





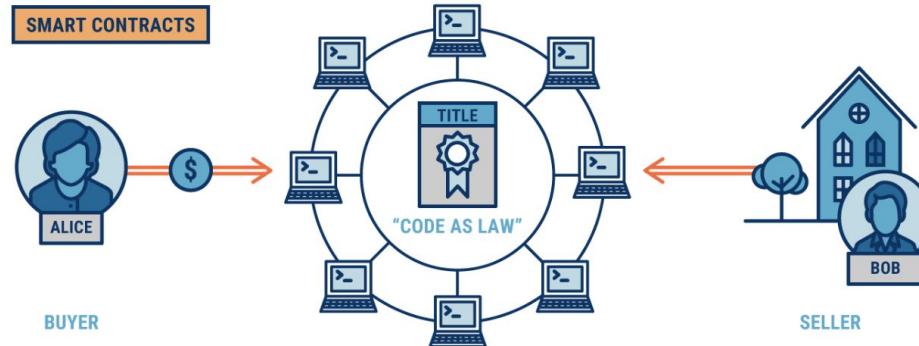
**BITCOIN**

**VS**

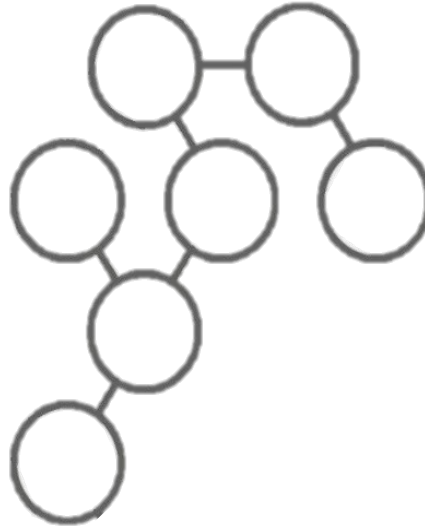


**ETHEREUM**

# What **Smart Contract** mean for the Future of Business

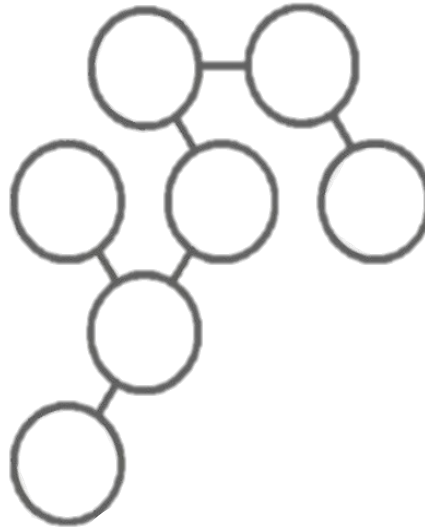


# How **Ethereum** works?



**Ethereum  
Network**

# How **Ethereum** works?

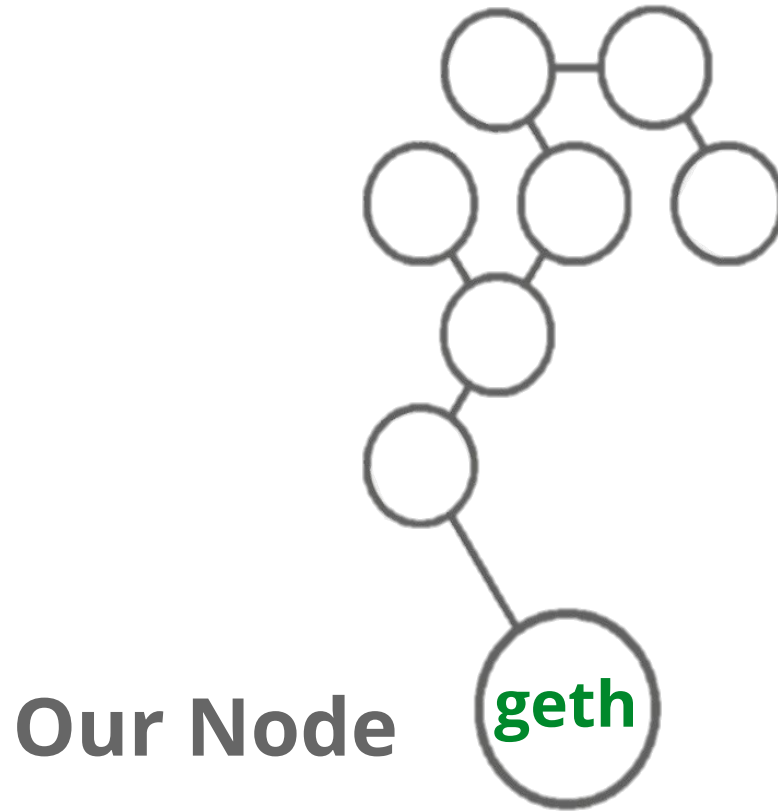


**Ethereum  
Network**

**Our Node**

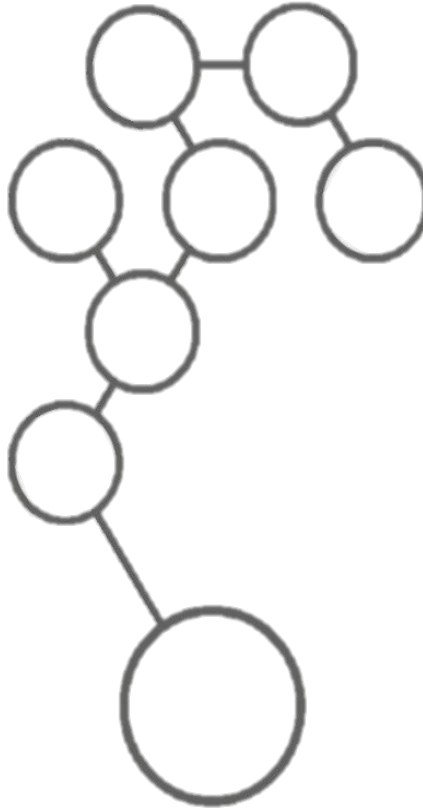
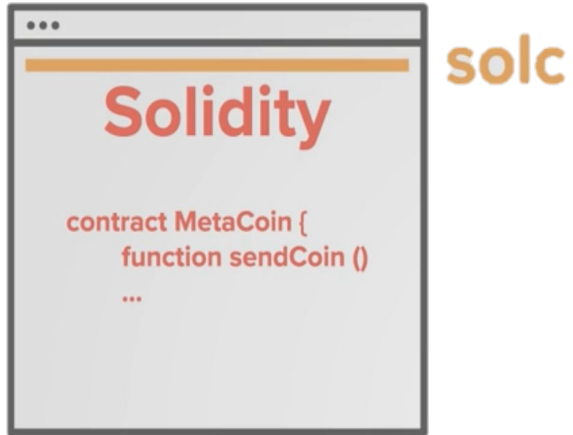
**geth**

