PIEMONTE
FONDI STRUTTURALI E DI INVESTIMENTO EUROPEI 2014/2020
fondo sociale europeo
ITS TECNOLOGIE DELL'INFORMAZIONE E DELLA COMUNICAZIONE

Introduzione a Ethereum

Corso Fintech Software Developer

Modulo **Tecnologie Blockchain**

Docente: Dott. Enrico Zimuel

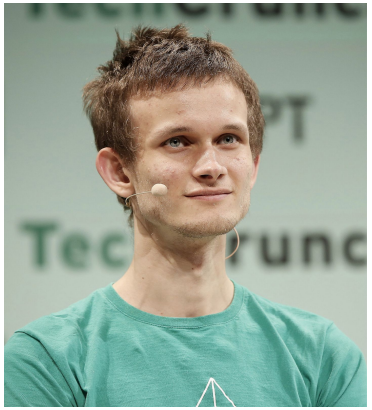
in collaborazione con:
MINISTERO DELL'ISTRUZIONE, DELL'UNIVERSITÀ E DELLA RICERCA
UNIONE EUROPEA
REGIONE PIEMONTE
per una crescita intelligente, sostenibile ed inclusiva
www.regione.piemonte.it/europa2020
INIZIATIVA CO-FINANZIATA CON FSE

Sommario

- Ethereum
- Proof of Stake
- Ether
- dapp
- Account
- Transazioni
- Blocchi
- Ethereum Virtual Machine
- Gas
- Nodi e network

Ethereum

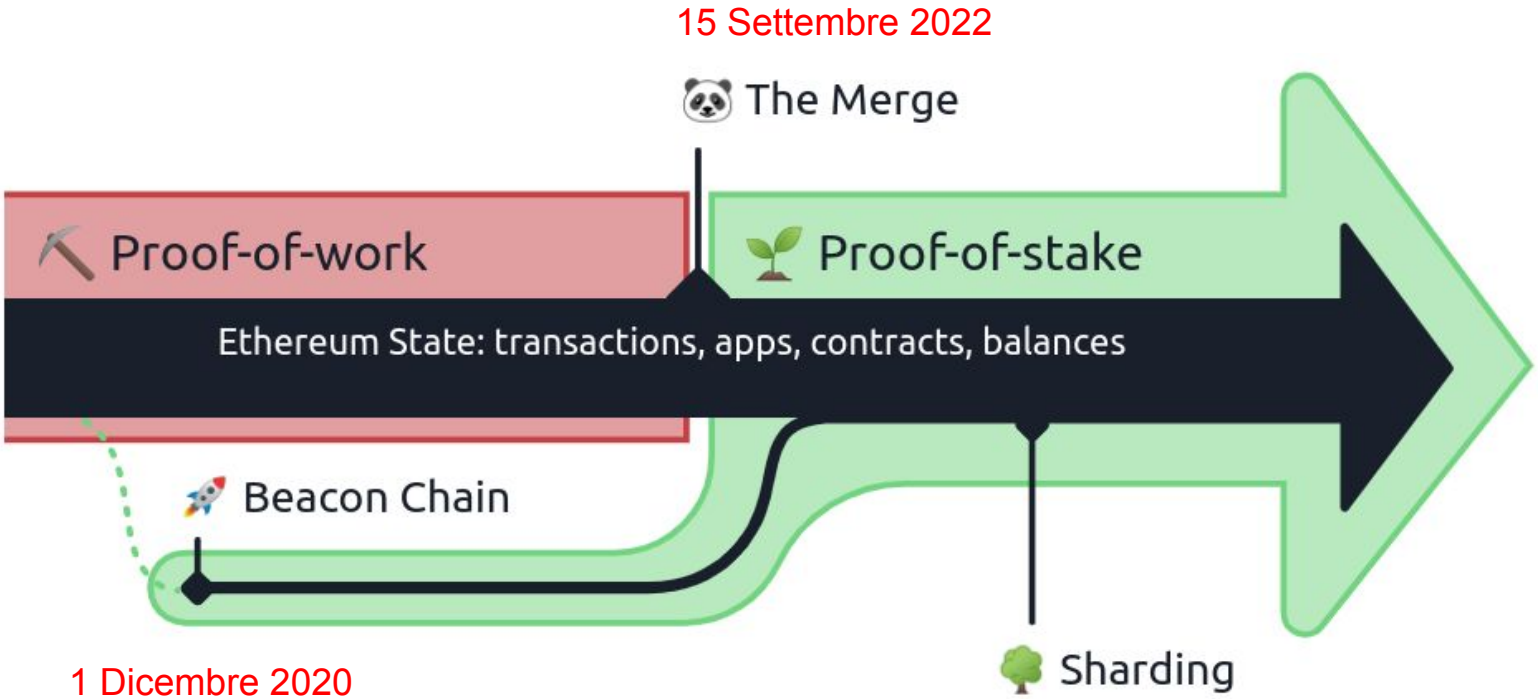
- Ethereum è una blockchain pubblica con la possibilità di eseguire applicazioni che girano all'interno della blockchain
- **Ethereum Virtual Machine** (EVM) è il "computer" che esegue queste applicazioni
- Le richieste di esecuzione di un'applicazione sono chiamate transazioni
- Ogni transazione modifica lo stato della EVM che viene memorizzato nel registro distribuito (blockchain)
- Ethereum è stata creata da Vitalik Buterin nel 2013



