Loteria Descentralizada em Blockchain EOSIO

Ricardo de Barros Marlière

Universidade Federal de Juiz de Fora

Dezembro de 2018

Histórico


THE TIMES

Saturday January 3 2009 timesonline.co.uk No 69523 £1.50

Eat Out from £5

More than 900 great restaurants, including four Gordon Ramsay favourites from £15

Start collecting tokens today Pullout inside

Israel prepares to send tanks and troops into Gaza

Israel allowed foreigners to flee the Gaza Strip as it prepared for a ground offensive. At least 430 Palestinians were killed in a week of airstrikes News, page 3

Michael Sheen Frost, Nixon and me

Working mums So that's how she does it

Detox in style The best spas on the planet

Salman Rushdie I won't marry again

Giant killing? Guide to the FA Cup third round

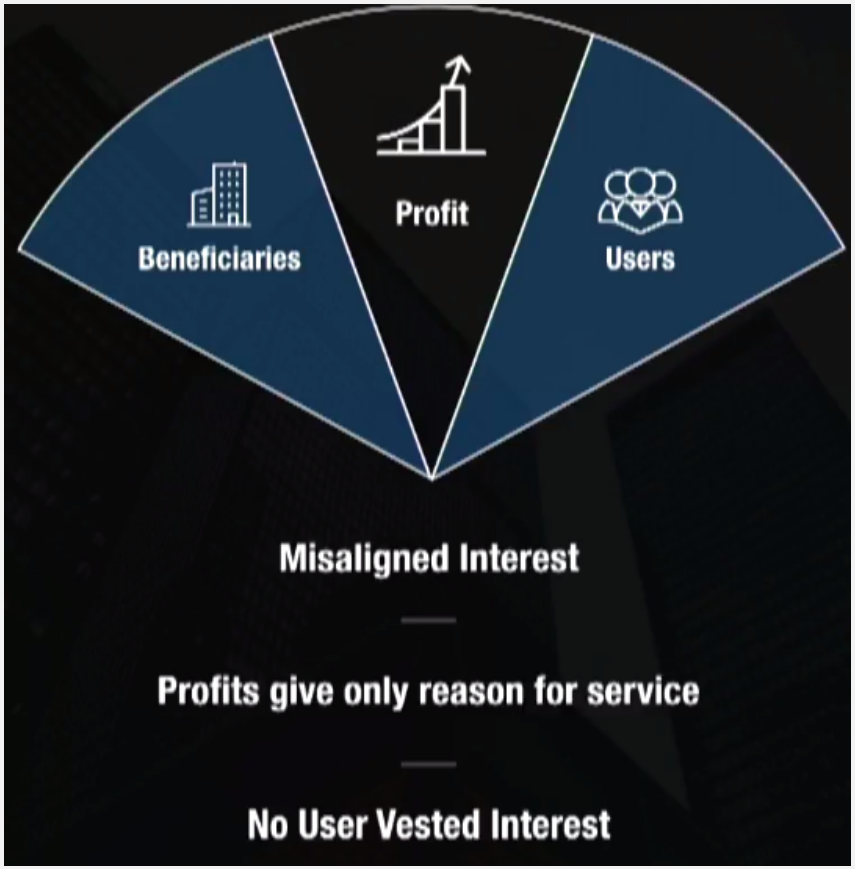
Chancellor on brink of second bailout for banks

Billions may be needed as lending squeeze tightens

99p

- Bitcoin é concebido em 2008 e implementado em 2009.
- Ethereum é concebido em 2013 e implementado em 2015.
- EOSIO é concebido em 2017 e implementado em 2018.

