Verbale interno del 2021-11-16
Progetto ShopChain

yakuzaishi.swe@gmail.com

Informazioni sul documento

Responsabile	Michele Filosofo
Redattori	Matteo Midena
Verificatori	Luca Busacca
Uso	Interno
Destinatari	Gruppo *Yakuzaishi*

Sommario
Vengono riportati gli argomenti discussi durante la riunione.

Contenuti

1 Generale

1.1 Informazioni sulla riunione

- **Luogo**: Chiamata Discord;
- **Ora di inizio**: 15:00;
- **Ora di fine**: 17:30;
- **Partecipanti**: Francesco Bugno, Luca Busacca, Luca Carturan, Michele Filosofo, Dario Furlan, Francesco Mattarello, Matteo Midena

1.2 Ordine del giorno

Durante la riunione sono stati trattati i seguenti temi:

- Discussione generale su cosa sia una blockchain e le caratteristiche che dovranno essere sfruttate per questo progetto;
- Discussione riguardo quale blockchain adottare;
- Esposizione dei pregi e difetti di alcune blockchain selezionate.

2 Resoconto

Poiché per il seguente capitolato è stata offerta una libera scelta riguardo le migliori tecnologie da adottare allo scopo, il gruppo ha svolto un'attenta valutazione riguardo le tecnologie compatibili con gli obiettivi da raggiungere. Qui sotto sono riassunte le principali caratteristiche e motivazioni che hanno decretato la scelta della blockchain di riferimento.

2.1 Cos'è una blockchain

La blockchain, che letteralmente significa "catena di blocchi", è una struttura dati condivisa e immutabile. Viene definita come un registro digitale le cui voci sono raggruppate a blocchi che vengono concatenati in ordine cronologico e la cui integrità è garantita dall'uso della crittografia. Si basa sul concetto in cui quando il contenuto viene scritto (tramite un processo normato), questo non è più modificabile né eliminabile, pena l'invalidazione dell'intero processo.
Le caratteristiche di base che caratterizzano questa tecnologia sono: digitalizzazione dei dati, decentralizzazione, disintermediazione (non è richiesto nessun intermediario), tracciabilità dei trasferimenti, trasparenza/verificabilità, immutabilità e programmabilità dei trasferimenti. Grazie a tali proprietà, la blockchain si pone come una valida alternativa in termini di sicurezza, affidabilità, trasparenza e costi alle banche dati e ai documenti gestiti in maniera centralizzata da autorità riconosciute e regolamentate (come banche, pubbliche amministrazioni, ecc...).

2.2 Ethereum

Ethereum è la blockchain più famosa dopo quella di Bitcoin$_{\mathbf{G}}$, prende la struttura di base di quest'ultima e aggiunge delle importanti modifiche. In particolare si basa su una blockchain programmabile, aprendo alla possibilità di effettuare altre attività oltre a semplici transazioni di criptovalute, parliamo in particolare di servizi finanziari, giochi o app decentralizzate, quindi completamente libere e non soggiogate ad un ente centrale.

2.3 Differenze tra blockchain di layer 0, 1, 2

Le blockchain possono essere catalogate in base al tipo di architettura su cui si basano, in particolare possono basarsi su un layer 0, 1 o 2. In base a questo tipo di differenziazione offrono un diverso grado di astrazione e funzionalità. Come funziona per un linguaggio di programmazione più si sale di livello, più ci si allontana dallo "strato fisico".

2.3.1 Layer 0

È il livello più basso, quindi permette di agire direttamente sull'architettura blockchain senza dover mettere mano a layer e protocolli sottostanti. Qui inoltre è possibile creare dApps (decentralized applications), validare schemi di dati, "coniare" criptovalute e altro ancora. Tutto l'hardware, server, nodi e qualsiasi device collegato ai nodi fa parte di questo livello. A questo livello agiscono direttamente gli algoritmi di consenso, come proof-of-work$_{\mathbf{G}}$, proof-of-stake$_{\mathbf{G}}$, proof-of-activity$_{\mathbf{G}}$.

2.3.2 Layer 1

Si occupano principalmente di rendere il protocollo base più scalabile. Si basa su due principali soluzioni:

- **Consensus protocol changes**: primo grande ed evidente cambiamento c'è stato con Ethereum$_{\mathbf{G}}$, in cui i progetti hanno iniziato a spostarsi dal protocollo di consenso Proof-of-Work$_{\mathbf{G}}$ (PoW) a un più veloce e leggero Proof-of-Stake$_{\mathbf{G}}$ (PoS). Questo è avvenuto su Ethereum$_{\mathbf{G}}$ con il protocollo Casper.

- **Sharding**: si basa sul semplice concetto di dividere le transazioni in "frammenti" più piccoli (traduzione di "shards") che possono essere processati parallelamente dalla rete.