Proof of Stake

- **Proof of Stake** (PoS) è il nuovo algoritmo di consenso utilizzato da Ethereum a partire dal 2022
- Ha sostituito il **Proof of Work** con un soft fork denominato The Merge
- Il Proof of Stake ha ridotto il consumo energetico necessario per far girare la rete Ethereum del 99,95%
- L'idea del Proof of Stake è riservare temporaneamente un deposito di almeno **32 ETH** (Ether, la moneta di Ethereum) ed essere inseriti in una coda di attivazione
- Quando un nodo validatore è attivato in questa coda esegue la validazione del blocco e invia un voto (attestato) in favore del blocco

The Merge


15 Settembre 2022
The Merge
Proof-of-work
Proof-of-stake
Ethereum State: transactions, apps, contracts, balances
Beacon Chain
1 Dicembre 2020
Sharding

Slot ed epoche

- Con il PoW il tempo di risoluzione dettava la frequenza di mining (ogni 10 minuti)
- Con il PoS il tempo è fisso e diviso in **slot** (ogni 12 secondi) ed **epoche** (contenenti 32 slot)
- Un nodo validatore **è scelto a caso** tra quelli in attesa per ogni slot. Validatori con depositi maggiori di riserva avranno più possibilità di essere eletti. Il validatore prescelto è responsabile per la creazione di nuovo blocco e per la trasmissione verso gli altri nodi
- In ogni slot è anche presente una **commissione di validatori** preposta a votare la validità del blocco appena creato

Ricompensa al validatore

- Quando un validatore viene selezionato ed esegue il suo compito viene ricompensato (con degli ETH coniati) + eventuali mance presenti nel blocco
- Se il comitato di verifica del blocco rileva delle manomissioni sulle transazioni presenti in un blocco il nodo validatore viene penalizzato, prelevando una percentuale di ETH dal suo deposito. Inoltre il validatore sarà escluso, per un periodo di tempo, dalla possibilità di far parte nuovamente del processo di validazione
- Ogni validatore è quindi incentivato economicamente a comportarsi correttamente

Ether

- **Ether** (ETH, simbolo Ξ) è la criptovaluta utilizzata per diversi scopi in Ethereum
- E' l'unica forma di pagamento per la tassa (fee) delle transazioni
- E' la moneta utilizzata nel mercato **DeFi** (Decentralized finance)
- E' lo strumento utilizzato come **gas fee** per l'esecuzione di transazioni
- Gli ETH vengono coniati per ogni blocco proposto e per ogni epoca verificata dal comitato di validatori
- Circa ⅛ degli ETH coniati va al validatore scelto a caso e gli altri sono equamente distribuiti tra tutti i partecipanti al processo di verifica

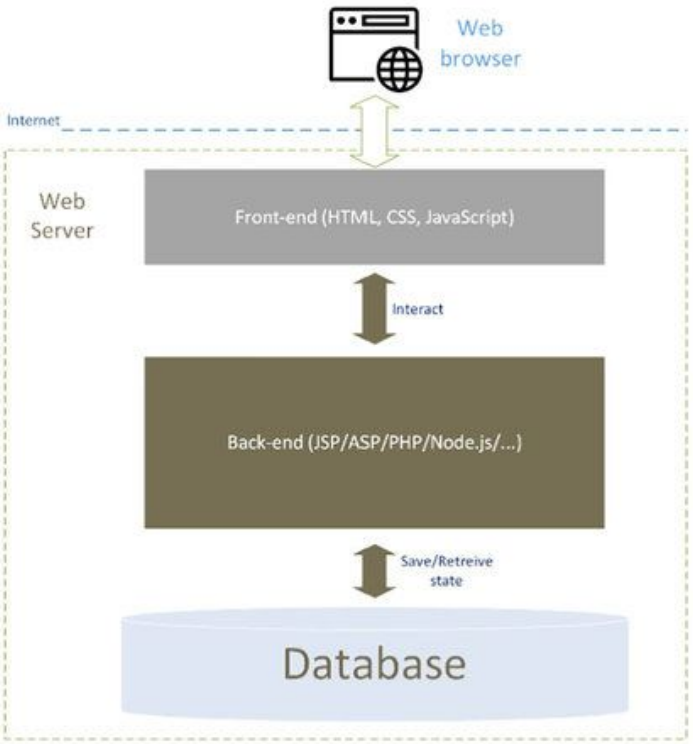
Burning ETH

- Ethereum oltre a coniare (**minting**) prevede anche di bruciare (**burning**) ETH
- Bruciare ether equivale a rimuoverlo permanentemente dalla circolazione
- L'operazione di **burning** avviene in ogni transazione Ethereum
- Quando un utente paga il **gas fee** per una transazione, questo viene bruciato
- Questo per evitare che i gas fee vengano utilizzati in qualche modo dai validatori
- Quando il traffico delle transazioni è elevato, i blocchi possono bruciare più ETH di quelli che verranno coniati
- Questo meccanismo tende a compensare il numero di ETH coniati, evitando possibili svalutazioni

