

TALLER DE BLOCKCHAIN

José Incera

Abril 2018

Agenda

Aspectos teóricos

- Introducción
- Bitcoin y su blockchain
- Descentralización y consenso
- Otros tipos de tecnologías de *ledger* distribuido (DLT)

Sesiones prácticas

- El blockchain de Bitcoin a través de un visualizador
- Multichain – DLT privado y permissionado
- Ethereum – Smart contracts
- Hyperledger – DLT para aplicaciones robustas

Confianza en el Mundo Digital: ¿Qué queremos?

Confidencialidad

- La información **no está disponible, accesible** o puede ser consultada **por individuos, entidades o procesos que no estén autorizados**



Integridad

- Tiene que ver con mantener y asegurar la precisión y completitud de la información a lo largo de todo su ciclo de vida. Esto quiere decir que **los datos no pueden ser modificados por alguien no autorizado o no detectado**



Disponibilidad

- La información tiene que **estar disponible cuando sea necesaria**



No-Repudiación

- Legalmente, la no-repudiación implica que **la parte transmisora de una transacción no puede negar el haberla enviado**



Confianza en el Mundo Digital – Algunos mecanismos

Confidencialidad

- **Encriptación.** Quien envía el mensaje lo “encripta” con **una llave** que solo el receptor deseado puede abrir



Integridad

- **Hash.** Una cadena (denominada “**hash**”) calculada con el contenido del mensaje original se utiliza para detectar si su contenido fue o no modificado en un tiempo posterior

Disponibilidad

- **Ledger replicado.** La información relevante se almacena en una cadena de bloques que es almacenada en muchos nodos

No-repudiación

- En el ecosistema de Bitcoin, la no-repudiación se logra por una **combinación de llave pública y llave privada**, y también porque el **registro está distribuido** y puede ser consultado y validado por todos (**transparencia**)

¿Qué es Bitcoin? / ¿Qué la distingue?

- Bitcoin es una **criptodivisa**.
 - En 2008, Satoshi Nakamoto (¿pseudónimo?) publica un artículo en la lista de criptografía de metzdowd.com donde describe el protocolo Bitcoin
 - Nakamoto Inicia el primer bloque (bloque génesis) con 50 Bitcoins
 - Primera transacción comercial: 2 pizzas de Papa John's por 10,000 BTC
- Bitcoin se caracteriza por funcionar con un **esquema descentralizado**, es decir, no está respaldado por ningún gobierno ni depende de la confianza en un emisor central
- Utiliza un sistema de **prueba de trabajo** para **impedir el doble gasto** y alcanzar el **consenso entre todos los nodos** que integran la red
- La información se intercambia sobre una **red no confiable y potencialmente comprometida**
- Las transacciones **no necesitan de intermediarios** (P2P) y el protocolo es código abierto



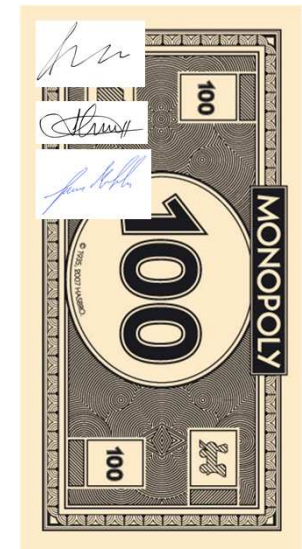
Bitcoin – Abstracción ¿Recuerdas el Turista?



¿Recuerdas hacer una pequeña trampa al jugarlo?

Bitcoin - Idea básica

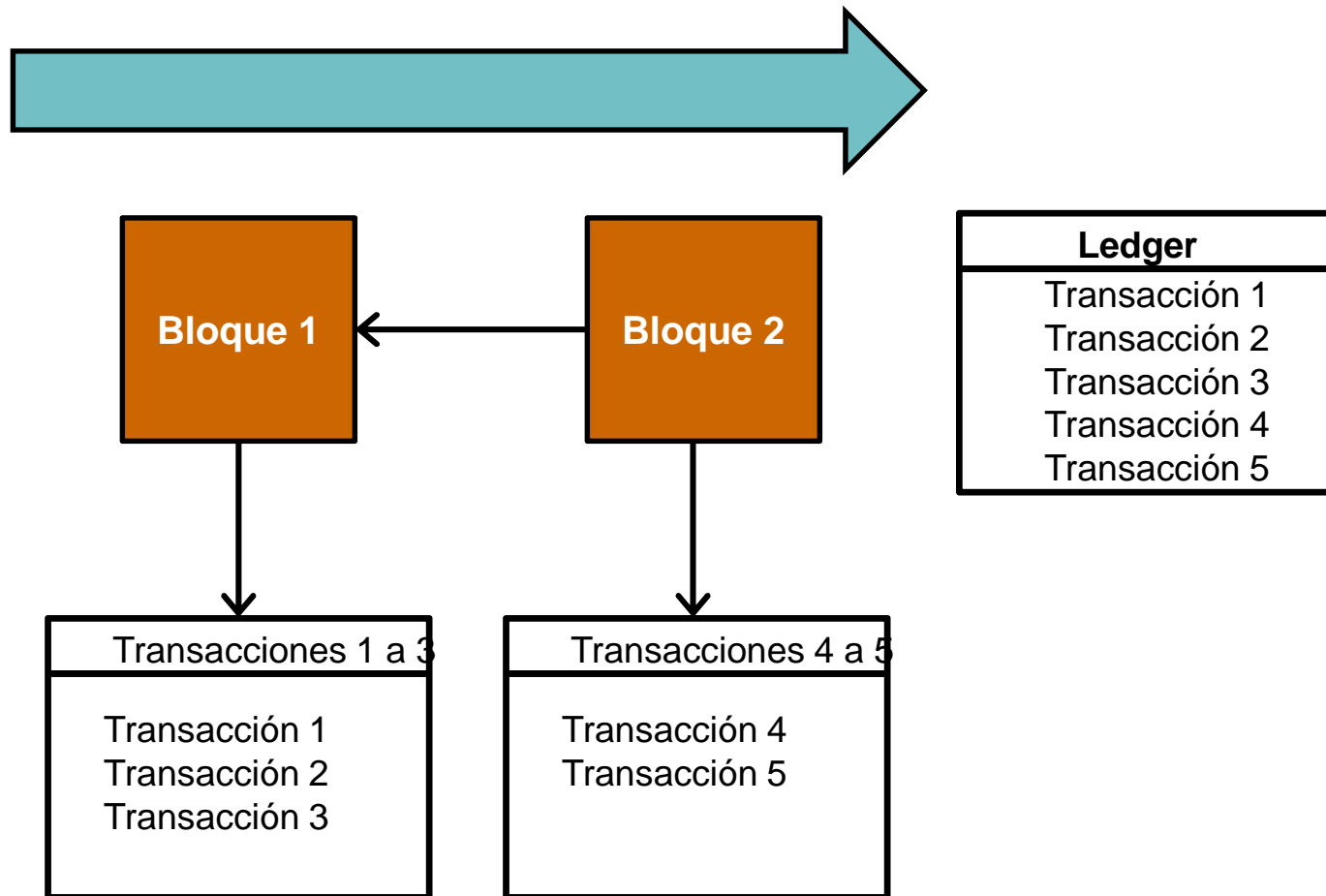
- En vez de tener *cuentas* en bancos y *liquidaciones* entre bancos (tal vez a través de un intermediario, como un banco central) cuando las transferencias de dinero (o de un *activo*) se ejecutan en Bitcoin – **en un blockchain**- los activos hacen referencia a cómo han ido cambiando de propietario
- Estas transacciones no pueden ser borradas ni modificadas



Conceptos principales

- *Ledger* distribuido (replicado)
 - No hay una autoridad central (si no se desea)
 - No hay un punto único de falla o de vulnerabilidad
 - Es muy difícil alterar los registros. La visión de la verdad es compartida
 - *World state* mantenido por varios miembros
 - Red Peer-to-peer
- Transacción
 - Operación atómica
 - Las transacciones se agrupan en bloques
 - Los bloques se agregan a la cadena. Nunca pueden ser eliminados ni alterados

