

🔗 2023

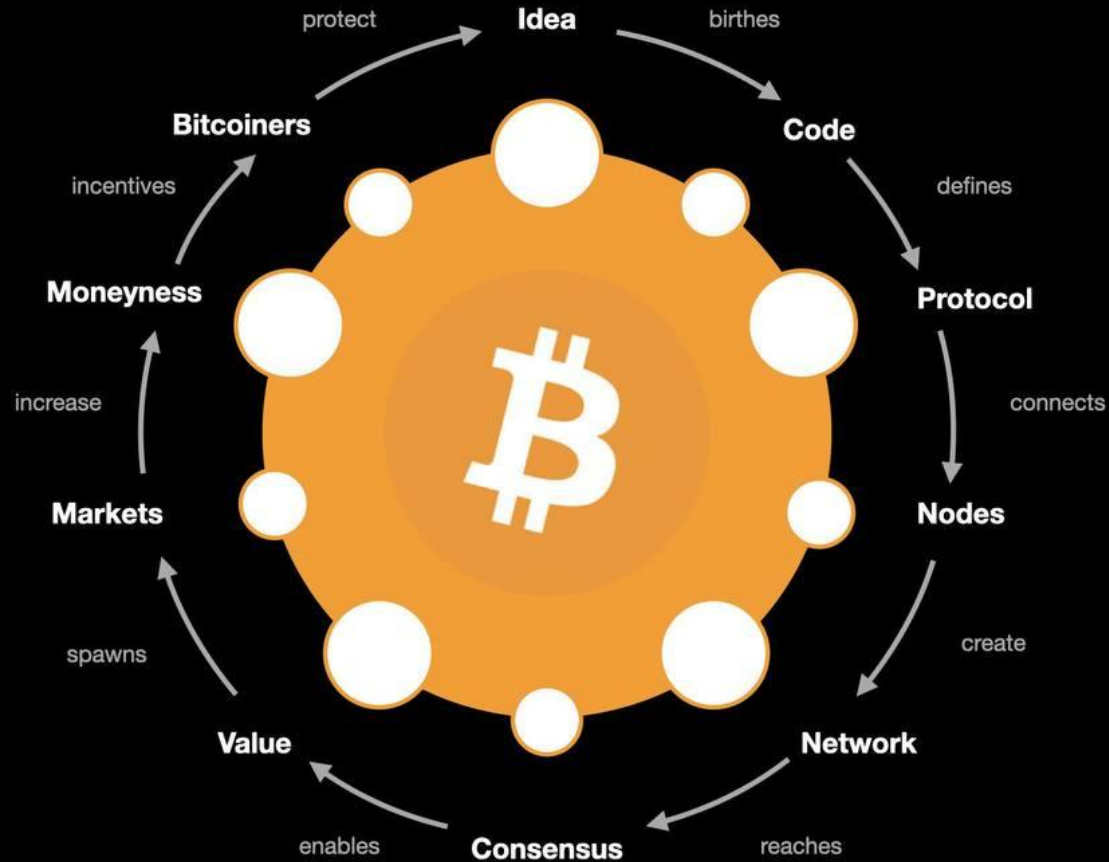
Scaling Bitcoin

Walter Maffione

AI Engineer / Head of R&D @BitPolito

Programma

- **Protocollo Internet, Protocollo Bitcoin, Scalabilità**
- **Transazioni in Bitcoin**
- **Scalare Bitcoin - problemi e soluzioni**
- **Lightning Network**
 - **Cos'è**
 - **Come funziona**
 - **Come si usa**
- **Pratica - BitPolito vi offre un caffè su Lightning Network**



moneta **bitcoin**

Protocollo **Bitcoin**

Scalare un Protocollo

Capacità di un sistema di gestire una **maggiore quantità di dati o di utenti** senza compromettere **prestazioni o stabilità**



Per Internet può significare **gestire un maggior numero di connessioni simultanee, maggiore velocità di trasmissione dati o maggiore capacità di elaborazione.**



Per Bitcoin la capacità della rete di gestire un **maggior numero di transazioni** al secondo e commissioni più basse, **senza compromettere la sicurezza e la decentralizzazione della rete.**

Protocollo Internet

Insieme di **regole e standard** che governano la comunicazione tra computer su Internet.

Internet utilizza una **gerarchia di protocolli** per trasmettere informazioni in modo efficiente e affidabile, garantendo la **connettività globale**.

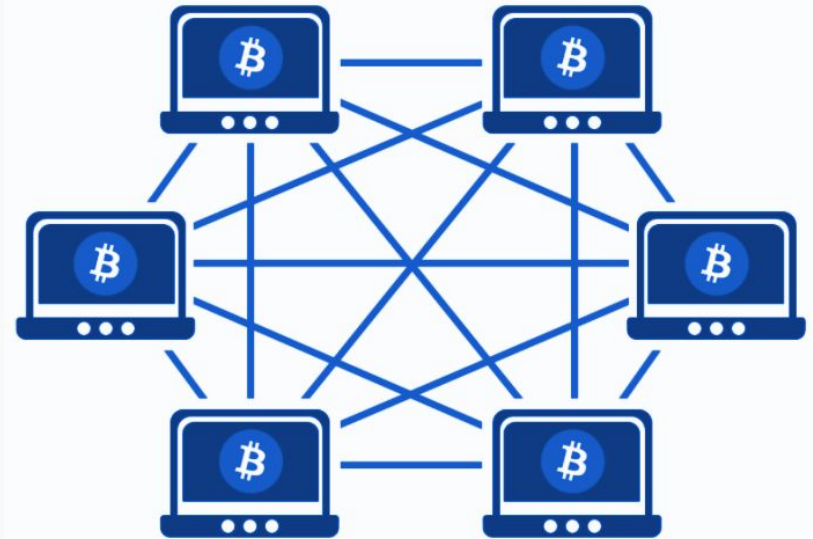
OSI Model	TCP/IP Model
Application Layer	Application layer
Presentation Layer	
Session Layer	
Transport Layer	Transport Layer
Network Layer	Internet Layer
Data link layer	Link Layer
Physical layer	



Protocollo Bitcoin

Insieme di regole che governano la rete peer-to-peer di Bitcoin

Permette transazioni sicure e affidabili senza la necessità di intermediari in un sistema distribuito, immutabile e trasparente



Scalare con layer superiori

In Ingegneria costruire delle **architetture a livelli/strati** permette di **creare** dei **sistemi** che sono molto più **stabili** e **robusti** (esempio architettura degli edifici)

Strati superiori:

- Innovazione rapida
- Competizione tra protocolli

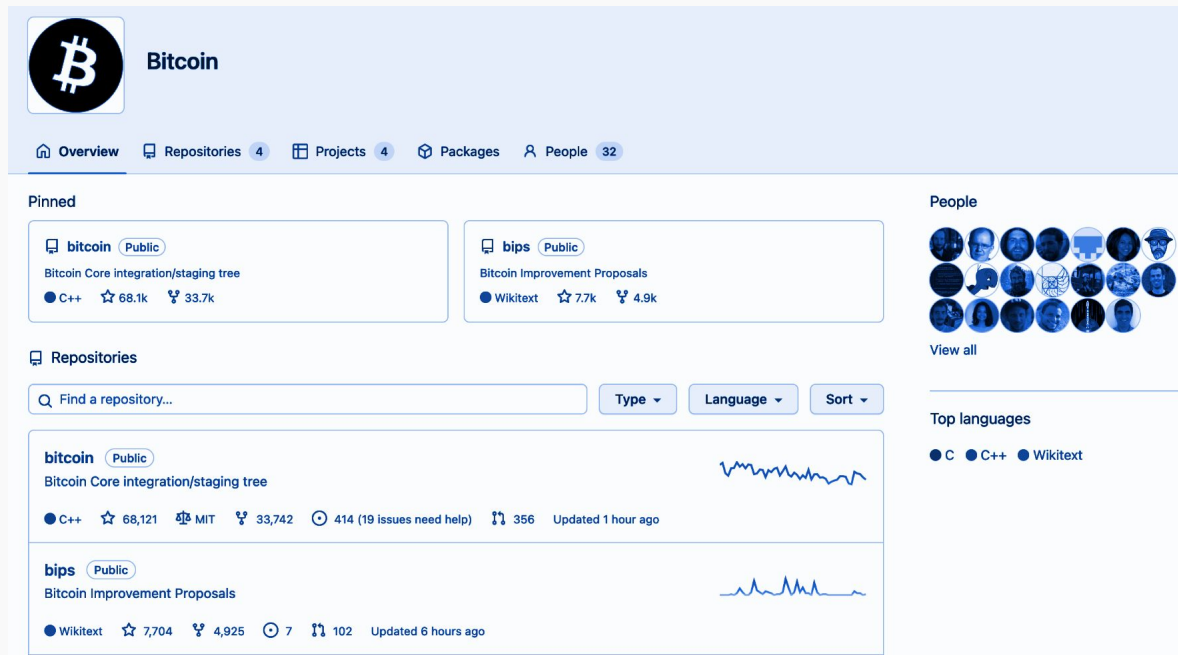
Strati inferiori:

- Stabilità
- Robustezza
- Ossificazione
- Convergenza



Bitcoin = Software

- Codice open-source
- Sistema decentralizzato
- Crittografia
- Proof-of-work
- Inflazione schedulata
- Algoritmo di aggiustamento della difficoltà
- ...



The screenshot shows the GitHub profile for Bitcoin. At the top is the Bitcoin logo and the name "Bitcoin". Below this is a navigation bar with links to Overview, Repositories (4), Projects (4), Packages, and People (32). The "Pinned" section features two repositories: "bitcoin" (Bitcoin Core integration/staging tree) and "bips" (Bitcoin Improvement Proposals). The "Repositories" section includes a search bar and a list of repositories, with "bitcoin" and "bips" highlighted. On the right side, there are sections for "People" (a grid of avatars) and "Top languages" (showing C, C++, and Wikitext).

