



TRY-CHAIN!

Sobre nosotros



Roberto García Álvarez

[LinkedIn](#)



Víctor Nieves Sánchez

[LinkedIn](#)

Sobre nosotros



Fuentes: <https://www.inetum.com.es/en/index.html>, <https://www.alastria.io/>

Contenido

01 Introducción

¿Qué es eso de blockchain?

05

02 Tipos de blockchain

Espera ... ¿hay varios tipos de blockchain?

09

03 Cómo levantar un nodo en Quorum

Nodos y más nodos

12

04 Geth sobre un nodo

API para trabajar con un nodo

14

05 Truffle, Ganache, Solidity y Remix IDE

Suite para desarrollar

16

06 ¿Blockchain para todo?

Póngame una de blockchain para llevar, ¡gracias!

21

1

Introducción

¿Qué es blockchain?

Blockchain es una tecnología que funciona como un **libro en el que se registran las transacciones** que se llevan a cabo.

Una **transacción es un mensaje que se firma digitalmente mediante criptografía** y se envía a toda la red.



1 A wants to send money to B



2 The transaction is represented online as a "block"



3 The block is broadcast to every party in the network



4 Those in the network approve the transaction is valid



5 The block then can be added to the chain, which provides an indelible and transparent record of transactions



6 The money moves from A to B

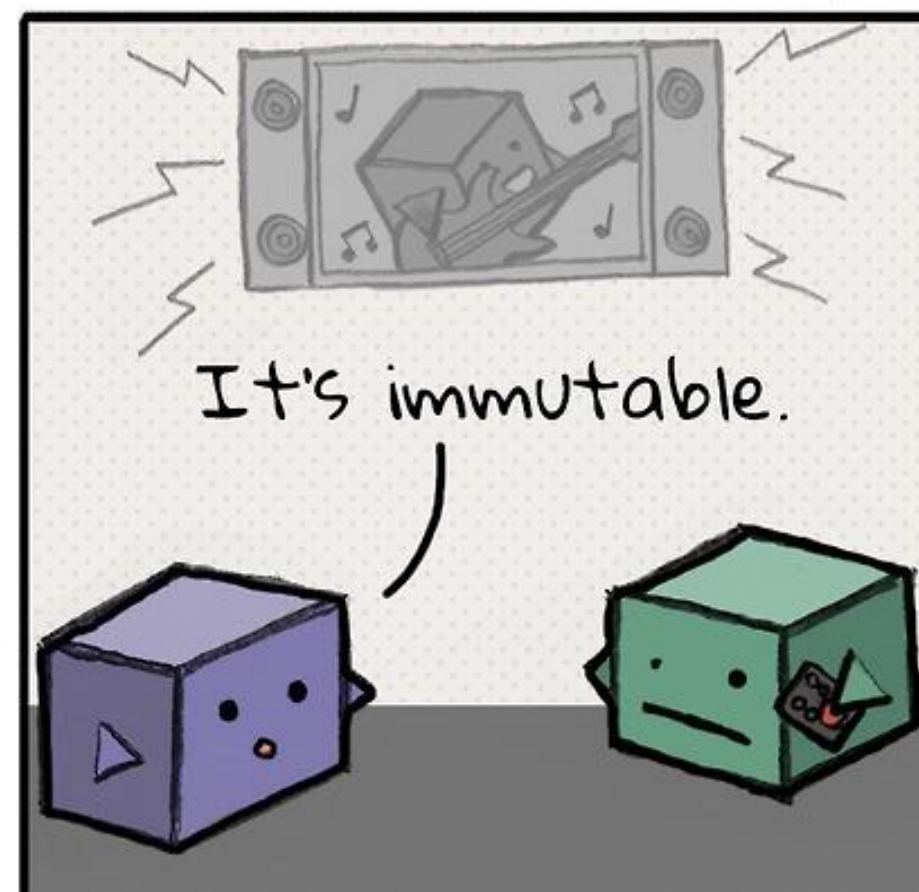
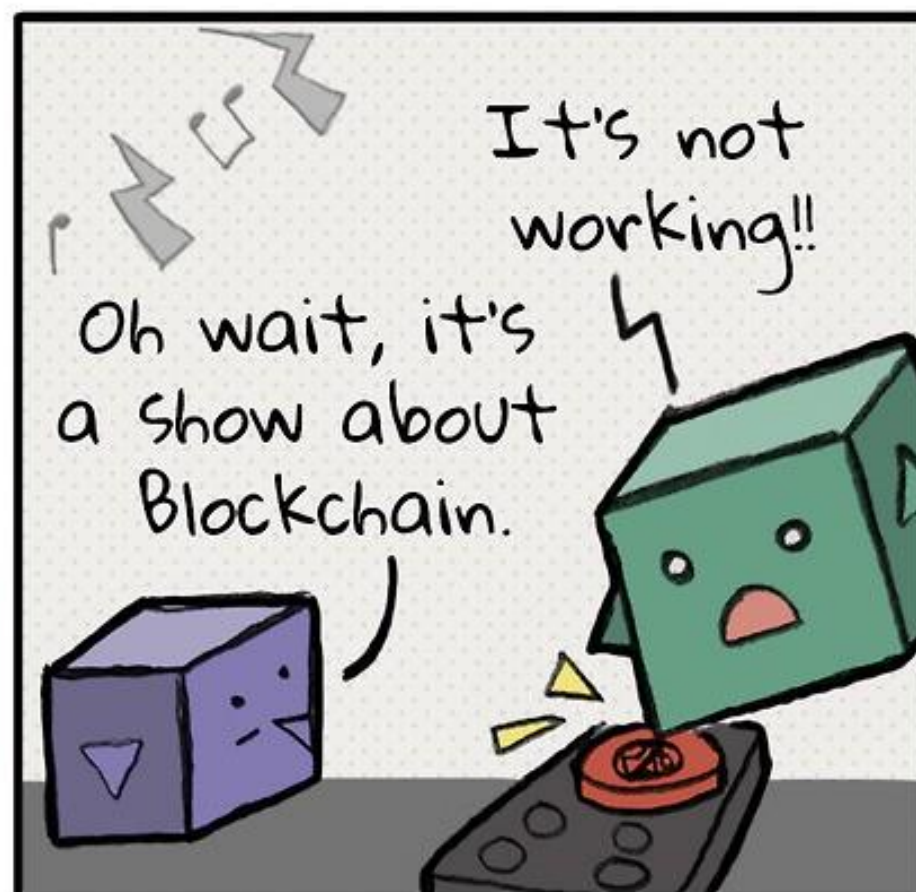
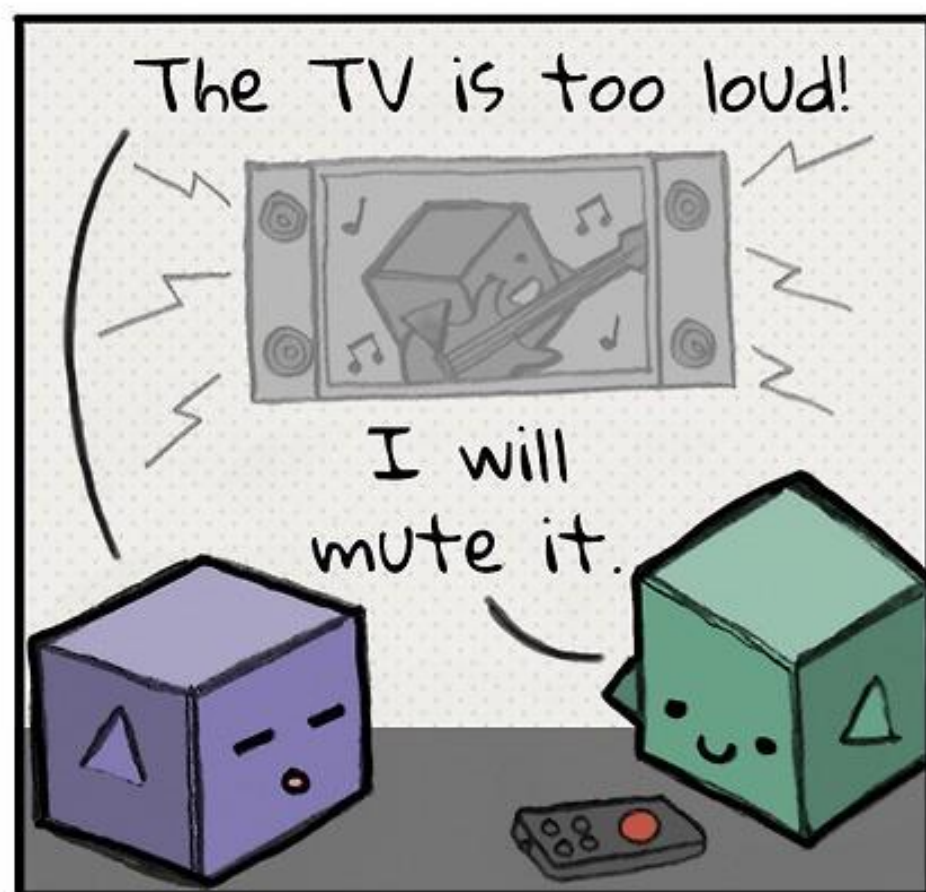


