BLOCKCHAIN.GAL

Anatomía de los Smart Contracts

CIBTC A Coruña — 14-15 de diciembre de 2018

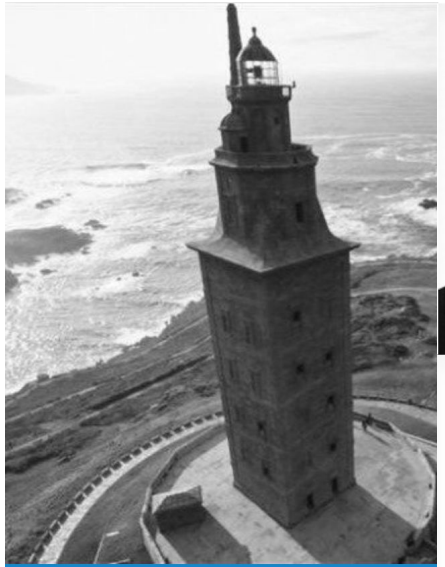
Luís M. Louzao González
Rodrigo Martínez Castaño

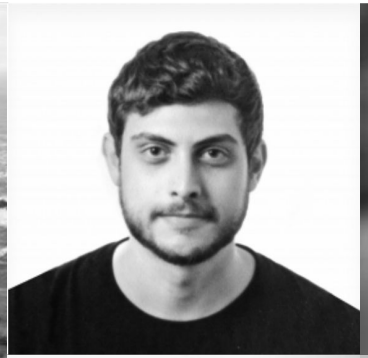
A Coruña CIBTC

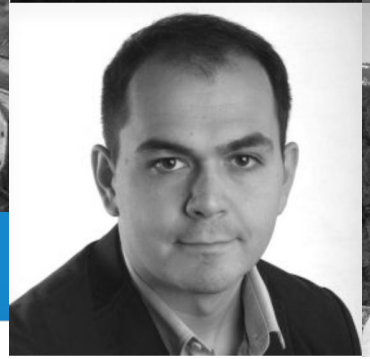

Santiago

BLOCKCHAIN.GAL

A Coruña CIBTC





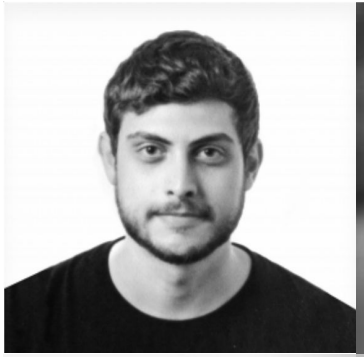


A Coruña

Vigo

BLOCKCHAIN.GAL

A Coruña CIBTC



BLOCKCHAIN.GAL

A Coruña CIBTC

@brunn3is





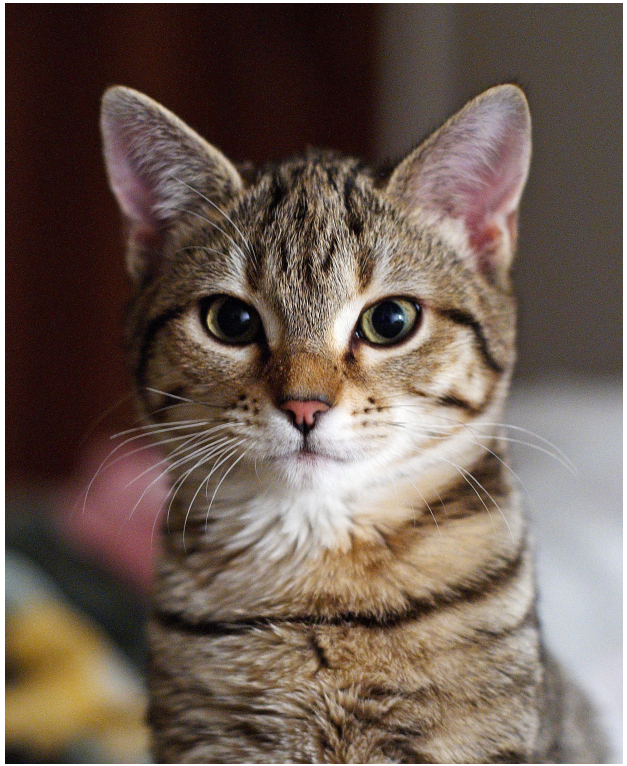


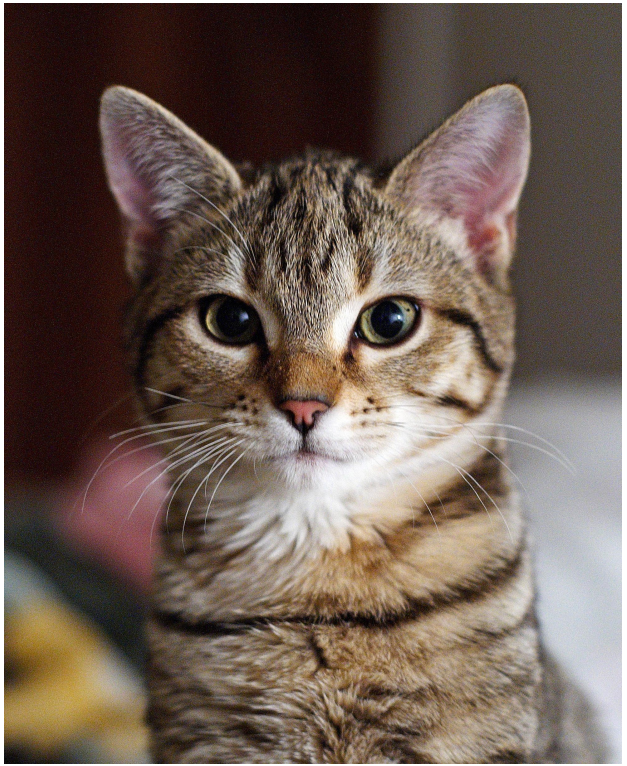
@_lou_z

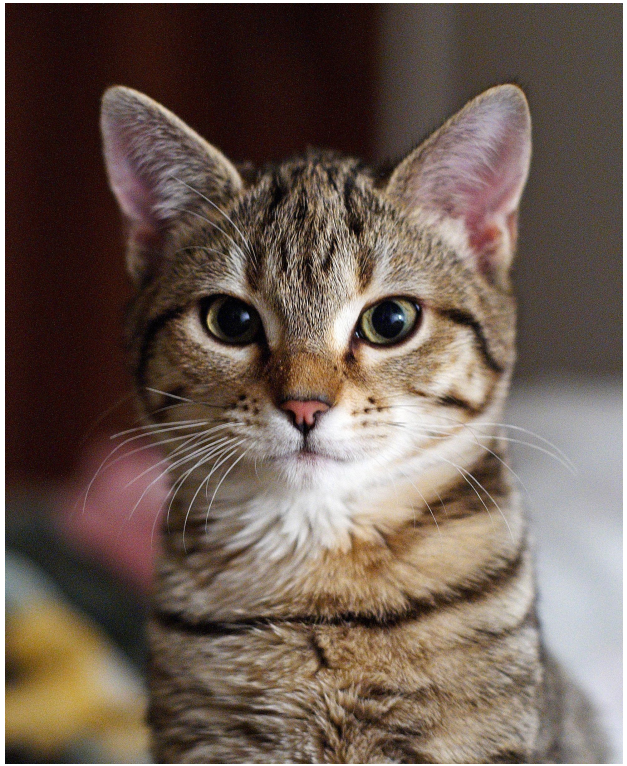
1 Conceptos previos

Función hash criptográfica

- Mensaje de cualquier tamaño se transforma en una representación de **tamaño fijo**
- **No** se puede **recuperar el mensaje** a través de su resumen
- Es **inviable** tratar de obtener el **mismo resumen con** dos **mensajes diferentes**
- La más **mínima variación** del mensaje produce un **resumen** totalmente **distinto**
- Ante un **mismo mensaje siempre** se obtiene el **mismo resumen**