Bitcoin

Overview Repositories 4 Projects 4 Packages People 32

Pinned

- bitcoin** (Public)
Bitcoin Core integration/staging tree
C++ 68.1k 33.7k
- bips** (Public)
Bitcoin Improvement Proposals
Wikitext 7.7k 4.9k

Repositories

Find a repository... Type Language Sort

- bitcoin** (Public)
Bitcoin Core integration/staging tree
C++ 68,121 MIT 33,742 414 (19 issues need help) 356 Updated 1 hour ago
- bips** (Public)
Bitcoin Improvement Proposals
Wikitext 7,704 4,925 7 102 Updated 6 hours ago

People

View all

Top languages

C C++ Wikitext

github.com/bitcoin

Transazioni Bitcoin

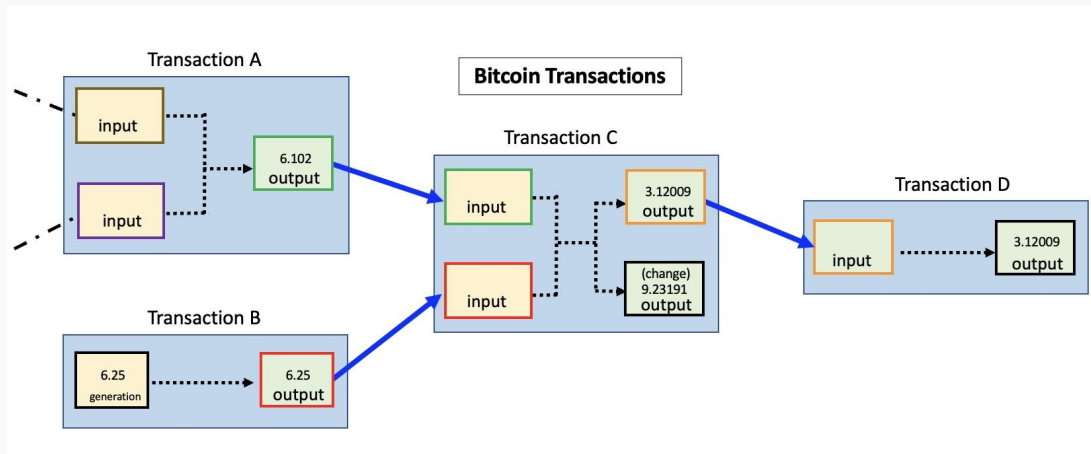
Definiscono come **inviare o ricevere bitcoin** da un indirizzo all'altro.



I fondi non ancora spesi in una transazione sono chiamati **UTXO (Unspent Transaction Outputs)**.

Un transazione è composta da:

- Input e Output con relativi amount
- Commissioni
- Script di spesa e di sblocco



Esempio di transazione

Bitcoin Script

Il linguaggio di programmazione per le transazioni in Bitcoin

Permette **attraverso un insieme di istruzioni predefinite** (OPCODE) di definire delle **condizioni di spesa** per delle transazioni in Bitcoin.

Esempi:

- Sposta questi bitcoin solo se:
 - C'è una firma valida
 - Sono passate X ore, giorni, anni
 - Ci sono almeno 2 firme su 3 possibili

Le operazioni sono limitate e per questo garantiscono un sistema **sicuro**.

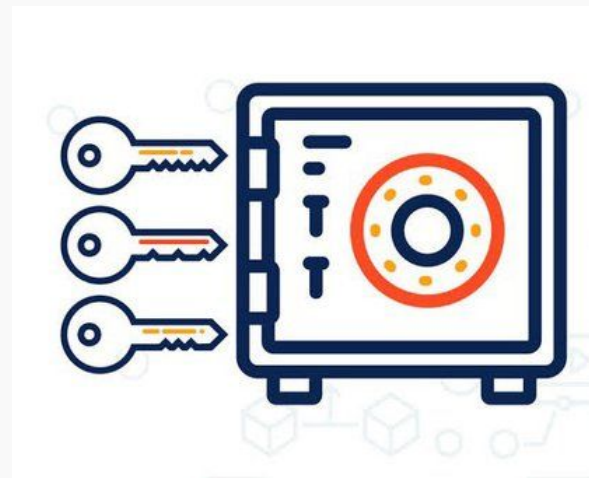
Transazioni multi-signature

Le transazioni multi-signature (**multi-sig**), sono un tipo di **transazione** Bitcoin che **richiede più di una firma per essere valida**.

È necessario il consenso di più parti coinvolte, generalmente più di una persona.

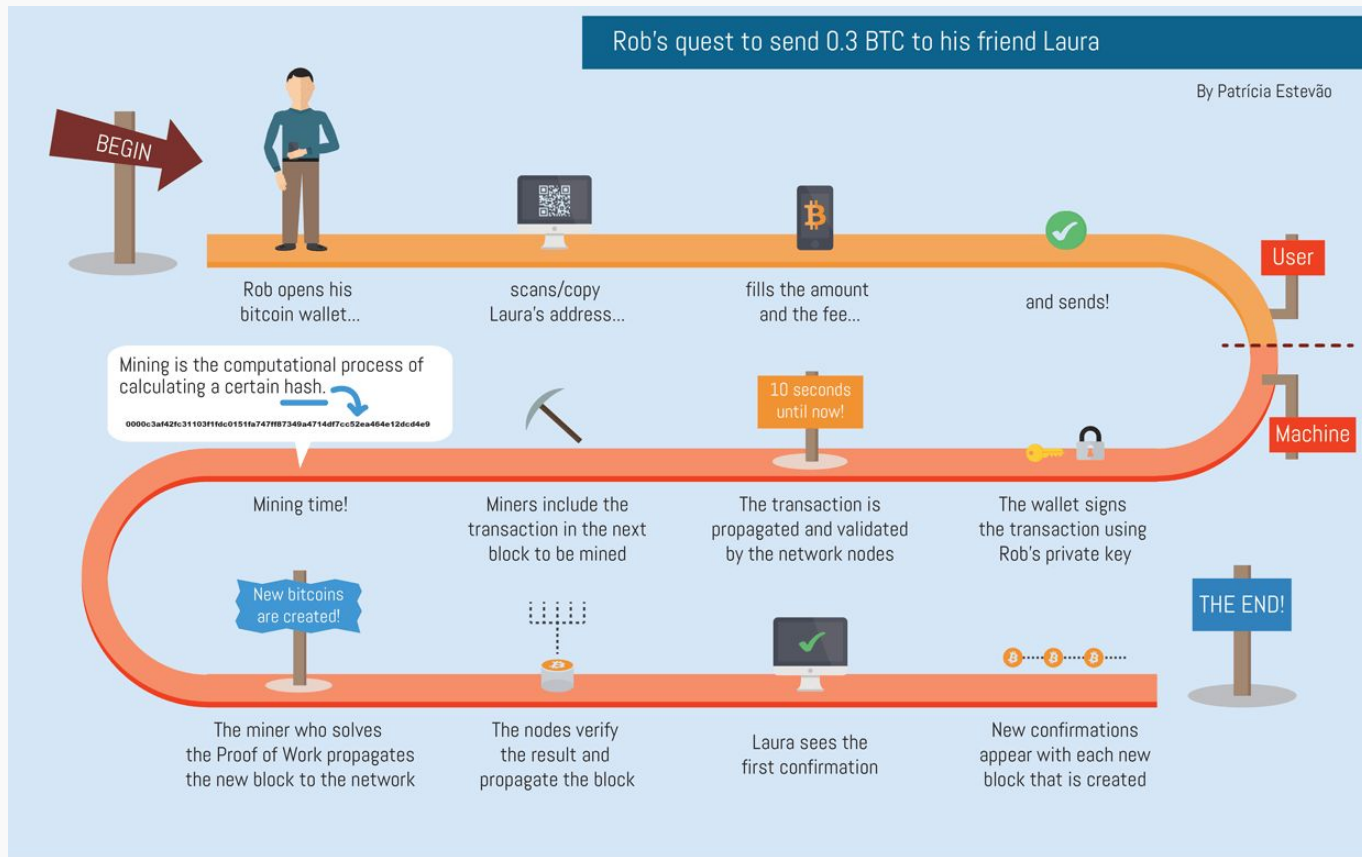
Esempio di utilizzo:

- Gestione di fondi di gruppo
- Gestione di beni digitali in un'azienda
- Canali di pagamento Lightning (vedremo dopo cosa sono)



Transazione N-M
Richiede almeno N firme su M
possibili

Ciclo di vita di una transazione



**Perchè
Bitcoin
non
scala?**

Colpa della Timechain!

La time(block)chain è una **struttura dati inefficiente** che ha il solo compito di **evitare il double spending**

Tutti i nodi devono scaricare, verificare e salvare la storia di tutte le transazioni per sempre.

Questo causa problemi di **scalabilità** e di **privacy**

Explorer:

mempool.bitpolito.it

mempool.space

Alcuni parametri della rete:

- Generazione di un **blocco in media ogni 10 minuti**
- Dimensione massima di un blocco teoricamente **4 MB**
- Se i blocchi sono pieni cresce circa di 1 TB ogni 4 anni, oggi 500 GB

Al crescere del numero di utenti il tempo medio necessario per una conferma e il costo di ogni transazione aumenta notevolmente!

