consensus components
- Blockchain structures
- Smart contract theory
- Smart contract application
- Research goals

30

Distinction between databases and blockchain ledgers

- **Distinction between databases and blockchain ledgers**
  - *It begins with architecture*



- **Databases**
- **Blockchain ledgers**

31

Distinction between databases and blockchain ledgers

- **Distinction between databases and blockchain ledgers**

Databases	VS	Blockchains
Databases have admins & centralized control		No on is the admin or in-charge
Only entities with rights can access database		Anyone can access (public) blockchain
Only entities entitled to read or write can do so		Anyone with right proof of work can write on the blockchain
Databases are fast		Blockchains are slow
No history of records & ownership of digital records		History of records & ownership of digital records

32

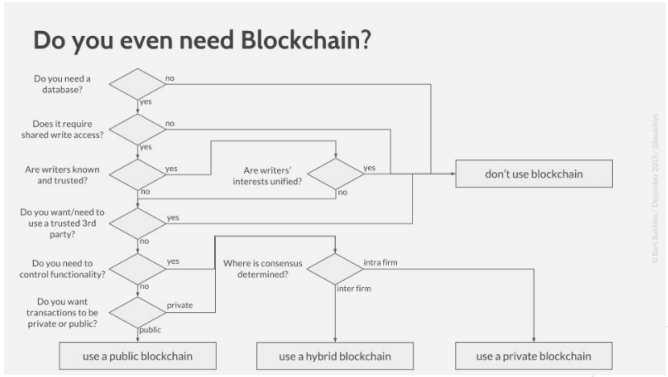
Introduction à la technologie Blockchain

- **Introduction**
  1. Quick history of Blockchains
  2. Blockchain terminologies
  3. Distinction between databases and blockchain ledgers
  4. **Why and when to use a Blockchain?**
- Cryptographic component
- Consensus components
- Blockchain structures
- Smart contract theory
- Smart contract application
- Research goals

Why and when to use a Blockchain?

- **Blockchain technology does not solve every problem**
  - Sometimes a centralized database is more suitable for a specific project
  - Many companies leverage the “Blockchain hype” to receive more funding with unrealistic projects

Why and when to use a Blockchain?



Source: https://medium.com/@sbmeunier/when-do-you-need-blockchain-decision-models-a5c40e7c9ba1

Why and when to use a Blockchain?

- **Blockchain can help with accountability and traceability**
  - Everyone can see what transactions are made on a public blockchain
  - Everyone can check they are valid
  - Example use case: the supply chain industry. You can easily track the provenance of products.

Why and when to use a Blockchain?

- **Blockchain can help with decentralization**
  - Nobody can control it
  - Everyone can check they are valid
  - Example use cases: currency creation, asset ownership management, casinos…

Introduction à la technologie Blockchain

- Introduction
- **Cryptographic component**
  1. Cryptography, hash functions and digital signatures
- Consensus components
- Blockchain structures
- Smart contract theory
- Smart contract application
- Research goals

Introduction à la technologie Blockchain

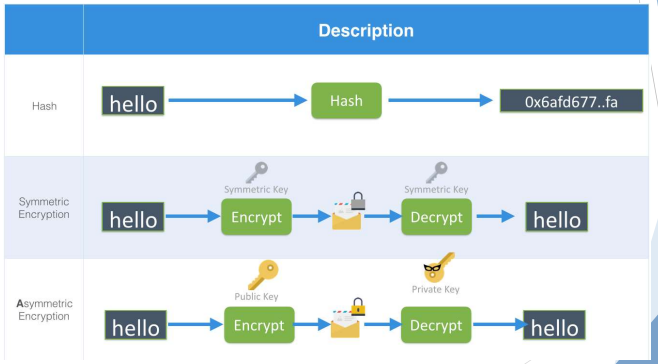
- Introduction
- Cryptographic component
  1. **Cryptography, hash functions and digital signatures**
- Consensus components
- Blockchain structures
- Smart contract theory
- Smart contract application
- Research goals