Veamos un ejemplo

<https://andersbrownworth.com/blockchain/>

CONGA COMICS

Block Height 1: "Loud Noises"



2

Tipos de blockchain

Tipos de blockchain



ethereum

corda



HYPERLEDGER



HYPERLEDGER
BESU



	Públicos Bitcoin, Ethereum, Litecoin	Privados Hyperledger, Corda, Quorum	Federados Hyperledger, Corda, Quorum	Blockchain as a Service IBM, Microsoft, Amazon
Cualquiera puede participar	✓	✗	✗	NA
Los participantes actúan, en general, como nodos	✓	✗	✗	NA
Transparencia	✓	≈	≈	NA
Hay un único administrador	✗	✓	✗	NA
Hay más de un administrador	✗	✗	✓	NA
No hay administradores	✓	✗	✗	NA
Ningún participante tiene más derechos que los demás	✓	✗	✗	NA
Se pueden implementar Smart Contracts	✓	✓	✓	NA
Existe recompensa por minado de bloques	≈	✗	✗	NA
Soluciona problema de falta de confianza	✓	✗	≈	NA
Seguridad basada en protocolos de consenso	✓	✗	≈	NA
Seguridad basada en funciones hash	✓	≈	≈	NA
Provee servicios en la nube	NA	NA	NA	✓

✓ Sí ✗ No ≈ A veces NA No Aplica

Quorum



J.P.Morgan



CONSENSYS

- Blockchain de Ethereum centrada en la empresa. Creado por J.P.Morgan.
- Código abierto.
(<https://github.com/jpmorganchase/quorum>).
- Algoritmo de consenso: RAFT.
- Smart Contracts: Solidity.



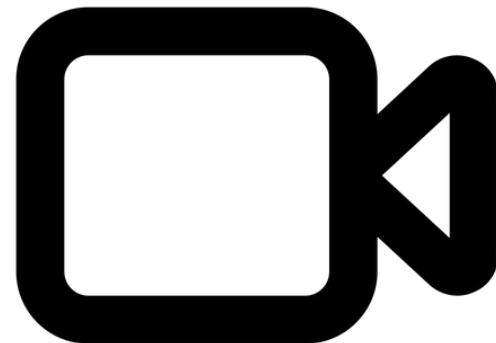
Quorum

3

Levantando un nodo en Quorum

Demo

Quorum Wizard: Herramienta de línea de comandos que permite crear redes Quorum para desarrollo de manera sencilla.



