Blockchain e criptovalute

Meccanismi di base, principali piattaforme di supporto e scenari applicativi

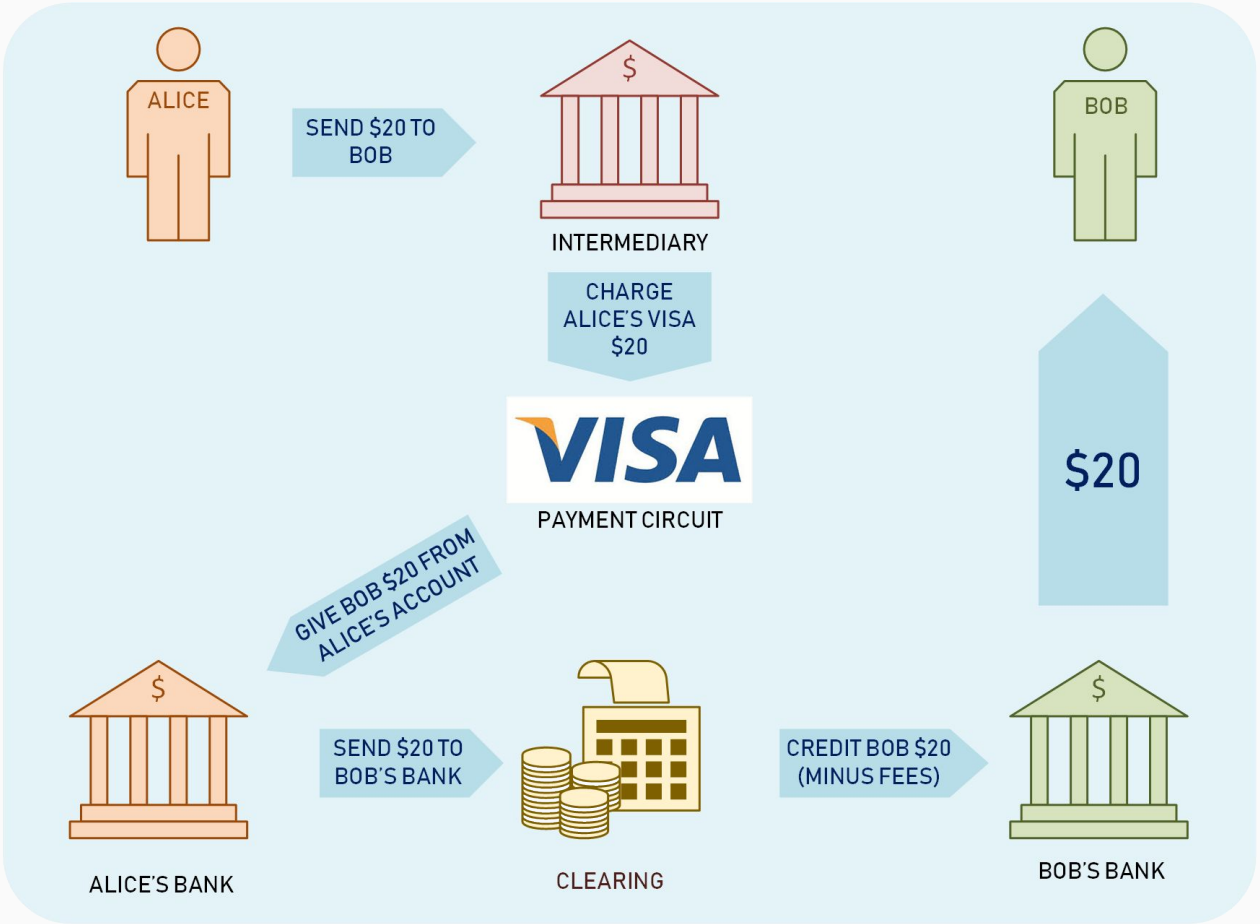
“ Administrators hold the power. They can doctor corporate accounts, delete titles from land registries or add names to party rolls. To stop the keepers from going rogue, and catch them if they do, society has come to rely on all sorts of tools, from audits to supervisory boards. [...] Now imagine a parallel universe in which lists have declared independence: they maintain themselves. This, broadly, is the promise of the blockchain”

“Disrupting the trust business”, The Economist (2017)

Il problema della fiducia

Il sistema economico e burocratico moderno si basa su **terze parti fidate** per garantire l'autenticità di documenti e la validità delle transazioni economiche

Ad esempio, prima di accettare un pagamento elettronico, è necessario che un intermediario assicuri possa andare a buon fine



Le banche garantiscono a Bob che Alice possegga realmente $20

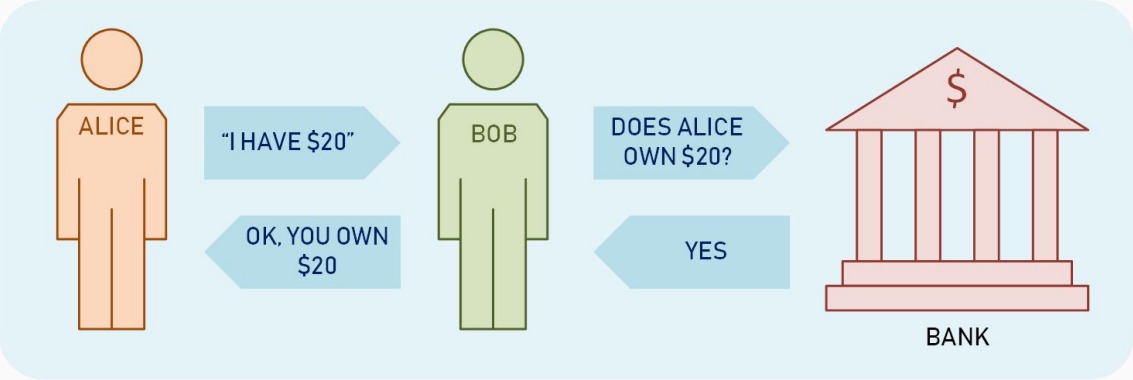
Fiducia: modello tradizionale

Questo approccio funziona, ma ha diverse problematiche:

- **Opacità**: il sistema non è trasparente (specialmente nel caso bancario)
- **Costi**: ciascun intermediario può esigere commissioni per il servizio che eroga, anche ingenti
- **Tempistiche**, che dipendono da procedure anche non digitali ed orari di apertura di banche e uffici
- L'utente non è il reale possessore dei suoi dati o del suo account; i servizi sono erogati a **discrezione** di banche o altre entità
- Possibilità di **errori**, umani o tecnici, fuori dalla responsabilità dell'utente

Fiducia: modello tradizionale

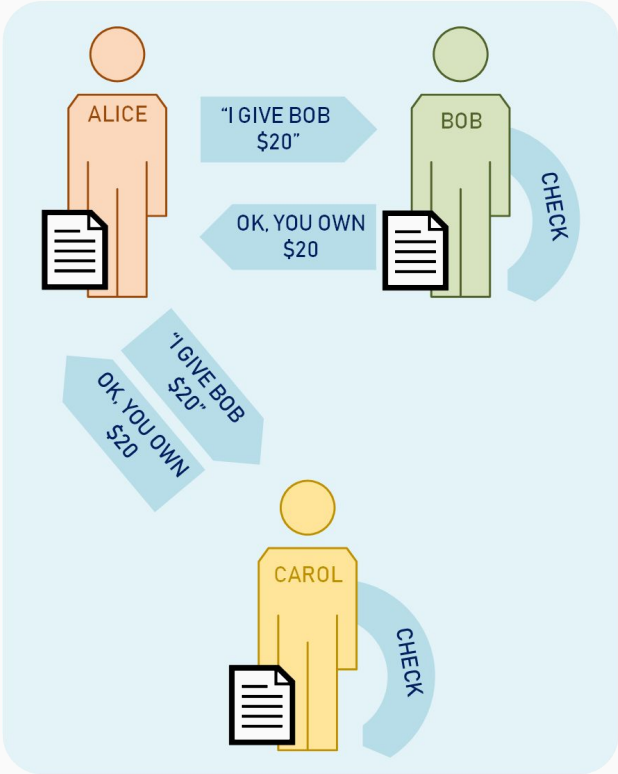
Dato che (in genere) le parti di una transazione non si fidano tra di loro, **è necessaria una vasta rete di intermediari fidati** per **garantire ed assicurare** la validità e l'adempimento di quanto viene promesso



Fiducia: ledger distribuito

Intuizione:

se tutti possiedono una copia del libro mastro (***ledger***), è sempre possibile verificare **che quanto viene promesso da una parte sia vero**, senza bisogno di intermediari esterni



Fiducia: ledger distribuito

Se…

- Tutti i partecipanti tengono traccia del saldo corrente di ognuno;
- Ciascuna operazione è registrata da ogni membro del sistema;
- Nessuna operazione può essere autorizzata senza essere stata prima accettata e registrata dagli altri partecipanti;

Allora…

- **Bob può sempre essere certo del fatto che Alice possegga $20;**
- **Alice non può spendere gli stessi soldi due volte *(double spending)***

I sistemi basati su blockchain, come **Bitcoin**, implementano con successo un ledger distribuito in grado di provvedere queste garanzie

Bitcoin

- Prima **criptovaluta**, nata come sistema di pagamenti completamente decentralizzato, anonimo e indipendente da entità governative
- **Completamente gestito, generato e controllato dai suoi utenti**, sulla base di un **protocollo** ben definito e chiaro a tutti
- **Blockchain** utilizzata come *ledger distribuito* per verificare la validità delle transazioni
- Creata nel 2008 da uno misterioso sviluppatore (o un collettivo) conosciuto *“Satoshi Nakamoto”*, la cui identità è ancora sconosciuta
- Non è infinita; possono esistere max ₿21mln

Blockchain

Una **blockchain** è, fondamentalmente, un database:

- ***decentralizzato*** (nessuna autorità centrale)
- **distribuito** (replicato su tutti i computer che ne fanno parte)
- composto da una catena di **blocchi immutabili**
- basato su **crittografia** e **firma digitale** per garantire riservatezza, immutabilità e "*proprietà*" dei dati

Originariamente progettata come **libro mastro distribuito** (***distributed ledger***) per **Bitcoin**, la prima e più significativa **criptovaluta decentralizzata**

Bitcoin

- Consiste in una rete globale di computer, detti ***nodi***, ciascuno dei quali mantiene una copia completa della blockchain
- Permette l'invio e ricezione di pagamenti verso altri utenti della piattaforma, identificati da **indirizzi** completamente anonimi e non tracciabili
- Fa estensivo uso di crittografia, sia per garantire integrità dei dati mediante *algoritmi di hash*, sia per l'autenticazione (mediante schemi di **firma a chiave pubblica/privata**)
- È potenzialmente libero ed indipendente da influenze governative, banche e industria; è una moneta completamente decentralizzata

Bitcoin: valore

In generale, il valore di una valuta può derivare da:

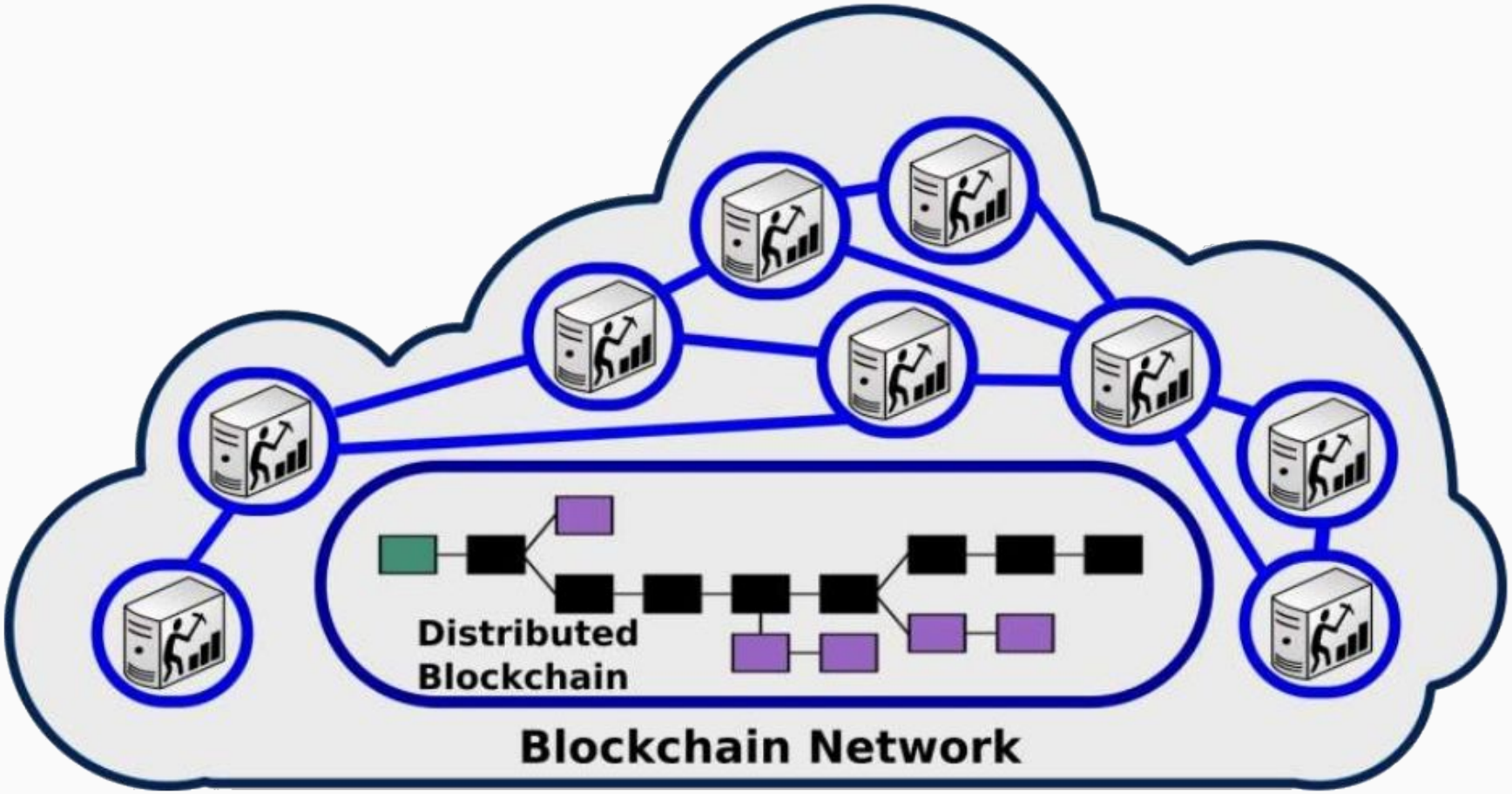
- Il suo valore intrinseco (*commodity money*, es: monete d'oro);
- La sua convertibilità in un bene o altra valuta che abbia un valore intrinseco (*representative money*, es: gold certificates)
- Una imposizione dall'alto (**fiat money**)

Una valuta **fiat** non ha valore intrinseco; il valore dipende dall'**obbligo di utilizzarla** e dalla sua scarsità (**se chiunque potesse stampare moneta, non avrebbe valore**)

Bitcoin: *Proof-of-Work*

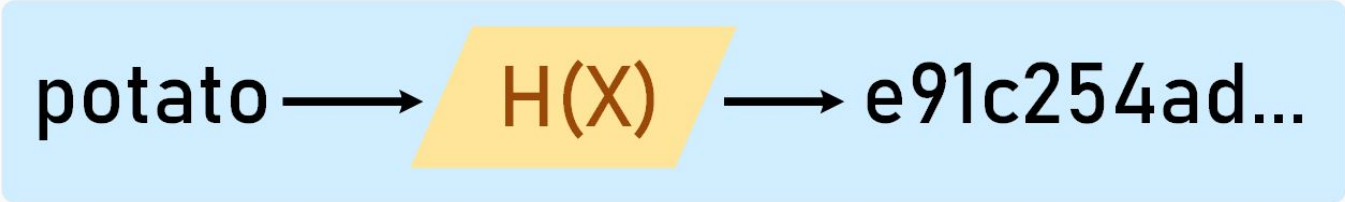
In assenza di un'autorità (come una banca centrale o governi), Bitcoin non può essere fiat money; dev'essere la rete stessa a "stamparlo" e distribuirlo.
La soluzione è ***Proof-of-Work*:**

- Alcuni nodi possono scegliere di divenire **miners**, e assumere su di sé il compito di *"scoprire"* nuovi blocchi da aggiungere in coda alla blockchain e di validare le transazioni in arrivo sulla rete
- I *miners* **investono tempo e risorse di calcolo** per **competere** fra loro, risolvendo una *challenge* matematica (ricerca di un *nonce* adeguato)
- Una volta che un nodo ha scoperto una soluzione, **il miner reclama e pubblicizza il suo blocco alla rete**, ricevendo **nuovi bitcoin** e **commissioni sulle transazioni** come premio; gli altri abbandonano il lavoro e ricominciano con un nuovo blocco.



*I nodi **miner** mantengono collettivamente la blockchain*

Algoritmi di hashing



Funzione di hash crittografica: algoritmo che, dato una serie di dati in input, ritorna un valore numerico univoco (detto **digest** o **hash**) avente, almeno, le seguenti proprietà:

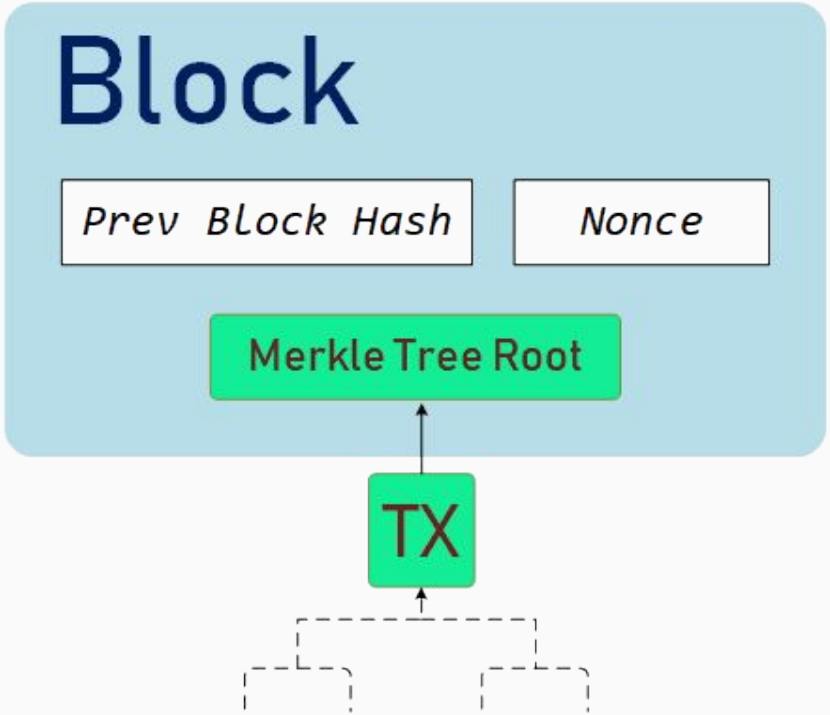
- **Assenza di collisioni:** dev'essere impossibile, o quasi, per due valori diversi avere identico hash
- **Non invertibilità**: ricostruire l'input partendo dal suo hash deve idealmente **richiedere la prova di tutti i valori possibili** (attacco ***brute force***)

Queste proprietà rendono ideali le funzioni di hash nel garantire **integrità dei dati**

Blockchain: struttura

- Ciascun blocco contiene da 1 ad N transazioni
- Ogni blocco fa riferimento all'*hash crittografico* del blocco precedente
- Una volta aggiunto, **un blocco è immutabile, e diviene traccia permanente delle transazioni a cui è associato**

Blockchain: blocchi

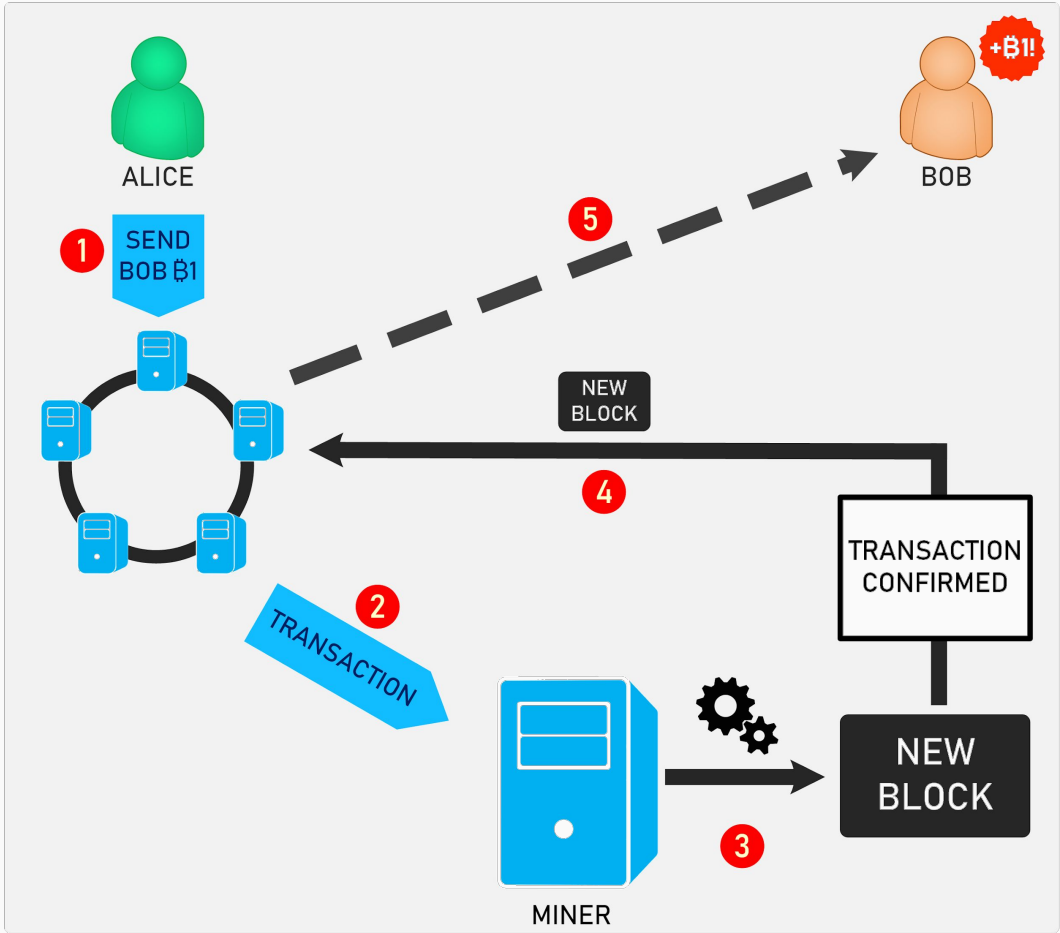


Ogni blocco contiene idealmente:

- **L'hash del blocco precedente**
- L'hash **rappresentante la radice dell'albero delle transazioni**, a loro volta **inalterabili** grazie ad hash *(Merkle Tree);* è quindi possibile lasciare **solo un riferimento alla transazione radice**, e recuperare in maniera inalterabile i dati successivamente, risparmiando notevolmente lo spazio su disco utilizzato

Bitcoin: miner e transazioni

- Quando un utente desidera inviare BTC ad un altro utente, genera una **transazione** utilizzando **il suo indirizzo**, e la invia ai miner che conosce; oltre all'ammontare desiderato, dev'essere aggiunta una commissione
- I miners **raccolgono le transazioni,** mentre ricercano un valore che soddisfi la challenge della *Proof-of-Work* per reclamare il blocco
  - **Il primo miner che riesce in questo intento crea e pubblica un nuovo blocco, che viene aggiunto alla blockchain da tutti i nodi della rete**
  - Le transazioni sono state aggiunte **immutabilmente** al blocco, ne rappresentano parte integrante e non possono più essere ripudiate o modificate
  - Il miner **guadagna la ricompensa** per l'operazione (attualmente ฿12.5), oltre che le **commissioni** incluse nelle transazioni minate