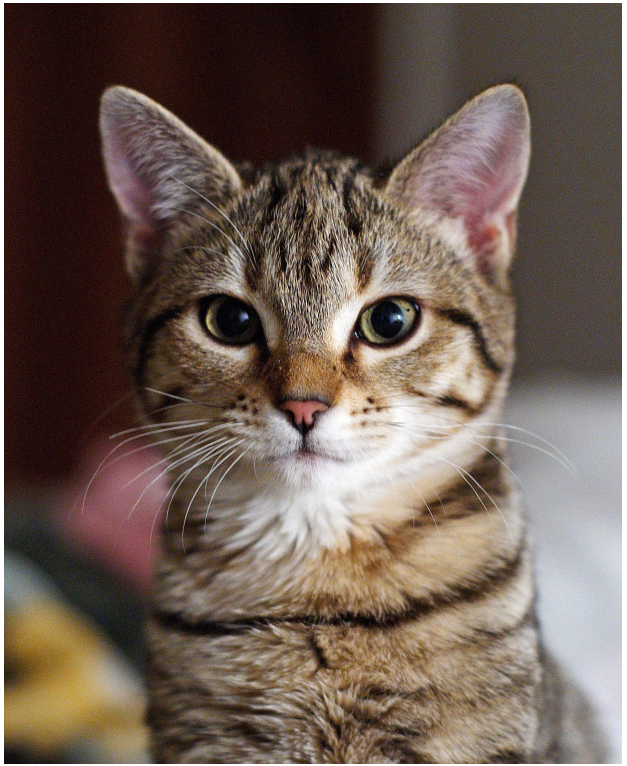
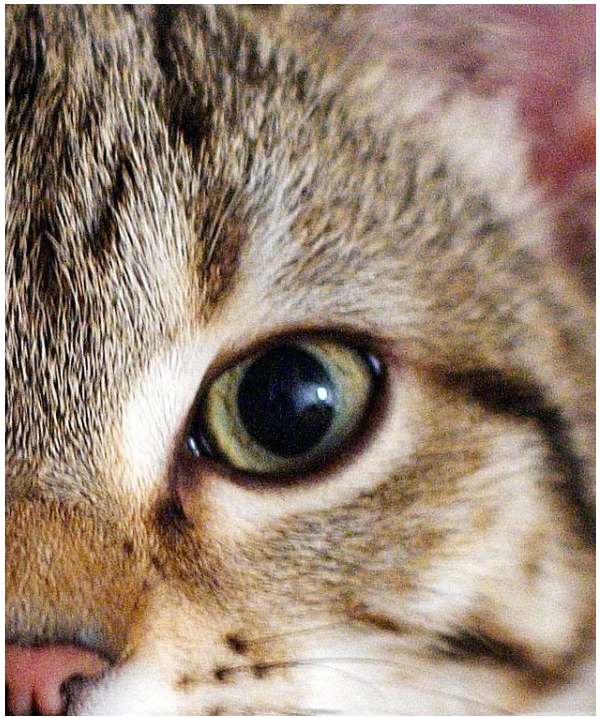


Figura 1:
0xd0bccf73ff5b015be0665c7a5924fbd44edf7d08044da92ca44189f9cbd7d026
Figura 2:
0x8ee6a5c6a914feb457a1af7e12a074735fa6ceb2439025863e58dee8341efd63



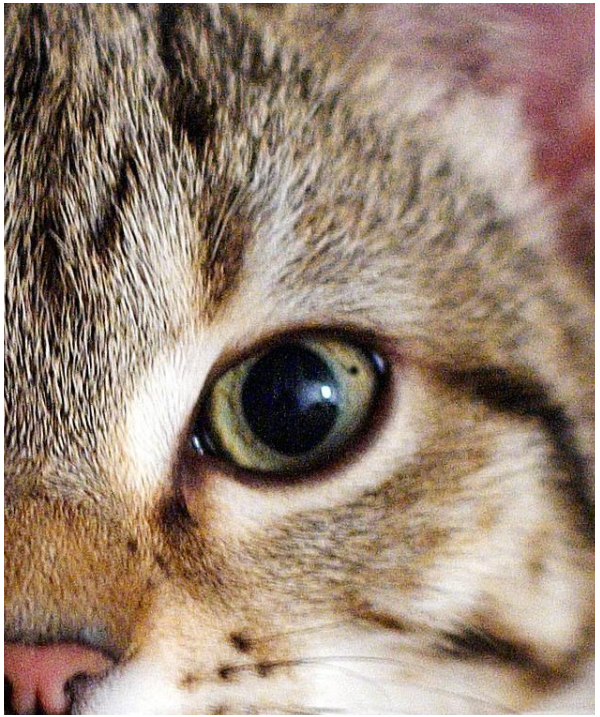


Figura 1:
0xd0bccf73ff5b015be0665c7a5924fbd44edf7d08044da92ca44189f9cbd7d026
Figura 2:
0x8ee6a5c6a914feb457a1af7e12a074735fa6ceb2439025863e58dee8341efd63

Firma digital con criptografía asimétrica

- Dos claves relacionadas matemáticamente: **pública y privada**
- Para garantizar el origen de un mensaje el emisor lo firma con su clave privada (**firma digital**)
- **Cualquiera puede verificar** que el contenido procede del emisor (obtener su clave pública) a través del contenido original y su firma.

1 Blockchain

Blockchain

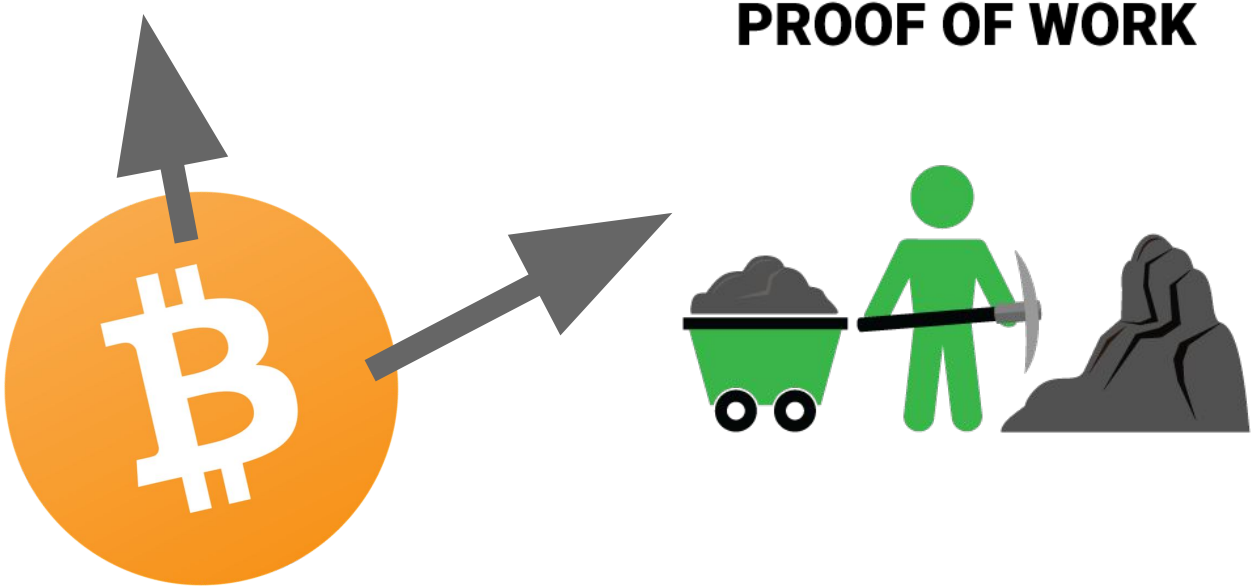
- Estructura de datos utilizada para construir bases de datos **distribuidas sin autoridad central**
- **Registros** secuenciales **agrupados** en **bloques**
- Se forma una **cadena**, cada **bloque** incluye una **referencia al** bloque **anterior** de forma recursiva
  - Hash
- La cadena de bloques se consensúa de forma distribuida a través de un protocolo de consenso (PoW, PoS...)