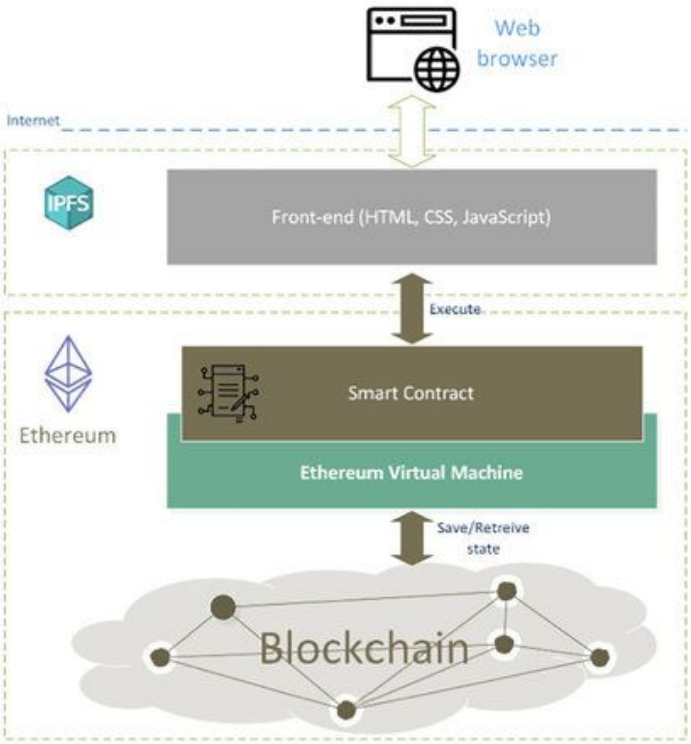
dapp

- Un’applicazione decentralizata (**dapp**) è un’applicazione creata su una rete distribuita che utilizza uno o più smart contract e un frontend, con un interfaccia utente
- Una dapp ha le seguenti caratteristiche:
  - **Decentralizzata**, vengono eseguite nella rete Ethereum
  - **Deterministica**, il risultato dell’esecuzione di una dapp deve essere lo stesso a prescindere dall’ambiente nel quale viene eseguito
  - **Turing completa**, le dapp possono eseguire qualsiasi azioni date le adeguate risorse
  - **Isolata**, l’esecuzione di una dapp avviene in un ambiente virtuale denomiato Ethereum Virtual Machine

Traditional Web Application



Decentralized Application (DApp)



Smart contract

- Gli **smart contract** rappresentano la parte backend di una dapp
- Uno smart contract è un programma che viene eseguito dalla blockchain Ethereum
- Una volta che uno smart contract è inserito nella blockchain non può essere modificato
- Pro:
  - **Zero downtime**, gli smart contract vengono eseguiti su una rete distribuita per cui saranno sempre disponibili in qualche nodo
  - **Privacy**, gli smart contract possono essere eseguiti in maniera pseudo-anonima (basta un l'indirizzo Ethereum)
  - **Resistenza alla censura**, uno smart contract non può essere censurato, bloccato da uno o più partecipanti alla rete

Smart contract (2)

- Pro:
  - **Data integrity**, gli smart contract non possono essere alterati grazie alle proprietà crittografiche della blockchain
  - **Trustless computation/verifiable behavior,** gli smart contract possono essere analizzati e la loro esecuzione è totalmente prevedibile, non c'è bisogno di fidarsi di un'autorità centrale
- Contro:
  - **Manutenzione**, difficile da eseguire poiché il codice non può essere modificato in una blockchain, è possibile solo rilasciare versioni più aggiornate
  - **Performance**, gli smart contract non possono essere troppo performanti perché la loro esecuzione è soggetta a tutti i meccanismi di verifica di una blockchain
  - **Congestione della rete**, se una dapp utilizza troppo risorsa computazionale può influenzare il traffico dell'intera rete

Smart contract (3)

- Contro:
  - **Esperienza utente**, è difficile progettare esperienze utenti user-friendly perchè all'utente finale può risultare difficile mettere in piedi tutta l'infrastruttura per interagire con la blockchain
  - **Centralizzazione**, le interfacce utenti delle dapp possono girare su server centralizzati e richiamare gli smart contract sulla blockchain. In questo modo una dapp non è più decentralizzata, perdendo di fatto tutti i vantaggi della blockchain

Account

- In Ethereum ci sono due tipologie di Account:
  - **Externally-owned account** (EOA), gestiti da qualcuno attraverso chiavi private;
  - **Contract account**, uno smart contract pubblicato nella rete Ethereum.
- Tutti gli account possono:
  - ricevere, tenere, e inviare ETH e token;
  - interagire con smart contract pubblicati nella rete Ethereum.

Account: EOA

- Nessun costo per la creazione di un account
- Può avviare transazioni
- Le transazioni possono essere di ETH o token
- Costruito tramite una coppia di chiavi: pubblica e privata

Account: smart contract

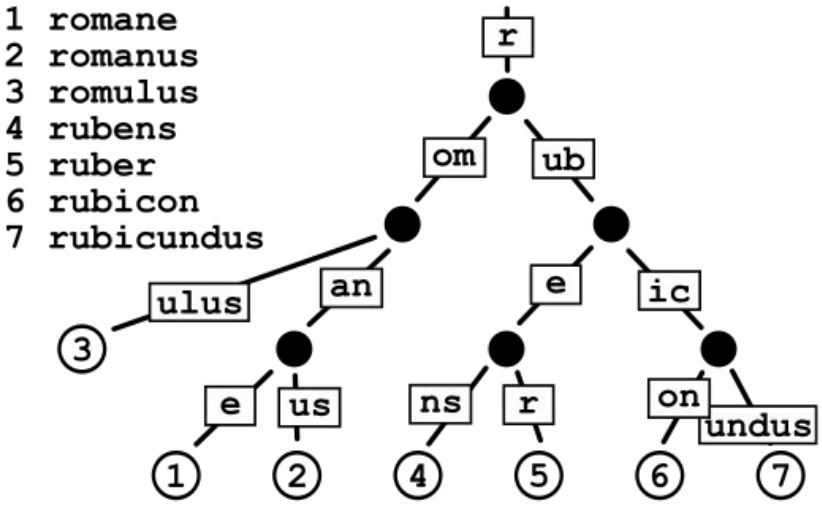
- La creazione di uno smart contrat ha un costo perché si utilizzano risorse della rete per la memorizzazione
- Possono eseguire transazioni solo in risposta a un’altra transazione
- Le transazioni da un account external possono attivare (trigger) del codice di uno smart contract che può eseguire diverse azioni, come trasferire dei token o creare un altro smart contract
- Gli account degli smart contract non hanno una chiave privata. Sono controllati dalla logica del codice