Modifica al Blocksize

Cambiare la dimensione massima di un blocco ha impatto su:

- **Il tempo di validazione e download**
- **Memoria (GB)**
- **Il tempo di propagazione**
- **Centralizzazione del mining**
- **La gestione delle fees**



The Block Size War

Momento cruciale nella storia di Bitcoin, ha portato alla risoluzione di un **importante dibattito tecnico e politico** riguardante la **dimensione di un blocco Bitcoin**



Storia della dimensione di un blocco Bitcoin:

- (2009 - 2010) Nessun limite inserito nel codice
- (2010 - 2017) Dimensione massima impostata a 1 MB da Satoshi Nakamoto
- (> 2017) Dimensione massima portata a 4 MB virtuali

Dibattito

Small Blockers

Sostenevano che **una dimensione troppo grande** dei blocchi **avrebbe aumentato i costi per gli utenti che gestiscono dei nodi validanti** sulla rete, rendendo Bitcoin più centralizzato

Big Blockers

Sostenevano che la **dimensione massima** dei blocchi doveva essere **aumentata** per migliorare la scalabilità

Fine guerra e upgrade a SegWit



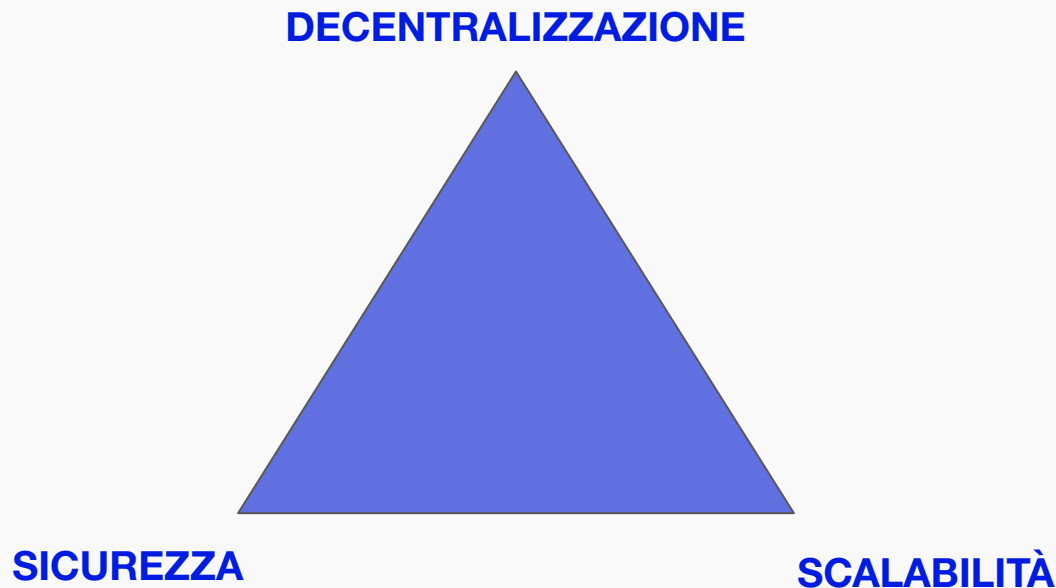
Il dibattito si chiude con l'introduzione di **Segregated Witness (SegWit)**, soft-fork che ha offerto una soluzione politica equilibrata.

Blocchi aumentati a 4 MB virtuali,
Risolve alcuni problemi andando a spostare i dati relativi alle firme delle transazioni in un nuova **struttura dati meno costosa**.

Ha permesso di sviluppare Lightning Network

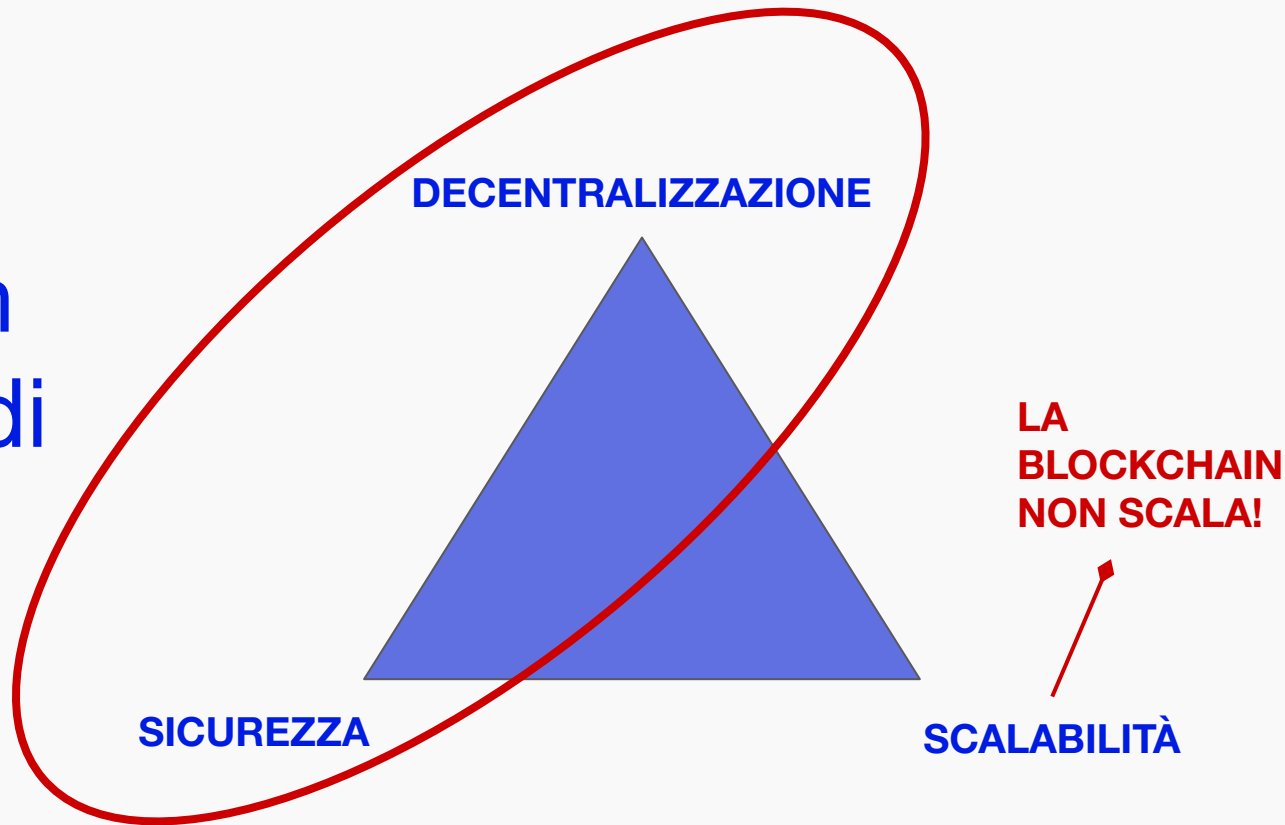
Trilemma delle blockchain

Se ne
possono
avere solo **2**
su **3...**



Trilemma delle blockchain

La
blockchain
al di fuori di
Bitcoin è
inutile



**Come
Scalare
Bitcoin?**

Bitcoin senza blockchain

Per risolvere il problema della scalabilità si utilizzano soluzioni **off-chain** chiamate **layer-2**.

L'obiettivo è utilizzare soluzioni che permettano di inviare bitcoin, senza passare dalla blockchain principale, in modo tale da garantire **transazioni più veloci, ad un costo minore e con più privacy**

Alcune soluzioni:

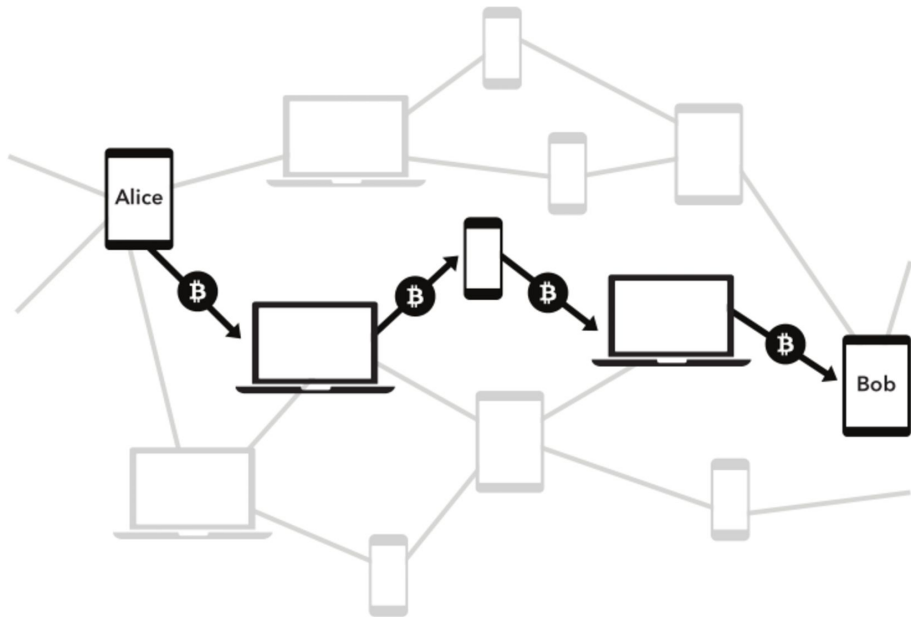
- **Lightning Network**
- **Sidechains**
- **Federazioni**
- **RGB**

Lightning Network

Tecnologia detta **Layer-2**, abilita l'invio e la ricezione di bitcoin in modo **istantaneo** e a **commissioni quasi zero**

Aumenta la **privacy** e la **scalabilità** di Bitcoin. Permette la creazione di nuovi mercati e tecnologie grazie ai **micropagamenti**

Risolve il problema della scalabilità di Bitcoin formando lo stack di protocolli **LNP/BP** analogo a **TCP/IP** per Internet

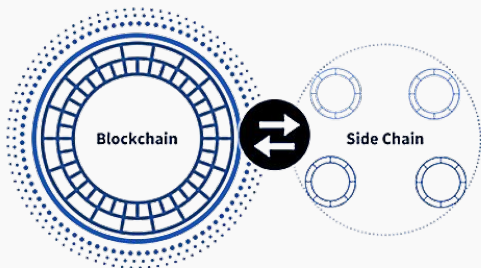


Sidechains

Rete separata ancorata alla blockchain principale attraverso un meccanismo **2-way-peg**.