Cryptography, hash functions and digital signatures

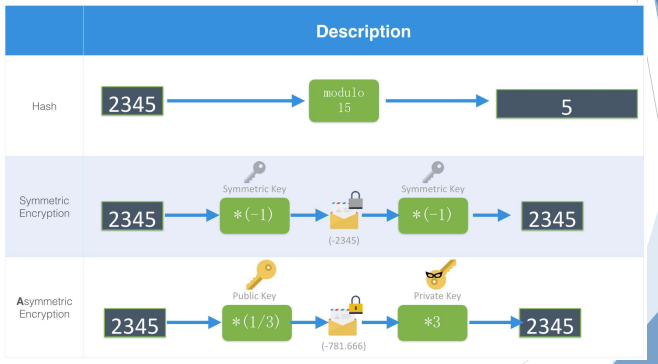
- **Cryptography**: the encryption and decryption of data
  - 2 main cryptographic concepts used in Blockchain:
    - Hashing
    - Digital Signatures
  - 3 forms of encryption that are widely used:

Symmetric cryptography	Asymmetric cryptography	Hashing
Same password to encrypt & decrypt	one password to encrypt, the other to decrypt	Maps to fixed size
2 ways function	Passwords come by pair	1 way function

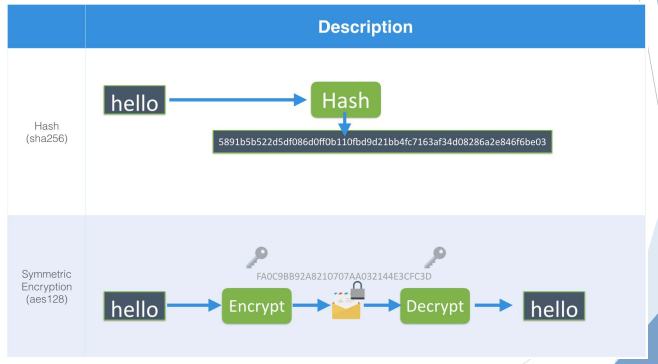
Cryptography, hash functions and digital signatures



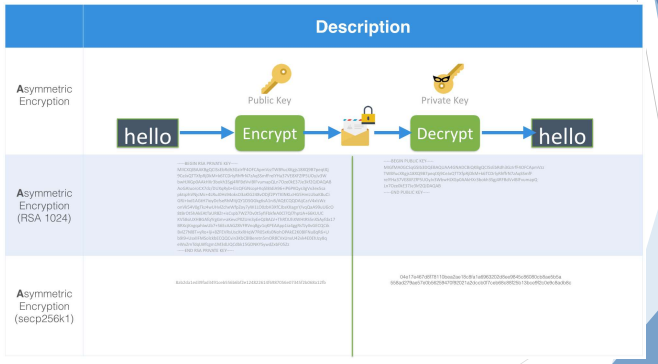
Cryptography, hash functions and digital signatures



Cryptography, hash functions and digital signatures



Cryptography, hash functions and digital signatures



Introduction à la technologie Blockchain

- Introduction
- Cryptographic component
- **Consensus components**
  1. Principles and paradigms of distributed systems
  2. Blockchain consensus algorithms
- Blockchain structures
- Smart contract theory
- Smart contract application
- Research goals

Introduction à la technologie Blockchain

- Introduction
- Cryptographic component
- Consensus components
  1. **Principles and paradigms of distributed systems**
  2. Blockchain consensus algorithms
- Blockchain structures
- Smart contract theory
- Smart contract application
- Research goals

Consensus components

- **Principles and paradigms of distributed systems**
  - ***Byzantine fault tolerance*** (BFT): the dependability of a fault-tolerant computer system, particularly distributed computing systems, where components may fail and there is imperfect information on whether a component has failed.
  - The objective of BFT is to defend against failures of system components with or without symptoms that prevent other components of the system from reaching an agreement among themselves, where such an agreement is needed for the correct operation of the system.
  - One example of BFT in use is bitcoin. The bitcoin network works in parallel to generate a blockchain with proof-of-work allowing the system to overcome Byzantine failures and reach a coherent global view of the system's state.

