AP

ERC20 Token

Ethereum Blockchain

16 FEBBRAIO 2021

01 Corso Token ERC20

IN BREVE

Panoramica

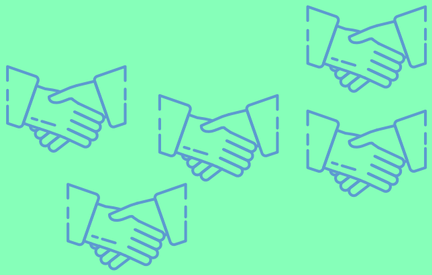
Token
Come nasce e cosa rappresenta il concetto di token su Ethereum

Perchè ERC20?
Cosa è e come nasce lo standard ERC20

Caratteristiche dello standard ERC20
Funzioni necessarie di un contratto affinchè sia calssificato come ERC20.

Esempio di ERC20 (essenziale)
Andiamo su Remix

TOKEN SU ETHEREUM



Come abbiamo visto, sulla macchina virtuale Ethereum, gli utenti possono interagire tra di loro o con la stessa EVM inviando delle richieste (transazioni). L'invio di richieste avviene tramite Smart Contracts.

Ogni interazione (quindi contratto) può generare quelli che sono chiamati **TOKEN** che possono a tutti gli effetti agire come valute "alternative" agli ETH.
Un esempio nel mondo reale sono ad esempio i punti spesa del supermercato. C'è un interazione tra me e il supermercato e in cambio ricevo dei punti che saranno poi equivalenti a € e potranno essere utilizzati per acquistare della merce.

Così come i punti del supermercato, anche i token su Ethereum non sono fisicamente detenuti dal proprietario.
Esistono solo all'interno di un particolare contratto (tessera supermercato) che ne regola le varie caratteristiche e i rispettivi bilanci. Ogni "prelievo" o utilizzo richiede da parte dell'utente un'interazione col contratto (senza invio di ETH, se non quelli necessari alle tasse).

Esattamente come utilizzando i punti spesa per pagare un prodotto non richiederà il pagamente di € alla cassa (se non quelli impliciti pagati come tasse sul prodotto).

PERCHÈ ERC20?

Il crescere continuo del numero di utenti e quindi di Smart Contracts ha fatto emergere una problematica.
Cosa succede se ognuno comunica nella propria "lingua". Ossia se ogni contratto creato è scritto secondo uno standard "soggettivo?

Si intuisce subito che nascerebbero delle incomprensioni tra gli utenti con la necessità di dover riscrivere ogni volta nuovi contratti che possano adattarsi o accettare il token di un altro contratto. (Un po' come se decidessi di spendere dei punti Carrefour nel supermercato Coop).

Da qui la necessità di creare uno standard comune, ERC20.

ERC - ETHEREUM REQUEST FOR COMMENT

Sono dei documenti tecnici che definiscono gli standard di programmazione su Ethereum. Sono vanno confuse con le EIP (Ethereum Improvement Proposal) che sono delle proposte di miglioramente dell'intero protocollo.
ERC20 tuttavia è nato durante la 20ima EIP (EIP20) e da qui il nome.

LO STANDARD ERC20

Affinchè un Token sia conforme allo standard ERC20 deve contenere le seguenti funzioni

totalSupply
Restituisce la fornitura totale di token nel contratto. (Include i token posseduti dai vari utenti)

balanceOf
Restituisce il bilancio di token di un particolare indirizzo (sia utente che contratto stesso)

transfer
Trasferisce token dal mandante (colui che crea la transazione) a un indirizzo. Genera anche un evento che viene scritto in modo permanente nella blockchain.

transferFrom
Trasferisce token da un indirizzo (non necessariamente colui che crea la transazione) a un altro. Genera anche un evento che viene scritto in modo permanente nella blockchain. Il mandante deve essere autorizzato a spendere quella cifra dall'indirizzo mittente.

approve
Autorizza qualcuno a utilizzare un certo ammontare del tuo bilancio. E' quindi direttamente collegata alla funzione transferFrom

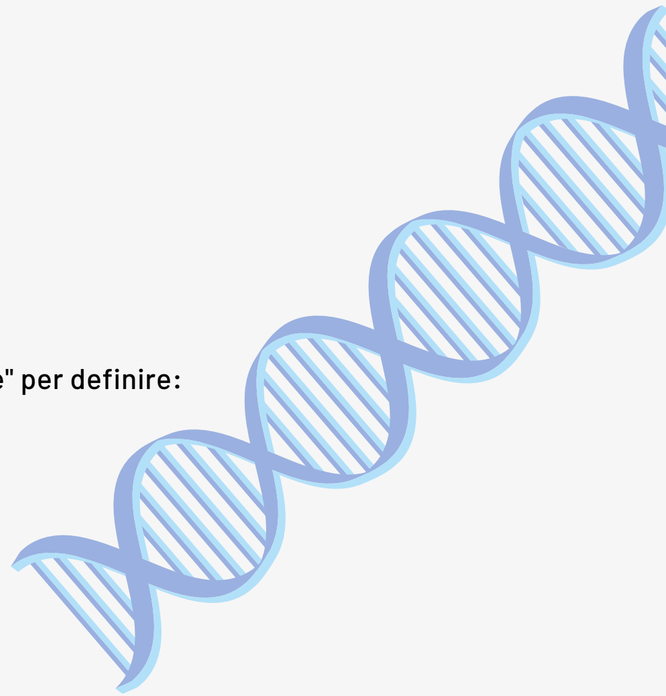
allowance
Visualizza un resoconto della funzione approve. Ossia quanti token un certo indirizzo può spendere su mandato di un altro.

UN PASSO INDIETRO

Analizziamo un contratto "completo" base che soddisfa lo standard ERC20.

Come vedremo, prima di definire le funzioni viste nella slide precedente, il contratto necessita di una serie di ulteriori contratti "base" per definire:

- Contesto di lavoro e classificazione degli utenti (Context)
- Interfaccia base esterna che soddisfi lo standard ERC20
- Sicurezza delle operazioni matematiche eventualmente utilizzate nel contratto (SafeMath)
- Caratterizzazione degli indirizzi (Address)



Sebbene non strettamente necessari, alcuni di questi contratti sono ormai parte di qualcunque Token ERC20.
Il principio sfruttato è l'ereditarietà tra contratti. Ossia, all'interno dello stesso "file" possono essere inclusi più contratti che potranno tra di loro ereditare funzioni, variabili e quant'altro.

VANTAGGI

- Maggiore organizzazione del codice
- Separazione delle funzioni e maggiore facilità di test
- Possibilità di frazionare i deployment

SVANTAGGI

- Complessità di programmazione
- Difficoltà di lettura per l'utente
- Rischi di sicurezza (eredità da contratti esterni - bachi)