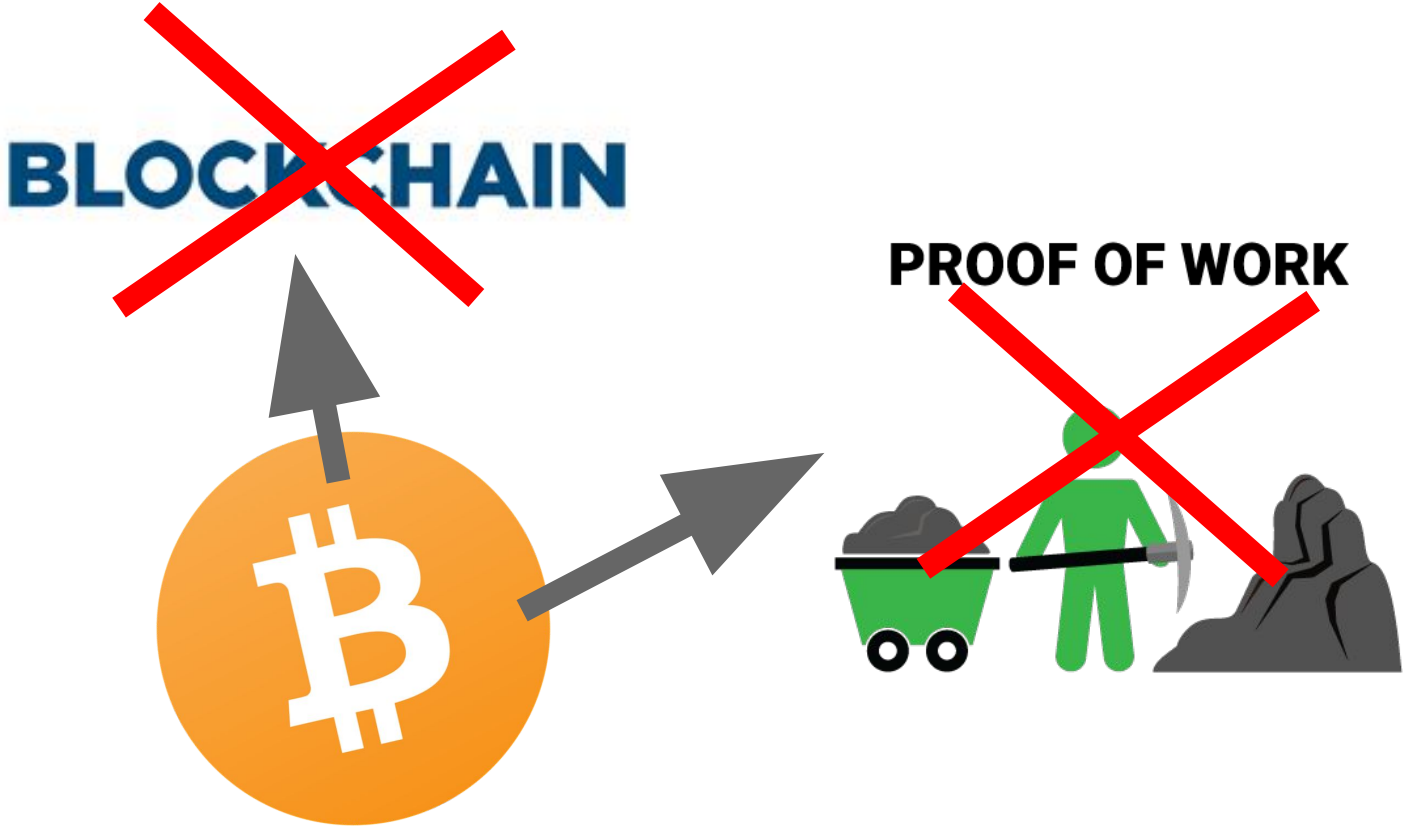
Blockchain

- Los **registros no pueden alterarse o eliminarse** una vez añadidos
- Están **firmados** digitalmente (**autenticación**)