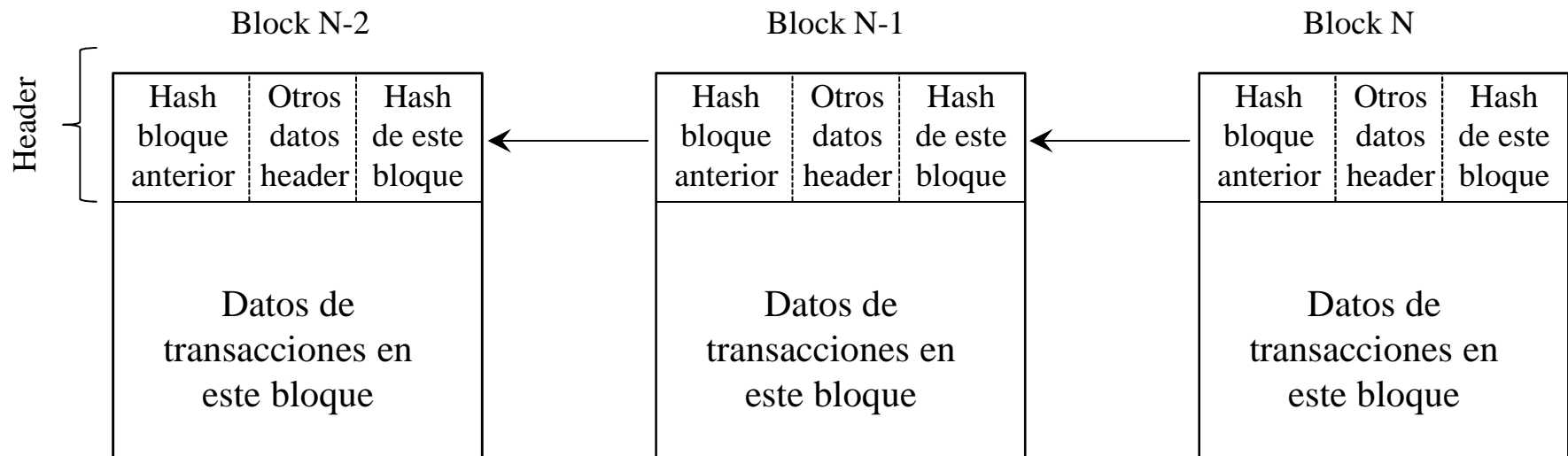
Conceptos principales



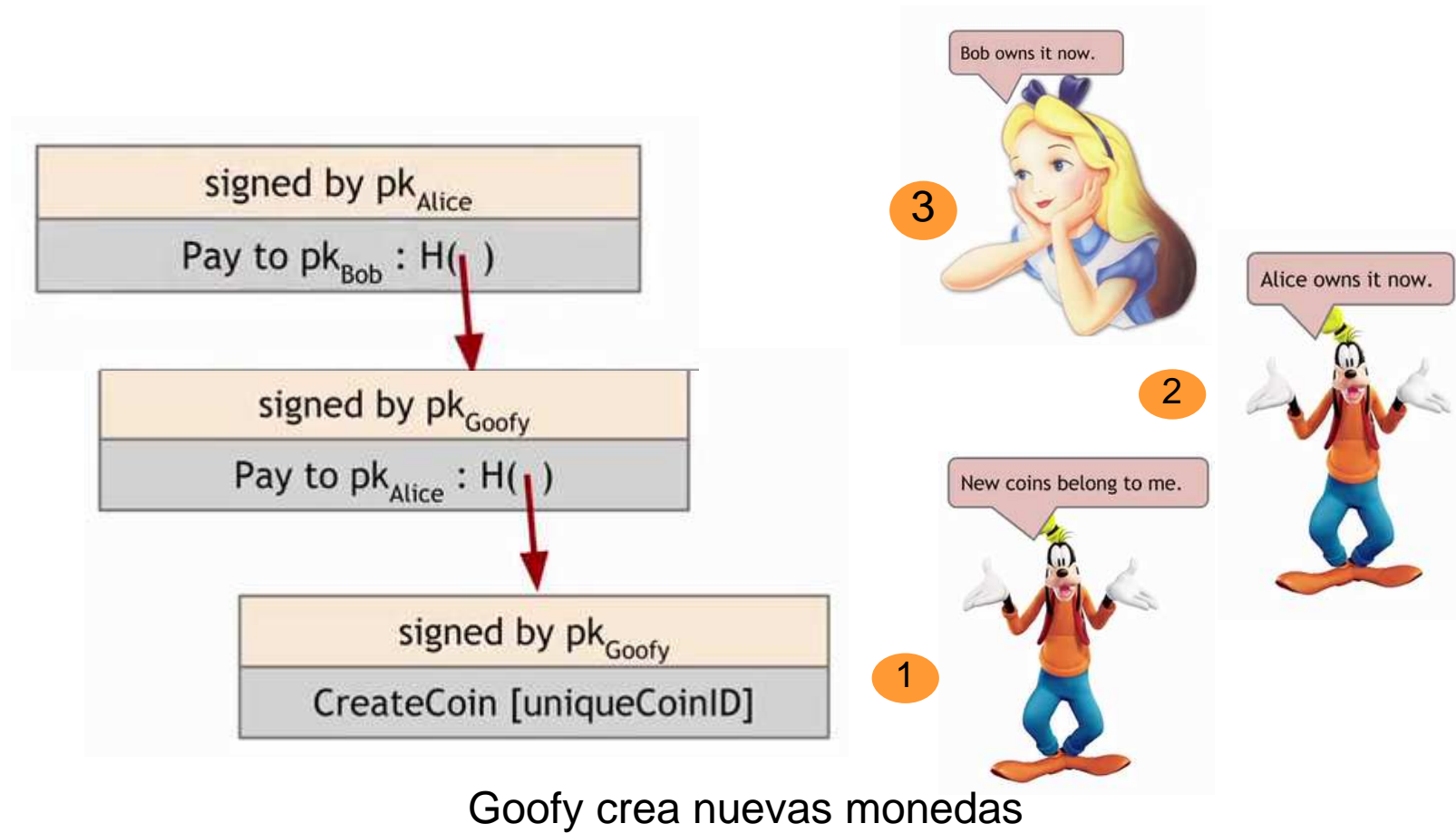
Blockchain = archivo replicado monótonamente incremental

Blockchain: Estructura e integridad

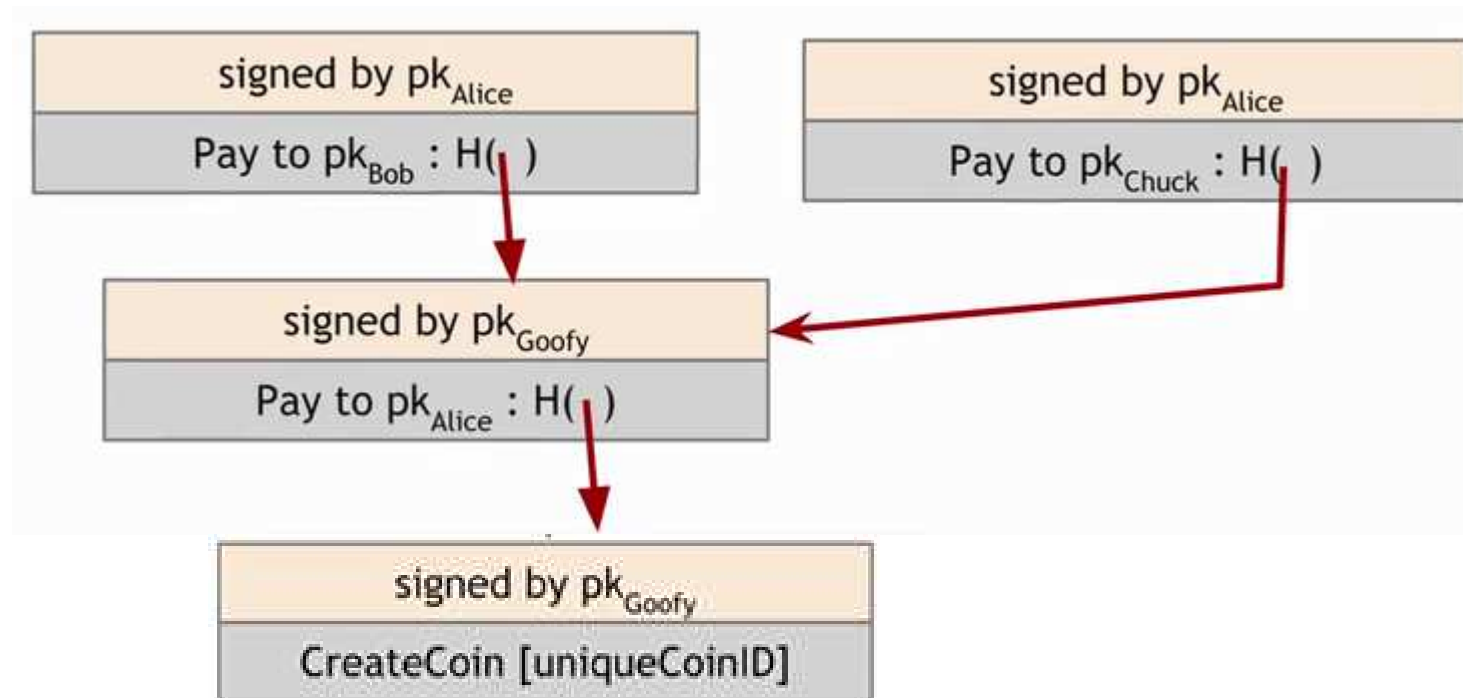
Lista de transacciones (LoT)	Hash de la LoT	Hash del bloque anterior	nonce	Time-stamp	Hash de todo el encabezado
------------------------------	----------------	--------------------------	-------	------------	----------------------------



Una Criptodivisa sencilla: *Goofy Coin*



Goofy Coin: El problema de “double-spending”

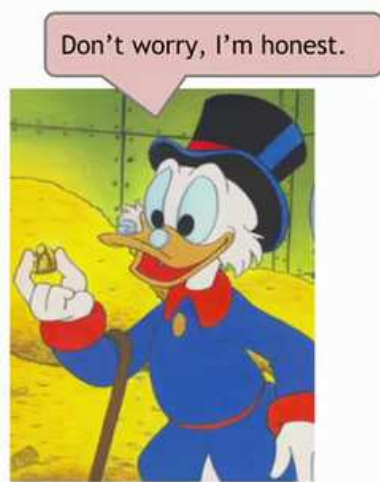


Resolviendo el “double-spending”

Scrooge publica **un registro de todas las transacciones válidas** (i.e., una especie de blockchain, firmada por Scrooge)



Chuck (y cualquiera) puede darse cuenta de que Alice está gastando dos veces las mismas monedas.



¿Podemos prescindir de Scrooge”?

Descentralización y consenso

¿Podremos deshacernos de Scrooge?