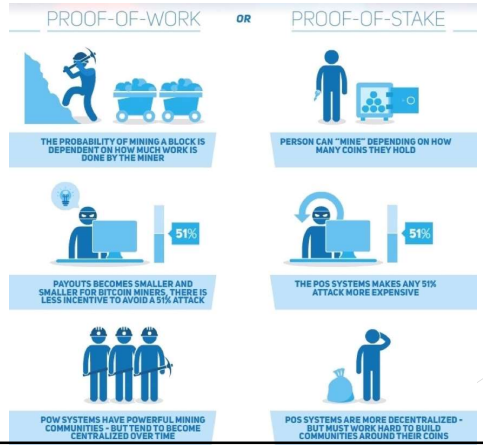
Introduction à la technologie Blockchain

- Introduction
- Cryptographic component
- Consensus components
  1. Principles and paradigms of distributed systems
  2. **Blockchain consensus algorithms**
- Blockchain structures
- Smart contract theory
- Smart contract application
- Research goals

Consensus components

- **Blockchain consensus algorithms**
  - Behind every cryptocurrency, there's a consensus algorithm. No consensus algorithm is perfect, but they each have their strengths. In the world of crypto, consensus algorithms exist to prevent double spending.
  - Proof of Work (PoW)
  - Proof of Stake (PoS)
  - Delegated Proof of Stake (DPOS)
  - Proof of Burn (PoB)
  - Practical Byzantine fault tolerant Mechanism (PBFT)
  - …

Consensus components



Introduction à la technologie Blockchain

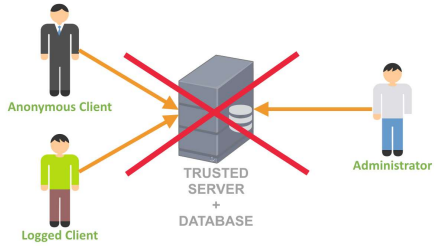
- Introduction
- Cryptographic component
- Consensus components
- **Blockchain structures**
  1. Blockchain structure
  2. Types of blockchain
- Smart contract theory
- Smart contract application
- Research goals

Introduction à la technologie Blockchain

- Introduction
- Cryptographic component
- Consensus components
- Blockchain structures
  1. **Blockchain structure**
  2. Types of blockchain
- Smart contract theory
- Smart contract application
- Research goals

Consensus components

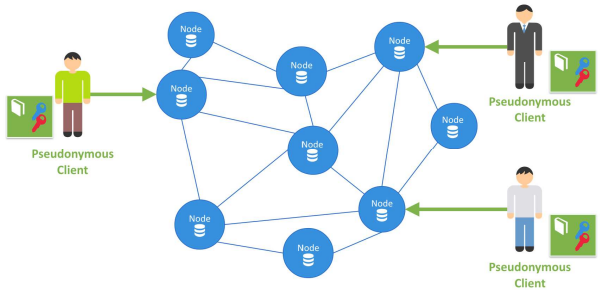
- **Blockchain structure**
  - No more client/server architecture with name roles



53

Consensus components

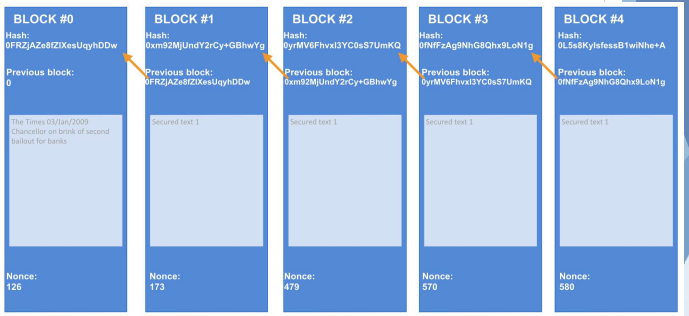
- **Blockchain structure**
  - Peer-to-peer Architecture with pseudonymous client bearing key pairs. Node each node as a database copy.