Formato degli account

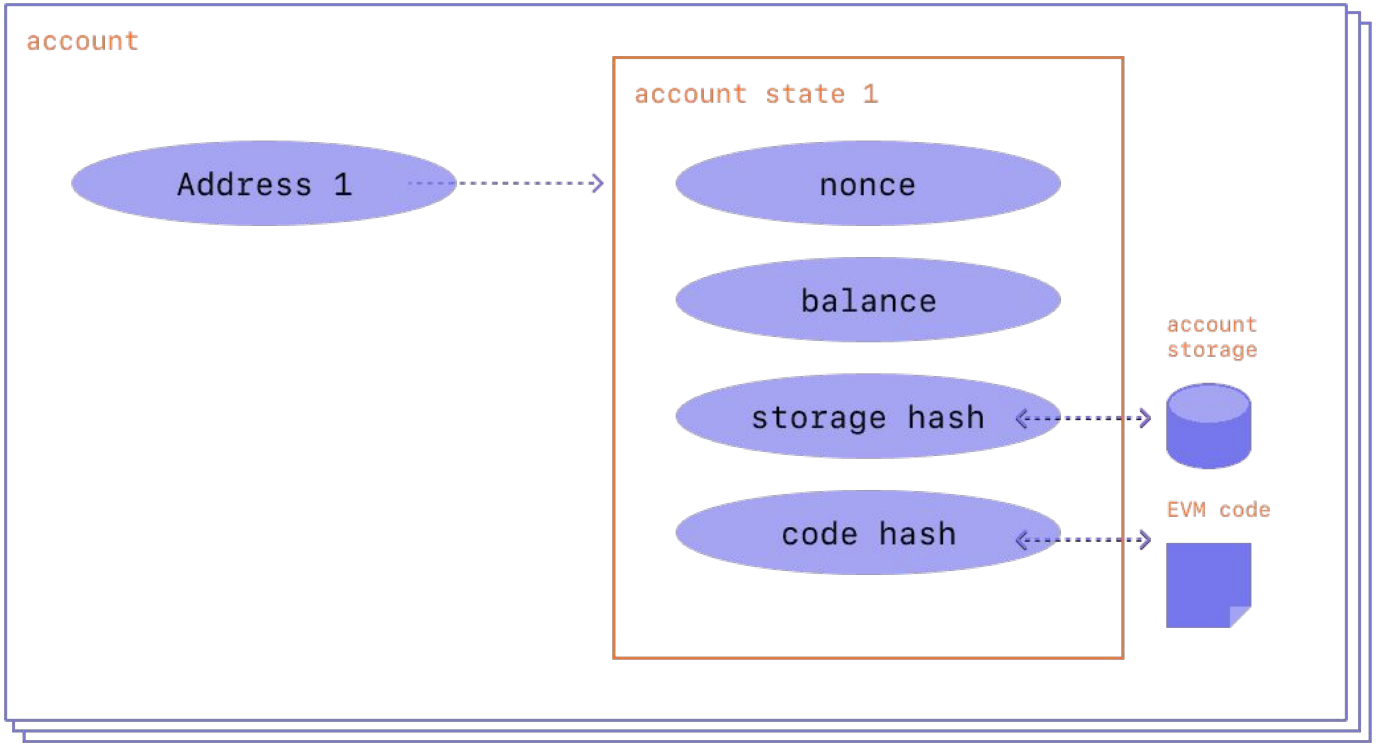
- Un account Ethereum è composto da 4 elementi:
  - **nonce,** un contatore del numero di transazioni inviate dall'account. Questo numero viene utilizzato per verificare che le transazioni vengano eseguito soltanto una volta. In un account contract identifica il numero di smart contract creati;
  - **balance**, il numero di **wei** (1 ETH = 10^{18} wei) posseduti dall'account;
  - **codeHash**, hash del codice che verrà eseguito sulla Ethereum Virtual Machine (EVM). Per gli account external questo hash è vuoto;
  - **storageRoot**, conosciuto anche come **storage hash** è un hash di 256 bit del root node di un Modified Merkle Patricia Trie (MPT) contenente i dati dello smart contract sotto forma di coppie chiave, valore.

Patricia Trie

- **Patricia trie** o **radix trie** o **radix tree** è una struttura dati che consente di memorizzare delle stringhe raggruppandole per caratteri comuni tramite un albero
- Curiosità: la parola inglese “trie” deriva da re**trie**val