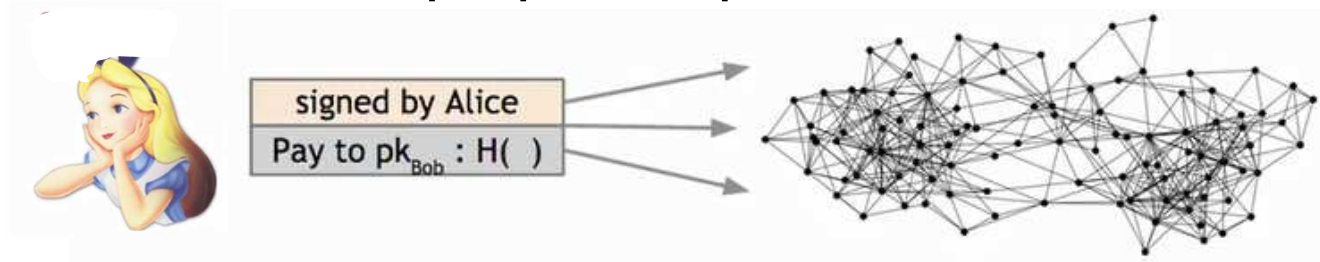
Aspectos de descentralización a considerar

- ¿Quién mantiene el ledger?
- ¿Quién decide qué transacciones son válidas?
- ¿Quién crea nuevas bitcoins?
- ¿Quién determina cambios en el sistema?
- ¿Cómo adquieren valor las bitcoins?
- Red P2P:
 - Abierta a cualquiera
- Minado:
 - Abierto a cualquiera, pero genera concentración de poder
- Actualizaciones de software:
 - Desarrolladores del núcleo tienen la confianza de la comunidad

Consenso distribuido

Se asume que hay n nodos entre los cuales m son maliciosos. Se ha alcanzado consenso si:

- Todos los nodos honestos coinciden en un valor tras verificar su validez
- Ese valor fue propuesto por un nodo honesto



En Bitcoin cuando Alice difunde una transacción, se busca que todos los nodos acuerden qué transacciones son válidas y en qué orden se ejecutan. Este consenso produce **un solo ledger global**

Retos para llegar a un consenso

- Nodos que pueden fallar
- Nodos maliciosos
- Imperfecciones de la red
 - No todos los nodos están conectados
 - Fallos en la red
 - Latencia
 - No se tiene una noción global de tiempo
 - Muchos algoritmos de sistemas distribuidos no se pueden aplicar
- En sistemas distribuidos *duros* se sabe que si $m > n/3$, no se puede llegar a un consenso

Consenso en Bitcoin

Las restricciones de los sistemas distribuidos se pueden relajar en Bitcoin:

- Se introduce un esquema de incentivos
- Hay un fuerte componente aleatorio al seleccionar el nodo que propone un valor (que agrega un bloque)
- El consenso ocurre en largos periodos de tiempo (1 hora)
- Las entidades no tienen identificador

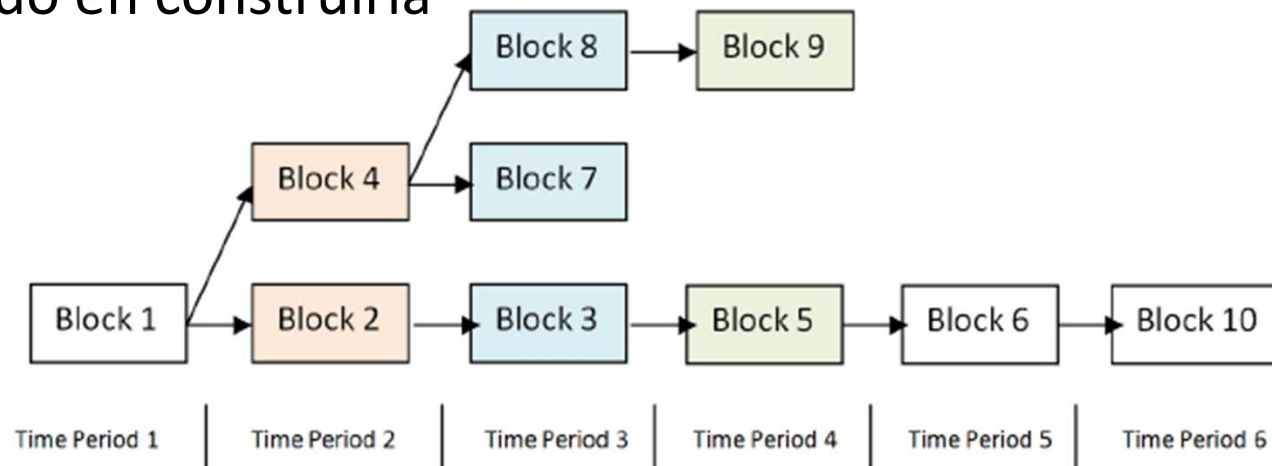
El blockchain de Bitcoin funciona mejor en la práctica que en teoría

Consenso implícito: la cadena más larga

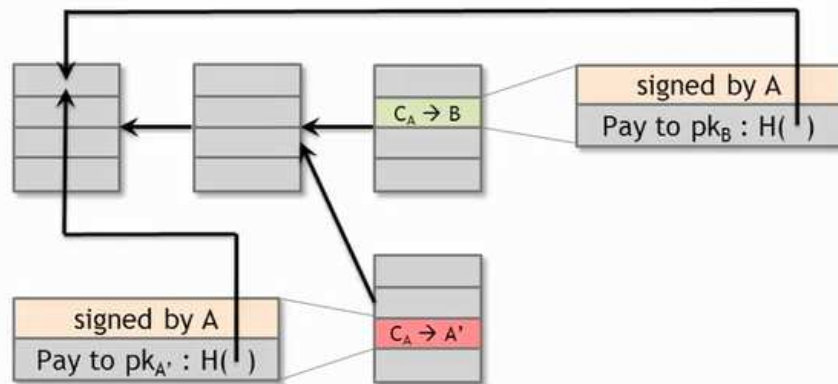
- Los nodos reciben **incentivos para “comportarse bien”**; el comportarse mal es costoso
- Mecanismo de **consenso implícito**:
 - En cada ciclo, **un nodo** gana **el derecho de agregar un bloque** a la cadena
 - El resto de los nodos verifica que las transacciones en el bloque propuesto sean válidas.
 - El resto de los nodos **implícitamente aprueban o desaprueban** la adición del nuevo bloque:
 - a) Extendiendo la blockchain con la adición del nuevo bloque (i.e., una **confirmación**) y tomando el hash de este bloque en el siguiente que crearán y formarán la **cadena más larga**.
 - b) No extendiendo la blockchain, y manteniendo el hash que estaba a la cabeza como el último

Obtener consenso

- Aunque la cadena es una lista, conviene visualizarla como un árbol
- La rama más larga (*longest path*) representa la cadena aceptada
- Un participante que desea extender una rama en la cadena de bloques, representa un voto para obtener consenso en esa rama. Entre más larga es la rama, más cómputo se ha invertido en construirla

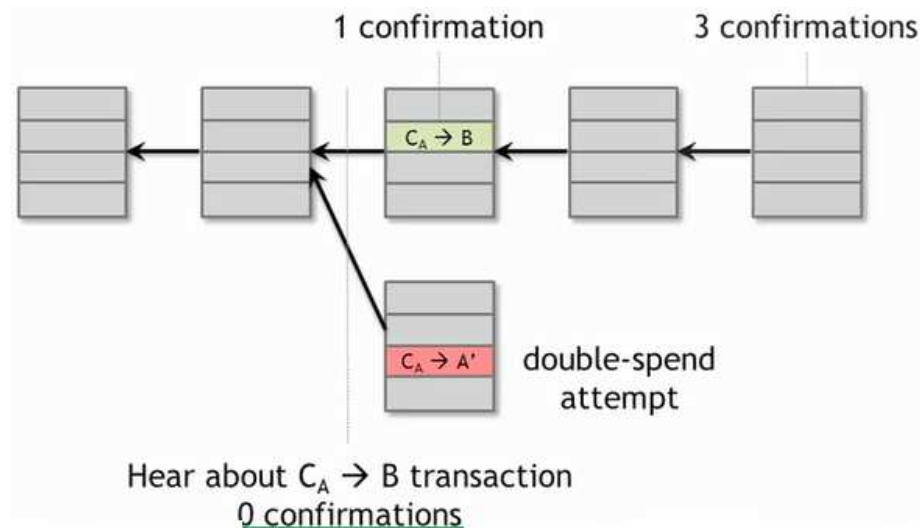


Evitando el “doble gasto” en Bitcoin



Double-spending attack

- Los nodos siempre extienden la cadena más larga.



- La probabilidad del “doble gasto” decae exponencialmente con el # de confirmaciones.
- Heurística: esperar por 6 confirmaciones.

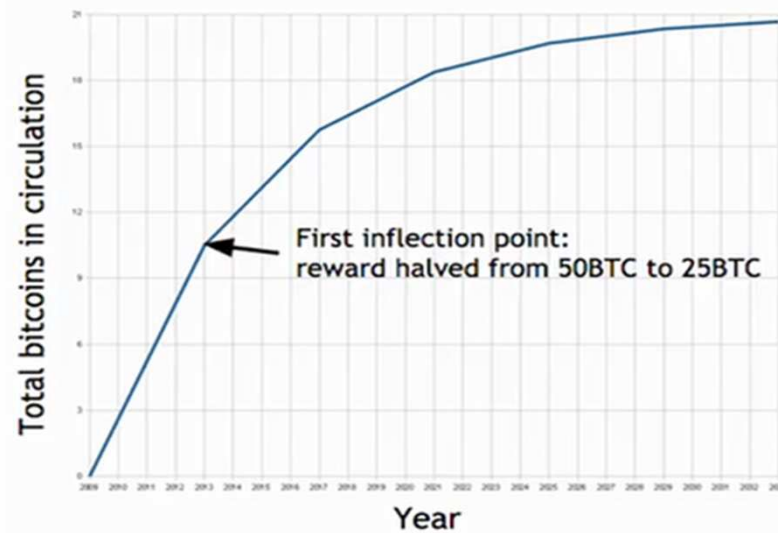
Confirmación de la transacción

- Una transacción aceptada en un bloque que potencialmente será agregado a la cadena, ha sido verificada. En particular, se confirma que sus entradas están disponibles (es decir, tiene fondos)
- Cada nuevo bloque encima de aquél en el que se registró la transacción, se considera una confirmación
- El mínimo para validar una transacción son dos bloques (20 min), son necesarios 4 bloques (40 min) y la transacción se considera madura e imborrable después de 6 bloques (1 hora)
- Las nuevas BTC creadas por el proceso de minado sólo son válidas tras 120 confirmaciones
 - Con esto se busca asegurar que un nodo con más de 51% de poder computacional no retira transacciones fraudulentas

Incentivos en Bitcoin

- 1) El nodo **minero** que **añade un bloque válido** a la blockchain recibe un premio, en la actualidad de **12.5 BTC** si y solo si el bloque pertenece a la cadena más larga. Cada 210,000 transacciones (aproximadamente cada 4 años) este premio se reduce a la mitad
- 2) Puede haber una **comisión** por cada transacción que se añade a un bloque de la blockchain.