Satoshi Nakamoto

BLOCKCHAIN



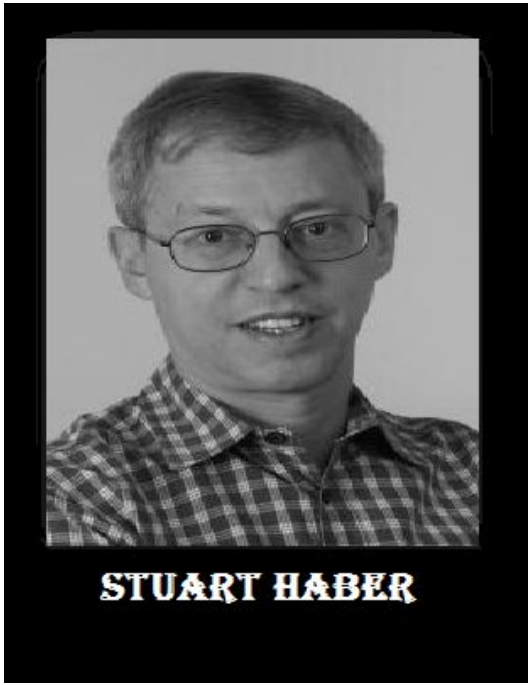

BLOCKCHAIN
PROOF OF WORK

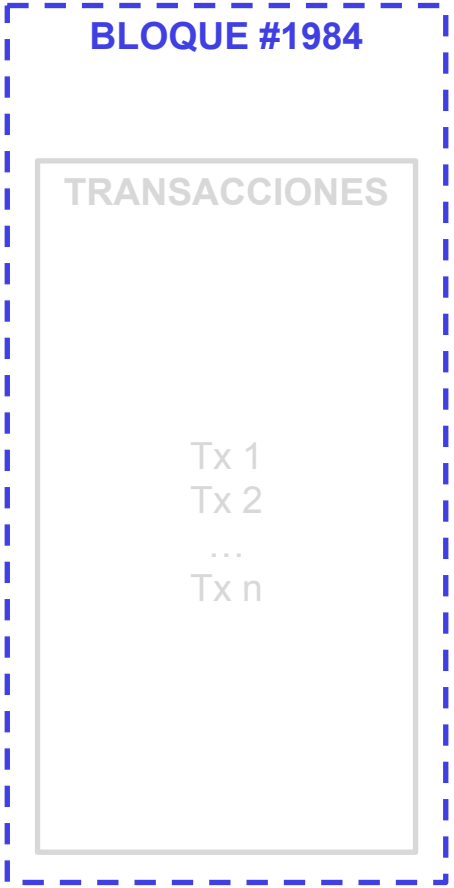
BLOCKCHAIN

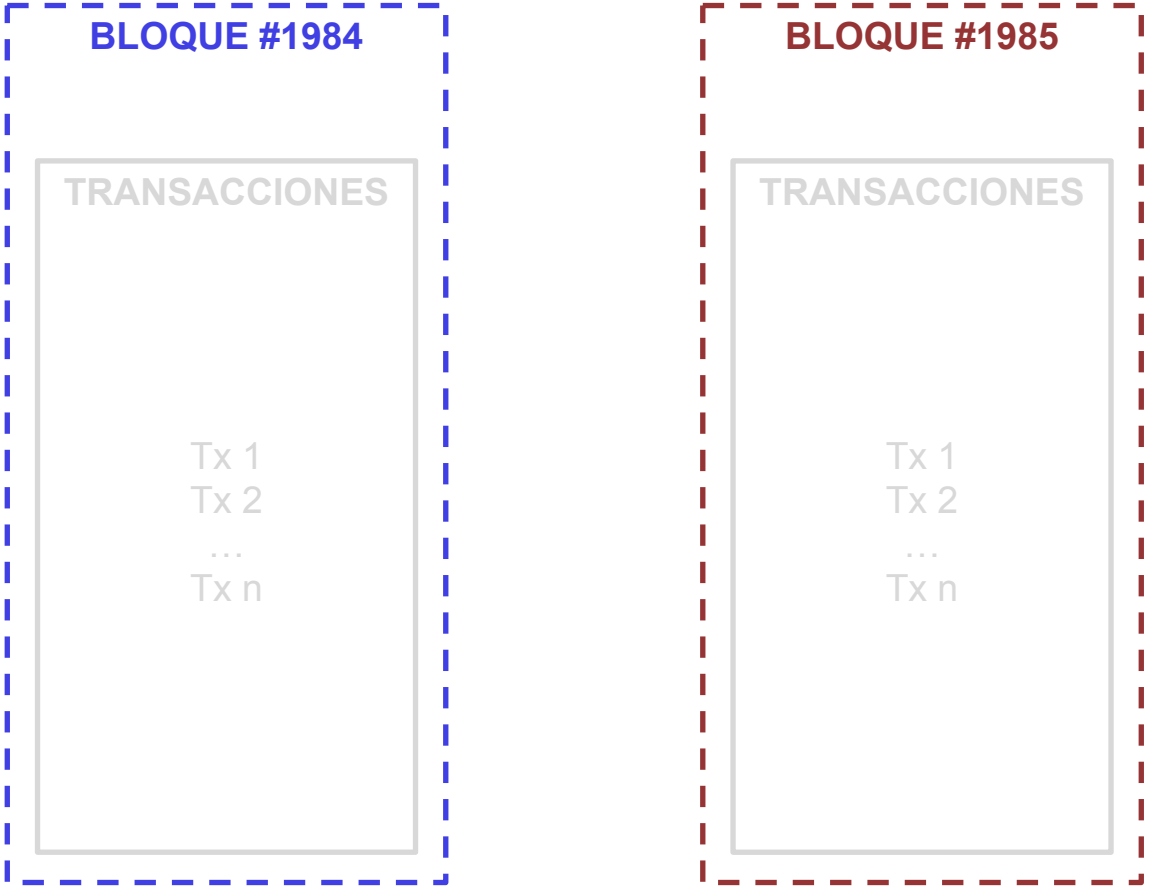
SINCE

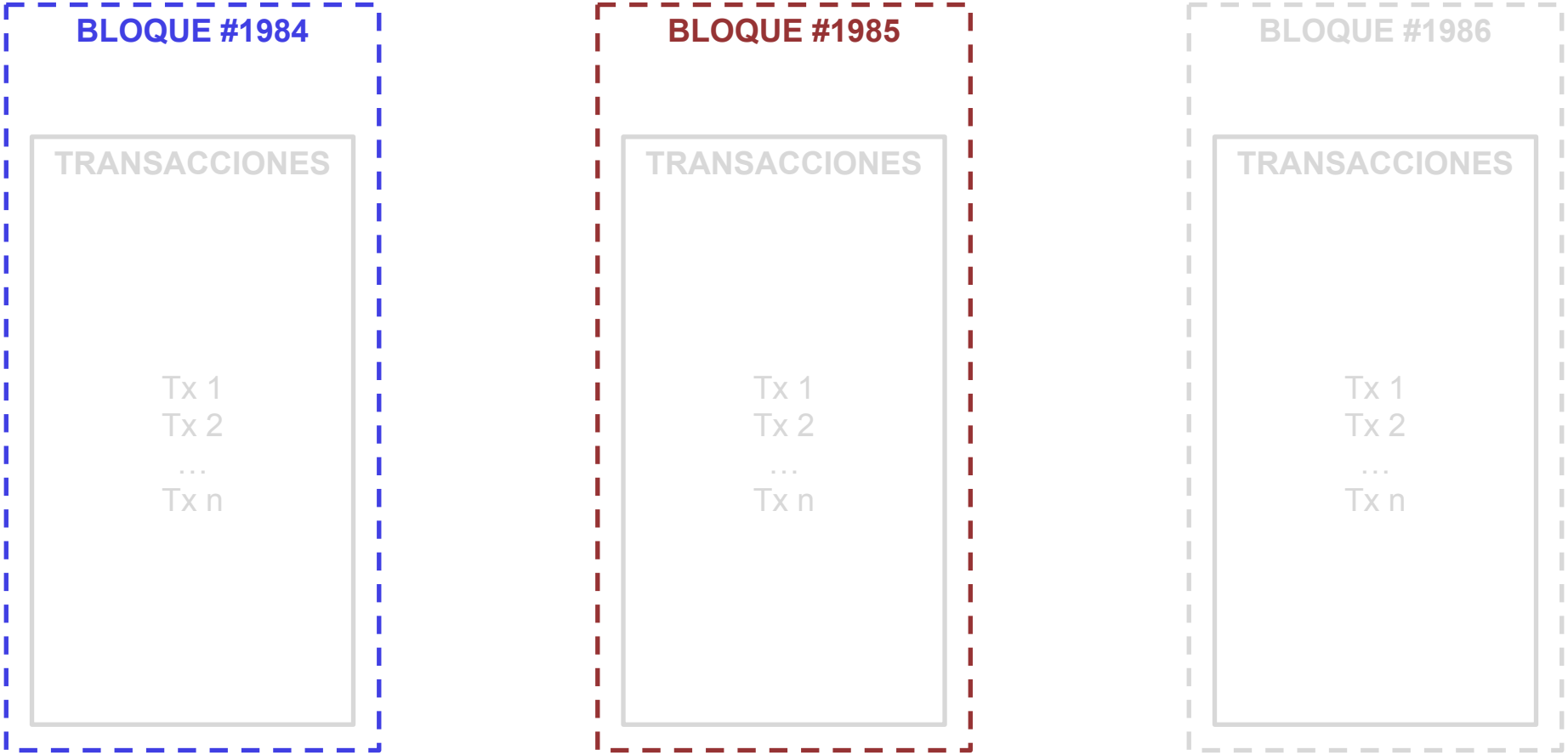
★ 1990 ★

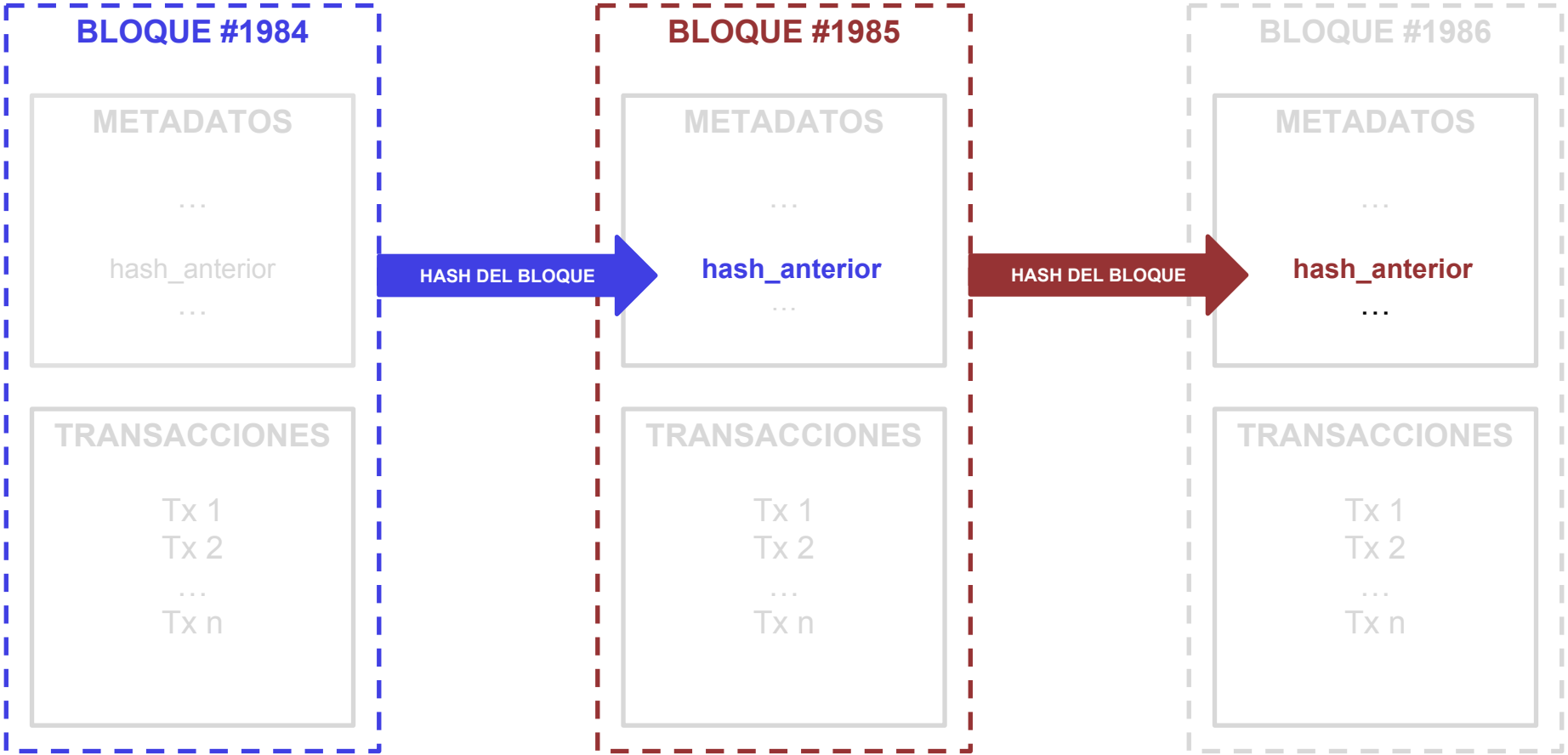

W. SCOTT STORNETTA

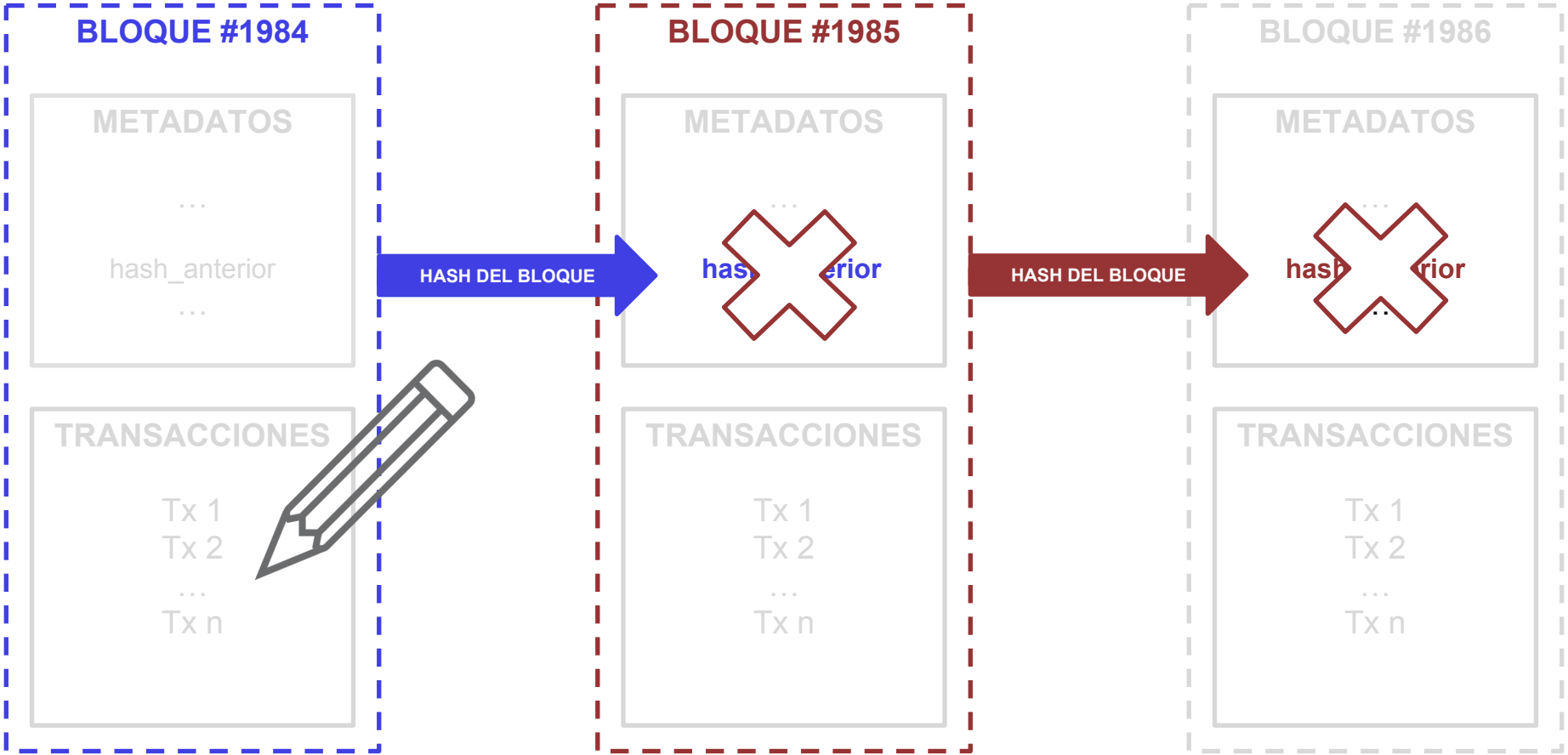

STUART HABER


BLOQUE #1984
TRANSACCIONES
Tx 1
Tx 2
...
Tx n


BLOQUE #1984
TRANSACCIONES
Tx 1
Tx 2
...
Tx n
BLOQUE #1985
TRANSACCIONES
Tx 1
Tx 2
...
Tx n


BLOQUE #1984
TRANSACCIONES
Tx 1
Tx 2
...
Tx n
BLOQUE #1985
TRANSACCIONES
Tx 1
Tx 2