Fuente: <https://github.com/ConsenSys/quorum-wizard>

4

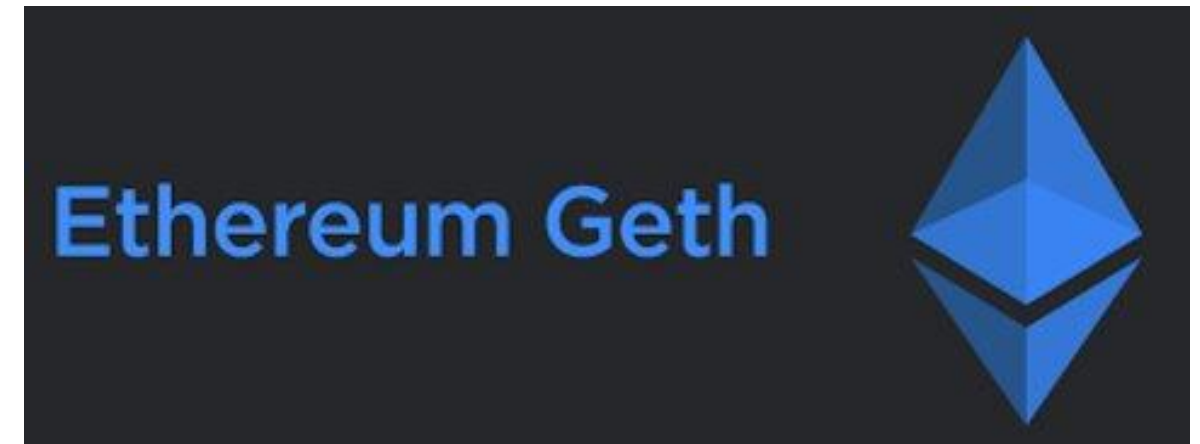
Geth sobre un nodo

Geth

Consola que expone el API de web3 en JavaScript.

Comandos útiles:

- `geth attach http://IP:PORT`
- `eth.accounts` → Consultar cuentas existentes.
- `personal.newAccount("password")` → Nueva cuenta
- `Personal.unlockAccount("address", "password", segundos)` → Desbloquear cuenta
- `admin-peers` → Consultar datos de los nodos.
- `eth.getTransaction("transactionHash")` → Consulta transacción.



5

Truffle, Ganache, Solidity y Remix-IDE

Truffle & Ganache

- Framework más popular para Ethereum.
- Permite conectarse a cualquier red blockchain que utilice Ethereum Virtual Machine (EVM) (Entorno de ejecución de Smart Contracts en Ethereum)
- Herramienta que permite generar una blockchain local de Ethereum para realizar pruebas en la fase de desarrollo.
- Proporciona 10 cuentas por defecto con su correspondiente par de claves.



Solidity

- Lenguaje utilizado para la creación de Smart Contracs en Ethereum
- Lenguaje de tipado estático.
- Cada tipo de variable tiene que ser especificada en tiempo de compilación.

```
pragma solidity ^0.4.25;

contract HelloWorld {

    string public hello;
    uint256[] private array;

    constructor() public {
        hello = "Hello World!";
    }

    event pushEvent(address caller, string functionName,uint256 element, uint256 timestamp);

    function push(uint256 element) public {
        emit pushEvent(msg.sender, "push", element, block.timestamp);
        array.push(element);
    }

    function getArray() public view returns (uint256[]){
        return array;
    }

}
```



SOLIDITY

Remix-IDE

- **Ethereum-IDE que permite desarrollar, compilar y desplegar Smart Contracts de manera más cómoda sobre un nodo de la blockchain.**
- **Se puede acceder desde una página web o levantarlo en local.**





Proyecto de ejemplo

<https://github.com/VictorNS69/Try-chain>

6

¿Blockchain para todo?