I *miner "minano"* nuovi blocchi, aggiungendo le transazioni pendenti alla chain

Bitcoin: *Proof-of-Work*

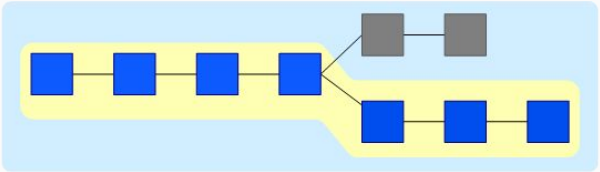
Il campo **nonce**, invece, rappresenta la challenge della *Proof-of-Work* di Bitcoin che i miners debbono vincere:

$$\text{hash}(\textit{block}) < \text{difficulty}$$

- Dopo avervi aggiunto le transazioni che ha ricevuto, il miner ricerca un valore di **nonce** tale che **l'hash crittografico del blocco intero sia minore di un dato valore** (**target**). Questa operazione richiede fortuna e molte risorse
- Un **livello di difficoltà** viene scelto ogni 2016 blocchi attraverso un algoritmo noto a tutti, per **mantenere il ritmo medio di un nuovo blocco ogni 10 minuti.**

Bitcoin: consenso distribuito

Per evitare che si formino chains in competizione, la rete di Bitcoin utilizza il principio della ***longest chain*** per decidere quale sia la futura sequenza di blocchi considerata ufficiale da tutti i blocchi:



- Viene privilegiata la catena con più lavoro dietro, ovvero la catena su cui i miners hanno **dimostrabilmente** utilizzato più risorse
- **La rete passa immediatamente ad un ramo più lungo**, scartando gli altri
- Quando un blocco finisce in un ramo *"morto"* diviene scartato, e diviene *stale*: le sue transazioni sono ritornate al *pool*, in attesa di un nuovo mining

Questo è l'algoritmo che permette il raggiungimento di **consenso distribuito**

Blockchain: oltre Bitcoin

Bitcoin attraverso la sua blockchain ha mostrato un nuovo modello:

- Basato su **consenso completamente distribuito,** basato su *Proof-of-Work*
- In grado di mantenere completa fiducia in assenza di terze parti
- Libero, “democratico” e senza autorità centrali

Questa prima implementazione però è risultata piuttosto *limitata* al di fuori del caso d’uso economico:

- Scarsa espressività, prettamente limitata a transazioni finanziarie, e non facilmente estendibile
- Impossibilità (o quasi) di esprimere dati e relazioni complessi

Blockchain: oltre Bitcoin

A partire dal 2014, con **Ethereum** sono emerse nuove implementazione di blockchain e criptovalute, focalizzate su un maggiore potere espressivo delle transazioni.

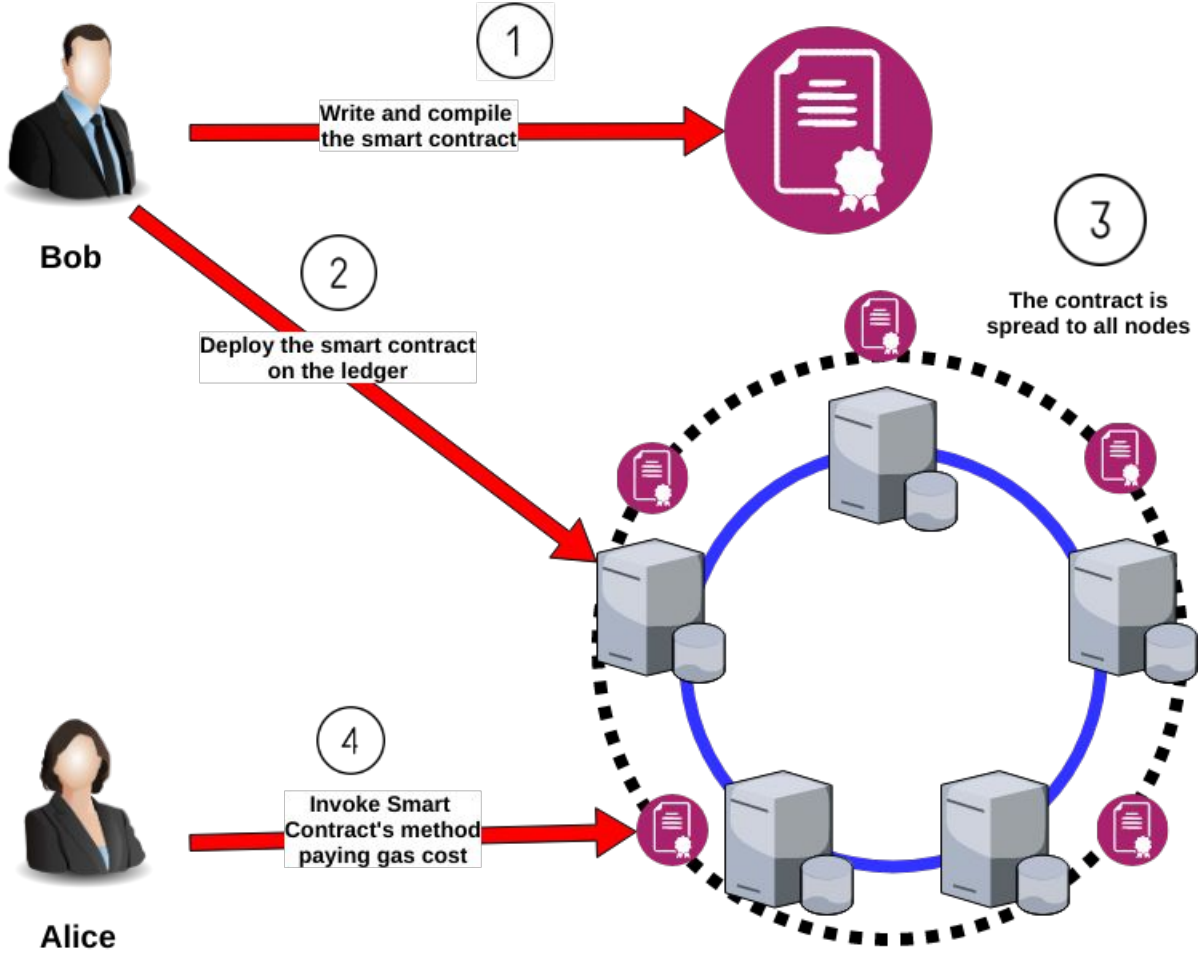
In particolare, queste si sono focalizzate su concetti chiave come:

- **Smart Contracts**
- **Exchange** decentralizzati
- **Decentralized Autonomous Organizations** (DAO)
- **Microtransazioni** e maggiore velocità di mining
- Creazione e trasferimento di risorse virtuali: **Smart Property**

Smart Contracts

Gli **Smart Contracts** sono alla base della nuova idea di **blockchain** introdotta da **Ethereum** e successivi progetti:

- Entità appartenenti alla blockchain in grado di **esprimere vincoli e obblighi contrattuali, senza bisogno di una terza parte**
- Capacità di utilizzare le transazioni per eseguire **codice arbitrario** (non solo movimenti economici)
- **Definiti** utilizzando linguaggi di programmazione specifici
- **Contratto**: entità avente uno stato, funzioni ed un indirizzo
- Ridefiniscono la transazione come **esecuzione di una funzione**, non più solo in termini economici
- Per ottenere il consenso **tutti i nodi eseguono le operazioni definite dal contratto**, e confrontano il risultato

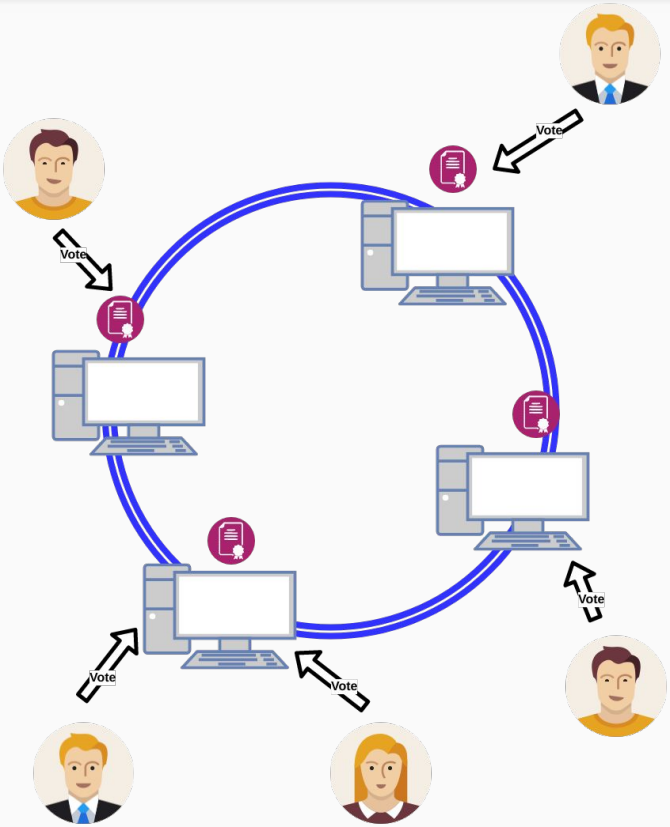


Ciclo di vita di uno smart contract

Smart Contracts: un esempio pratico

Un esempio lampante è un sistema di **voto** basato su blockchain:

- Sfruttando l'anonimato delle identità su blockchain, **il voto rimane segreto**
- Falsificare il risultato è estremamente complesso, dato che le operazioni del contratto sono ripetute su ogni nodo appartenente al ledger
- Non può essere **ripudiato**, dato che una volta validato il risultato rimane permanentemente scritto sul ledger



Smart Contracts: possibili problematiche

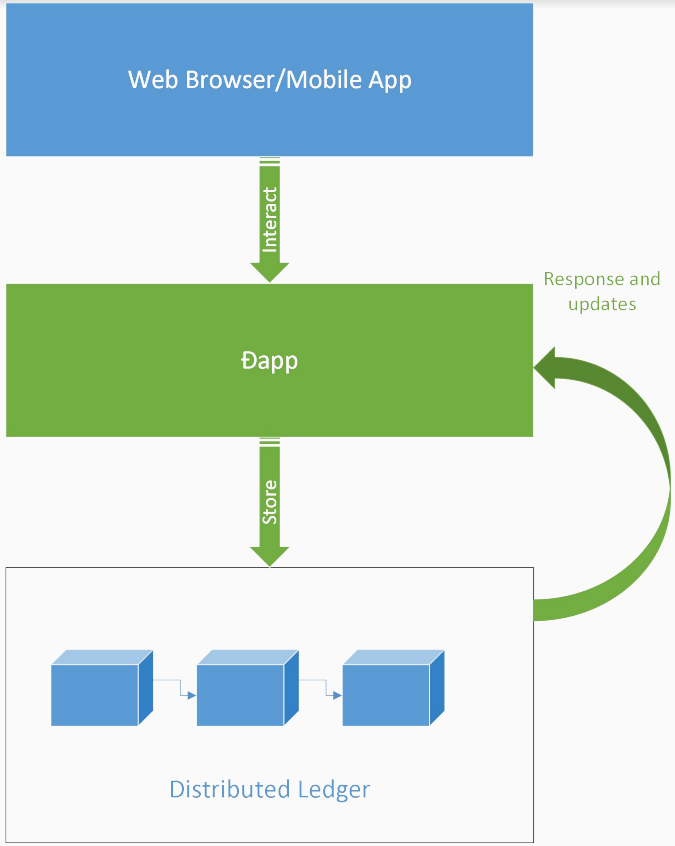
Gli Smart Contracts sono dei software, come tali sono soggetti a bug con l'effetto collaterale che le versioni "errate" non possono essere cancellate