54

Consensus components

- **Blockchain structure**
  - Data structure:



55

Consensus components

- **Blockchain structure**
  - Blocks of data:

```
yallet@tyler:~/.bitcoin/blocks$ find . -name 'blk*.dat' -mtime -7 -ls
26610095 130688 -rw-------   1 yallet   yallet   133819048 Nov 23 20:37 ./blk00688.dat
26610563 130556 -rw-------   1 yallet   yallet   133682935 Nov 25 16:30 ./blk00690.dat
26611820 130992 -rw-------   1 yallet   yallet   134128511 Nov 24 17:53 ./blk00689.dat
26609041 131076 -rw-------   1 yallet   yallet   134217422 Nov 22 21:51 ./blk00687.dat
26610902 130840 -rw-------   1 yallet   yallet   133975212 Nov 21 20:41 ./blk00686.dat
26612258 130460 -rw-------   1 yallet   yallet   133583976 Nov 26 13:46 ./blk00691.dat
26611825 114692 -rw-------   1 yallet   yallet   117440512 Nov 28 09:34 ./blk00693.dat
26611491 130112 -rw-------   1 yallet   yallet   133230159 Nov 27 14:49 ./blk00692.dat
yallet@tyler:~/.bitcoin/blocks$ hexdump -C blk00691.dat | head -n 15
00000000  f9 be b4 d9 53 3a 0f 00  00 00 00 20 f3 48 e2 80  |....S:..... .H..|
00000010  bb 89 03 22 dd e9 93 ad  9e bc fd 7e 53 14 45 7a  |...".......~S.Ez|
00000020  b5 f2 97 00 00 00 00 00  00 00 00 00 1f 5b e2 c0  |.............[..|
00000030  d1 7d cb 96 9a 37 86 21  c4 a8 af 5a ad a0 ad 0b  |.}...7.!...Z....|
00000040  b2 d2 ef 15 75 c3 3a c6  67 6e 46 0e de 58 38 58  |....u.:.gnF..X8X|
00000050  d4 e6 03 18 3e c5 4e e3  fd 45 0b 01 00 00 00 01  |....>.N..E......|
00000060  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |................|
*
00000080  ff ff ff ff 49 03 d0 b8  06 2f 48 61 6f 42 54 43  |....I..../HaoBTC|
00000090  2f e7 94 bb e5 9b be e7  9c 81 e8 af 86 e6 98 a5  |/...............|
000000a0  e9 a3 8e e9 9d a2 ef bc  8c e7 8e af e4 bd a9 e7  |................|
000000b0  a9 ba e5 bd 92 e6 9c 88  e5 a4 9c e9 ad 82 e3 80  |................|
000000c0  82 2f 06 74 7d 3d e3 b3  1d 9c f7 99 01 00 ff ff  |./.t}=..........|
000000d0  ff ff 01 4b 1d d3 4e 00  00 00 00 19 76 a9 14 bf  |...K..N.....v...|
000000e0  d3 eb b5 48 5b 49 a6 cf  16 57 82 46 23 ea d6 93  |...H[I...W.F#...|
yallet@tyler:~/.bitcoin/blocks$
```

56

Introduction à la technologie Blockchain

- Introduction
- Cryptographic component
- Consensus components
- Blockchain structures
  1. Blockchain structure
  2. **Types of blockchain**
- Smart contract theory
- Smart contract application
- Research goals

Consensus components

- **Types of blockchain**
  - There mainly three types of Blockchains that have emerged after Bitcoin introduced Blockchain to the world.
  - ✓ Public Blockchain:

  no one in charge, anyone can participate in reading/writing/auditing the blockchain (i.e. Bitcoin, Litecoin, etc.)
  - ✓ Private Blockchain:

  a private property of an individual or an organization, there is one in charge of important things such as read/write or whom to selectively give access to read or vice versa (i.e. Bankchain)
  - ✓ Consortium or Federated Blockchain:

  More than one in charge. A group of companies or representative individuals come together and make decisioins for the best benefit of the whole network (i.e. r3, EWF)