Problema de Plataformas Centralizadas



Exemplos:

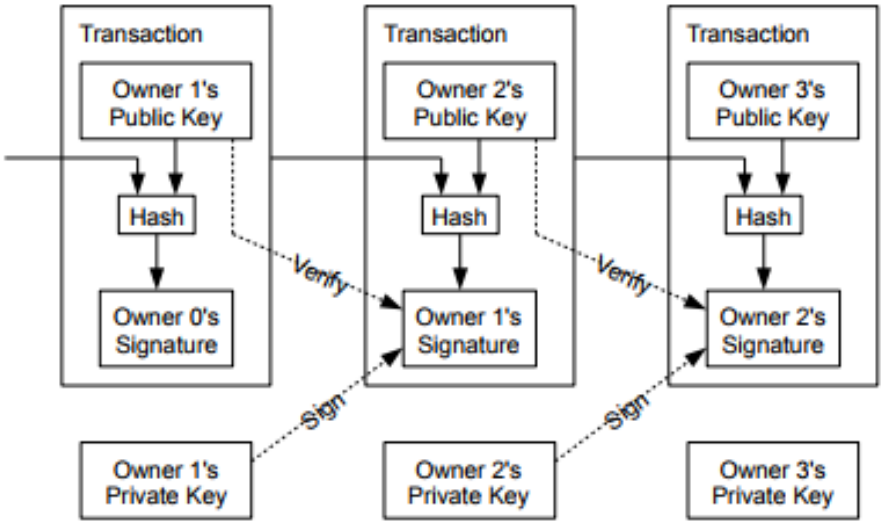
- Uber não possui carros.
- Airbnb não possui imóveis.
- Spotify não cria música.
- Facebook não cria conteúdo.
- Alibaba não possui estoque.
- iFood não produz alimentos.

Objetivo das Blockchains

Descentralizar e remover intermediários

Blockchain

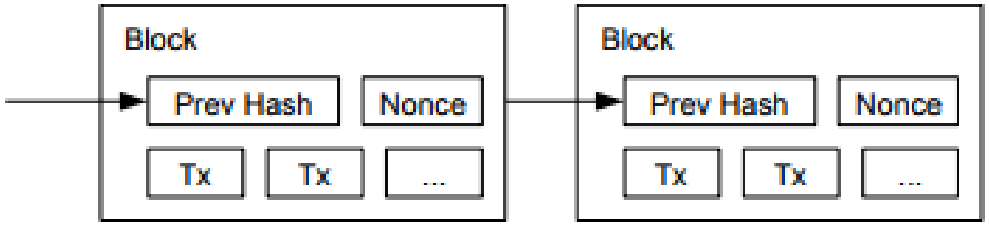
Transferência de Propriedade



- Dono anterior anuncia o novo dono.
- Prova-se criptograficamente que num determinado momento (bloco) a propriedade foi alterada.
- Assim, o dono anterior não consegue gastar as mesmas moedas novamente.

Blockchain

Lista Encadeada de Blocos



- Mantém um livro-razão com o timestamp de cada transação.
- Registro é mantido irreversível por via de Proof of Work.
- Retrabalho para alterar o histórico é inviável.

Blockchain

Mineração e Proof-of-Work


```
{"block_num":101,"previous":"aca376f206b8fc25a6ed44dbdc66547c36c6c33e3a119ffbeaef943
642f0e906","producer":"miner1","producer_signature":"SIG_K1_111111111111111111111111
1111111111116uk5ne","timestamp":"2018-06-08T08:08:08.500","transactions":["tx1","tx2
","tx3","tx4","tx5"],"nonce":1}

39ff35560c0cc34bb8ffa2ad12edf8efe9bec76ff403ba2ff93256e62e88b52e  next_block.1st_try
```

```
{"block_num":101,"previous":"aca376f206b8fc25a6ed44dbdc66547c36c6c33e3a119ffbeaef943
642f0e906","producer":"miner1","producer_signature":"SIG_K1_111111111111111111111111
1111111111116uk5ne","timestamp":"2018-06-08T08:08:08.500","transactions":["tx1","tx2
","tx3","tx4","tx5"],"nonce":2}

3aeb8f64dc36c213dc6c9713c2b6abf1f802fb9b99cb8fdf19e98e2ffc0f8d00  next_block.2nd_try
```


- Por definição, primeira transação bota em circulação novas moedas para quem o descobriu.
- Oferta limitada em 21 milhões, emitida de forma descentralizada.
- Recompensa incentiva nós e evita que se tornem bizantinos.