I nodi della rete possono essere decentralizzati o governati da una federazione. Aumenta la scalabilità e funzionalità

Le più famose sono Liquid Network (Blockstream) e Rootstock



Vantaggi:

- **Scalabilità:** Transazione + veloci, - costo
- **Privacy:** Crittografia avanzata aumenta la privacy delle transazioni
- **Smart Contract:** Possibili transazioni e operazioni più complesse

Svantaggi:

- **Centralizzazione:** maggior rischio di censura
- **Sicurezza:** Minore rispetto a Bitcoin, rischio maggiore di attacchi

Federazioni - Fedimint

Sistemi basati su **e-cash** (D. Chaum 1982) permettono la creazione di token basati 1:1 su bitcoin grazie ad una “**mint**” e convertibili in qualsiasi momento.

È possibile creare **federazioni** per decentralizzare il sistema

Fedimint è un protocollo open-source che permette mini-federazioni fondandosi sull'idea di “seconda parte fidata” ad esempio **amici** o **parenti**.

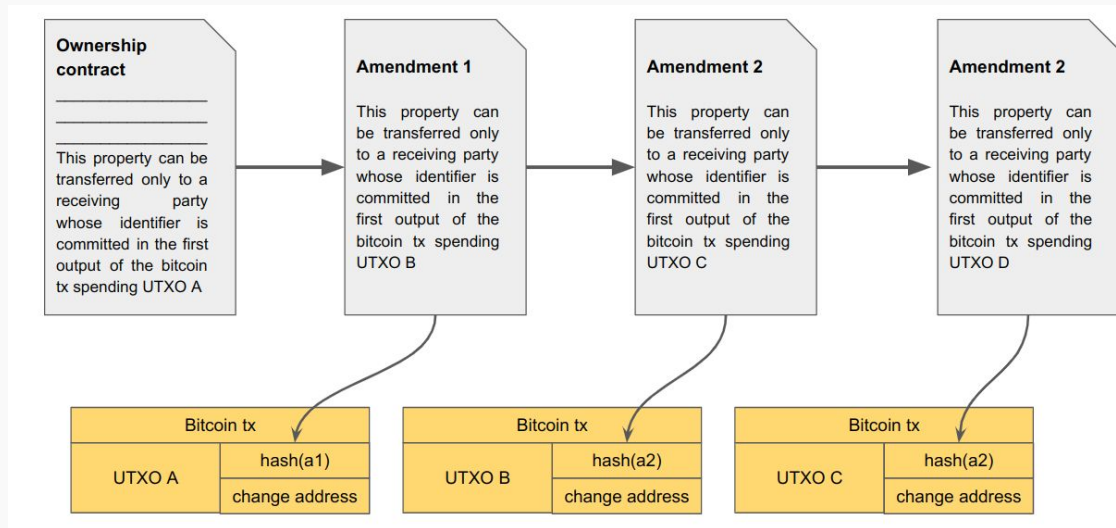
Soluzione **compromesso** rispetto alla self-custody totale, utile **per utenti inesperti** o **non disposti ad assumersi rischi di custodia**

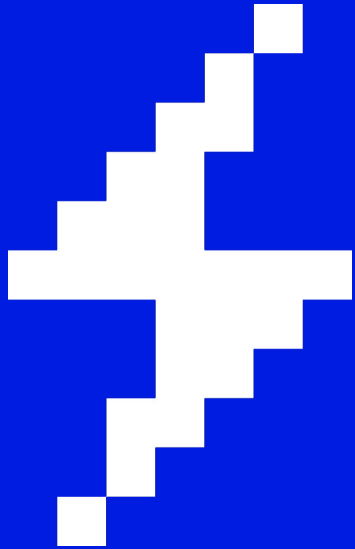


RGB

Protocollo per **emettere e trasferire asset e token** attraverso l'uso di smart contract off-chain programmabili.

Validazione lato client mantiene tutti i dati relativi alle transazioni RGB fuori dalla blockchain. Blockchain di Bitcoin usata solo come layer di **commitment**, permette di ottenere **maggior scalabilità, migliore privacy e più flessibilità**.



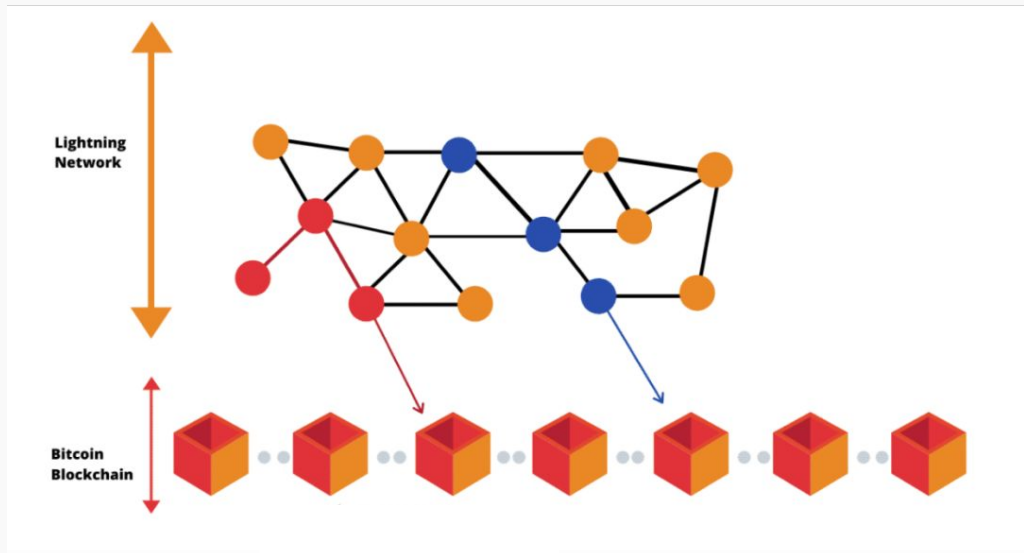


Lightning Network

Cos'è Lightning

Modo più intelligente di usare Bitcoin

- Lightning Network è una **rete di pagamento peer-to-peer** che si basa sulla blockchain di Bitcoin
- Sfrutta i **canali di pagamento** per aumentare la capacità di elaborazione delle transazioni sulla rete
- Consente di effettuare **transazioni veloci e a basso costo**
- Contribuisce alla **scalabilità** e alla **privacy** di Bitcoin



Satoshis

Su Lightning Network si spostano “**satoshi**”, chiamati anche **sats**. Sono **frazioni di bitcoin**

Conversione:

1 BTC = 10^8 SAT = 100,000,000 SAT

1 SAT = 10^{-8} BTC = 0.00000001 BTC

Satoshi	Bitcoin
1	0.00000001
10	0.00000010
100	0.00000100
1,000	0.00001000
10,000	0.00010000
100,000	0.00100000
1,000,000	0.01000000
10,000,000	0.10000000
100,000,000	1.00000000

Lightning - Use Cases

- **Consumatore**

Alice vuole effettuare pagamenti veloci, sicuri, economici e privati per piccoli acquisti al dettaglio come un caffè al bar

- **Commerciante**

Bob possiede un bar, usa Lightning per accettare pagamenti istantanei e senza commissioni

- **Attività di servizi software**

Chan vende i propri servizi di AI su internet, ogni richiesta di generare un'immagine con AI avviene tramite micropagamenti di satoshi su Lightning Network

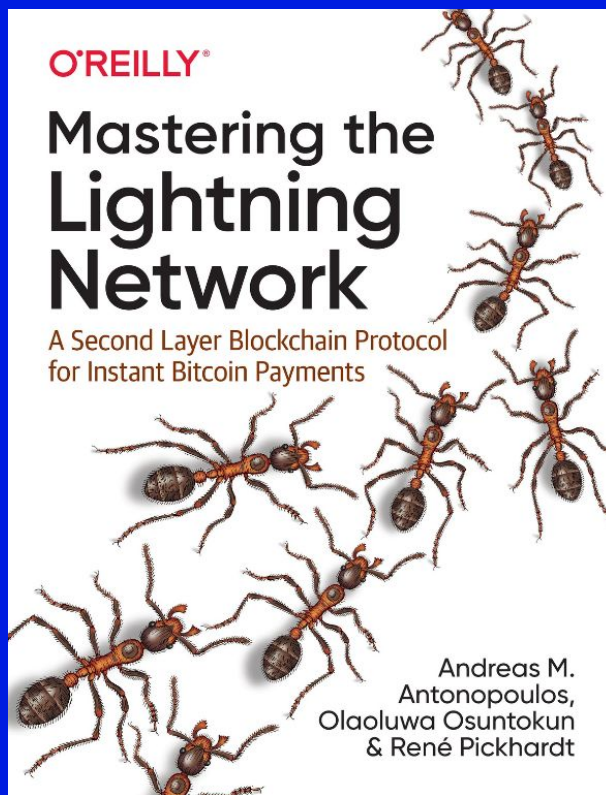
- **Gamer**

Dina gioca molto ai videogiochi, ma preferisce quelli che hanno una “economia di gioco”, mentre gioca guadagna piccole somme di sats completando missioni o vendendo oggetti virtuali

