

# Introdução a Ethereum

O que vem depois do Bitcoin?



# Steve Wozniak compara Ethereum à Apple na Conferência de Viena



*"Ethereum me interessa porque pode fazer coisas e porque é uma plataforma."*

<https://br.cointelegraph.com/news/steve-wozniak-compares-ethereum-to-apple-at-vienna-conference>

# 94 dos 100 Maiores Projetos de Blockchain Partiram da Ethereum, diz Pesquisador

## Jovens mulheres no Afeganistão abraçam Ethereum para mudar situação no país

Por Cassio Gusson - 18 de junho de 2018



Compartilhar no Facebook



Tweet



Ethereum Chega aos 35 Milhões de Endereços e Supera o Bitcoin em Endereços Ativos

# Recapitulando



Foco em transações

# Recapitulando

Foco em transações



Bob

# Recapitulando

Foco em transações



**Bob**

# Recapitulando

Foco em transações



**Bob**



**Auaulice**

# Recapitulando

## Foco em transações



**Transaction** View information about a bitcoin transaction

72cbd36622b302576a6aa20211727ae99dbba9244f331fb6eb1d67a16cfb883

32qBrvRK1c79ByBASZZMLhSdwe6Wvb7vcy (0.09302905 BTC - Output) → 19kX2vNF45JxKtVStUQax685XySXc5d4J - (Unspent) 0.03789902 BTC  
32qBrvRK1c79ByBASZZMLhSdwe6Wvb7vcy - (Spent) 0.05006051 BTC

5 Confirmations 0.08795953 BTC

**Input** **UTXO**

Summary	
Size	372 (bytes)
Weight	828
Received Time	2018-01-20 07:25:59
Included In Blocks	<a href="#">505131</a> ( 2018-01-20 07:27:30 + 2 minutes )
Confirmations	5 Confirmations
Visualize	<a href="#">View Tree Chart</a>

Inputs and Outputs	
Total Input	0.09302905 BTC
Total Output	0.08795953 BTC
Fees	0.00506952 BTC
Fee per byte	1,362.774 sat/B
Fee per weight unit	612.261 sat/WU
Estimated BTC Transacted	0.03789902 BTC
Scripts	<a href="#">Hide scripts &amp; coinbase</a>



# Recapitulando

## Foco em transações



*Bitcoin uses a scripting system for transactions. Forth-like, **Script** is simple, stack-based, and processed from left to right. It is intentionally not Turing-complete, with no loops.*

Estrelando...



ethereum

# Foco: Executar Smart Contracts

```
/* Allow another contract to spend some tokens in your behalf */
function approve(address _spender, uint256 _value)
    returns (bool success) {
    allowance[msg.sender][_spender] = _value;
    return true;
}

/* Approve and then communicate the approved contract in a single tx */
function approveAndCall(address _spender, uint256 _value, bytes _extraData)
    returns (bool success) {
    tokenRecipient spender = tokenRecipient(_spender);
    if (approve(_spender, _value)) {
        spender.receiveApproval(msg.sender, _value, this, _extraData);
        return true;
    }
}

/* A contract attempts to get the coins */
function transferFrom(address _from, address _to, uint256 _value) returns (bool success) {
    if (balanceOf[_from] < _value) throw; // Check if the sender has enough
    if (balanceOf[_to] + _value < balanceOf[_to]) throw; // Check for overflows
    if (_value > allowance[_from][msg.sender]) throw; // Check allowance
    balanceOf[_from] -= _value; // Subtract from the sender
    balanceOf[_to] += _value; // Add the same to the recipient
    allowance[_from][msg.sender] -= _value;
    Transfer(_from, _to, _value);
    return true;
}

/* This unnamed function is called whenever someone tries to send ether to it */
function () {
    throw; // Prevents accidental sending of ether
}
```

# Foco: Executar Smart Contracts

```
/* Allow another contract to spend some tokens in your behalf */
function approve(address _spender, uint256 _value)
    returns (bool success) {
    allowance[msg.sender][_spender] = _value;
    return true;
}

/* Approve and then communicate the approved contract in a single tx */
function approveAndCall(address _spender, uint256 _value, bytes _extraData)
    returns (bool success) {
    tokenRecipient spender = tokenRecipient(_spender);
    if (approve(_spender, _value)) {
        spender.receiveApproval(msg.sender, _value, this, _extraData);
        return true;
    }
}

/* A contract attempts to get the coins */
function transferFrom(address _from, address _to, uint256 _value) returns (bool success) {
    if (balanceOf[_from] < _value) throw; // Check if the sender has enough
    if (balanceOf[_to] + _value < balanceOf[_to]) throw; // Check for overflows
    if (_value > allowance[_from][msg.sender]) throw; // Check allowance
    balanceOf[_from] -= _value; // Subtract from the sender
    balanceOf[_to] += _value; // Add the same to the recipient
    allowance[_from][msg.sender] -= _value;
    Transfer(_from, _to, _value);
    return true;
}

/* This unnamed function is called whenever someone tries to send ether to it */
function () {
    throw; // Prevents accidental sending of ether
}
```