...
Tx n
BLOQUE #1986
TRANSACCIONES
Tx 1
Tx 2
...
Tx n


BLOQUE #1984
METADATOS
...
hash_anterior
...
TRANSACCIONES
Tx 1
Tx 2
...
Tx n
HASH DEL BLOQUE
BLOQUE #1985
METADATOS
...
hash_anterior
...
TRANSACCIONES
Tx 1
Tx 2
...
Tx n
HASH DEL BLOQUE
BLOQUE #1986
METADATOS
...
hash_anterior
...
TRANSACCIONES
Tx 1
Tx 2
...
Tx n

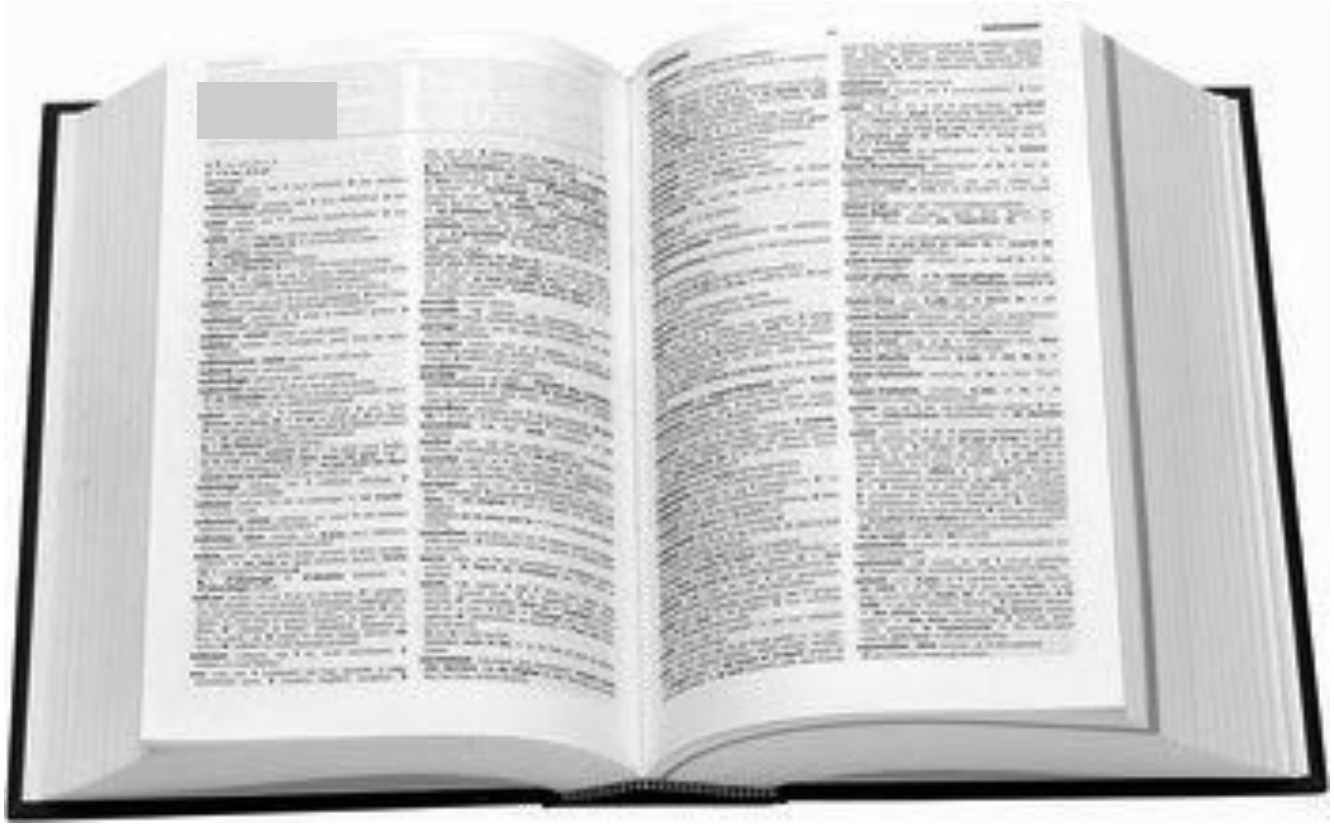

BLOQUE #1984
METADATOS
...
hash_anterior
...
TRANSACCIONES
Tx 1
Tx 2
...
Tx n
HASH DEL BLOQUE
BLOQUE #1985
METADATOS
...
TRANSACCIONES
Tx 1
Tx 2
...
Tx n
HASH DEL BLOQUE
BLOQUE #1986
METADATOS
...
TRANSACCIONES
Tx 1
Tx 2
...
Tx n