Storia

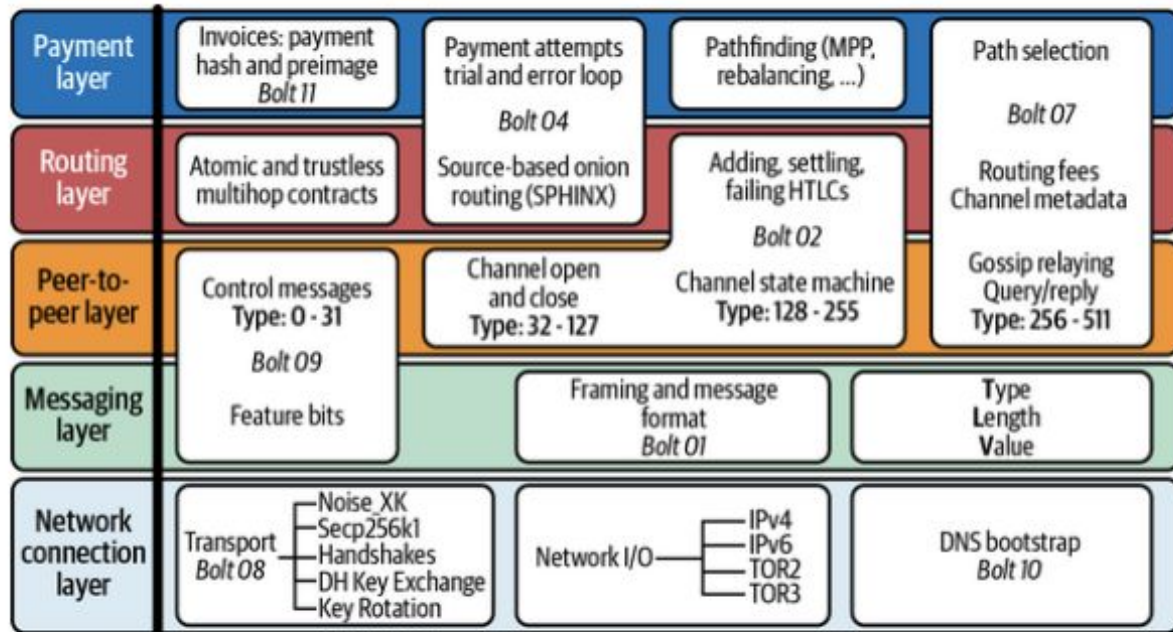


Come Funziona?



Protocollo Lightning

Composto da uno **stack** di protocolli e specifiche chiamate **BOLT** (*Basis of Lightning Technology*)



Nodo Lightning

Software che gira su un server, PC o mobile, **scambia messaggi p2p** con altri nodi della rete parlando il **protocollo Lightning**

Implementazione	Azienda	Linguaggio
LND	Lightning Labs	Go
Core Lightning	Blockstream	C
Eclair	ACINQ	Scala
LDK	Spiral	Rust



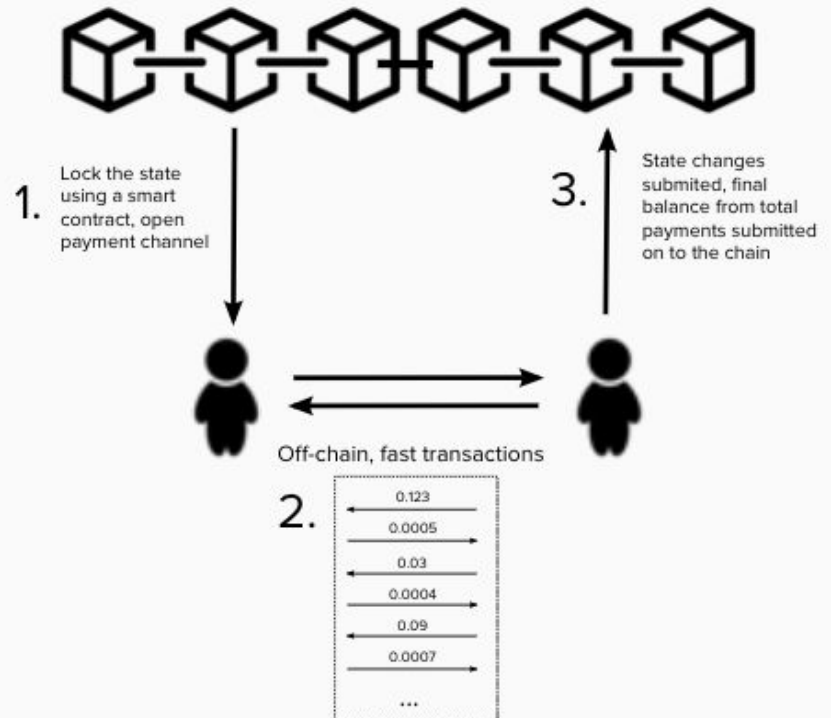
Canali di Pagamento

Un canale di pagamento (**Payment Channel**) è una relazione finanziaria tra due nodi, che attraverso uno smart-contract riescono ad effettuare **pagamenti** tra loro **senza dover registrare ogni transazione sulla blockchain di Bitcoin.**

Viene salvata **on-chain** solo la transazione di **apertura** e quella di **chiusura**

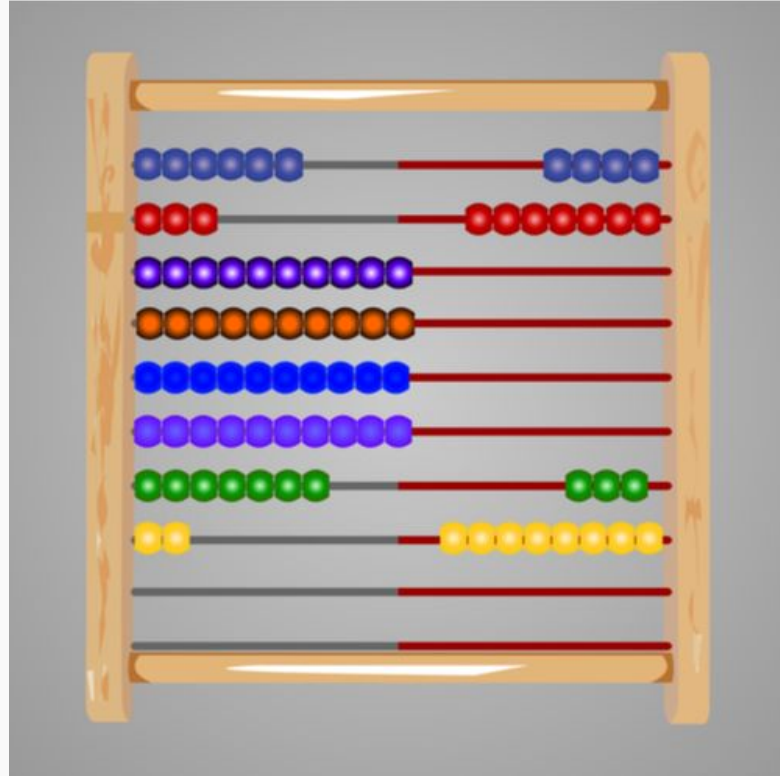
Payment Channels

nichanank.com



Canali di Pagamento

Pensate al canale di pagamento di un LN come ad un **abaco**, dove le perline rappresentano i **bitcoin** all'interno del canale.

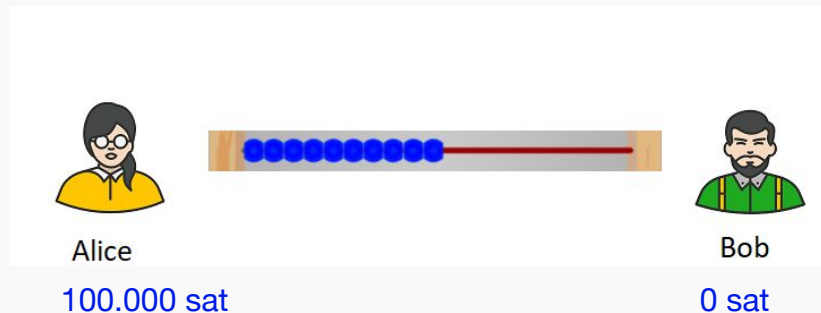


Apertura di un canale

Per creare un canale di pagamento, una delle due parti deposita una somma di bitcoin verso un indirizzo multi-sig 2-di-2 (**Funding Transaction**).

L'importo depositato nel canale rappresenta la **capacità** del canale e l'importo massimo trasferibile.

Alice prima di confermare l'apertura del canale si assicura di poter riprendersi i fondi in caso Bob sparisca

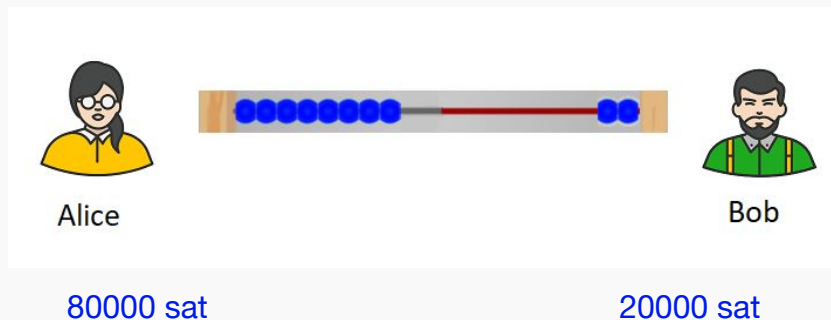


Apertura di un canale

Per creare un canale di pagamento, una delle due parti deposita una somma di bitcoin verso un indirizzo multi-sig 2-di-2 (**Funding Transaction**).