# How **Ethereum** works?

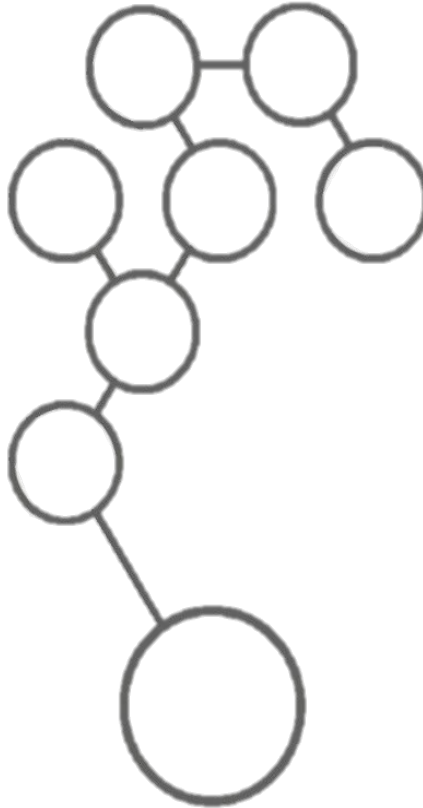
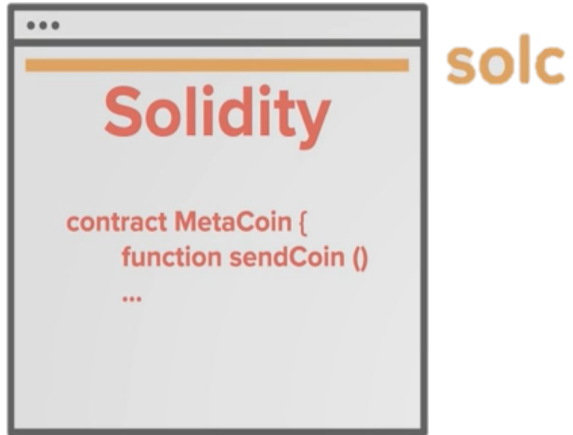


**Ethereum  
Network**

# How **Ethereum** works?



# How **Ethereum** works?

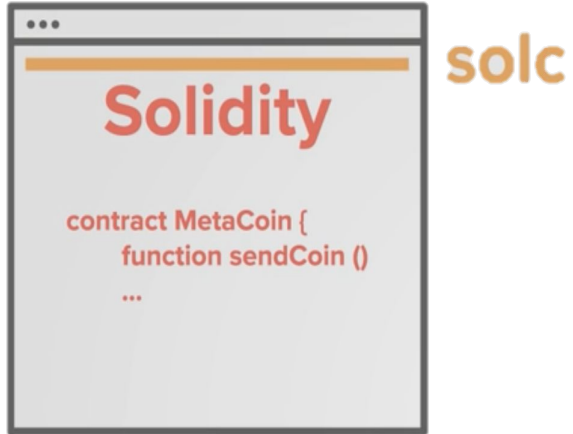


**bytecode**

0x370777fabcea2  
12hdyt321jgasda  
389676413984sad  
123ads36ad5sa67  
sa5665as457a4as  
as657as56a4a3a7  
4as4a5sa6s7a2...



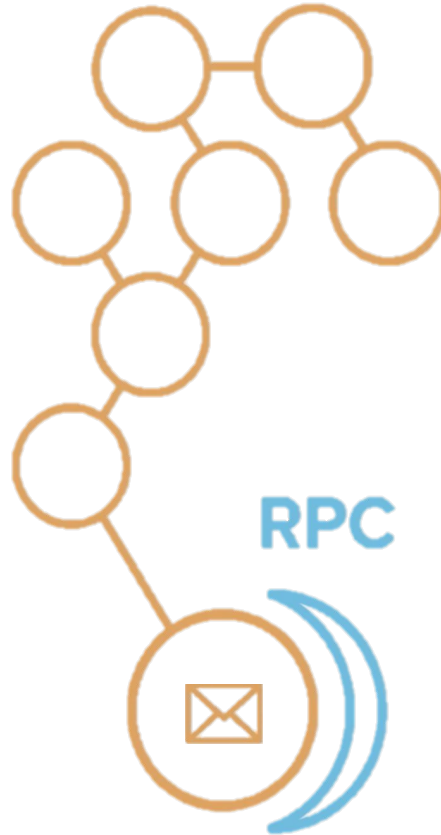
# How **Ethereum** works?