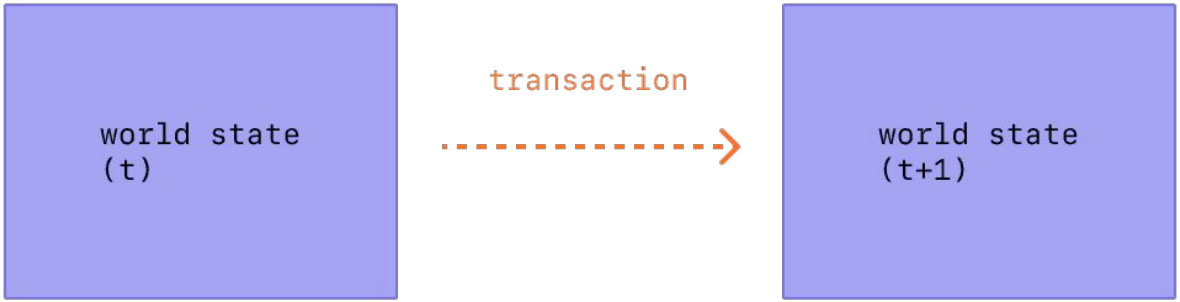


Ethereum account



Transazioni

- Una transazione Ethereum si riferisce a un'azione compiuta da un account externally-owned, quindi un account gestito da una persona
- Esempio, Bob invia ad Alice 1 ETH; il saldo di Bob sarà -1 e quello di Alice +1
- Una transazione produce un cambio di stato



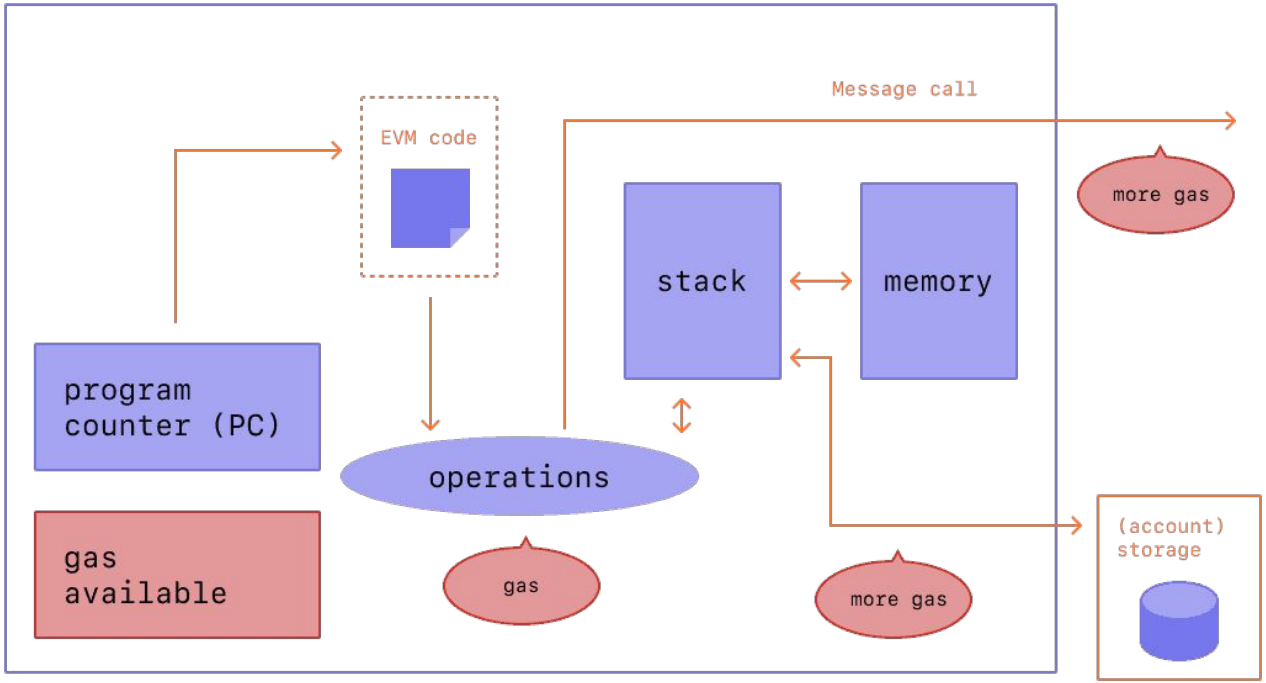
Transazioni (2)

- Un cambio di stato deve essere propagato in tutta la rete
- Tutte le transazioni richiedono una tassa (fee) per essere eseguite
- Le transazioni contengono le seguenti informazioni:
  - **recipient**, indirizzo del destinatario
  - **signature**, firma della transazione da parte del mittente (con la sua chiave privata)
  - **nonce**, un numero incrementale che identifica la transazione per l'account
  - **value**, l'ammontare di ETH oggetto della transazione (espresso in WEI)
  - **data**, qualsiasi dato si voglia specificare (opzionale)
  - **gasLimit**, il massimo numero di gas unit che la transazione può consumare
  - **maxPriorityFeePerGas**, il massimo numero di gas come mancia da dare al validatore
  - **maxFeePerGas,** la quantità massimo di gas disponibile per pagare la transazione (inclusa la maxPriorityFeePerGas)