- Uno smart contract può contenere bug più o meno critici che possono essere **sfruttati da altri sviluppatori** tramite altri smart contracts
- Un bug critico può portare al **furto di risorse** come criptovalute o altre risorse più o meno sensibili
- Una **correzione di un bug** comporta il **ri-deployment** di uno smart contract (e di altri ad esso associati) con conseguente costo in termini di commissioni (**gas** nel gergo di Ethereum)

Đapps

Decentralized applications (Đapps) sono applicazioni che vengono *“eseguite”* sulla blockchain:

- Con logica principale realizzata mediante **smart contracts**
- Con Blockchain come backend decentralizzato e pubblicamente gestito
- Intrinsecamente resistenti a censura e a controllo esterno
- Che possono interagire con dati provenienti dal mondo esterno mediante sorgenti di dati dette **oracoli**



Đapps: caso d'uso

uPort è un esempio molto noto di **Đapp**, è un sistema di **identità digitale** completamente decentralizzato e distribuito

- Ha come punto di ingresso **un'applicazione mobile**
- Basato sulla blockchain di **Ethereum**
- Utilizza i **sensori biometrici** dello smartphone come sistema di autenticazione forte
- Ha un sistema di **claims** e **attestations** per ricevere e diffondere informazioni personali (anche sensibili)
- Ha un sistema di **recupero dell'identità** basata sulla propria reputazione sulla blockchain

Smart Property and virtual asset

Gli *smart contracts* hanno in sé la potenzialità di **eliminare intermediari terzi** anche in altri campi oltre che quello economico:

- Ogni proprietà è **registrata** sul ledger pubblico come **risorsa**
- Ogni **risorsa** è associata ad un **unico utente** registrato sulla blockchain che sarà l'unico a poterla gestire
- **Trasferire** la proprietà di una proprietà sarà una semplice **transazione** con una chiamata ad uno **smart contract**
- Possibilità di avere potenzialmente **completa trasparenza** dei dati

Questa idea ovviamente si appoggia ad una alta diffusione ed uso dei ledger pubblici, come la blockchain di Bitcoin o Ethereum

Decentralized Autonomous Organization

Le **Decentralized Autonomous Organization** (**DAO**) riprendono il concetto di **organizzazione** e le traspongono sulla blockchain:

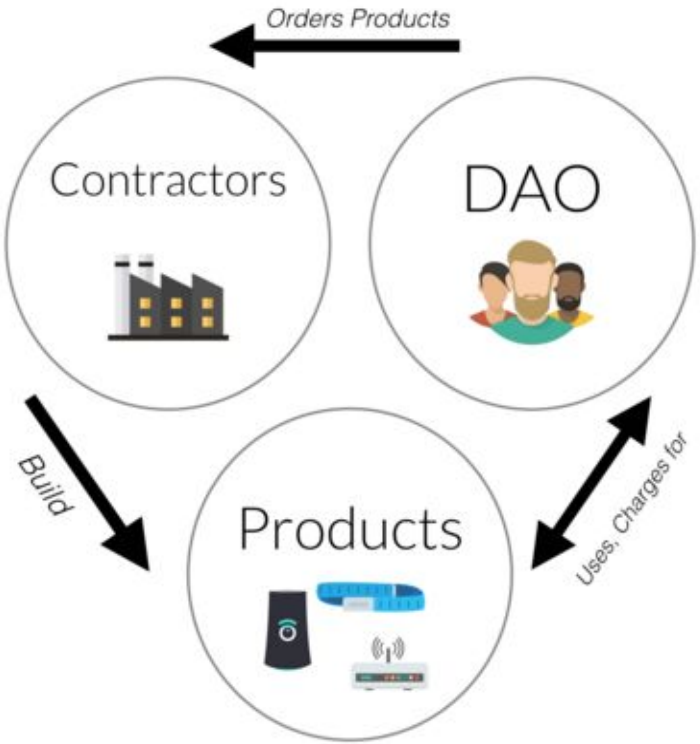
- **Smart Contracts** utilizzati per **regolamentare** il funzionamento dell'organizzazione
- **Procedure di gestione automatizzate e decentralizzate** dal codice del DAO stesso
- Il **DAO** può ricevere, effettuare **transazioni** regolate dalle linee guida decise dai membri dell'organizzazione (ad es. pagare stipendi)

Dai DAO si possono derivare *idealmente* anche altri modelli come le corporation e le società

Decentralized Autonomous Organization

Un esempio di **DAO** può essere un **marketplace** per la vendita di prodotti

- Il DAO può **ordinare** dei prodotti dai contractors e pagarli
- il DAO può metterli a **disposizione sul marketplace**, una volta che questi prodotti sono stati recapitati
- Il DAO *"incassa"* il pagamento e crea l'ordine di spedizione, ricevuto un ordine
- Il DAO agisce come fosse un'**organizzazione autonoma**



DAO: possibili problematiche

Il DAO va a sostituire un'entità fisica definendo regole ben precise a livello di smart contracts, questo può comportare diversi problemi

- Se le regole non sono ben definite posso avere delle **situazioni di instabilità** che porta a risultati completamente inaspettati
- Il DAO ha l'amministrazione di risorse economiche e fondi, un eventuale bug negli smart contracts che lo regolano potrebbe causare la **perdita totale di queste risorse**
- **Riparare** un eventuale bug potrebbe essere **molto oneroso**, e in alcuni casi potrebbe richiedere l'intervento dell'intera community, rompendo l'originale idea di decentralizzazione totale

The DAO

- Un **DAO** (chiamato *“The DAO”*) è stato creato il 30 Aprile 2016 e finanziato sulla blockchain pubblica di **Ethereum**, ricavando fondi per oltre $150m
- **Numerose vulnerabilità** furono trovate nel **codice degli smart contract che regolavano il DAO**, più volte minimizzate dagli sviluppatori di questo
- Il 18 Giugno 2016 un utente anonimo riuscì a **sfruttare alcune di queste vulnerabilità e rubare** 3.6M di Ether dai fondi del DAO
- Per rimediare a questo la **Ethereum Foundation** propose ed attuò un “fork” della blockchain pubblica, per annullare le transazioni
- L’attaccante minacciò **cause legali** poiché gli ether da lui ottenuti non erano stati sottratti in modo illegale, in quanto lui non aveva violato alcuna legge.
- Alcuni utenti rifiutarono di applicare le modifiche richieste, separandosi e dando origine ad una nuova criptovaluta (*Ethereum Classic*)

Chi utilizza realmente la blockchain?