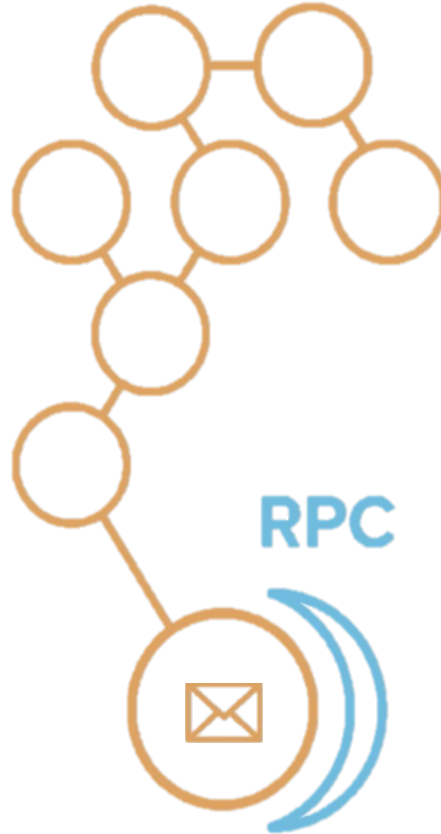
## bytecode

0x370777fabcea2  
12hdyt321jgasda  
389676413984sad  
123ads36ad5sa67  
sa5665as457a4as  
as657as56a4a3a7  
4as4a5sa6s7a2...