Introduction à la technologie Blockchain

- Introduction
- Cryptographic component
- Consensus components
- Blockchain structures
- **Smart contract theory**
  1. Smart Contract Theory and architecture
  2. Architectures and decentralized autonomous systems
- Smart contract application
- Research goals

Introduction à la technologie Blockchain

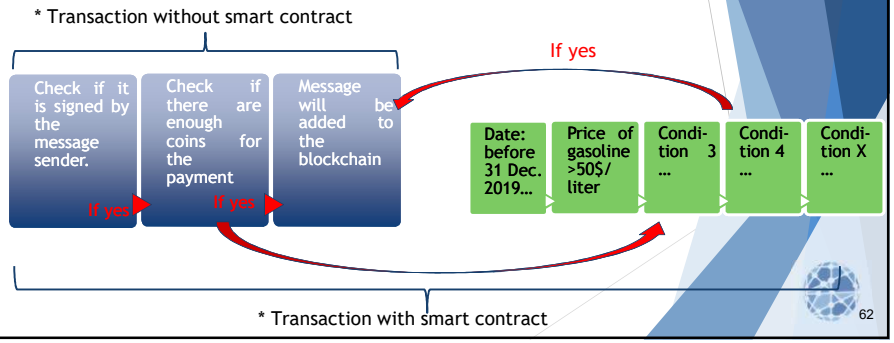
- Introduction
- Cryptographic component
- Consensus components
- Blockchain structures
- Smart contract theory
  1. **Smart Contract Theory and architecture**
  2. Architectures and decentralized autonomous systems
- Smart contract application
- Research goals

Smart Contract Theory and architecture

- **Smart Contract Theory**
  - A computer protocol designed digitally facilitate, verify, or enforce the negotiation or performance of a contract.
  - It allows the performance of credible transactions without the third parties.
  - The transactions are traceable and irreversible.

Smart Contract Theory and architecture

- **Smart Contract architecture**



Introduction à la technologie Blockchain

- Introduction
- Cryptographic component
- Consensus components
- Blockchain structures
- Smart contract theory
  1. Smart Contract Theory and architecture
  2. **Architectures and decentralized autonomous systems**
- Smart contract application
- Research goals

Architectures and decentralized autonomous systems

- **DAO** (Decentralized Autonomous Organization)
  - An organization represented by rules encoded as a computer program, which is transparent, controlled by shareholders and not influenced by a central government.
  - It's notionally like the example for getting funds for a small conference, except that it includes much more. Members buy shares in the DAO and can vote on things according to the number of shares they have. The dreamers have the idea they'll replace Democracy and run entire countries this way.
  - The DAO was the largest crowdfunding in history, having raised over $150m from more than 11,000 enthusiastic members. (ICO)
  - A DAO's financial transaction record and program rules are maintained on a blockchain.

Introduction à la technologie Blockchain

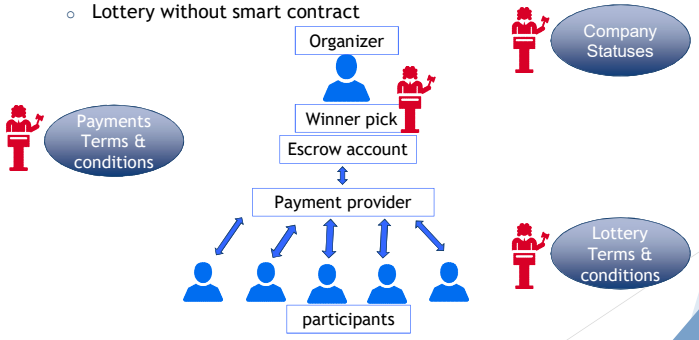
- Introduction
- Cryptographic component
- Consensus components
- Blockchain structures
- Smart contract theory
- **Smart contract application**
  1. Existing blockchain applications, related structures and architectures
- Research goals

Introduction à la technologie Blockchain

- Introduction
- Cryptographic component
- Consensus components
- Blockchain structures
- Smart contract theory
- Smart contract application
  1. **Existing blockchain applications, related structures and architectures**
- Research goals