- El No. de Bitcoins está topado en 21 millones y se agotarán en 2140



BITSO

1 BTC = 317974.93 MXN

Fecha: 4-Ene-2018



Prueba de Trabajo en Bitcoin

- La selección de un nodo es **proporcional a la fracción de poder de cómputo** que tiene con respecto a todos los demás nodos (i.e., “mineros”) de la red
- Los **mineros compiten** para resolver un **acertijo criptográfico** el cual sirve como **prueba de trabajo** del minero que resolvió la solución
- El acertijo consiste en **encontrar un nonce** (i.e., una cadena de bits) tal que, cuando se calcula el hash del (nonce || prev-hash || tx || tx...) el **resultado** nos da **un número menor que un determinado valor**:

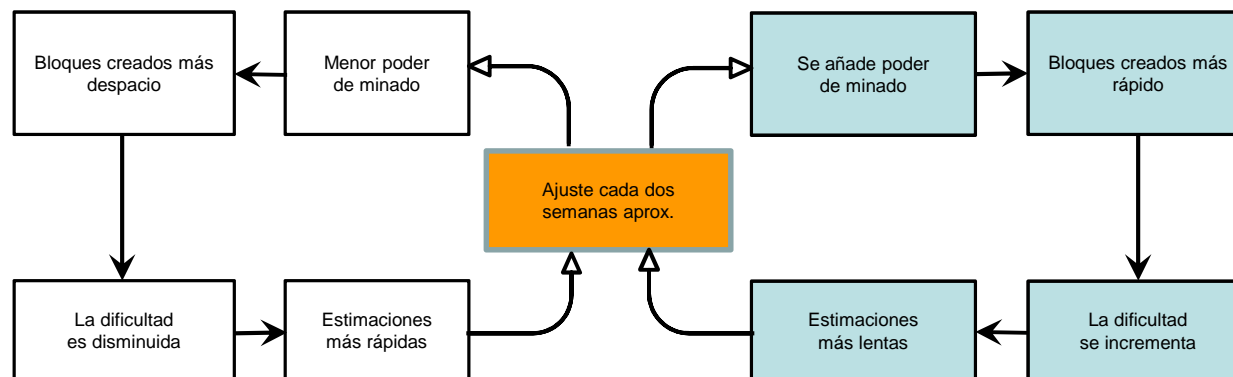
Un 0 => Prob 50%; Dos => 25%... 10 => 0.097%

Actualmente (Ene 2018) hay que encontrar ¡70 ceros!



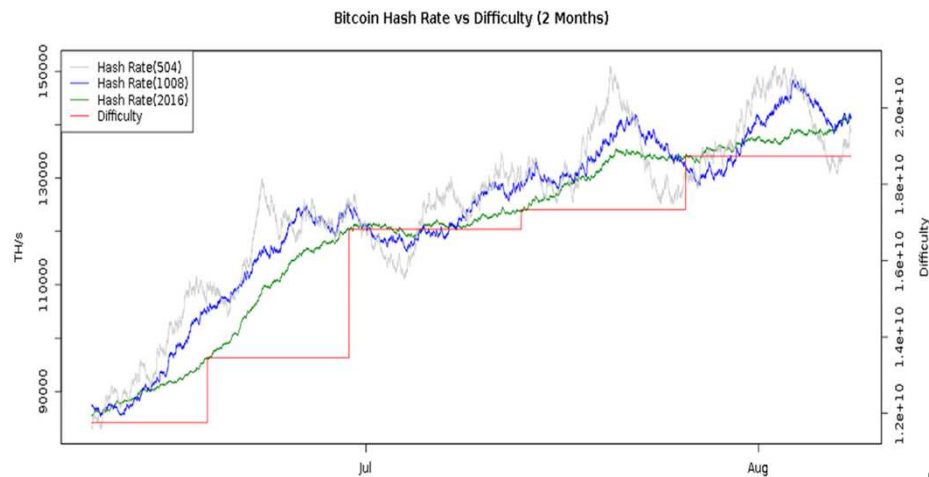
Propiedades del *Proof of Work* en Bitcoin

- Es computacionalmente costoso. En enero 2018, el espacio de búsqueda es de 10^{70}
 - Muy pocos nodos se molestan en competir. Esos son los **mineros**
- El costo computacional es **parametrizable**
 - Todos los nodos recalculan el reto objetivo cada 2,160 bloques, aproximadamente cada dos semanas

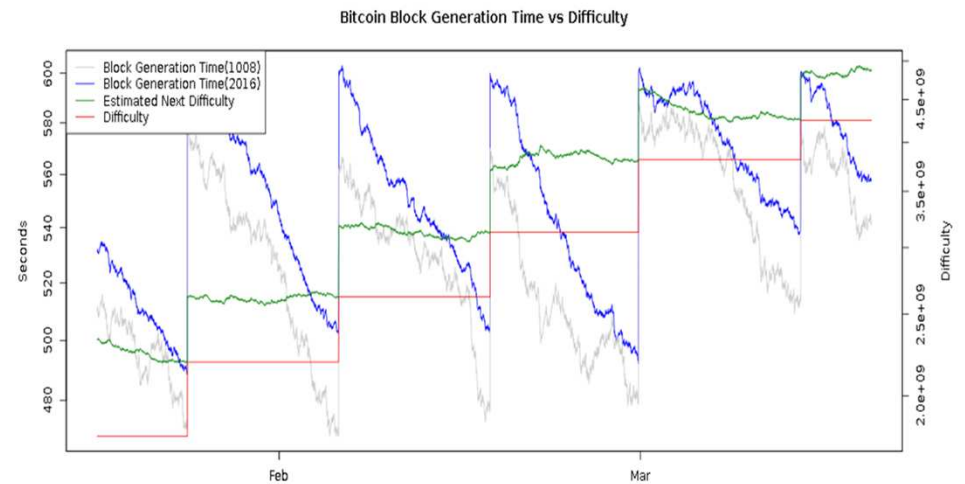


- La verificación del trabajo hecho es trivial
 - Los ganadores anuncian el nonce en el encabezado. La verificación se reduce a calcular el hash con él

Evolución de la dificultad de minado



Tiempo para agregar un bloque



Evolución de los mecanismos de minado

CPU



GPU



FPGAs y
ASICs



Centro profesional
de minado
(en Georgia,
Europa)

Otras formas de consenso

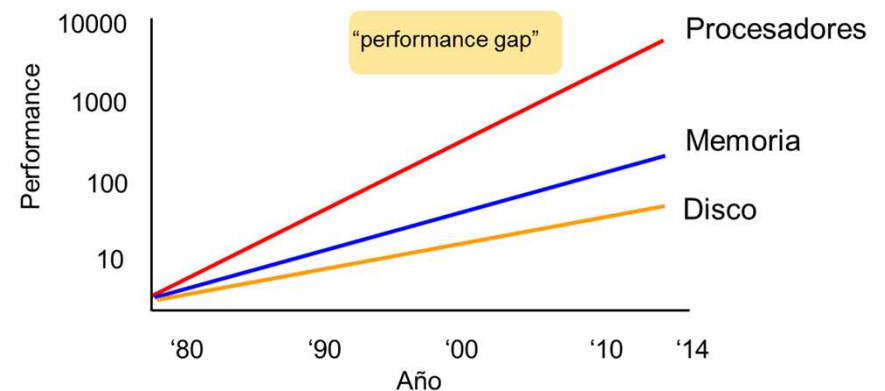
¿Es necesario el *Proof of Work*?

Proof of *useful* work

- La prueba de trabajo de Bitcoin consume mucha energía (entre 300 y 900 MW en agosto de 2014). Una pregunta natural es buscar algo útil que hacer con esos ciclos de CPU
 - Síntesis de proteínas, búsqueda de vida extraterrestre... pero es muy difícil ligar esto a un bloque
 - Primecoin
 - Busca números primos: p_1, p_2, \dots, p_n . p_i es un primo divisible por el hash $H(\text{previo} || \text{mrkl_root} || \text{nonce})$
 - Permacoin
 - Aprovecha el hardware de los mineros para almacenar y cobra por ello.

Ethereum Ethash

- También es *Proof of work* pero basado en accesos a memoria. La premisa es que **el desempeño para acceso a memoria** es más estable que el desempeño de procesamiento crudo.
- A partir de una **semilla** se obtiene una **cache de 16 MB** con números pseudoaleatorios. Clientes ligeros (laptops, tabletas) almacenan esta cache.
- De la cache se puede generar un **set de datos de 1GB** con la característica de que cada ítem en ese set depende únicamente de un pequeño número de ítems en la cache. Clientes robustos y mineros almacenan todo el dataset.
- El minado consiste en calcular el hash de pequeños bloques aleatorios del dataset. La verificación puede hacerse con poca memoria usando la cache para regenerar solo los bloques requeridos del dataset



Proof of Stake

- Quién tiene derecho a agregar un nuevo bloque, es función de una participación económica (*stake*) que puede ser total de criptodivisas, cantidad de criptodivisas depositadas, cantidad de CPU/memoria, etc.
- El voto tiene el peso de la participación. Comparable ligeramente a una votación de accionistas
- Para evitar centralización, se separa quien propone un nuevo bloque de quien **lo valida**. Estos validadores son los que tienen la participación. Se les paga por comisión.
- La participación puede variar de ronda en ronda
- Debe haber un resultado tras un periodo corto de tiempo
- Ninguna blockchain grande lo ha implementado. Ethereum tiene intención de migrar a PoS

Proof of Burn

- En vez de gastar dinero en equipo caro para minar, el nodo *quema* una cierta cantidad de criptodivisas
- Con ello gana el derecho a minar (un tiempo después) un bloque. Se espera que lo colectado en comisiones compense lo quemado
- La probabilidad de minar es proporcional a la cantidad de monedas quemadas (¿invertidas?)
- Concepto sencillo pero también desperdicia recursos.
- No se sabe bien el efecto de la natural inflación
- Ninguna blockchain grande lo ha implementado

Proof of Elapsed Time (PoET)

- Busca garantizar equidad entre todos los nodos para agregar un bloque
- Cada participante solicita un tiempo de espera de su *enclave* local y confiable (su CPU). Aquél con el menor tiempo de espera es quien gana el derecho a proponer un nuevo bloque
- El *enclave* local firma la función y el valor para que los demás puedan verificar que el nodo no hizo trampa anunciando un tiempo de espera deliberadamente corto
- Concepto propuesto por Intel con base en su tecnología de instrucciones protegidas SGX (*software guard extensions*) para HyperLedger Sawtooth, aunque no es propietario de Intel pues pueden usarse enclaves de AMD con *SME*, *Secure memory extensions* o *SEV*, *Secure Encrypted Virtualization*
- Debe considerarse que:
 - El concepto de enclave es complejo y relativamente desconocido, por lo que es potencialmente inseguro
 - Se requiere de una autoridad certificadora para validar las firmas de los enclaves