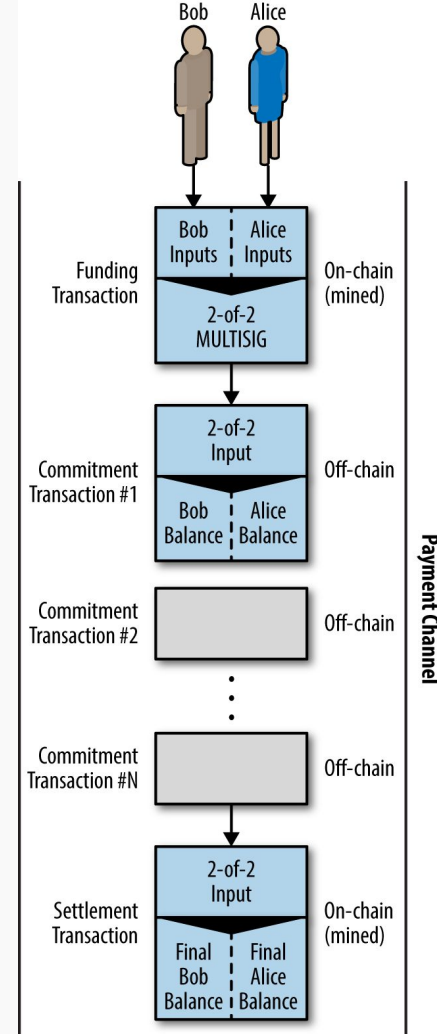
L'importo depositato nel canale rappresenta la **capacità** del canale e l'importo massimo trasferibile.

Alice prima di confermare l'apertura del canale si assicura di poter riprendersi i fondi in caso Bob sparisca



Commitment Transaction

- Transazioni che **aggiornano lo stato del canale** e non vengono propagate finchè non si vuole chiudere il canale.
- Ad ogni cambiamento di stato le due parti si scambiano una **transazione di punizione**
- Se una delle due controparti prova a fregare l'altra, propagando una transazione di uno stato precedente, **se scoperta**, perde tutti i fondi del canale



Commitment Transaction

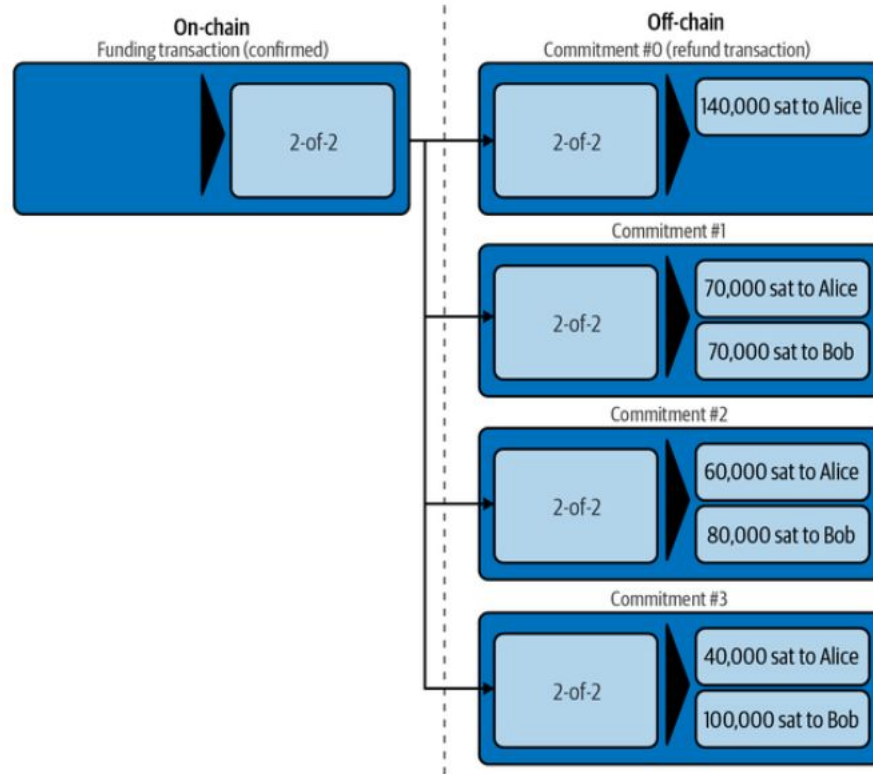


Figure 7-6. Multiple commitment transactions

Chiusura di un canale

Transazione che viene propagata sulla rete da una delle due parti e chiude il canale riportando il saldo finale sulla blockchain.

Se possibile è preferibile non chiudere i canali

- **Chiusura cooperativa**

- Le controparti si mettono d'accordo e spendono la transazione di apertura col saldo finale

- **Chiusura non cooperativa**

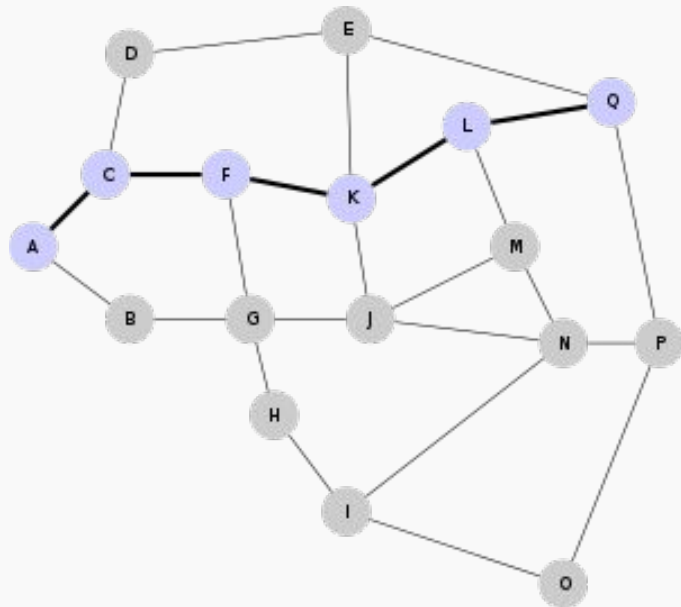
- Non c'è truffa, propagato ultimo stato valido
- Truffa che non viene scoperta in tempo
- Justice Transaction: è stata scoperto un tentativo di truffa, si ottiene tutta la capacità del canale

Routing

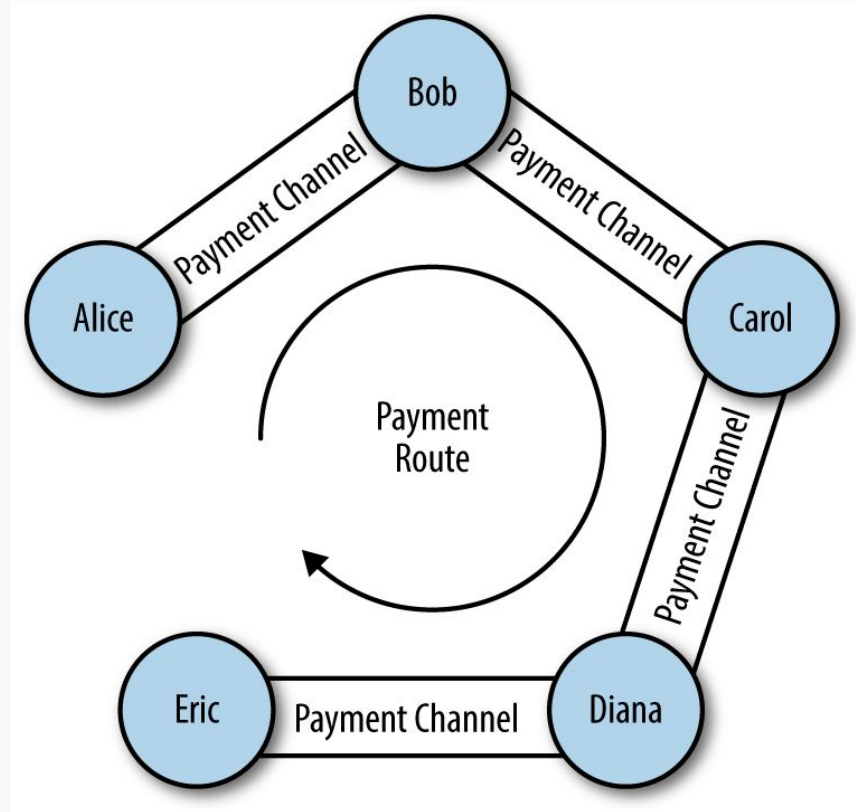
I canali di pagamento possono rimanere **privati** oppure **annunciati** a tutta la rete.

Annunciare i canali permette al nodo collegarsi ad altri nodi e **costruire un grafo della rete, salvandolo in memoria**

Due nodi non devono necessariamente avere un canale fra di loro per poter fare un pagamento, **basta che sia presente un percorso** che li colleghi e che tutti i canali intermedi abbiano abbastanza **liquidità**



Routing

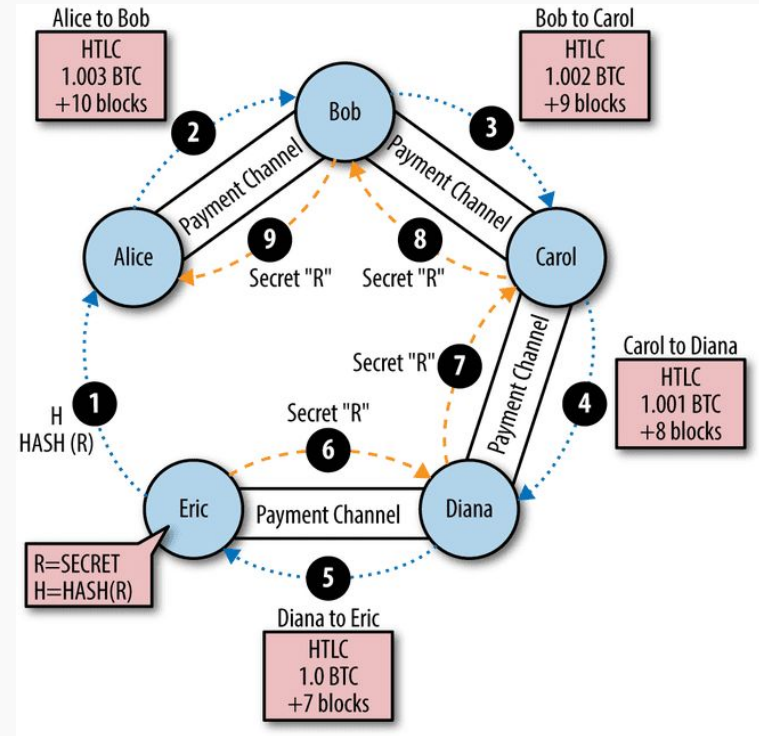


Come funziona il routing

Il routing nella rete Lightning Network si basa sul passaggio di **hash time-locked contract (HTLC)**, tra diversi nodi fino a raggiungere la destinazione che ha richiesto il pagamento.