# How **Ethereum** works?

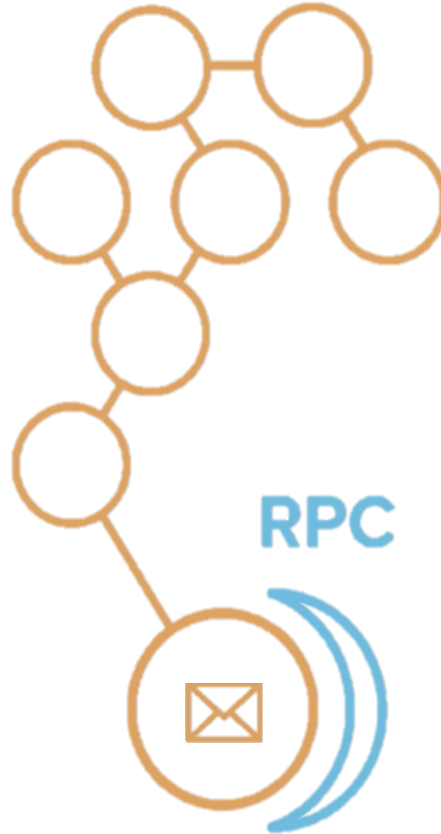


# How **Ethereum** works?

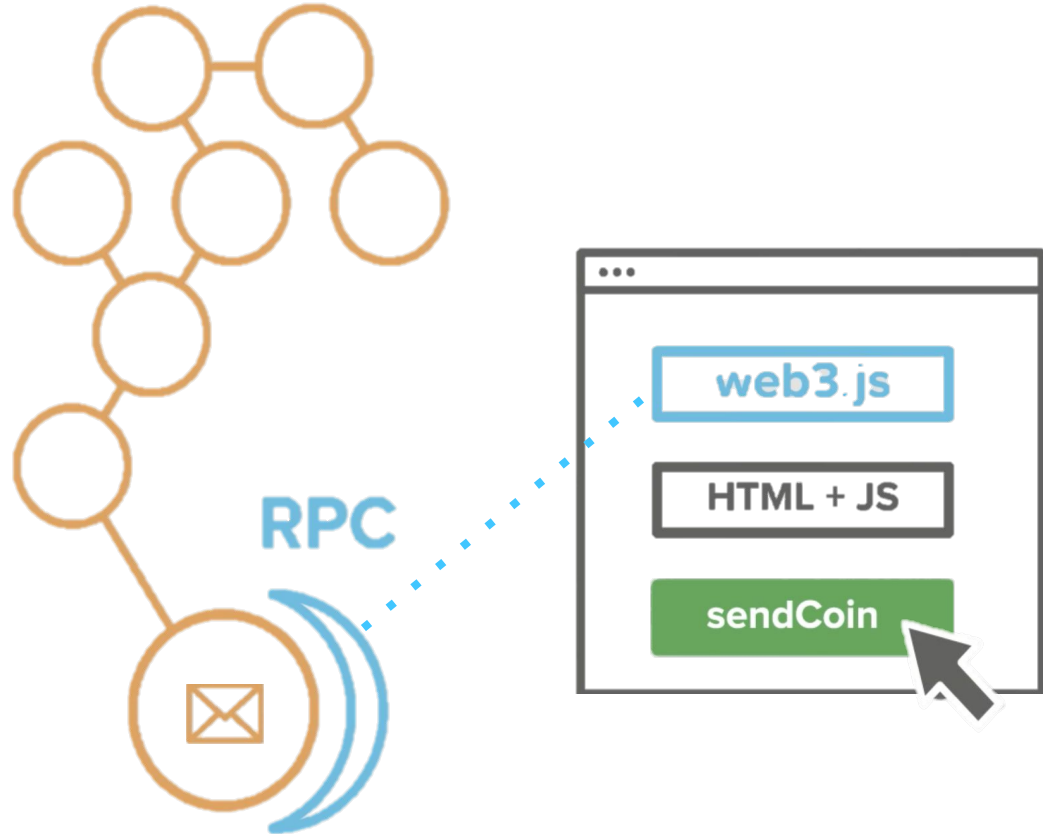


web3.js

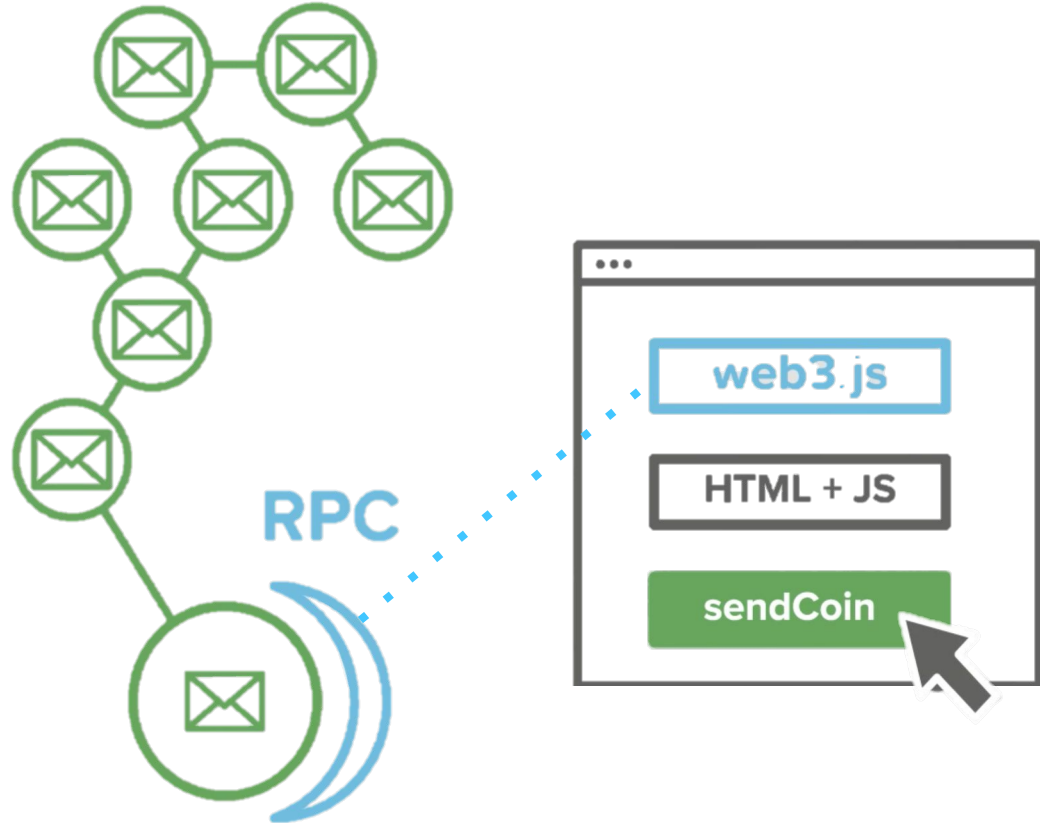
# How **Ethereum** works?