DLT privados y permissionados

En blockchains privados, donde se conoce la identidad de los participantes y permissionados, donde distintos participantes pueden tener distintos roles, es posible implementar mecanismos de consenso mucho más sencillos, como

- **Round robin** – con tolerancia de espera para considerar el caso de nodos apagados o descompuestos
- **Practical Byzantine Fault Tolerant (PBFT)** - un protocolo bien conocido en sistemas distribuidos que puede ser implementado en redes relativamente confiables y con un número reducido de nodos
- **Hashgraph** – Ledger distribuido que garantiza consenso en BFT asíncrono. Genera una gráfica acíclica dirigida, votación virtual y un protocolo *gossip*. Puede manipular miles de transacciones por segundo

Otros tipos de *Ledger distribuido*

Algunas ventajas de blockchain para los negocios

- Ahorra tiempo. Pasamos de días a casi tiempo real (por ejemplo, letras de crédito)
- Remueve costos. Overheads e intermediarios
- Reduce el riesgo de ciber ataques
- Mayor confianza porque todo es más transparente, se usa de la misma manera por todos
- Los *activos* no tienen que ser cripto-divisas. De hecho, no necesariamente deben registrarse intercambios de activos
- Ejemplos de usos
 - Títulos de propiedad, validación de diamantes y obras de arte
 - Trazabilidad de alimentos, materias primas
 - Identidad digital, pasaportes, certificados de boda, diplomas

Blockchain de Bitcoin. Algunas limitaciones

- Pseudo anonimato – no permite auditorías. No se siguen lineamientos KYC
 - Se ha usado para venta de drogas en línea y pagos de *ransomware*
- Latencia: Transacciones se validan en mínimo 20 min, necesario 40 min y permanente: 60 min
 - No sirve para aplicaciones de alta transaccionalidad o tiempo casi real
- No escalable: Menos de 100 transacciones por segundo; se requiere de al menos dos órdenes de magnitud más
- Consumo energético PoW:
 - En jun 2017, un minero generaba 1×10^{18} de hashes por segundo y consume 500 MW, el equivalente de 325,000 hogares.
 - En ene 2018, el consumo total se estima en 2100 MW, que es aproximadamente 0.01% del consumo mundial. Algunos autores sobreestiman los cálculos y obtienen 43 TW, más o menos lo que consume Perú.
- Costo
 - En un bloque caben aproximadamente 2,000 transacciones. Para que una transacción sea incluida en el bloque, se otorga una comisión al minero.
 - En 2018, esta comisión rondaba los \$14 USD independientemente del monto

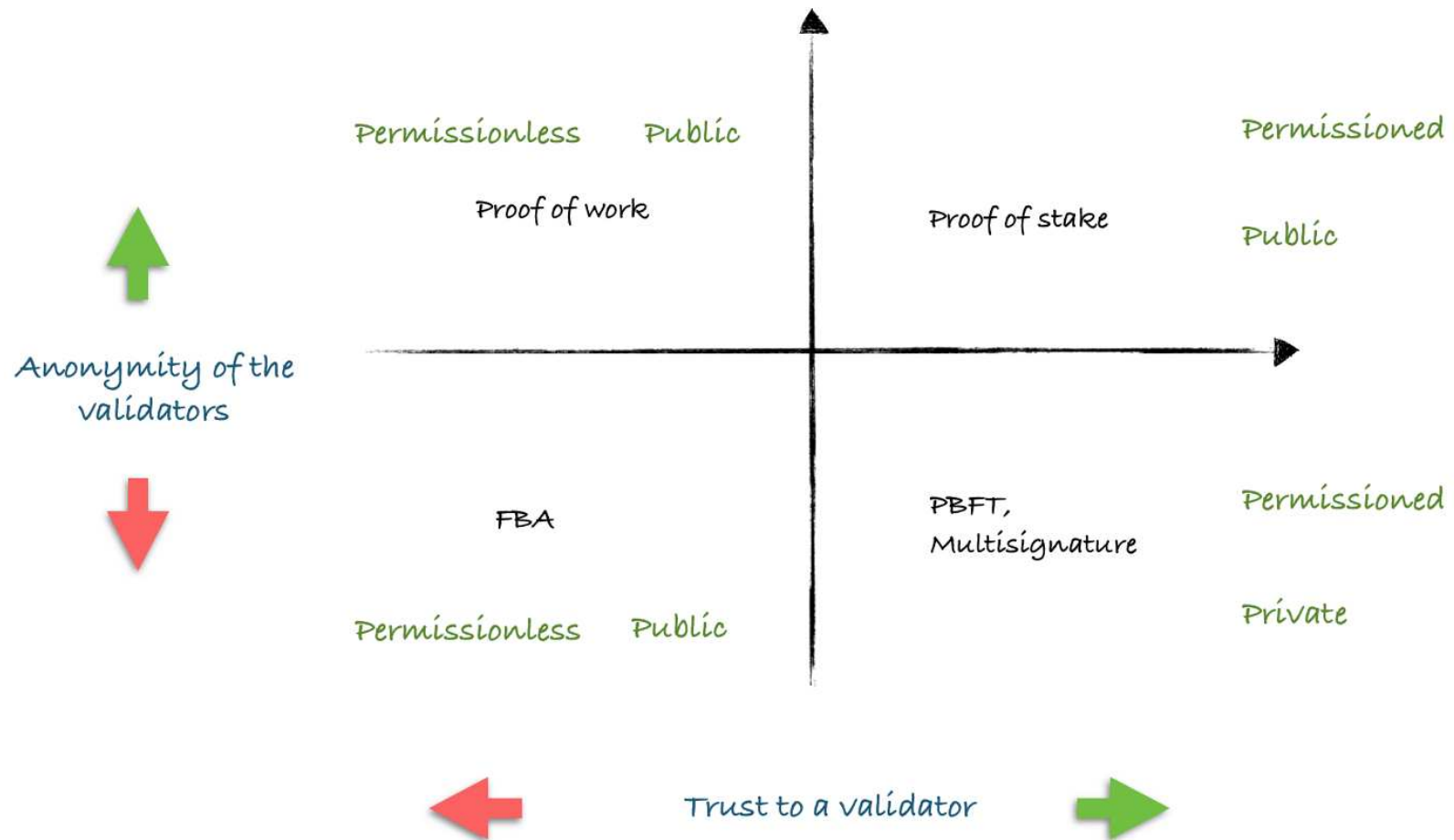
Evolución de blockchains

- Blockchain 1.0 – Crypto divisas
- Blockchain 2.0 – Smart contracts, blockchains permissionadas
 - Ethereum, Hyperledger y otros
 - La cadena de bloques también puede almacenar código que se ejecuta bajo ciertas condiciones para controlar las transacciones
 - Dado que el código está en la cadena, es inviolable
 - Contratos legales pueden traducirse a reglas de negocio que se codifican en un *smart contract*
 - P. Ej., Liberar pago si la carga llegó antes de las 23:40
 - Cadenas permissionadas (o privadas) sólo aceptan usuarios previa autenticación (no hay anonimato). Puede haber distintos roles, notablemente un rol de auditor
- Blockchain 3.0 - Aplicaciones

Blockchain privado

- Se mantiene intercambio P2P entre nodos
 - De ser necesario, el contenido es cifrado, solo accesible a las partes interesadas
- No necesariamente cripto divisa
 - Típicamente, la red está implementada para mercados específicos donde puede haber o no intercambio de activos
- Varios mecanismos de consenso
 - Proof of Stake, PBFT, Proof of Elapsed Time, Round Robin, ...
 - Las confirmaciones se dan en segundos
 - Escalable
- Varios roles
 - Administrador, auditor, regulador, etc.