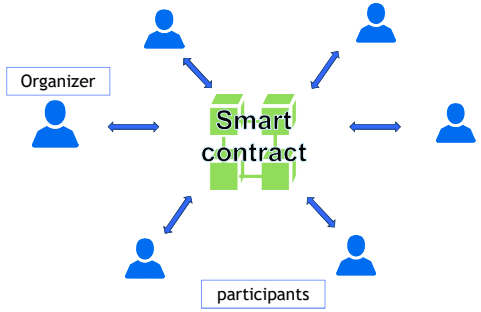
Smart contract application

- **Example 1: Lottery**
  - Lottery without smart contract



Smart contract application

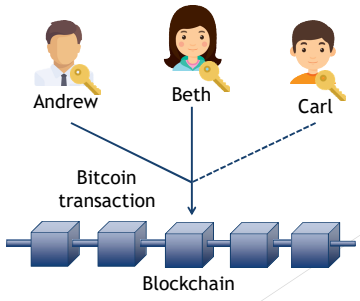
- **Example 1: Lottery**
  - Lottery with smart contract



Smart contract application

- **Example 2-1: Group wallets**
  - Enforcing at least 2 out of 3 people of a group to agree to create a valid transaction

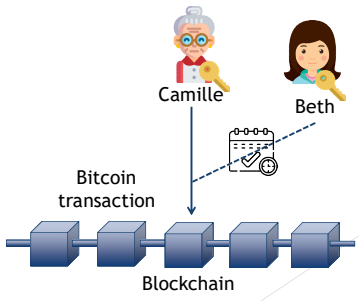
```
2 <pubKeyAndrew>
<pubKeyBeth>
<pubKeyCarl> 3
CHECKMULTISIG
```



Smart contract application

- **Example 2-2: Heritage wallets**
  - Enforcing that a transaction must be signed either by Camille OR by Beth after 5 years

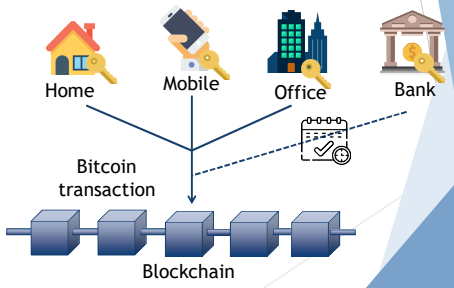
```
IF
  <pubKeyCamille>
  CHECKSIG
ELSE
  <5 y> CLTV DROP
  <pubKeyBeth>
  CHECKSIG
ENDIF
```



Smart contract application

- **Example 2-3: Secure storage**
  - Enforcing that a transaction must be signed by either 3 devices in different locations OR a recovery key deposited in the bank after 8 months

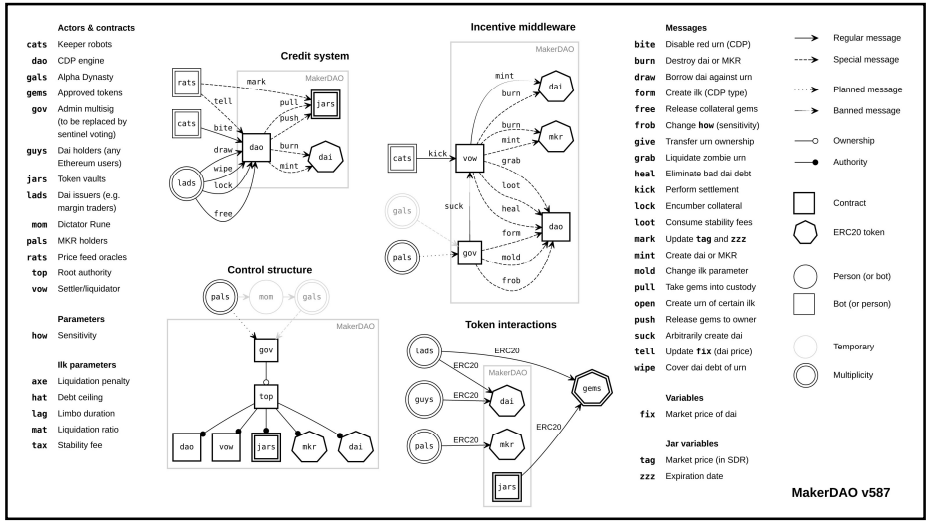
```
IF
    3 <pubKeyHome>
     <pubKeyMobile>
     <pubKeyOffice> OP_3
     CHECKMULTISIG
ELSE
    <8 m> CLTV DROP
    <pubKeyBank>
    CHECKSIG
ENDIF
```