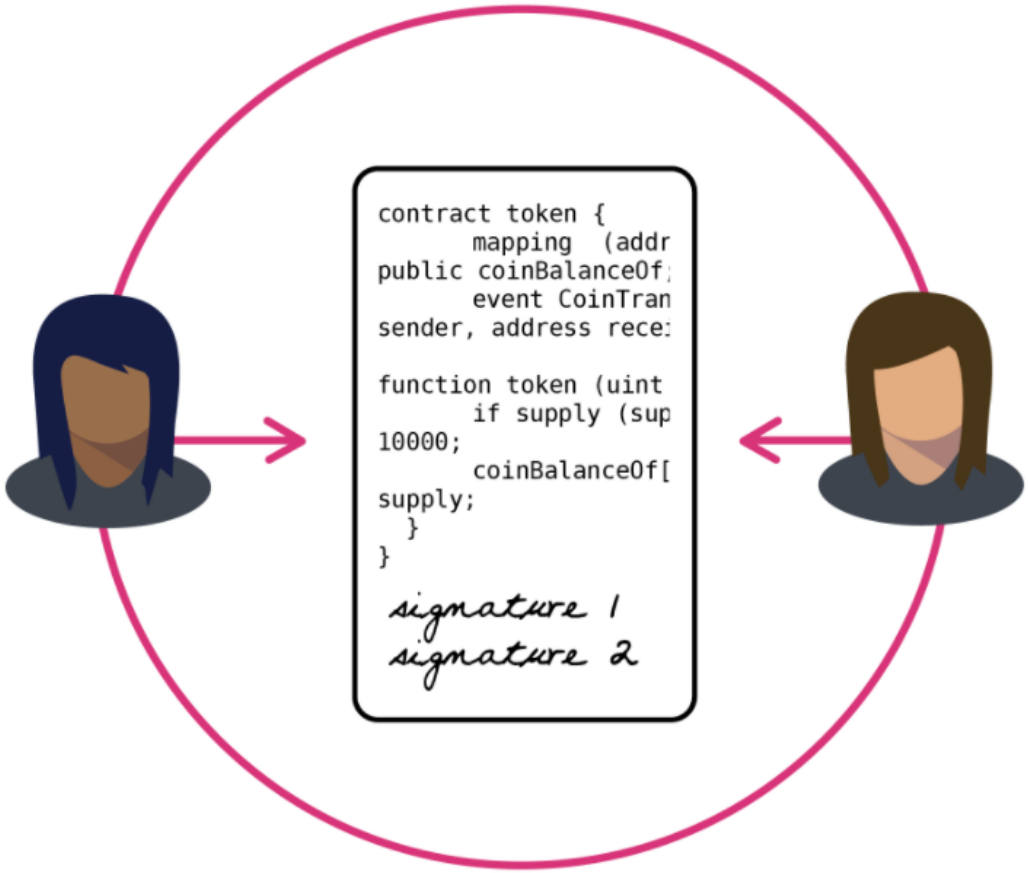
Smart Contracts

- Conceito concebido na década de 90 e incorporado em blockchains com o advento do Bitcoin.
- Cláusulas contratuais podem ser embutidas em hardware ou software, de forma a automatizar e garantir sua execução.
- Um contrato inteligente é um programa cuja execução é autônoma, transparente e imutável.

Smart Contracts

- Um contrato "tradicional" é definido em papel.
- Leis prévias existem para que sua execução seja garantida, caso contrário deve haver juízo.
- A auditoria deve ser feita (semi) manualmente, através da recoleção de dados.

Smart Contracts



- Um contrato inteligente é definido em software.
- Sua execução é feita de consensualmente pelos nós validadores e suas entradas e saídas tornam-se imutáveis.
- A auditoria é automatizada visto que os dados ficam transparentemente disponíveis.