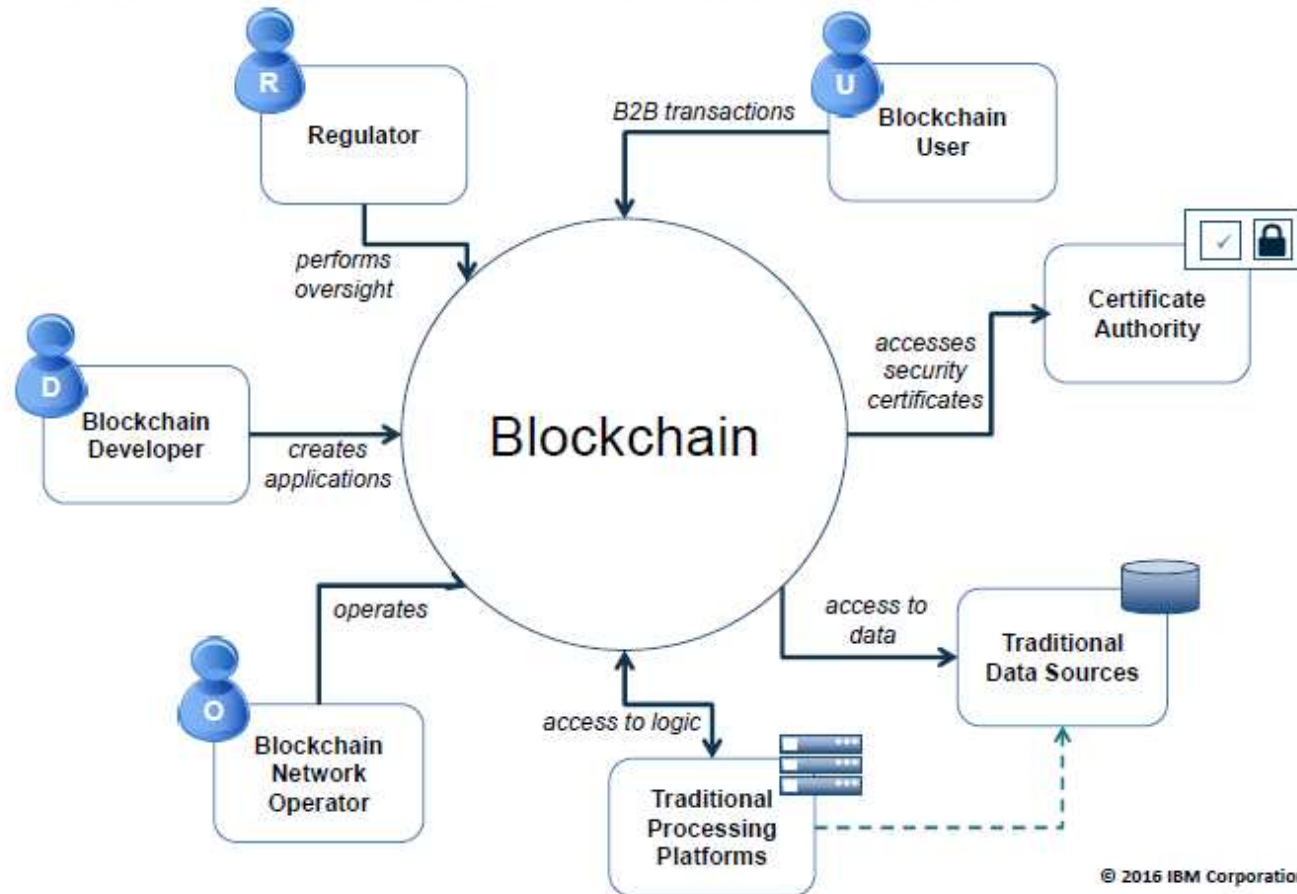
Tipos de blockchain y consenso



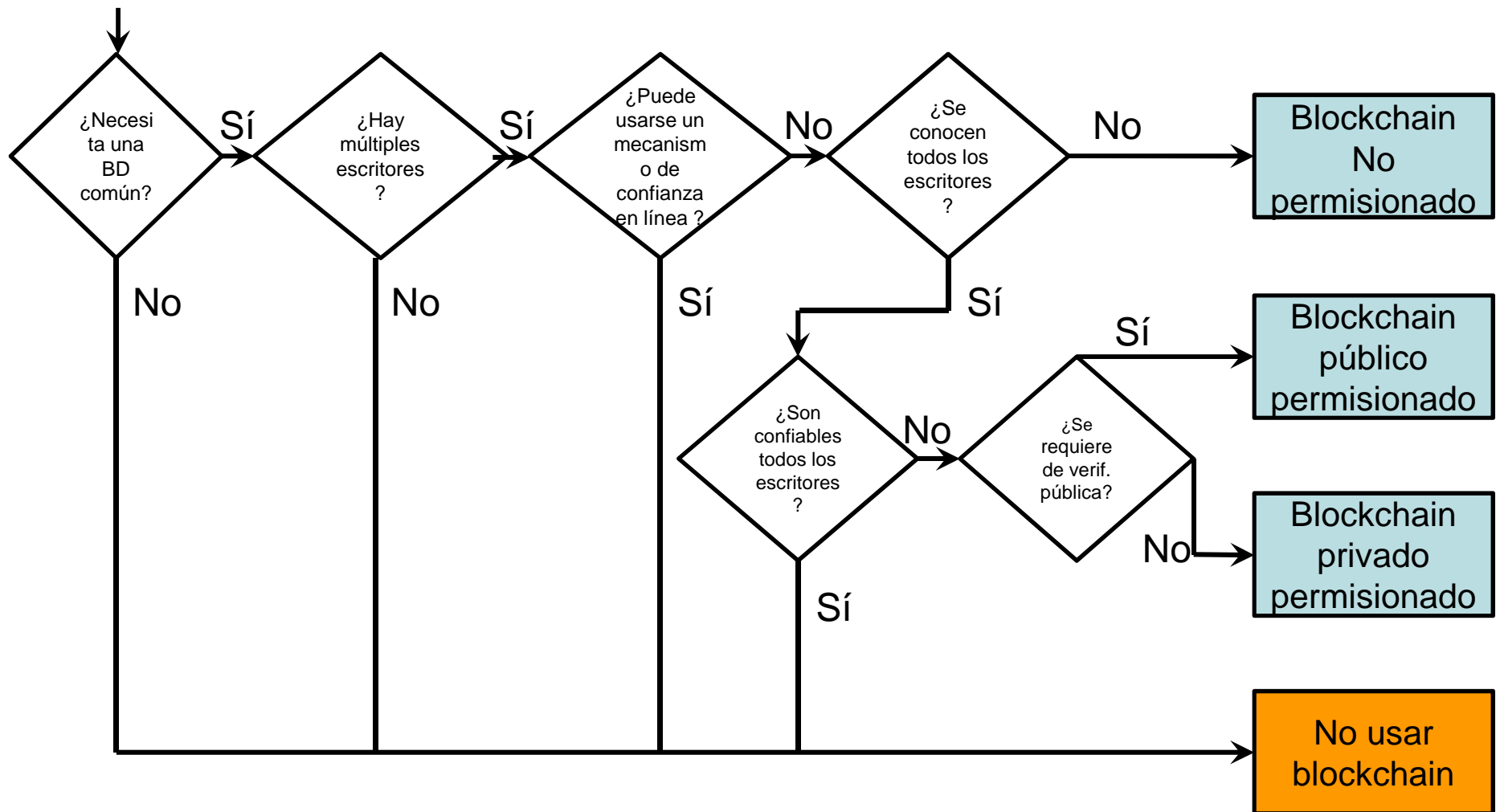
Participantes en una blockchain permissionada

IBM developerWorks

The participants in a blockchain network



¿Cuándo usar blockchain?



Blockchain: El Protocolo de Confianza

- Desde 1981 se habían querido resolver los problemas de **privacidad, seguridad e inclusión** de Internet con criptografía.
- La tecnología detrás de Bitcoin **asegura la integridad sin necesidad de un tercero de confianza.**
- Marc Andreessen (Netscape): “This is it!. **Whoever he is should get the Nobel prize! He is a genius!**”
- El nuevo “ledger” puede ser utilizado para registrar **cualquier cosa de valor** o importancia.



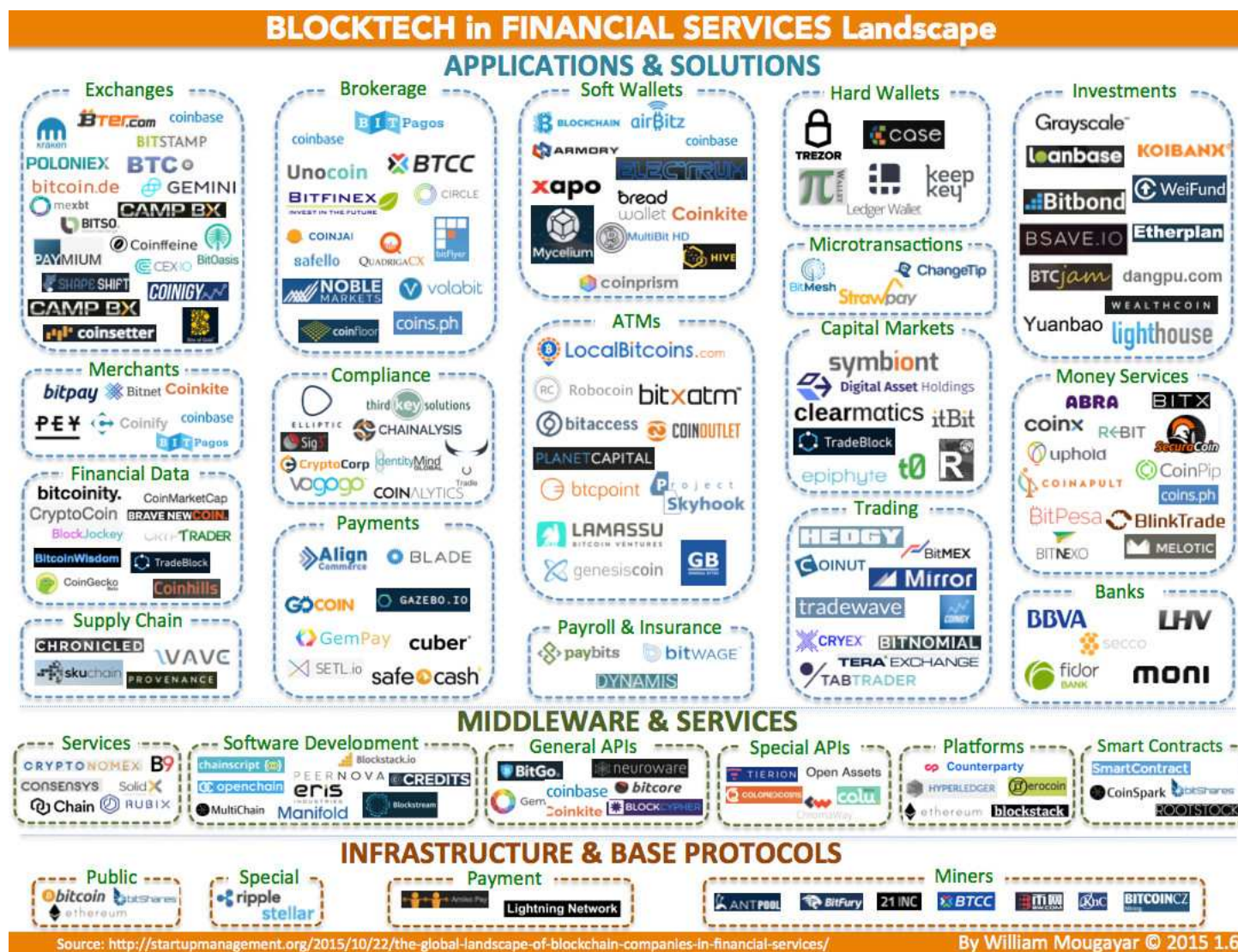
El Interés ha sido enorme



Instituciones Financieras invirtiendo en blockchain



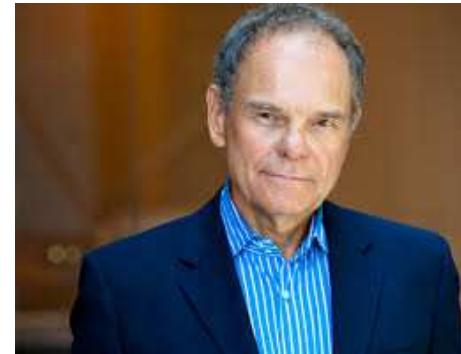
Mapa tecnológico de blockchain para el sector financiero