2.3.2.1 Pro

- Aggiunge semplicemente un livello superiore all'architettura, senza apportare grandi modifiche;
- Permette di eseguire un maggior numero di operazioni a parità di tempo;

2.3.2.2 Contro

- Inefficient Consensus Protocol: alcuni protocolli di consenso non sono adatti a gestire grandi traffici di transazioni e conseguentemente queste strutture non sono pronte per un'adozione di massa;
- Excessive Workload: il carico di lavoro su questo livello può diventare eccessivo e provocare grandi rallentamenti (è il caso di Ethereum$_{\mathbf{G}}$ e le sue conseguenti elevate gas fee).

2.3.3 Layer 2

Sono lo strato più alto di astrazione, e attualmente si pongono l'obiettivo di risolvere gran parte dei problemi presenti nel layer 1. Le soluzioni si dividono in due possibilità:

- **State Channels**: ha come componente principale un canale full-duplex tra due partecipanti che permette di svolgere compiti che tipicamente sono on-blockchain, off-blockchain. Questo permette di diminuire di molto i tempi di attesa visto che viene a mancare la componente terza, come ad esempio le azioni dei miner$_{\mathbf{G}}$. Come funziona:
  - Una porzione di blockchain viene ritagliata (viene usata l'espressione "sealed off", ovvero sigillata) tramite multi-firma o sotto smart-contract, previo accordo tra le due parti coinvolte;
  - Quest'ultime possono interagire tra loro senza preoccuparsi di firmare le transazioni eseguite
  - Quando la sequenza di scambi termina, solo lo stato finale viene registrato sulla blockchain.

  La metafora più facile è quella del conto, è come se una persona aprisse un conto con un'altra e poi lo saldasse "a rate". Invece di registrare ogni singola rata sulla rete, viene registrato solo il saldo finale.
  Le soluzioni di questo tipo più comuni sono: la "Bitcoin's Lightning Network" (permette di eseguire tante microtransazioni) e la "Ethereum's Raiden Network" (permette anche di eseguire smart contracts sui canali). Entrambe utilizzano Hashed Timelock Contracts (HTLCs) per creare state channels.

- **Nested blockchains**: si basa sulla soluzione chiamata Plasma, caratterizzata come segue:
  - La rete principale (la main chain) stabilisce le regole di base dell'intero sistema e non prende parte alle operazioni svolte a meno che non debba risolvere dispute;
  - Si basa su un sistema in cui la main chain si appoggia a multipli livelli di blockchain. Questi livelli sono connessi tra loro in modo da formare una struttura del tipo padre-figlio. La chain padre delega il lavoro alla chain figlia in grado di eseguire quella particolare istruzione e che ritornerà poi il risultato indietro al padre.
  - Questo sistema permette di ridurre il carico della root chain e se costruita e usata correttamente è in grado di aumentare la scalabilità della rete esponenzialmente

2.3.3.1 Pro

- Evita problemi di disordine per quanto riguarda i compiti di una blockchain;
- Evita di pagare inutili fees su microoperazioni e microtransazioni;

2.3.3.2 Contro

- Minore controllo sui dati a basso livello;

2.4 Blockchain alternative

Visto i grandi difetti presenti nella blockchain Ethereum$_{\mathbf{G}}$, in particolare il costo elevato delle sue gas fee e la lenta scalabilità, abbiamo preso in esame diverse blockchain e valutato la migliore rispettivamente alle esigenze del problema. Qui sotto sono riportati i pro e contro di ogni soluzione.

2.4.1 Avalanche

Avalanche è un protocollo che integra diverse blockchain, che permette la creazione di asset tokenizzati, l'esecuzione di smart contract e anche la creazione di blockchain private. AVAX è il suo token di riferimento, utilizzato per governance, pagamento commissioni e gas per i contratti.

2.4.1.1 Pro

- Blockchain ad alta scalabilità;
- Fee molto basse.

2.4.1.2 Contro

- Tecnologia meno diffusa e conosciuta rispetto ad Ethereum$_{\mathbf{G}}$;
- Progetto discretamente complesso.

2.4.2 Solana

Solana è una blockchain open source$_{\mathbf{G}}$, permissionless che garantisce un alto numero di transazioni al secondo. Queste caratteristiche sono ottenute principalmente dall'architettura a basso livello che utilizza ad esempio un proprio algoritmo di consenso "Proof-of-History" e una cifratura SHA-256.

2.4.2.1 Pro

- Transazioni molto veloci ($>$ 710000 tps);
- Fee molto basse;
- Progetto recente ma dal grande potenziale.

2.4.2.2 Contro

- Non è una blockchain Ethereum$_{\mathbf{G}}$ compatibile, quindi gli smart contract non possono essere scritti in Solidity$_{\mathbf{G}}$. Il linguaggio necessario assomiglia al C$_{\mathbf{G}}$ e a RUST$_{\mathbf{G}}$ e risulta essere più complesso per chi si trova alla prima esperienza con queste tecnologie;
- I suoi punti di forza la rendono anche una scelta azzardata per un primo approccio al mondo delle blockchain.