¿Cuándo usar blockchain?

Es crítico entender que blockchain pese a ser una tecnología muy prometedora, no puede cubrir todas las necesidades de negocio.

En definitiva, blockchain es vital para hacer los procesos más eficientes, con mayor confianza y seguros.



Casos de uso para blockchain

Record keeping: storage of static information

Transactions: registry of tradeable information



1 Static registry

- Distributed database for storing reference data

- Example**
- Land title
 - Food safety and origin
 - Patent



2 Identity

- Distributed database with identity-related information
- Particular case of static registry treated as a separate group of use cases due to extensive set of identity-specific use cases

- Example**
- Identity fraud
 - Civil-registry and identity records
 - Voting



3 Smart contracts

- Set of conditions recorded on a blockchain triggering automated, self-executing actions when these predefined conditions are met

- Example**
- Insurance-claim payout
 - Cash-equity trading
 - New-music release



4 Dynamic registry

- Dynamic distributed database that updates as assets are exchanged on the digital platform

- Example**
- Fractional investing
 - Drug supply chain



5 Payments infrastructure

- Dynamic distributed database that updates as cash or cryptocurrency payments are made among participants

- Example**
- Cross-border peer-to-peer payment
 - Insurance claim

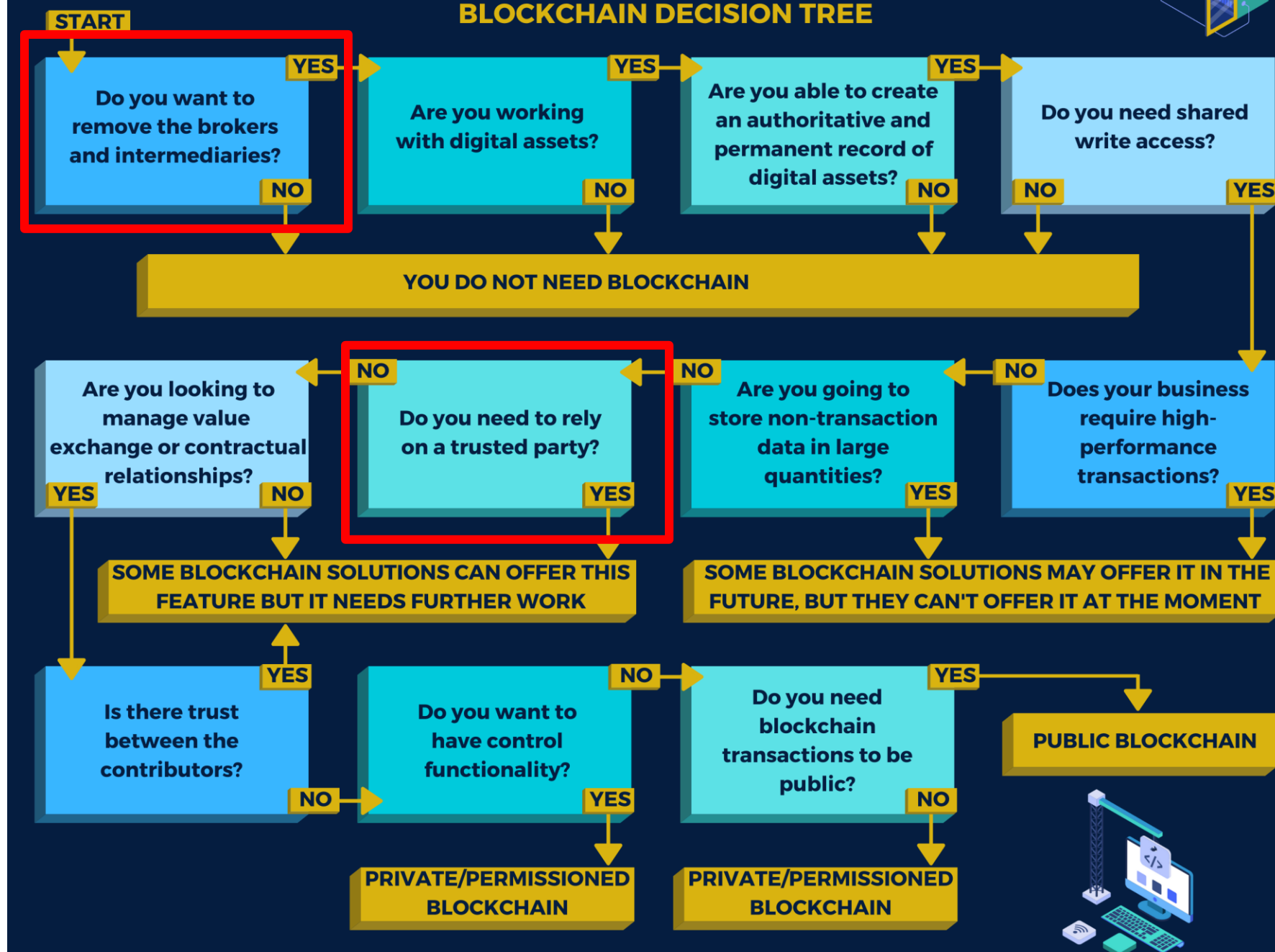
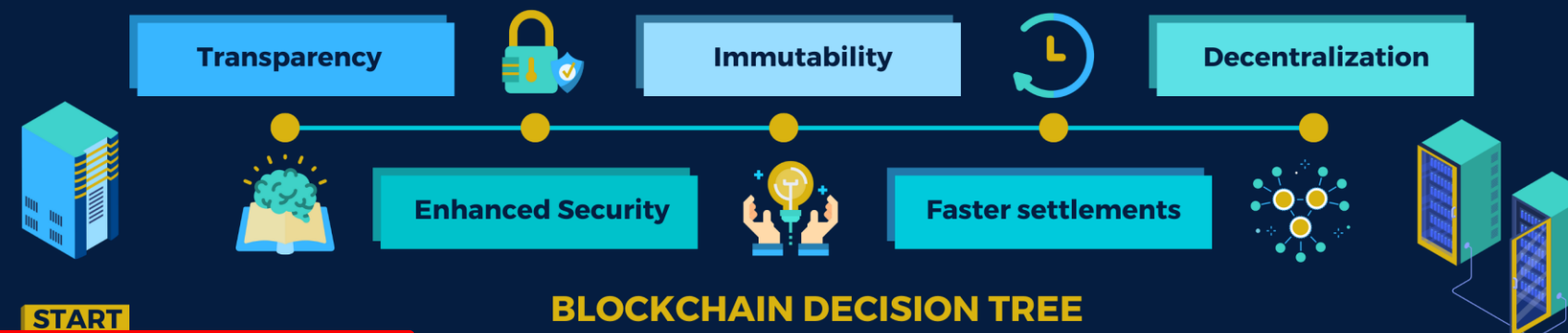


6 Other

- Use case composed of several of the previous groups
- Standalone use case not fitting any of the previous categories

- Example**
- Initial coin offering
 - Blockchain as a service

KEY FEATURES OF A BLOCKCHAIN-POWERED NETWORK



TRY-CHAIN!



inetum.world

FRANCE | SPAIN | PORTUGAL | BELGIUM | SWITZERLAND | LUXEMBOURG | ENGLAND |
POLAND | ROMANIA | MOROCCO | TUNISIA | SENEGAL | CÔTE D'IVOIRE | ANGOLA |
CAMEROON | USA | BRAZIL | COLOMBIA | MEXICO | RP OF PANAMA | PERU | CHILI |
COSTA RICA | DOMINICAN REPUBLIC | ARGENTINA | SINGAPORE | UAE