Smart contract application

- **Example 3: Decentralized Autonomous Organizations**
  - A smart-contract that acts like a company!
  - Human shareholders can vote to change the rules of the smart-contract
  - MakerDAO: Manages a “stablecoin” token called Dai. The currency is issued when people want collateralized loans, and destroyed when people do not have enough collateral to cover their loans. This mechanism keeps Dai’s value pegged to $1.

MAKER



Smart contract application

- **Example 3: Decentralized Autonomous Organizations**
  - Dash: Dash is a blockchain and a DAO at the same time. The DAO is used to fund projects that can improve the Dash blockchain.

Dash

Existing blockchain applications, related structures and architectures

- **ERC-20**
  - Proposed on November 19, 2015 by Fabian Vogelsteller.
  - A technical standard used for smart contracts on the Ethereum blockchain for implementing tokens. (ERC: Ethereum Request for Comment, 20: the number that was assigned to this request.)
  - It defines a common list of rules that an Ethereum token has to implement, allowing developers to program how new tokens will function within the Ethereum ecosystem. These rules include how the tokens are transferred between addresses and how data within each token is accessed.
  - + 142,200 ERC-20 token contracts (as of November 19, 2018): EOS, Bancor, Qash, etc…

Existing blockchain applications, related structures and architectures

- **ERC-721: a class of unique tokens**
  - A free, open standard that describes how to build non-fungible or unique tokens on the Ethereum blockchain. While most tokens are fungible (every token is the same as every other token, i.e.ERC-20), ERC-721 tokens are all unique.
  - It defines a minimum interface a smart contract must implement to allow unique tokens to be managed, owned and traded.
- **ERC-725: Ethereum Identity Standard**
  - A proposed standard for blockchain-based identity which lives on the Ethereum blockchain.
  - It describes proxy smart contracts that can be controlled by multiple keys and other smart contracts, it can describe humans, groups, objects and machines.
  - Users should be able to own and manage their identity instead of ceding ownership of identity to centralized organizations.

Introduction à la technologie Blockchain

- Introduction
- Cryptographic component
- Consensus components
- Blockchain structures
- Smart contract theory
- Smart contract application
- Research goals

Introduction à la technologie Blockchain

- Introduction
- Cryptographic component
- Consensus components
- Blockchain structures
- Smart contract theory
- Smart contract application
- Research goals
  1. Current research and challenges faced by Blockchains

Current research and challenges faced by Blockchains

- **Blockchain scalability is limited**
  - Most decentralized blockchains don't handle more than ~20 transactions per second – for the whole network!
  - You can increase scalability if you use a more centralized consensus mechanism. There is always a tradeoff between decentralization and scalability.
  - Some solutions that are being worked on are sharding, state channels and sidechains
- **Blockchain interoperability is limited**
  - The blockchain industry is not very mature
  - Different data structures, consensus mechanism and implementations are not interoperable. With thousand of different ledgers, not everyone can talk to each other!

Current research and challenges faced by Blockchains

- **Regulation is hard to implement**
  - Lawmakers do not know how to manage cryptocurrency assets
  - Usual regulatory frameworks are not adapted to the blockchain tech
  - For example, in theory you should check the ID and provenance of funds of everyone you transact with on the blockchain for anti money laundering purpose!
- **Smart contracts security is hard to get right**
  - Hacks have caused hundred of billions of dollars of loss so far
  - Formal verification tools and auditing solutions are actively being developed