2.4.3 Moonriver

Questo tipo di soluzione non presenta particolari caratteristiche se non quella di implementare gli smart contract su una parachain. Una parachain è una blockchain "specializzata" che si connette ad esempio a Polkadot (una blockchain layer 0, un altro esempio è Kusama), quest'ultimo è un protocollo di rete che permette di trasferire non solo tokens ma anche dati tra più blockchain. Quindi Polkadot è a tutti gli effetti un'applicazione che permette di avere registri e computazioni cross-chain. Le operazioni tra queste vengono eseguite in parallelo e non sequenzialmente, tramite l'uso di relay chain (blockchain minori a cui viene delegata parte della computazione richiesta) che fungono in certi casi da main chain.

2.4.3.1 Pro

- Sviluppo su una parachain, tecnologia molto interessante e valida per impieghi futuri;
- Fee più basse rispetto ad Ethereum$_{\mathbf{G}}$;
- Ambiente Ethereum$_{\mathbf{G}}$ compatibile, quindi possibilità di usare Solidity$_{\mathbf{G}}$ per lo sviluppo di Smart Contract;
- Progetto recente ma dal grande potenziale.

2.4.3.2 Contro

- Blockchain molto meno conosciuta, quindi probabilmente si avrà una community di supporto più piccola.

2.4.4 Polygon

È un protocollo e un framework per costruire e connettere blockchain Ethereum$_{\mathbf{G}}$ compatibili. Nasce per risolvere i principali problemi legati alla rete Ethereum$_{\mathbf{G}}$, ovvero basso throughput, gas fee elevate, mancanza di sovranità. Nonostante siano nate delle blockchain ethereum compatibili che sfruttano le qualità migliori dell'ecosistema ethereum$_{\mathbf{G}}$, non c'è ancora un protocollo o framework che si occupi di metterle in comunicazione ad alto livello. Proprio a questo scopo nasce Polygon.

2.4.4.1 Pro

- Possibilità di sviluppare su una tecnologia ad alto livello;
- Ambiente Ethereum$_{\mathbf{G}}$ compatibile, quindi possibilità di usare Solidity$_{\mathbf{G}}$ per lo sviluppo di Smart Contract.

2.4.4.2 Contro

- Blockchain molto meno conosciuta, quindi probabilmente si avrà una community di supporto più piccola;
- Polygon si appoggia comunque a Ethereum$_{\mathbf{G}}$, questo significa che ottimizza le transazioni solo tra le chain connesse, in caso contrario si hanno comunque fee alte.

2.4.5 Scelta: Fantom

è una piattaforma che ha i suoi punti di forza su high-performance, scalabilità e sicurezza negli smart contract. Nasce per risolvere le limitazioni delle precedenti piattaforme blockchain (come molte altre sue coetanee). Fantom è permissionless$_{\mathbf{G}}$, decentralized e open-source$_{\mathbf{G}}$.

Usa un suo algoritmo di consenso BFT(Byzantine Fault-tolerant) "Lachesis" che permette di avere delle eccellenti prestazioni per quanto riguarda velocità, costo e sicurezza. E' un tipo di algoritmo di consenso che permette di creare delle applicazioni peer-to-peer senza che lo sviluppatore debba creare il proprio network layer. Lachesis è:

- **Asincrono**: i partecipanti hanno la libertà di processare comandi in momenti diversi;
- **Leaderless**: nessun partecipante nella rete ha un ruolo speciale;
- **Byzantine fault-tolerant**: supporta fino a un terzo dei nodi difettosi inclusi comportamenti dannosi;
- **Finale**: l'output dell'algoritmo di consenso può essere usato immediatamente, non c'è bisogno di aspettare per la conferma di blocchi (tempo medio di 1-2 secondi).

2.4.5.1 Pro

- Transazioni molto veloci;
- Fee molto basse;
- Blockchain ethereum$_G$ compatibile, quindi possibilità di usare Solidity$_G$ come linguaggio per gli smart contract;
- Modulare, ovvero permette di convertire dApps sviluppate sulla rete Ethereum$_G$, di essere adattate in poco tempo.

2.4.5.2 Contro

- Meno conosciuta rispetto ad Ethereum$_G$.

2.5 Conclusioni

Concludendo quanto riportato da quest'analisi, le blockchain che hanno suscitato maggior interesse nel gruppo, relativamente agli aspetti positivi e negativi contestualizzati per lo scopo di questo progetto, sono Avalanche e Fantom. Entrambe sono due blockchain giovani, la prima è nata nel 2020, mentre la seconda nel 2018. La prima risulta essere più decentralizzata per via della presenza di un numero molto maggiore di nodi validatori, mentre la seconda offre costi per transazione molto minori.
Ambedue si presentano quindi come valide soluzioni, poiché differiscono di piccoli aspetti legati alle scelte e agli obiettivi ai quali gli sviluppatori delle due reti vogliono aspirare. Alla fine dopo diversi incontri con il proponente, si è giunti alla scelta di sviluppare l'applicazione su Fantom.

2.6 Tracciamento delle decisioni

ID	Decisione
VI-2021-11-16-1	È stata scelta come blockchain di riferimento Fantom, la quale verrà presentata al proponente per essere approvata