Un HTLC è una transazione che viene eseguita solo se vengono soddisfatte determinate condizioni, come il passaggio di una chiave segreta entro un determinato periodo di tempo.

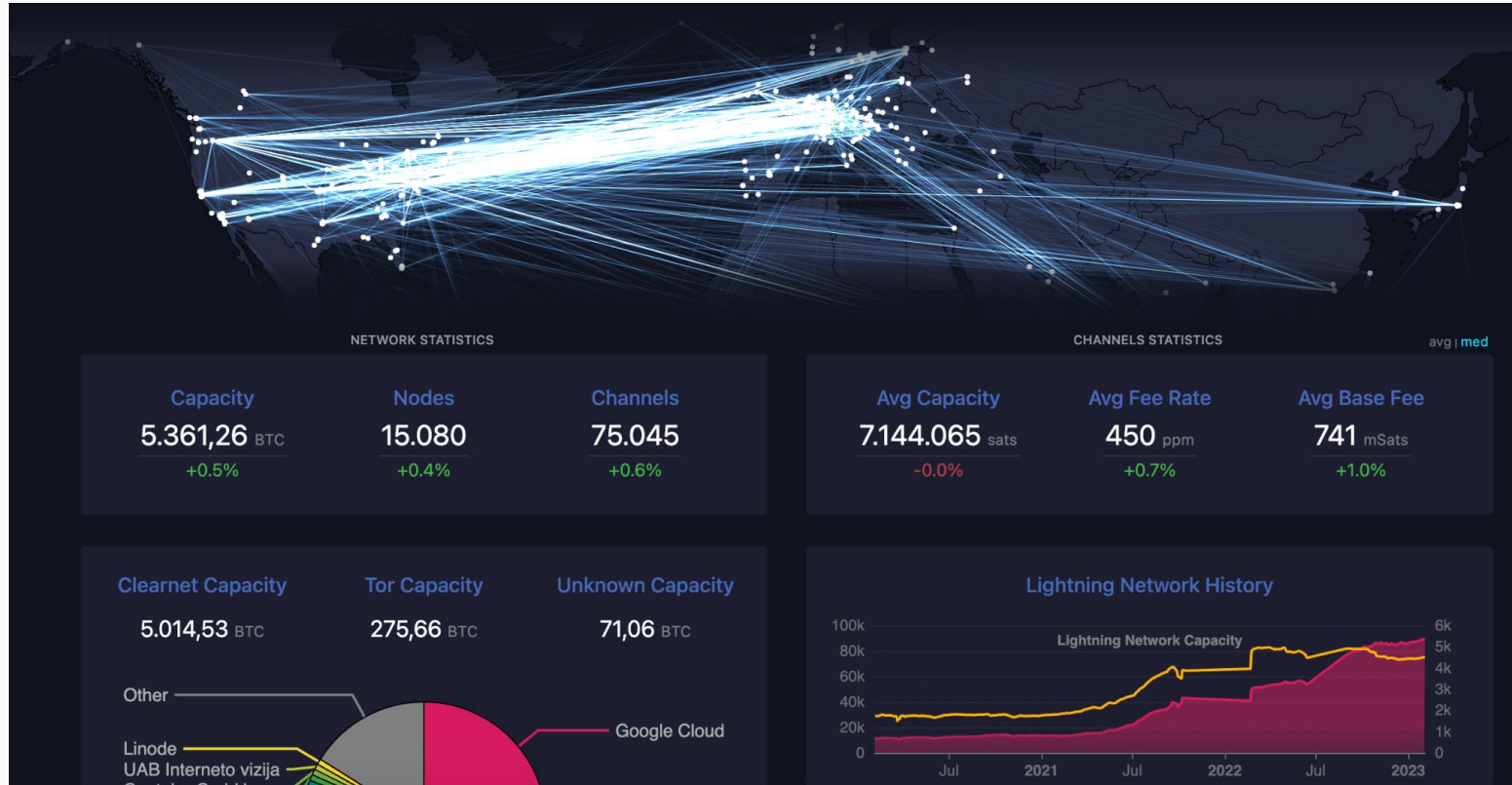
I nodi intermedi possono **guadagnare delle commissioni** per il lavoro svolto



Stato della rete Lightning

<https://mempool.space/it/lightning>

<https://amboss.space/>



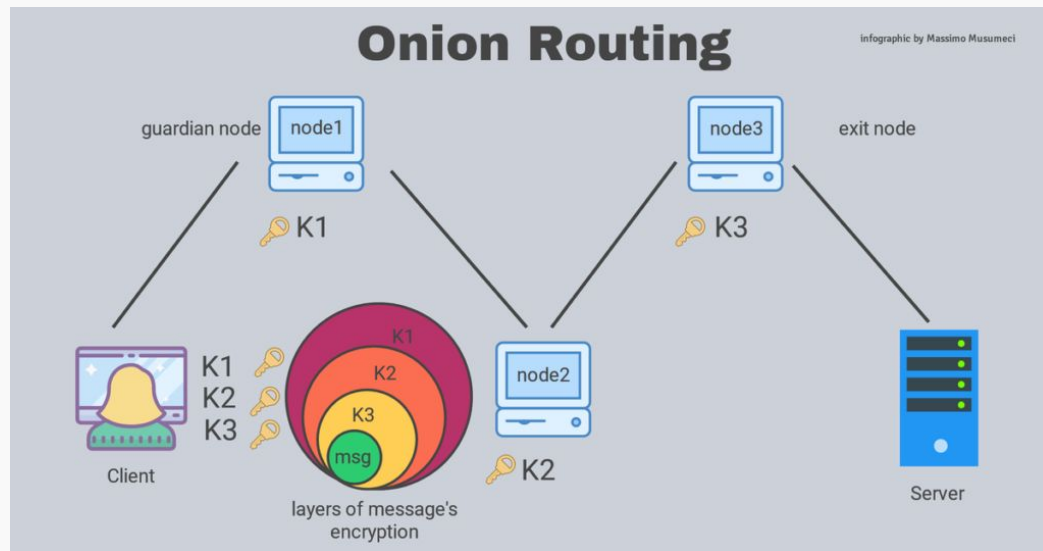
Privacy su Lightning Network

Lightning utilizza l'**onion routing** per migliorare la privacy nei pagamenti

Cifrando i dati del pagamento tramite la **crittografia asimmetrica** permette di leggere il messaggio in chiaro solo al destinatario.

Sono comunque possibili delle euristiche a livello di rete (Indirizzi IP, metadati, ...)

Lightning supporta anche la **rete TOR**, che permette ulteriormente di **migliorare la privacy degli utenti**, proteggendo dalla sorveglianza su internet



Gestire un nodo Lightning

Problemi:

- Lavoro che richiede diversa esperienza, risorse e tempo
- Gestire il backup dei canali è rischioso. File “tossico”
- Serve hardware affidabile e sempre on-line
- Sicurezza: gestendo un hot-wallet bisogna garantire una buona sicurezza per il nodo e le chiavi private
- Difficile gestione della liquidità nei canali (ribilanciamenti)
- Ad oggi complesso andare a break-even (spese > guadagni)

Alcune possibili possibili soluzioni:

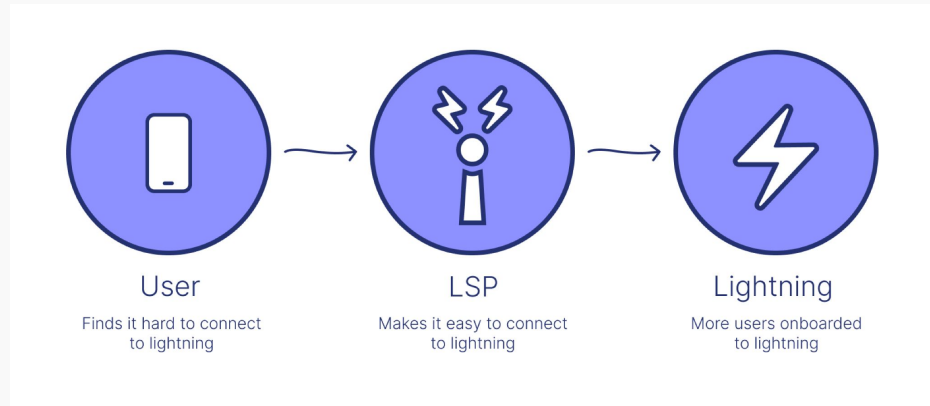
- Studiare e sperimentare
- Farsi aiutare con librerie esterne e tool automatici (facendo attenzione alla sicurezza)
- Utilizzare sistemi hardware resilienti (Dischi in RAID, UPS, VPS)

Lightning Service Providers

Hub della rete Lightning che **forniscono dei servizi semplificati** per utenti che vogliono utilizzare la rete Lightning.