Blockchain: Implicaciones

- “**True Peer-to-Peer sharing economy**”

- En lugar de compañías centralizadas como Uber o Airbnb, ¿qué tal una blockchain?
- Derechos de propiedad
- Remesas sin tener que pagar altos costos de transacción
- Compensación a los creadores de valor
- Contratos inteligentes y empresas abiertas distribuidas
- Internet Of Things (IOT) –intercambio de unidades de valor



https://www.ted.com/talks/don_tapscott_how_the_blockchain_is_changing_money_and_business#t-1112274

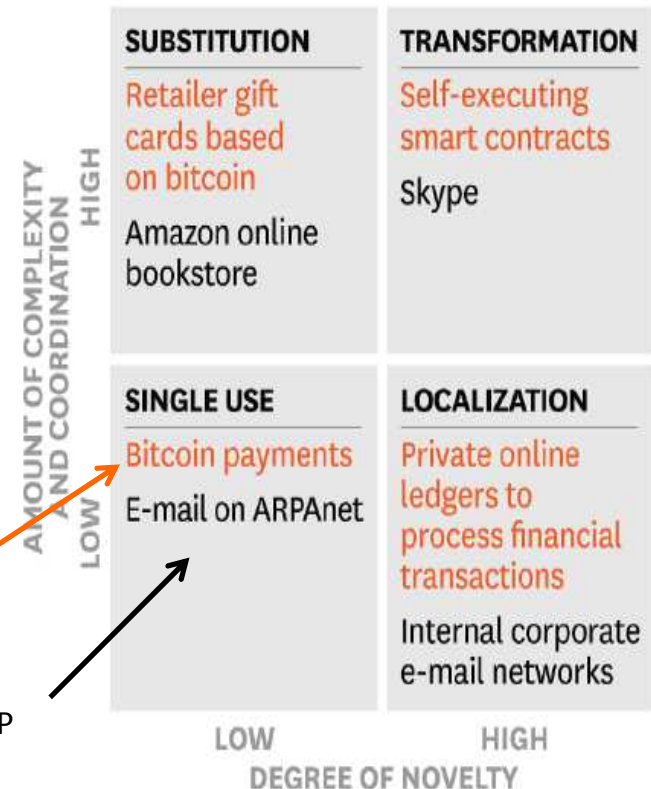
- **De un Internet** cuyo desarrollo fue impulsado por la colaboración, eficiencias en búsquedas, colección de datos y toma de decisiones donde el juego era **monitorear, mediar y monetizar información**....
- **A un Internet** impulsado por menores costos de negociación y el cumplimiento de acuerdos comerciales y sociales en donde el juego será **integridad, seguridad, colaboración, privacidad, creación y distribución de valor**.

Blockchain: Implicaciones

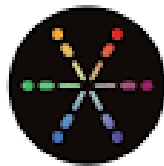
- Iansiti y Lakhani señalan que **Blockchain** debe entenderse como una tecnología “fundacional”, similar a lo que fue el protocolo TCP/IP de Internet introducido en 1972.
- Las implicaciones que requieren un nivel de coordinación bajo entre los distintos jugadores son las primeras en adoptarse.

Innovaciones sobre blockchain

Innovaciones sobre TCP/IP



Multichain



DLT privado, con permisos y perfiles

Principales características



- Cadena de bloques **privada**
 - Los miembros deben solicitar permiso de conexión al administrador
- Cadena de bloques **permissionada**
 - Permisos para conectarse; crear, enviar y recibir activos; crear y monitorear *streams*, gestionar permisos
- Derivación de bitcoin, casi todas **sus APIs son compatibles con las de bitcoin**
 - Es más fácil encontrar información sobre una función buscando en la web para bitcoin, que en el manual de Multichain
- Consenso: Acuerdo por mayoría
- Inserción de bloques (minado): Round robin
- **Streams**
 - Base de datos key-value.
 - Se puede consultar Qué, Quién, Dónde, En qué bloque, cuándo se guardó algo en el *stream*

Ethereum






Características

- Ethereum es mucho más que una criptodivisa
 - *Smart Contracts (SC)* y *Ethereum Virtual Machine (EVM)*
 - SC son como clases en un lenguaje de programación orientada a objetos.
 - Son “funciones” que se “invocan” con la dinámica de las transacciones y:
 - Almacenan y mantienen datos de entorno (tokens, pertenencia a la organización, etc.)
 - Administran las relaciones entre usuarios no confiables (**contienen las reglas de negocio**)
 - Proveen funciones a otros SC
 - Autorizaciones complejas, como negociaciones MxN
 - EVM. Los SC se ejecutan en **todos los nodos** de la red
 - SC distribuidos en la cadena de bloques, son casi imposibles de modificar o de bloquear
 - Dado que se ejecutan en todos los nodos, se requiere de una compensación.
 - » **Gas** en función del espacio en memoria, complejidad, y ancho de banda requeridos)

Algunas diferencias entre Bitcoin y Ethereum



		
Inserción de bloques	Cada 10 min	De 10 a 15 seg
Recompensa	12.5 BTC (se reduce c/4 años)	5 ETH (2 a 3 ETH a tíos)
Consenso	Proof of Work	Ethash PoW, migrando a PoStake
Costo	Comisiones voluntarias	Gas bien definido
Divisa	Criptodivisa BTC	Token ETH
División divisa	10^{-8} BTC = Satoshi	10^{-18} ETH = Wei

Hyperledger

Redes permissionadas para negocios



Proyecto Hyperledger – Linux Software foundation



Frameworks

**Hyperledger
Indy**

**Identidades
descentralizadas**

**Hyperledger
Fabric**

**El más completo:
Estructura modular
Plug&Play
Transacciones
confidenciales**

**Hyperledger
Iroha**

**Proyectos simples
Aplicaciones móviles
Integración a
infraestructura**

**Hyperledger
Sawtooth**

**Plataforma modular
Alta escalabilidad
PoET
IoT**

**Hyperledger
Burrow**

**“Smart contract
permisionada”
Implementa
Ethereum EVM**

Herramientas

**Hyperledger
Composer**

**Lenguaje de
modelado**

Evita usar Go

**Hyperledger
Explorer**

**Interfaz para
visualización**

**Hyperledger
Cello**

**Blockchain
as a Service**

**Hyperledger
Quilt**

**Interoperabilidad entre
ledgers**

Proyecto Hyperledger



- Blockchain entre industrias y sectores económicos
- Auditoria, transparencia, confianza inmutabilidad



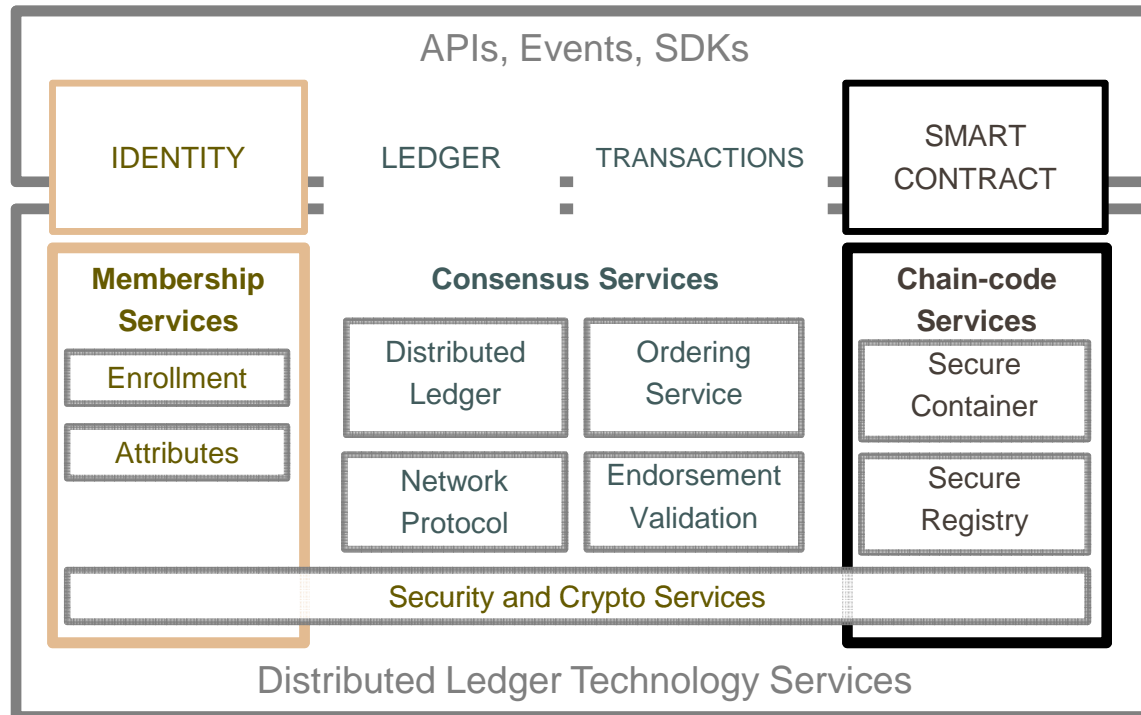
Mercado objetivo:

Redes de negocios

- Permisionada o no permisionadas, dependiendo del caso de uso
- Altamente modulares y escalables
- Privacidad entre actores a través de *canales*



Arquitectura de referencia



IDENTITY

Pluggable, Membership, Privacy and Auditability of transactions.