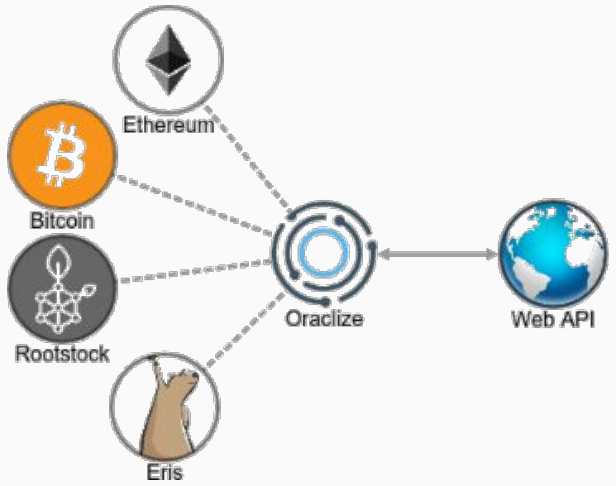
Ci sono molte compagnie che hanno fatto di blockchain il loro **core business**, adattandolo a molti use case ed **estendendo le funzionalità** messe a disposizione dalle piattaforme

- **Oracoli** per smart contracts che accedono a servizi esterni
- **Assicurazioni** online decentralizzate
- Sistemi di **micropagamenti** con crypto valute e non
- Business basati su **sharing economy**, ove blockchain è utilizzata per garantire e forzare il rispetto degli accordi pattuiti
- **Decentralized computing**

Oraclize

Oraclize è una startup che implementa oracoli per tutte le più note piattaforme blockchain:

- Utile per interrogare servizi esterni con un **risultato certificato**
- Viene utilizzato alla base di moltissimi servizi su blockchain che hanno bisogno di accedere ad una fonte di dati esterna **fidata**
- Il loro business si sta spostando anche verso ledger **permissioned** e **privati**, per permetterne l'uso anche in applicazioni sensibili



Etherisc

Etherisc è un'azienda che si occupa di **assicurazioni decentralizzate** sulla blockchain pubblica di **Ethereum**

- Vari tipi di assicurazione: voli, social security, assicurazione del raccolto
- Si basano sull'interrogazione di servizi esterni tramite **oracolo**
- Utilizzano degli smart contracts pubblici come ogni **Đapp**
- Il sistema di rimborso è **completamente automatizzato**: il contratto interroga l'oracolo **autonomamente** ed emette rimborsi se necessario

ETHERISC

Blockchain oltre Bitcoin ed Ethereum

Questa presentazione ha mostrato Bitcoin ed Ethereum, *e la loro accezione di Blockchain*, per diverse ragioni:

- C'è molto fermento dietro al mondo delle criptovalute e dei ledger distribuiti, con una forte disinformazione su cosa realmente siano
- Tutto questo è ulteriormente aggravato da fortissima speculazione economica e "*fear of missing out*" da parte di investitori e imprese
- Non esiste una definizione univoca di cosa sia e come debba funzionare una *"Blockchain"* (ma esistono delle proposte di standard)

Ripple e **Hyperledger Fabric** sono esempi di sistemi distribuiti basati su ledger distribuiti, che si ispirano ma non rispecchiano l'accezione di *"blockchain"* tradizionale

Ripple

Ripple è una implementazione di ledger distribuito, **exchange** e criptovaluta (**XRP**, **attualmente terza per capitalizzazione**), che non fa uso di *proof-of-work*:

- **Ripple protocol**: consenso distribuito per scambi economici fra istituzioni finanziarie, in XRP o valute fisiche (come EUR o USD)
- No mining, e zero commissioni su transazioni
- Bassissima latenza sulle commissioni
- Principalmente rivolto al mondo bancario
- Ripple può usare XRP come **valuta intermedia** per il forex, abbattendo tempi e costi di cambio

IBM Hyperledger Fabric

Hyperledger Fabric, sviluppato da IBM, è un **ledger permissioned** appartenente al progetto Hyperledger della Linux Foundation

- Implementa un **ledger permissioned**, ovvero dove l'accesso al network ed al registro è subordinata all'approvazione dei partecipanti
- Ha un sistema di **Smart Contracts** simile a quello messo a disposizione di Ethereum
- Fornisce le garanzie offerte da un ledger distribuito, senza renderne pubblico il contenuto
- Utile in ambienti industriali ed eterogenei (ad. es. filiera, tenere traccia di proprietà, ...)

HYPERLEDGER

Bibliografia e riferimenti

Bitcoin and Cryptocurrency Technologies - https://www.coursera.org/learn/cryptocurrency

Understanding the DAO attack - https://www.coindesk.com/understanding-dao-hack-journalists/
https://www.bloomberg.com/features/2017-the-ether-thief/

Paritiy Multi Sig wallet attack - https://medium.com/blockcat/on-the-parity-multi-sig-wallet-attack-83fb5e7f4b8c

S. Nakamoto, *"Bitcoin - A Peer-to-Peer Electronic Cash System"* - https://bitcoin.org/bitcoin.pdf

Ethereum Whitepaper - https://github.com/ethereum/wiki/wiki/White-Paper

Etherisc Whitepaper - https://etherisc.com/whitepaper

Storj - https://storj.io/

Golem - https://golem.network/

Streamr - https://www.streamr.com/

Bibliografia e riferimenti

Bitcoin: The End of Money as We Know It - http://www.imdb.com/title/tt4654844/

The Verge, *“Blockchain is Meaningless”* - https://www.theverge.com/2018/3/7/17091766/blockchain-bitcoin-ethereum-cryptocurrency-meaning