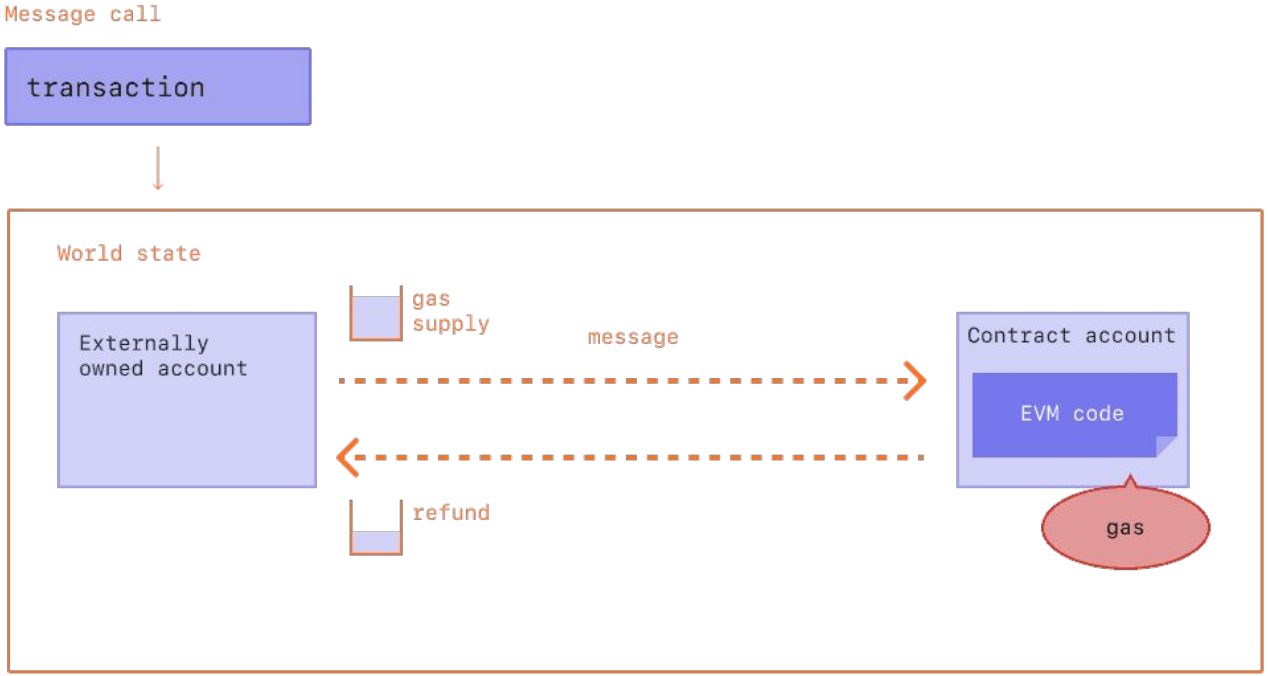
Gas

- Il **gas** è un'unità di misura relativa alla computazione necessaria per processare una transazione da parte di un nodo validatore
- Il **gasLimit** e il **maxPriorityFeePerGas** determinano il valore massimo pagato al validatore
- Se il gas non viene tutto consumato la parte eccedente viene recuperata
- Calcolo del costo di una transazione:
  - **units of gas used * (base fee + priority fee)**
  - Es: Alice paga 1 ETH a Bob, gas limit = 21'000 unità, base fee = 10 gwei, priority fee (mancia) di 2 gewi (1 gwei = 10^{-9} ETH = 0.000000001 ETH),
  - 21'000 * (10 + 2) = 252'000 gwei, 0.000252 ETH ≈ 0.38 $
- Stima dei costi delle transazioni su Ethereum: Ethereum Gas Tracker

Consumo del gas

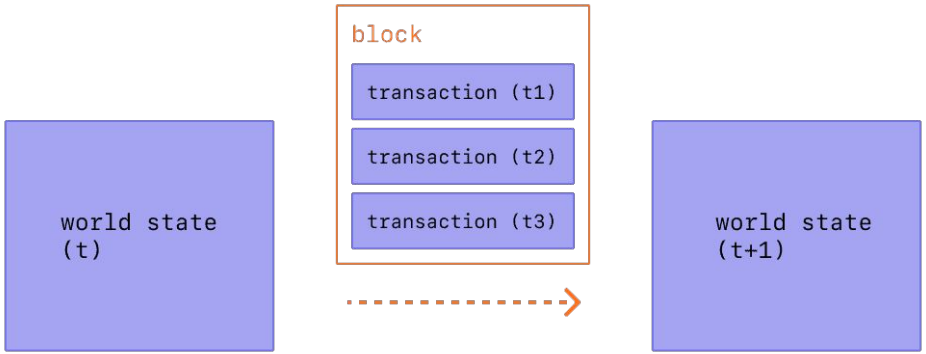


Gas refund


Message call
transaction
World state
Externally
owned account
gas
supply
message
Contract account
EVM code
gas
refund

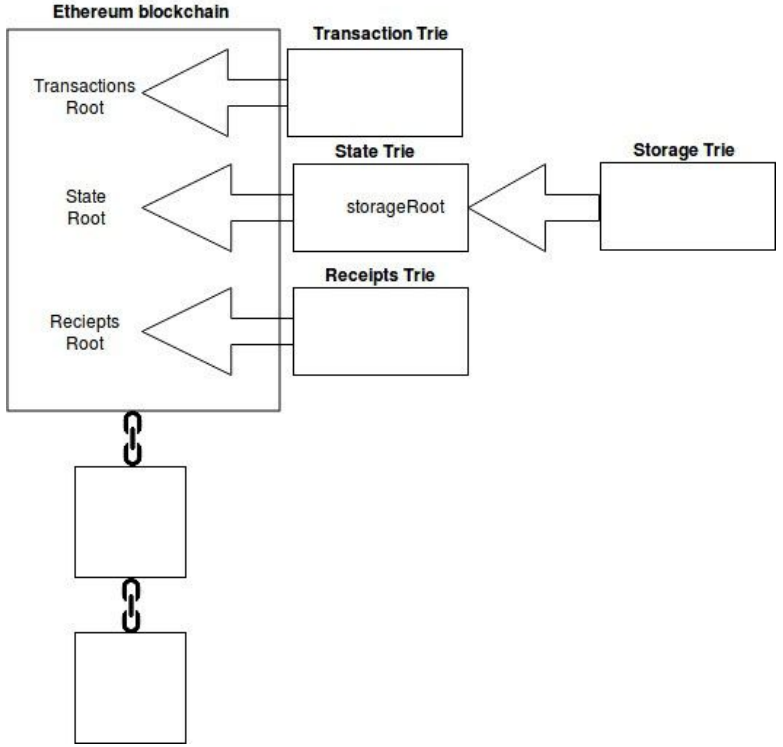
Blocchi

- Per garantire che tutti i partecipanti alla rete Ethereum mantengano una copia sincronizzata dello stato, le transazioni vengono raggruppate in blocchi (decine o centinaia)
- Anche se le transazioni possono essere inserite decine di volte al secondo, i blocchi vengono ogni **12 secondi** per consentire la sincronizzazione dei dati tra i nodi