Contratos em EOSIO

Hello World!

```
#include <eosiolib/eosio.hpp>

using namespace eosio;
using namespace std;

class [[eosio::contract("hello")]] hello : public contract
{
    public:
        using contract::contract;

        [[eosio::action]] void printact( string s )
        {
            print( s );
        }
};

EOSIO_DISPATCH( hello, (printact) )

```

Contratos em EOSIO

Implantação

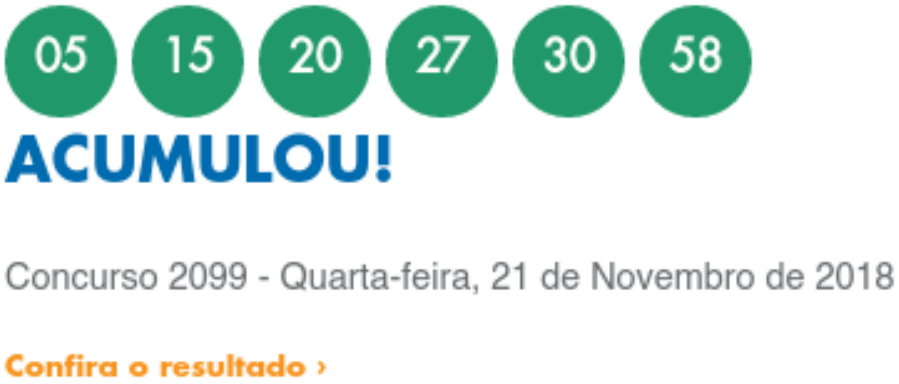
```
[00:15:44 eos@eos:~/git/hello_world]
% cleos set code accountnum11 hello.wasm
Reading WASM from hello.wasm...
Setting Code...
executed transaction: 41bbbb585b83fe95b9249cad2fd35b5a31beeb1d69d024e3b55cc74cd958ff08  280
0 bytes  678 us
#         eosio <= eosio::setcode               {"account":"accountnum11","vmtype":0,"vmver
sion":0,"code":"0061736d01000000015c1060027f7f00600000600...
warn  2018-11-15T00:15:49.957 thread-0  main.cpp:482                  print_result
warning: transaction executed locally, but may not be confirmed by the network yet

[00:15:49 eos@eos:~/git/hello_world]
% cleos set abi accountnum11 hello.abi
Setting ABI...
executed transaction: 7dd0fccb8585f3fae336cb1c0bd950683007ff2a130e5121847105badd5718cd  144
 bytes  601 us
#         eosio <= eosio::setabi                {"account":"accountnum11","abi":"0e656f7369
6f3a3a6162692f312e3000010870726 96e7461637400010173067374 7...
warn  2018-11-15T00:15:54.705 thread-0  main.cpp:482                  print_result
warning: transaction executed locally, but may not be confirmed by the network yet

[00:15:54 eos@eos:~/git/hello_world]
% cleos push action accountnum11 printact '["hello world"]' -p accountnum11
executed transaction: 0b59ca1eb0292fbffe43eaaff2ebb010dbaf7162b11482640d0bc7c4e7ac54e2  104
 bytes  556 us
#  accountnum11 <= accountnum11::printact       {"s":"hello world"}
>> hello world
warn  2018-11-15T00:15:58.802 thread-0  main.cpp:482                  print_result
warning: transaction executed locally, but may not be confirmed by the network yet
```