PROOF OF WORK

Cynthia Dwork & Moni Naor

Proof of Work | 1992


ADAM BACK
HASHCASH | 1997



Diccionario
2 páginas

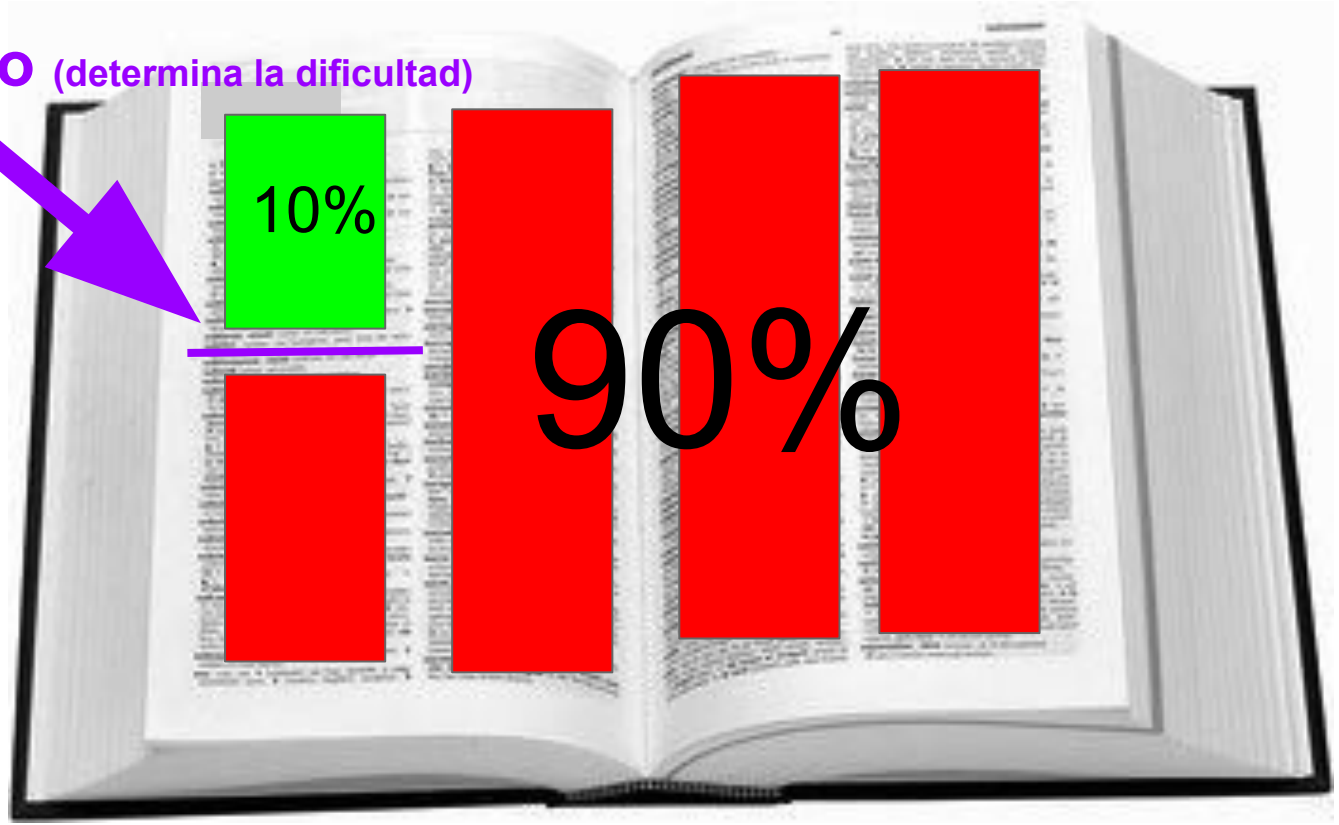

Todas las palabras


Todas las palabras
(orden alfabético)


Todos los hashes
(orden lexicográfico)

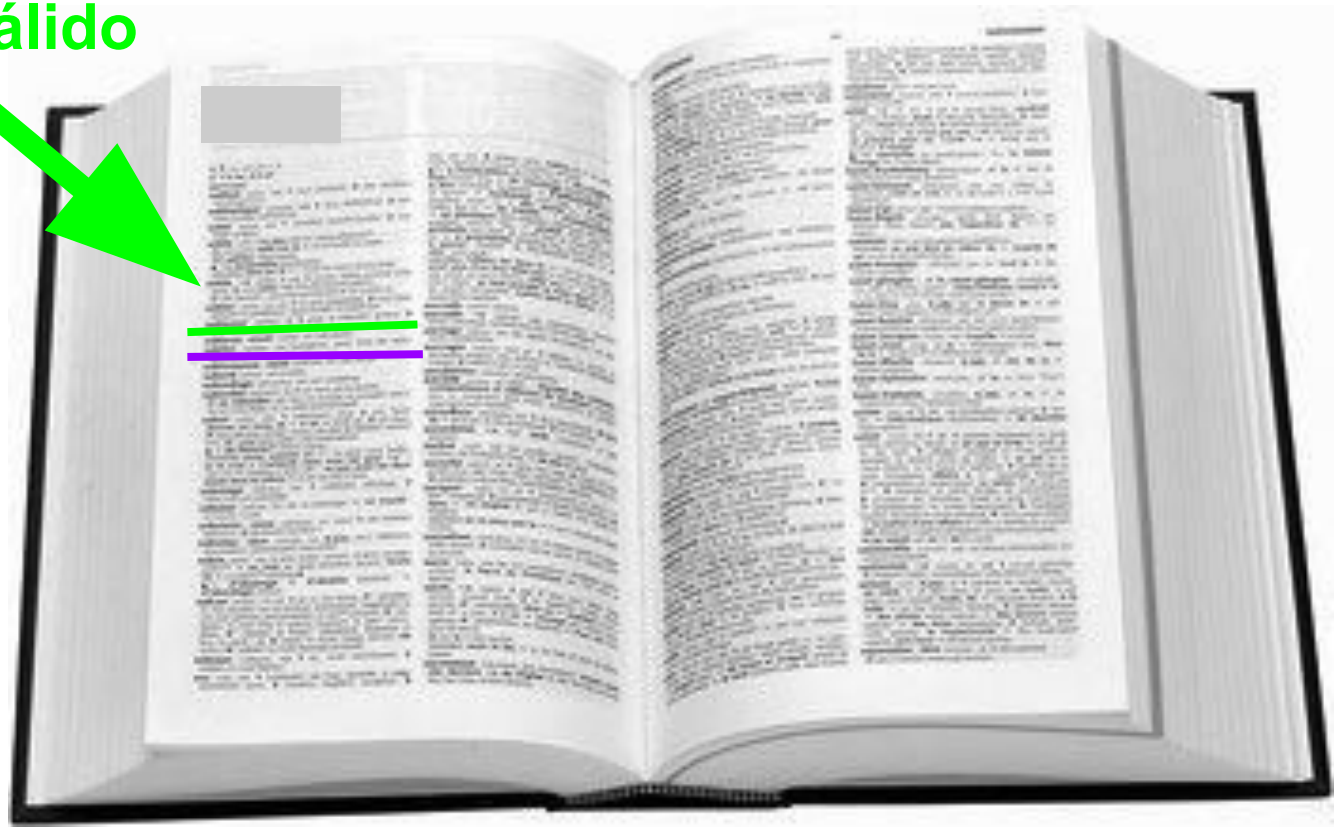
Objetivo (determina la dificultad)

Objetivo (determina la dificultad)



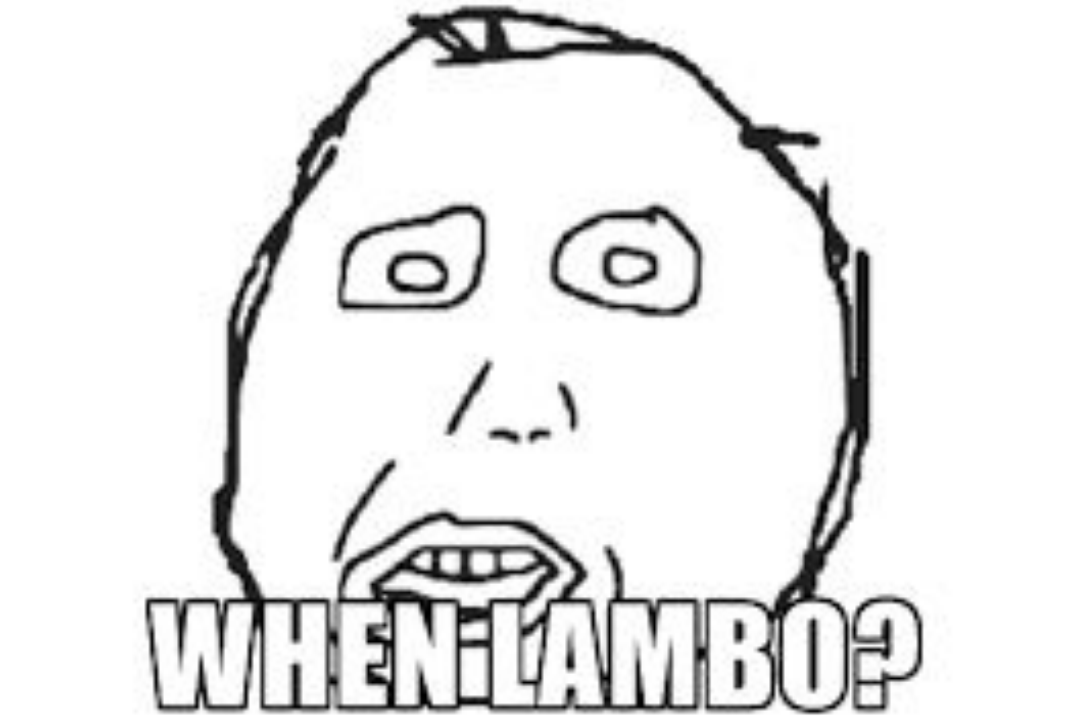
Objetivo (determina la dificultad)