# How **Ethereum** works?



# How **Ethereum** works?



# Ethereum stack



# Solidity in depth

constructor

event

mapping, structs

public / internal / private

views / pure / payable

require / modifiers

[More documentation](#)

```
pragma solidity ^0.4.24;

contract Election {
    struct Candidate {
        uint id;
        string name;
        uint voteCount;
    }

    event votedEvent(uint candidateId, string candidateName);
    event candidateVoteCount(uint candidateId, string candidateName, uint candidateCount);

    mapping(uint => Candidate) public candidates;
    mapping(address => bool) public voters;

    uint public candidatesCount;

    constructor() public {
        addCandidate("Equipo 1");
        addCandidate("Equipo 2");
        addCandidate("Equipo 3");
    }

    modifier isValidCandidate(uint _candidateId) {
        require(_candidateId > 0 && _candidateId <= candidatesCount);
        _;
    }

    function addCandidate(string _name) private {
        candidatesCount++;
        candidates[candidatesCount] = Candidate(candidatesCount, _name, 0);
    }

    function vote(uint _candidateId) public isValidCandidate(_candidateId) {
        require(!voters[msg.sender]);

        voters[msg.sender] = true;
        candidates[_candidateId].voteCount++;
        emit votedEvent(_candidateId, candidates[_candidateId].name);
    }

    function getVotes(uint _candidateId) public view isValidCandidate(_candidateId) returns(uint, string, uint) {
        emit candidateVoteCount(_candidateId, candidates[_candidateId].name, candidates[_candidateId].voteCount);
        return (_candidateId, candidates[_candidateId].name, candidates[_candidateId].voteCount);
    }
}
```



# Workshop 1. Creating a private network.

Private network as a development environment with two nodes.

What are we going to do?

- Create the network.
- Deploy the SmartContract.
- Interact with the SmartContract.



What do we need?

- Geth (GoEthereum). Most basic Ethereum client.
- Solc (Solidity Compiler).



solidity

# Workshop 2. My first project.

Solidity  
Truffle  
Ganache  
web3.js  
Metamask

## Resultados Hackatón

#	Nombre Equipo	Votos
1	Equipo 1	0
2	Equipo 2	0
3	Equipo 3	0

Selecciona tu Equipo favorito

Equipo 1



Votar

Your Account: 0xe240fb40e9641217a4e785d95c770ce545f5e5b5

# Development tools



To try out your smartcontracts: remix

<http://remix.ethereum.org>

Compile and migrate smartcontract: truffle

<https://truffleframework.com/>

Development IDE: vscode

<https://code.visualstudio.com/>

Ethereum development node: ganache-cli

<https://github.com/trufflesuite/ganache-cli>


Ethereum tesnet node: infura <https://infura.io/>



solidity



# Interesting Projects ( take a look [truffleframework.com/boxes](https://truffleframework.com/boxes) )

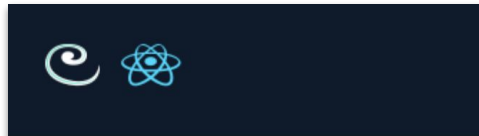


**pet-shop**

★ 101

A box containing all you need to get started with our Pet Shop tutorial.

💎 official, official



**react**

★ 337

Truffle, Webpack and React boilerplate.

💎 official, official, ethereum, ethereumjs, truffle, webpack, react, solidity




**Cheshire**

**endless-nameless-inc/cheshire**

★ 52

A sandbox for CryptoKitties dApp developers

💎 community, community, cryptokitties, dapp, ethereum




**Quintor/angular-truffle-box**

★ 37

Truffle Box for Angular is a quick-and-easy way to get your Dapp on the road with Truffle and Angular

💎 community, community



**DOkwufulueze/eth-vue**

★ 96

💎 community, community, vuejs, ethereum, ethereum-contract, dapp,