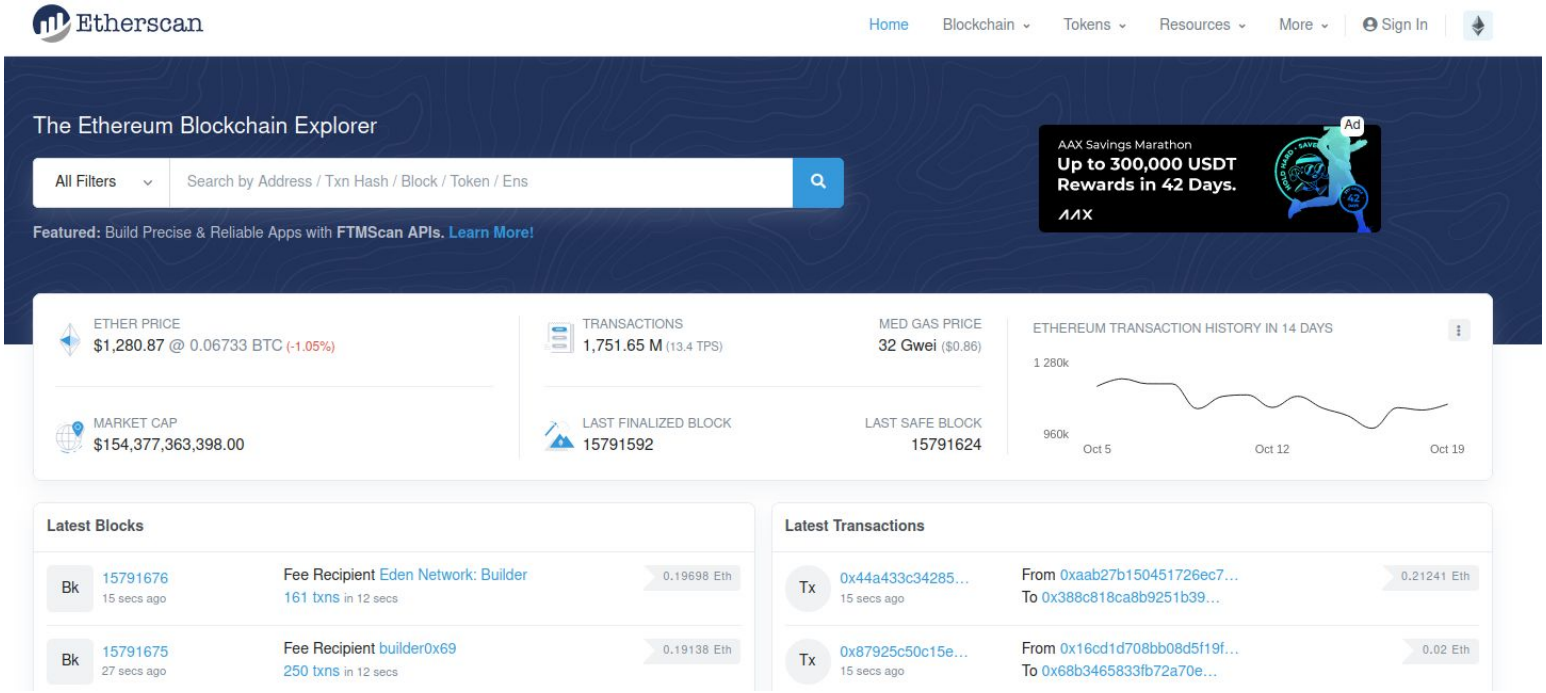


Dove sono memorizzati i dati?

- Gli smart contract, o meglio il codice EVM opcode è memorizzato nella blockchain Ethereum
- L'archivio degli account è memorizzato nello **Storage trie** (database di tipo chiave, valore)
- Lo **storageRoot** è memorizzato nella blockchain Ethereum