Hash válido



Bitcoin

- **No existe** un proceso de **registro** de cuentas
- **Se generan de forma aleatoria**
- La **clave pública** se utiliza como **identificador** de la cuenta (dirección)
- **Para transferir** bitcoins es necesario **firmar** la transacción **con la clave privada** ligada a la cuenta
- Las **transacciones** firmadas **se propagan** por la red a los distintos nodos de forma progresiva


WHEN LAMBO?

2 Smart Contracts

Smart contracts

*"Un contrato inteligente es un **protocolo computarizado** que **ejecuta los términos de un contrato**. Los objetivos generales de su diseño son satisfacer condiciones contractuales, minimizar excepciones tanto maliciosas como accidentales, y **minimizar la necesidad de intermediarios de confianza**."*

Nick Szabo

Smart Contracts, 1994

Smart contracts

De **transacciones limitadas** (monetarias) a
transacciones de invocación de software (arbitrario)

Smart contracts

- **Software distribuido e inmutable** registrado en una blockchain.
- Definen **reglas y consecuencias estrictas** a las que los usuarios se someten al interactuar con ellos.
- Se **invocan** a través de **transacciones** (reactivos)
- **Pueden almacenar** el **estado** que puede alterarse a través de las invocaciones

Smart contracts

- Aplicaciones que se ejecutan exactamente como fueron programadas
  - Compilador, máquina virtual (xVM)
- El **código fuente puede ser contrastado** contra un software desplegado
- El usuario puede (debería) **ver y auditar las reglas y consecuencias** del software en el código

Smart contracts en Bitcoin

- Capacidad para crear smart contracts limitados
  - Pagos condicionados
- Operation codes en ScriptSig (opcode)
- Lightning network

Smart contracts

- **Plataformas** blockchain como Ethereum permiten realizar **no solo transacciones específicas**
  - **Despliegue** de software
  - **Invocación** de software
  - **Turing completo**
    - Teóricamente cualquier cómputo
    - No solo condiciones para desbloquear el saldo
    - Reactivos

Smart contracts

- Además de almacenar estados **pueden actuar como una cuenta automatizada**
  - Recibir y enviar fondos
  - Interacción con otros contratos

3 Casos de uso

Transmisión de valor “fungible”

- Registro de balances (token)
- Operaciones:
  - Transferencia de fondos
  - ...

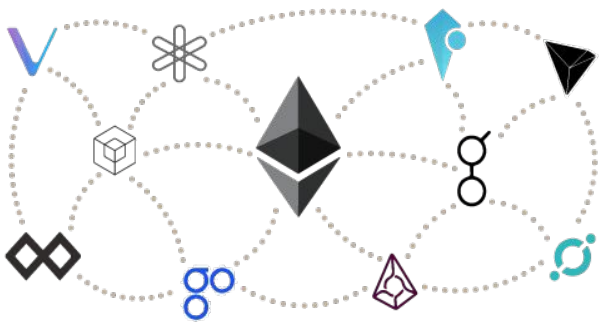


Imagen: https://decentralbox.com/wp-content/uploads/2018/03/erc20-token-liste.png

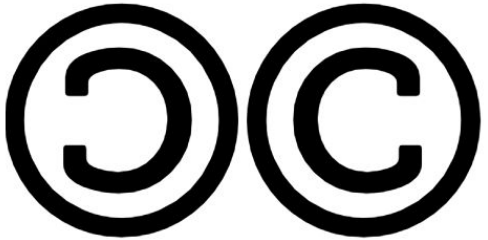
Transmisión de valor no fungible

- Los tokens tienen características diferentes
- Transmisión de propiedad
- Cada registro tiene propiedades únicas
- Pueden simbolizar un activo físico

Imagen: https://medium.com/@tarasca_dao/ana-santiso-primeira-artista-galega-que-registra-e-tokeniza-as-suas-obras-na-blockchain-71594137bfc1

Registro de información

- Registro de copyright
  - Metadatos
  - Hash -> Prueba de existencia (PoE)



Registro de trazabilidad

- Productos únicos
- Trazas
  - Conjunto de eventos asociados a un producto
- Alimentaria, historial académico, etc.

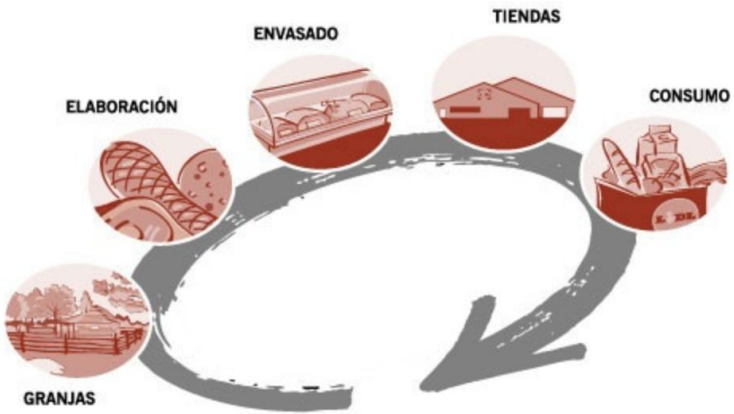


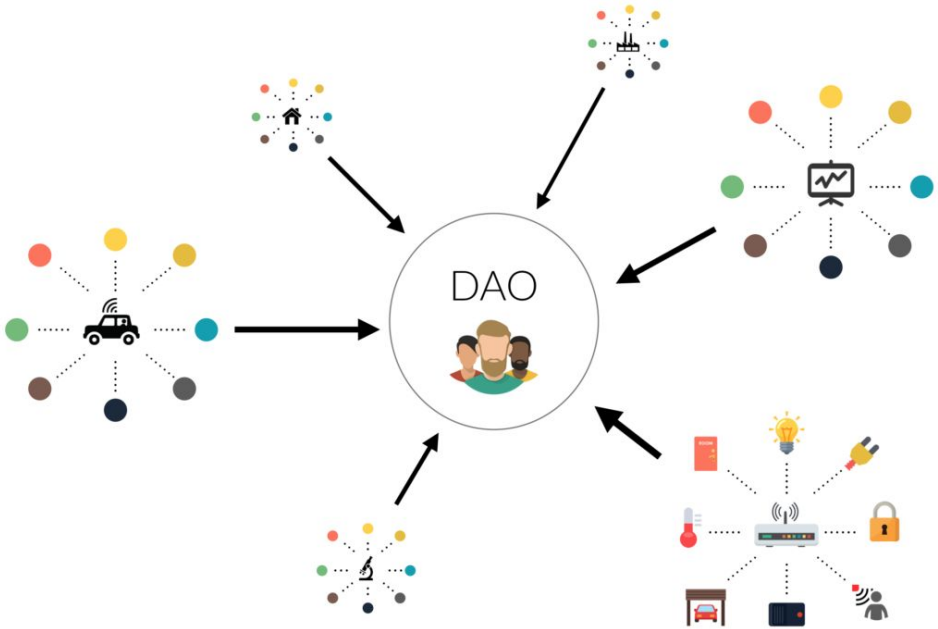
Imagen: http://blog.misfolletos.com/trazabilidad-de-los-alimentos.html

Gestión de identidad

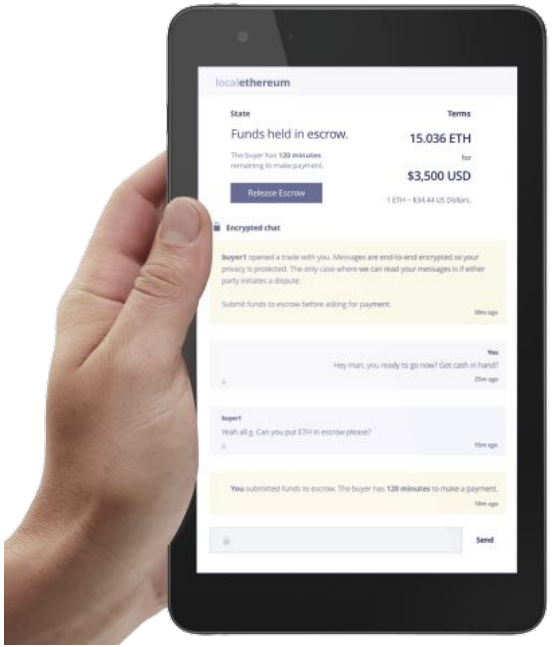
- Corroboraciones de autoridades y terceros
- Pérdida y recuperación de la identidad

ICO / Crowdfunding

DAO (Distributed Autonomous Organization)



Resolución de conflictos



localethereum

Ether's local private marketplace.