Alcuni dei servizi offerti:

- **Apertura di canali a richiesta**
- **Trampoline node**
- **Watchtower**
- **Backup dei canali**
- **Swap LN-BTC / BTC-LN**



Possibile **punto di centralizzazione** della rete, ma garantiscono per alcuni servizi utili per una migliore scalabilità

Problemi di Lightning Network

Alcuni dei possibili punti critici:

- Not your node, not your coins
- Richiede di essere on-line per poter ricevere un pagamento (e fare routing)
- Se il nodo va offline per molto tempo c'è il rischio che la controparte possa fregarsi la liquidità
- Per pagamenti molto grandi conviene preferire una transazione on-chain
- Rete TOR sotto attacco significa minore privacy nei pagamenti

Use Lightning Network



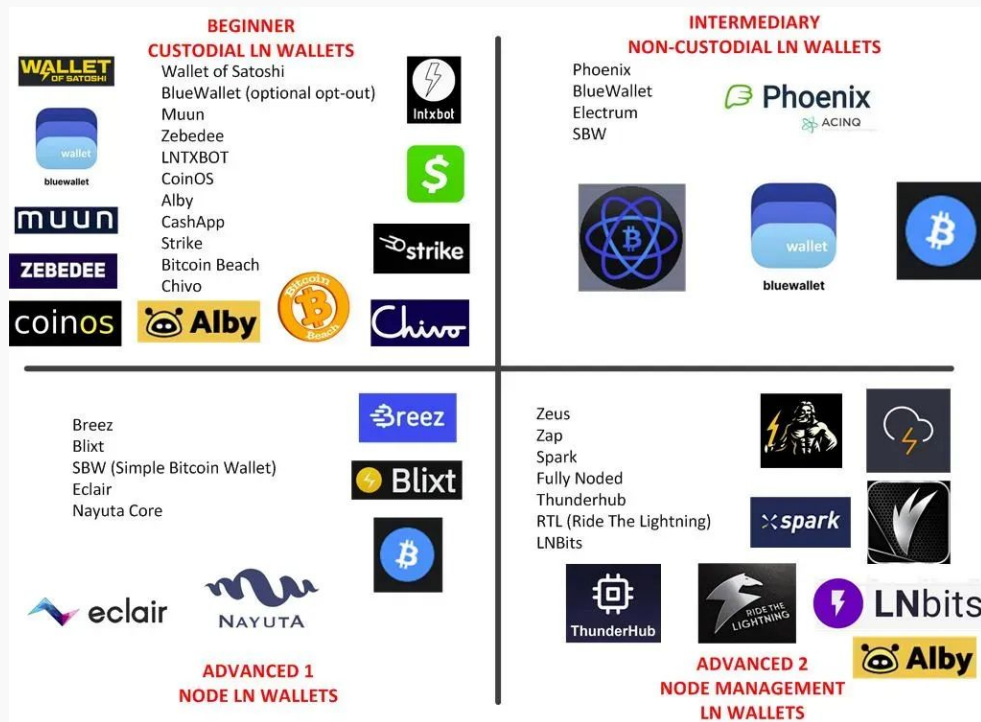
Lightning “Wallet”

App di pagamento mobile/web che permettono di ricevere e inviare bitcoin sulla rete Lightning

Si dividono in **custodial** o **non-custodial**

Non esiste il wallet migliore, ma dipende dalle **esigenze** ed **esperienza** di ognuno

È chiaramente preferibile **usare sempre wallet open-source**



Lightning Invoice

BOLT11

Stringa generata dal ricevente per poter essere pagato, utilizzabile una sola volta e con un periodo di scadenza.

Ogni invoice è firmata dal destinatario e contiene:

- Importo
- Data di scadenza
- Pubkey di destinazione,
- Funzioni supportate e altro

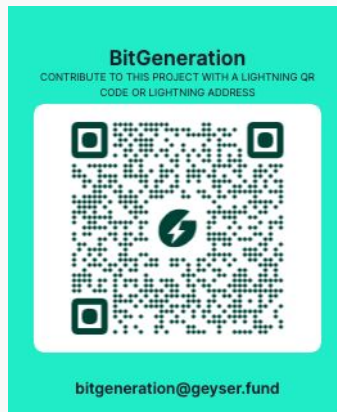
LNURL, è uno standard proposto per migliorare i pagamenti su Lightning con ad esempio metodi per prelievo automatico o link di pagamento statici. Presenta alcune criticità ma si sta sviluppando una soluzione migliore (*BOLT12*)

Invoice:



[Lightning Decoder](#)

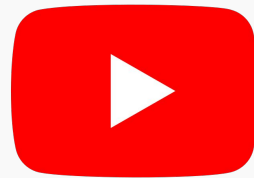
LNURL:



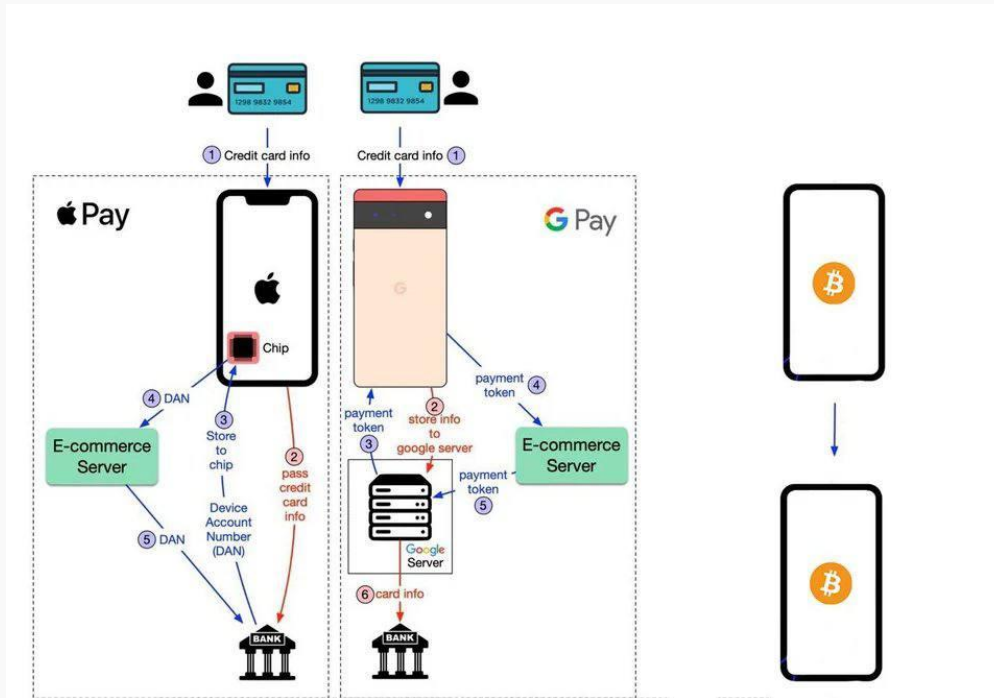
Payments Examples

Alcuni video ed esempi di pagamenti che utilizzano Lightning Network

- [Lightning Vending Machine](#)
- [A curated list of awesome Inurl things.](#)



Lightning Network vs Fiat



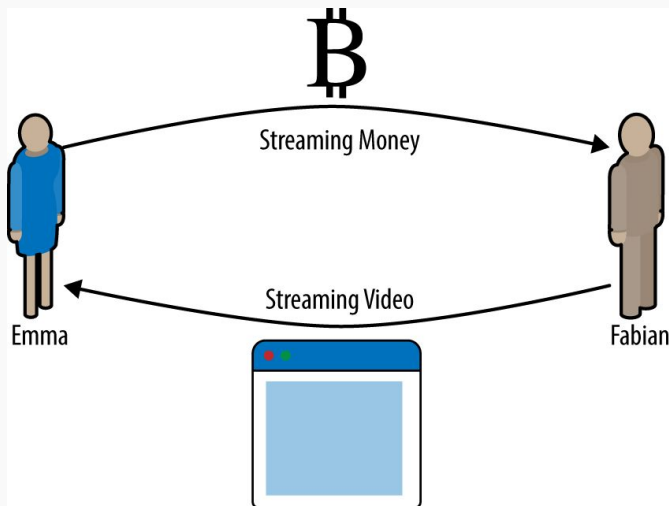
- Veloce come i sistemi di pagamento tradizionali.
- Commissioni più basse rispetto Visa/Mastercard
- Irreversibilità dei pagamenti
- Basta un cellulare e una connessione ad internet
- Moneta bitcoin:
 - Non-inflazionabile
 - Non-censurabile
 - Non-confiscabile
 - Libera e aperta a tutti

Lightning Apps

Lightning Network abilita la creazione di nuovi tipi di applicazioni grazie ai **micropagamenti**

Esempi:

- **Podcasting 2.0**
 - Breez, Fountain
- **Gaming**
 - Zebedee, THNDR
- **Video Streaming**
 - Impervious, Keet.io
- **Finance**
 - Kollider, LNM, Bolz















Risorse Bitcoin in Italiano

VENTUNO 



ventuno.space

Benvenuti su Ventuno
Una collezione delle migliori risorse Bitcoin,
in lingua italiana 
Impara di più approfondendo ogni categoria

 Cos'è Bitcoin?	 Risorse Didattiche	 Conferenze	 Podcasts
 Meetups	 Video	 Libri	 Wallets
 Privacy	 Hardware	 Ottieni Bitcoin	 Spendi Bitcoin
 Fatti pagare in Bitcoin	 Explorers & Dashboards	 Sviluppo	 Requisiti Listing

Bibliografia

- [Mastering Lightning Network](#)
- [LNP/BP-una-dolce-introduzione - G. Zucco](#)
- [The History of Lightning: From Brainstorm to Beta - Bitcoin Magazine](#)
- [Wallet Lightning a confronto](#)

Grazie!

Now get your sats!

walterm21@proton.me
bitpolito.it

**Paga 21 sat a
questo LNURL
per ricevere un
link da BitPolito**