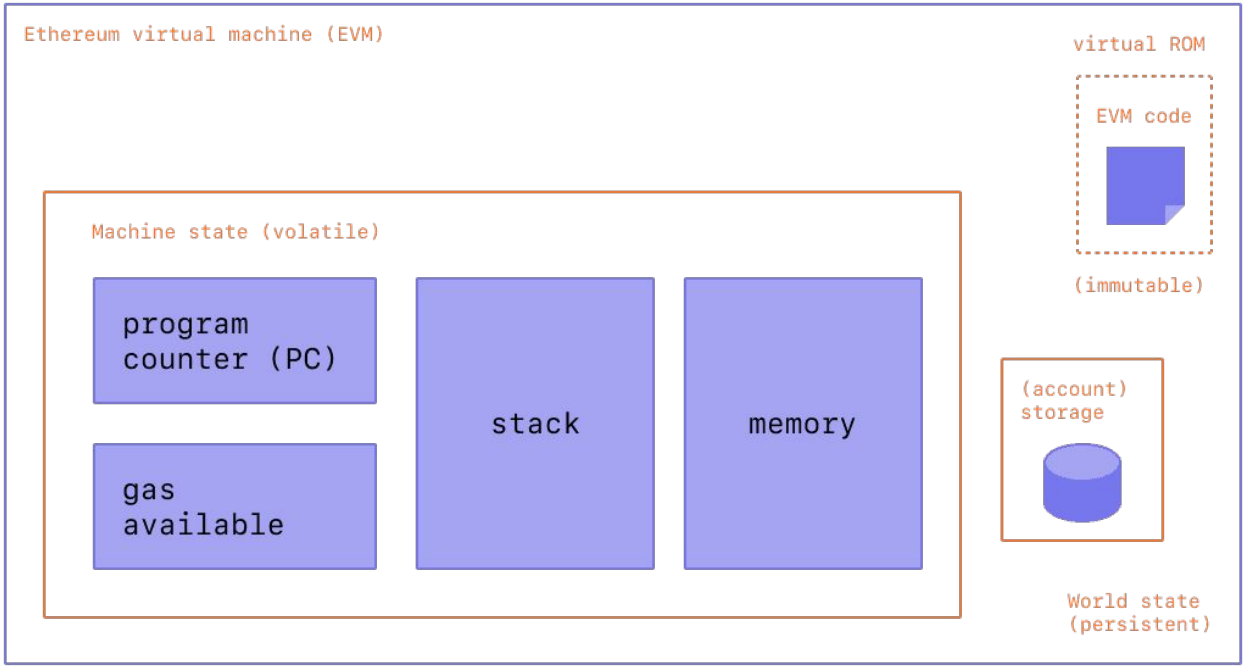
Etherscan.io



Ethereum Virtual Machine

- La Ethereum Virtual Machine (EVM) è il computer virtuale basato su stack per l'esecuzione degli smart contract
- Gli smart contract vengono scritti in un linguaggio ad alto livello, come Solidity e compilati in opcode
- L'opcode viene inserito nella blockchain Ethereum per poter essere eseguito da chiunque
- Il codice inserito nella blockchain è dunque immutabile (non può essere corretto)

EVM


Ethereum virtual machine (EVM)
virtual ROM
EVM code
(immutable)
Machine state (volatile)
program counter (PC)
gas available
stack
memory
(account) storage
World state (persistent)

Nodi

- Un nodo è un'istanza di un client Ethereum che è connesso ad altre istanze che girano su computer differenti, formando una rete
- Dopo The Merge il client Ethereum svolge due compiti:
  - **Execution Engine**, gestisce le nuove transazioni sulla rete, esegue il codice nella EVM, e sincronizza lo stato e i database di tutti i dati Ethereum
  - **Consensus client** (conosciuto anche come Beacon Node), implementa l'algoritmo di consenso proof-of-stake
- Il tutto utilizzando una sola rete (prima del The Merge si utilizzavano due reti diverse)

Tipi di nodi

- Full node
  - memorizza i dati di tutta la blockchain
  - partecipa alla validazione dei blocchi
  - tutti gli stati possono essere ricostruiti da un nodo full
  - è al servizio della rete, fornisce dati su richiesta
- Light node
  - memorizza soltanto gli header dei blocchi
  - non partecipano al meccanismo di consenso
  - non sono così diffusi come i full node
- Archive node
  - memorizza tutti i dati di un full node compreso lo storico degli stati
  - tipicamente terabyte di dati, non adatti a tutti gli utenti ma a coloro che sono interessati a realizzare servizi come block explorer, wallet e analisi dei dati

Network

- **Mainnet**, è la rete pubblica di Ethereum, quella dove vengono utilizzati ETH “reali”
- **Testnets**, sono delle reti pubbliche di test, dove gli ETH non hanno un valore economico
  - Goerli, rete di test basata su proof-of-stake
  - Sepolia, rete di test basata su proof-of-stake
  - Ropsten, rete di test deprecata

Bitcoin vs. Ethereum

	Bitcoin	Ethereum
Creator(s)	Satoshi Nakamoto	Vitalik Buterin, Charles Hoskinson, Gavin Wood, Jeffrey Wilcke, Mihai Alisie, Anthony Di Iorio and Amir Chetrit
Launch date	January 2009	July 2015
Currency vs. platform	A credible alternative to traditional fiat currencies (medium of exchange, store of value)	A platform to run programmatic contracts and applications via Ether
Consensus algorithm	Proof-of-Work	Proof-of-Stake
Block time	10 minutes on average	15 seconds on average
Transaction throughput	7 transactions per second	30 transactions per second
Supply	Finite supply-capped at 21 million BTC	Infinite supply

Riferimenti

- Andreas Antonopoulos, Gavin Wood, Mastering Ethereum: Building Smart Contracts and Dapps, O'Reilly, 2018
- Binance Academy, What is Proof of Stake? Youtube video
- Finematics, DEFI - From Inception To 2021 And Beyond (History Of Decentralized Finance Explained), Youtube video
- Laurent Senta, Where and how application data is stored in Ethereum?, Blog, 2021
- David Eisler, Trie and Patricia Trie Overview, Carleton University
- Vasa, Getting Deep Into Ethereum: How Data Is Stored In Ethereum?, HackerNoon, 2018
- Bernard Peh, Solidity Bytecode and Opcode Basics, Medium, 2017

Grazie dell’attenzione!

Per informazioni:
enrico.zimuel@its-ictpiemonte.it

POR Piemonte
FSE 2014-2020