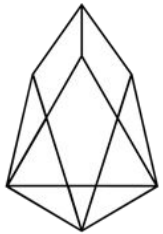
Oráculos

- Los smart contracts solo pueden comunicarse con otros smart contracts
- Se encuentran aislados
- Para acceder a información del mundo real alguien debe introducir la información en un smart contract
- Confianza en el **oráculo**

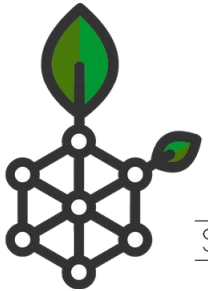
ethereum

NEO

Smart economy



EOS™



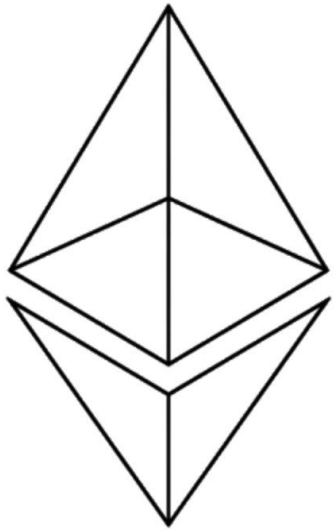
RSK

SMARTER BITCOIN

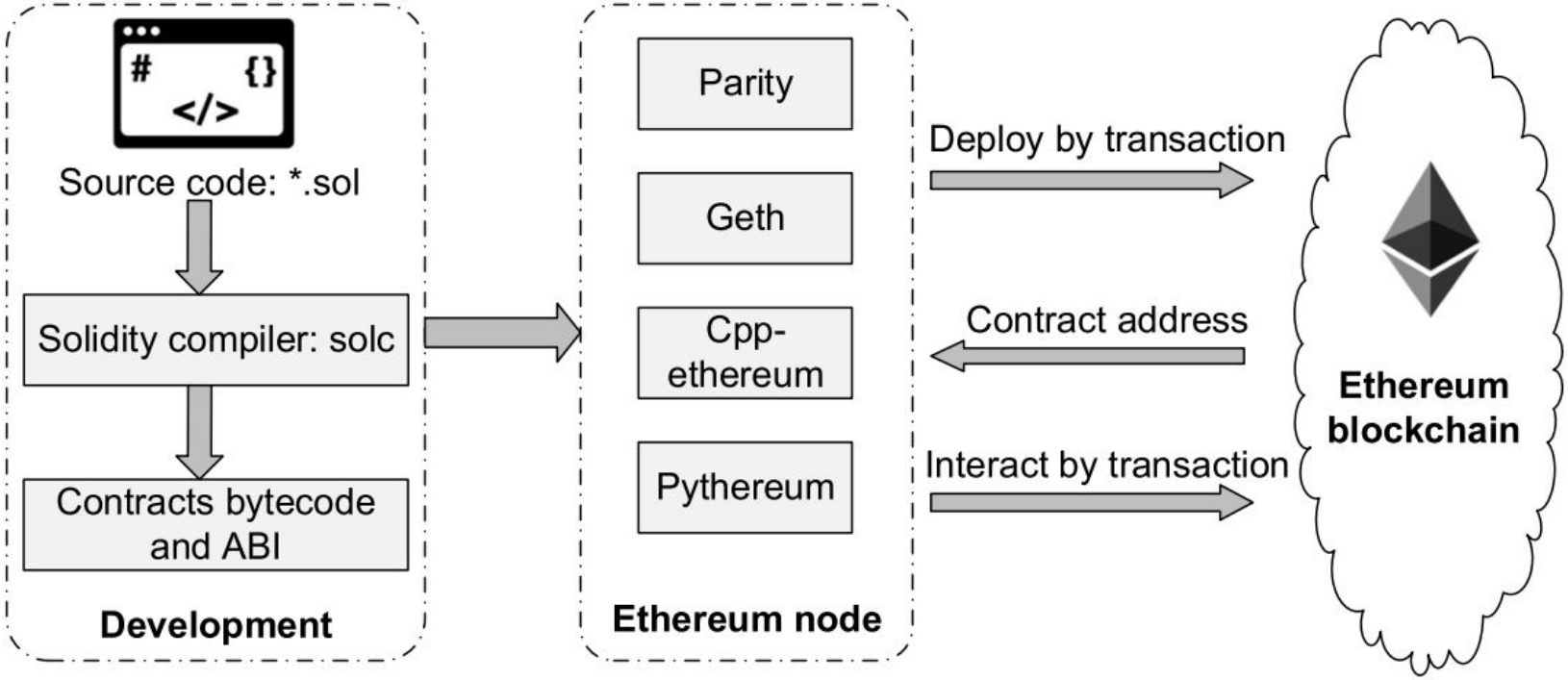
Lisk

ethereum

solc



web3{j, js, py}



https://www.sciencedirect.com/science/article/pii/S0167739X17318332?via%3Dihub

Interacción con un smart contract

- Cada smart contract tiene una interfaz ABI (Application Binary Interface)
- Mediante la ABI, se puede interactuar con un smart contract a través de un nodo mediante su interfaz JSON RPC
- 2 tipos de operaciones
  - Lectura (no hay transacción)
  - Escritura

Consideraciones finales

- Seguridad
  - Errores de programación
  - Cuentas comprometidas
- Actualización de Smart Contracts
  - Migraciones a nuevos *Smart Contracts*

Consideraciones finales

- Coste por número de transacciones por segundo
  - Especial atención en entornos IoT
- Coste almacenamiento datos (volumen alto)
  - Almacenar un resumen
- Blockchain privadas y permisionadas
- Privacidad y anonimato

5 **DEMO**

Demo Nivel Zero

- Desde Remix IDE hasta Ropsten (test)
- 2 contractos muy prácticos:
  - Poll: encuestas personalizadas
  - Copyright registry: autoría de obras
- Github: https://github.com/MrLouzao/cibtc-blochckaingal-workshop

Demo 1 - Chat basado en Blockchain

- Mensajes guardados en un Smart Contract
- Ineficiente: tiempo de espera (tx al contrato)
- Perfecto para ilustrar conceptos en una demo
- Github: https://github.com/MrLouzao/blockchain-gal-chat

Demo 1 - Chat basado en Blockchain

- Qué necesitamos?
  - Remix IDE (Solidity)
  - NodeJS + NPM
  - Truffle Framework
  - Bootstrap
  - Infura
  - Metamask


OKAY
LET'S
DO
THIS

Demo 2 - Registro de eventos

- Estados de un producto guardados en un Smart Contract
- Público y persistente
- Traza de estados de un producto
- Github:
  https://github.com/MrLouzao/cibtc-blochckaingal-workshop/tree/master/traceability_example

Demo 2 - Registro de eventos

- Qué necesitamos?
  - Remix IDE (Solidity)
  - NodeJS + NPM
  - Truffle Framework
  - Angular
  - Infura
  - Metamask

Demo 2 - Registro de eventos

- Qué necesitamos?
  - Remix IDE (Solidity)
  - NodeJS + NPM
  - Truffle Framework
  - Angular
  - Infura
  - Metamask


REGISTRATOR GAL
GOTTA SCAN'EM ALL!
imgflip.com

¡Gracias!

BLOCKCHAIN.GAL

A Coruña CIBTC