LEDGER | TRANSACTIONS

Distributed transactional ledger whose state is updated by consensus of stakeholders

SMART-CONTRACT

“Programmable Ledger”, provide ability to run business logic against the blockchain (aka smart contract)

APIs, Events, SDKs

Multi-language native SDKs allow developers to write DLT apps

Algunas diferencias entre Bitcoin, Ethereum y Hyperledger



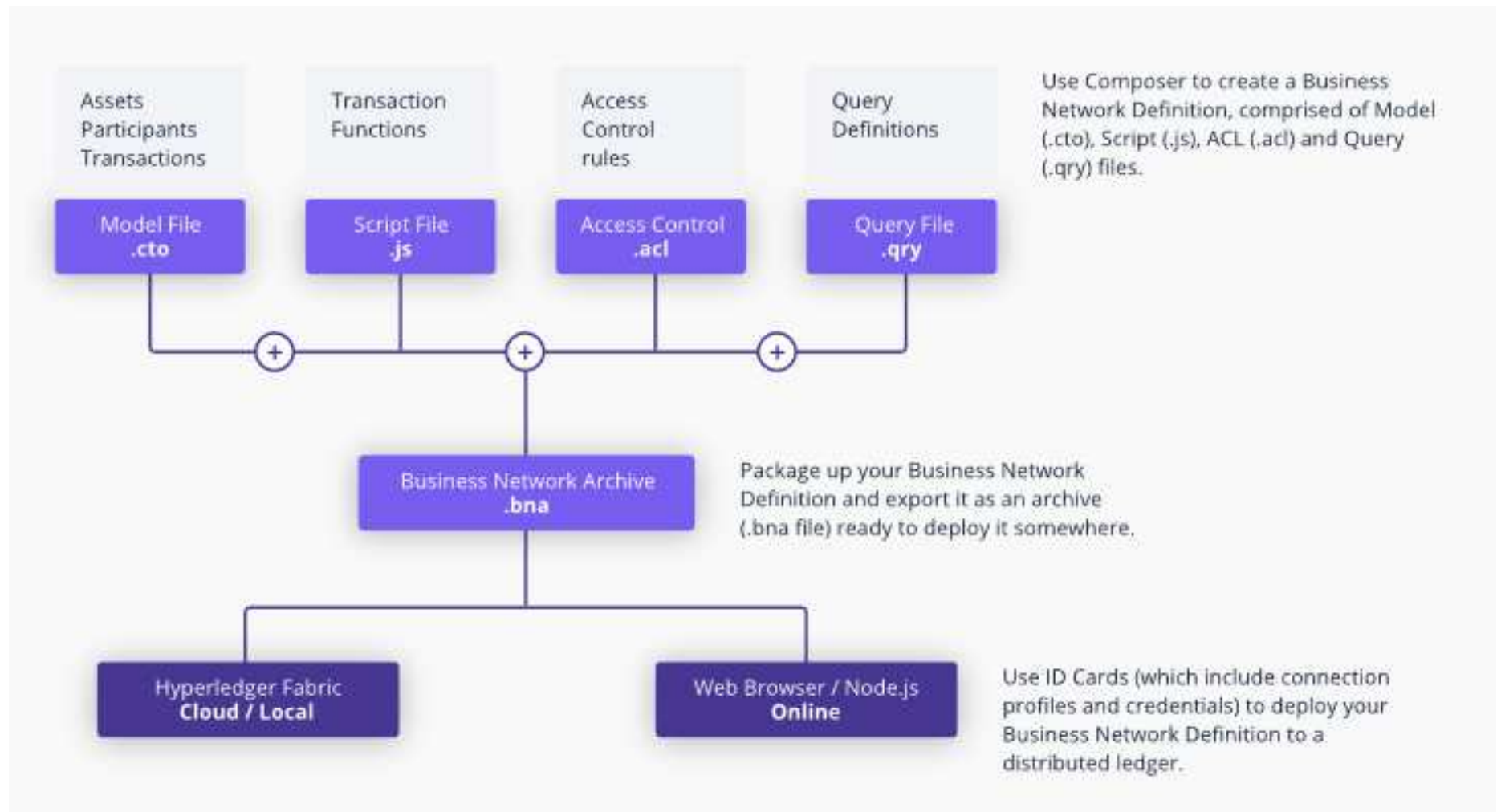
			
Basado en criptodivisa	Sí	Sí	No
Permisionado	No	No	Sí*
Pseudo-anónimo	Sí	No	No
Auditable	Sí	Sí	Sí
Inmutable	Sí	Sí	Sí
Modularidad	No	No	Sí
Smart contracts	No	Sí	Sí
Consenso	PoW	PoW->PoS	Configurable

Hyperledger composer



- Permite modelar redes de negocio. Lenguaje de modelado
 - Definir activos, participantes, transacciones
 - Creación de aplicaciones mucho más rápida y con menor riesgo
- Genera APIs REST para interactuar con la aplicación
 - Crear, actualizar y eliminar recursos
 - Emitir transacciones
- Produce una plantilla Angular JS para desarrollo de aplicaciones
- Ofrece un procesador de transacciones que se ejecutan sobre Hyperledger Fabric
- *Playground* en Web composer-cli para desarrollo

Hyperledger composer



Apéndice. Fundamentos criptográficos

Funciones hash y firmas digitales

Fundamentos Técnicos: Firmas Digitales

- Una **firma digital** debe cumplir con **dos condiciones**:
 - **Sólo el usuario puede “firmar”**... pero **cualquiera puede verificar** que la firma corresponde al firmante
 - La **“firma” está ligada al contenido** que se firma (es decir, no se puede “copiar” la firma de un documento a otro)
- Bitcoin utiliza el mecanismo de **“llave pública”** y **“llave privada”**:
 - Solamente yo conozco mi “llave privada”
 - Todos los que yo quiero conocen mi “llave pública”
 - Cuando se encripta con una llave, se puede desencriptar y abrir con la otra llave



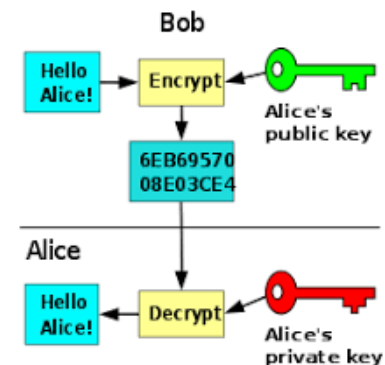
Llaves Pública y Privada

Para **demostrarles** a todos que **yo soy quien envía** un mensaje:

- Uso mi “**llave privada**” para “**firmar**” o **encriptar** el mensaje
- Cualquiera puede usar mi “**llave pública**” para abrir y **verificar** que soy el que envió el mensaje

Para **evitar que nadie**, con excepción del **destinatario deseado**, pueda **ver el contenido** del mensaje:

- El **emisor** usa la “**llave pública**” del destinatario deseado para **encriptar** el mensaje
- **Sólo el destinatario** puede abrir el mensaje con su “**llave privada**”

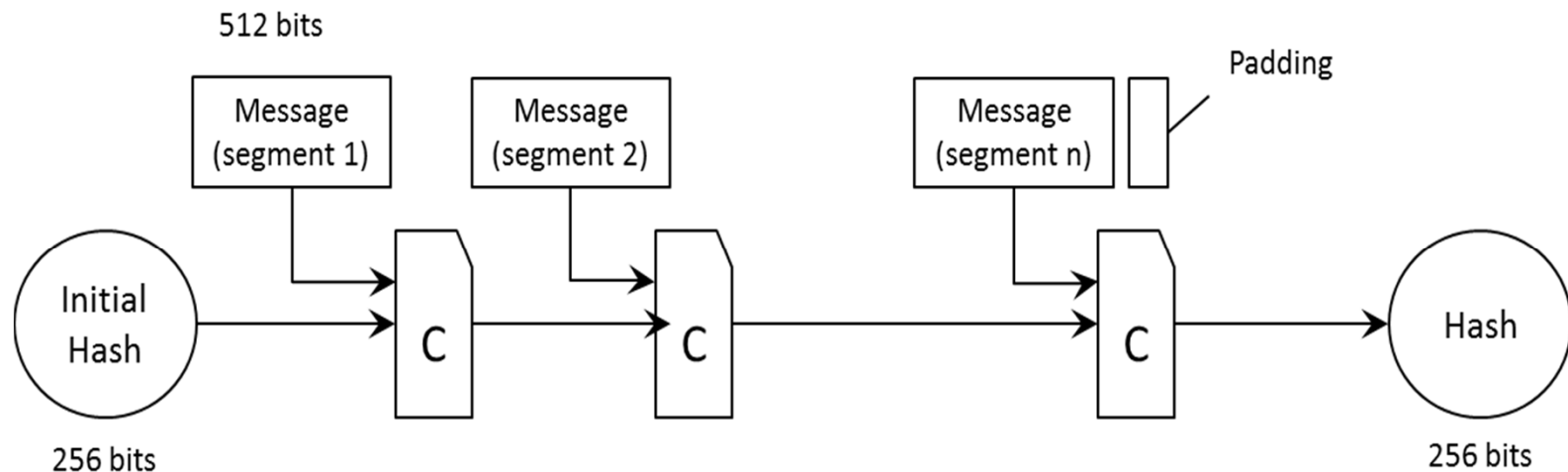


Fundamentos: Funciones Hash Criptográficas

- Una Función Hash Criptográfica es un **caso especial de una función hash**. Tiene propiedades que la hacen ideal para usarse en criptografía
- Es un **algoritmo matemático** que mapea un **cadena de bits** de cualquier tamaño a un cadena de bits de longitud fija y está diseñado para que la función sea **sólo en un sentido** (i.e., no se puede invertir)
- Cinco propiedades principales:
 1. Es **determinística**, para que el mismo mensaje siempre dé el mismo valor hash
 2. Es **fácil y rápido calcular** el valor hash de cualquier mensaje
 3. Es **“imposible” generar el mensaje a partir del valor hash**, a menos que se generen todos los posibles mensajes
 4. Un **cambio pequeño en el mensaje cambia dramáticamente el valor hash**, de tal forma que el nuevo valor hash no está correlacionado con el anterior
 5. Es **“imposible” encontrar dos mensajes distintos con el mismo valor hash**

Bitcoin utiliza la Función Hash Criptográfica SHA-256

- Desarrollada por la *National Security Agency* (NSA).
- Si la función de compresión (C) es “libre de colisiones”, entonces la función SHA-256 también es “libre de colisiones”



Algoritmo Hash

- En una sola dirección
- Probabilidad de colisión extremadamente baja
- MD5, SHA1, SHA256
- Ejemplos MD5:

abc: 0bee89b07a248e27c83fc3d5951213c1

abC: 2217c53a2f88ebadd9b3c1a79cde2638

“The Quick Brown Fox Jumped Over the Lazy Dog”

2dfd75162490ed3b4c893141f9ab37cf

Sin importar el tamaño de archivo original, se comprime en un compendio de 64-bytes

Ejemplo para lograr Integridad, Autenticidad y Transparencia