Objetivo do Trabalho

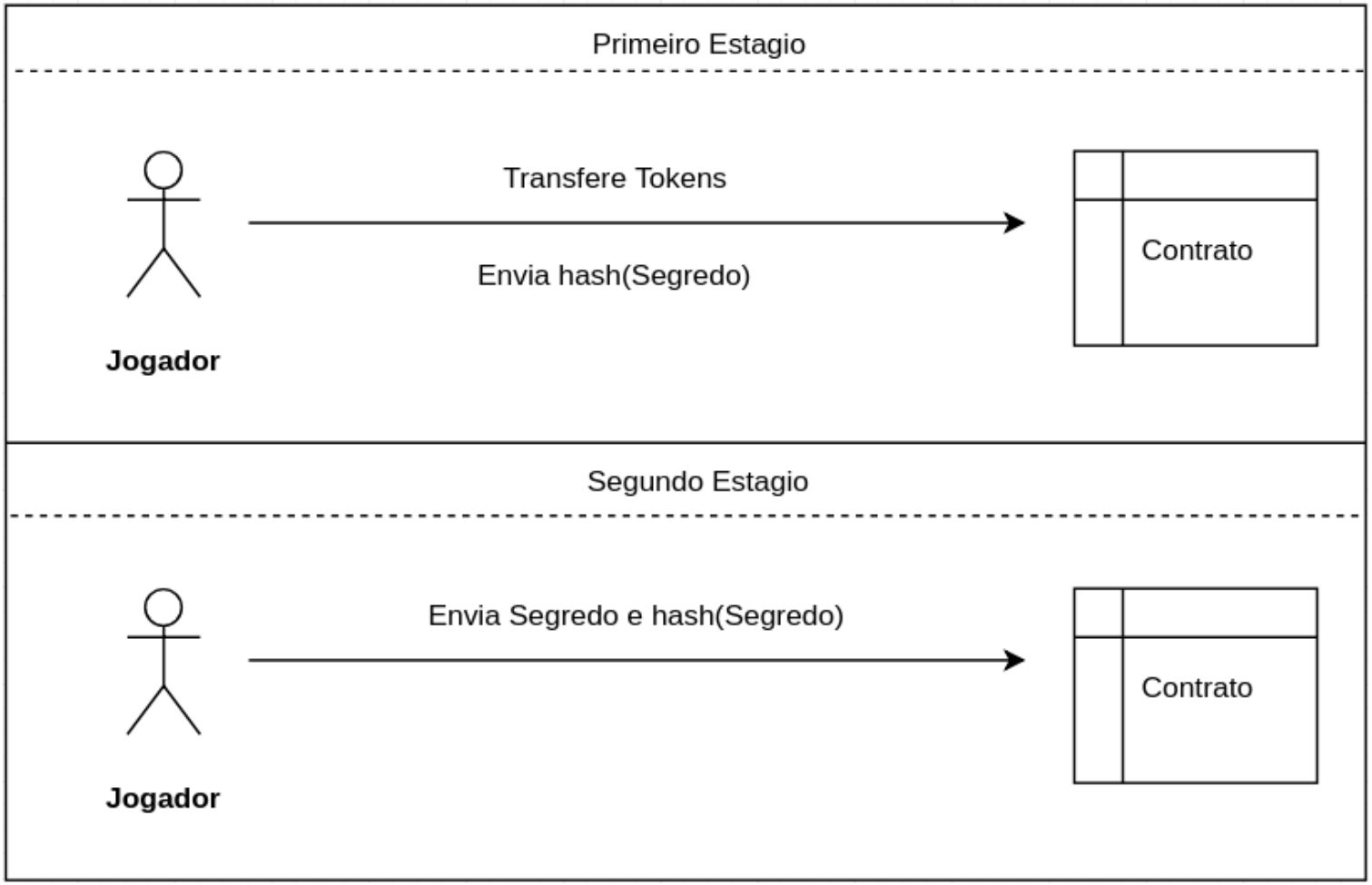
Descentralizar uma Loteria



- Ao invés de depender de "Caminhões da Sorte", programa-se um contrato para gerenciar as apostas e escolher os vencedores.
- Usuários interagem com o contrato através de chamadas criptograficamente autorizadas às suas ações.

Modelo da Aplicação

Fluxo do Jogo



Modelo da Aplicação

Ações do Contrato

```
[[eosio::action]] void setgame( uint64_t max_players,
                                uint32_t interval,
                                uint8_t  stage );

[[eosio::action]] void reset( name reset );

[[eosio::action]] void submithash( name             player,
                                   asset            quantity,
                                   capi_checksum256 hash );

[[eosio::action]] void submitboth( name             player,
                                   capi_checksum256 hash,
                                   capi_checksum256 secret );

```

Modelo da Aplicação

Tabelas do Contrato

```
1  struct [[eosio::table]] game
2  {
3      time_point_sec deadline;
4      uint64_t       max_players;
5      uint32_t       interval;
6      uint8_t        stage;
7      asset          total_pot;
8  };
9
10 struct [[eosio::table]] player
11 {
12     name             player;
13     capi_checksum256 hash;
14     capi_checksum256 secret;
15     asset            quantity;
16     boolean          active;
17     auto primary_key() const { return player.value; }
18 };
19
```