Redes P2P y Bitcoin

Guillermo Galindo Ortuño
Carlos Santiago Sánchez Muñoz

Fudamentos de Redes

Redes P2P

Bitcoin

- La cadena de bloques
- Bitcoin como red P2P
- Minado
- Demo de una cadena de bloques

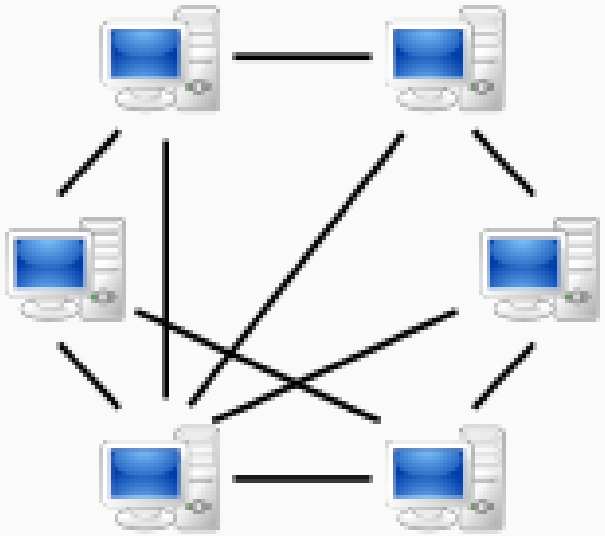
Bibliografía

Redes P2P

Definición

P2P es una arquitectura de red que consiste en que cada nodo funciona simultáneamente como cliente y como servidor.

Esta fue utilizada por aplicaciones como Napster, Spotify, Skype, etc.

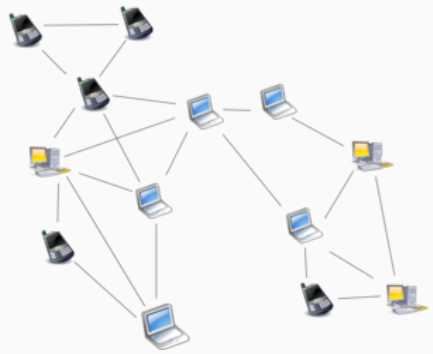


Características

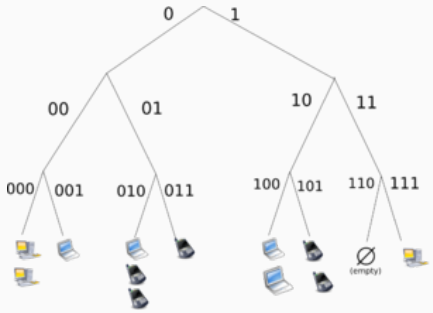
- Escalabilidad
- Robustez
- Descentralización
- Distribución de costes entre los usuarios
- Anonimato
- Seguridad

Clasificación

- Redes desestructuradas



- Redes estructuradas



- Redes híbridas

Problemas

- Encontrar nodos ya conectados a la red
- Cómo se conectan los nodos sin IP entre ellos

Bitcoin

Introducción

Bitcoin es una criptodivisa que utiliza un sistema totalemente descentralizado. Se llama bitcoin tanto a la unidad monetaria como al protocolo y a la red que lo sustentan.
Ahora mismo un bitcoin está valorado en aproximadamente 5000$.

Bitcoin

La cadena de bloques

Bloques

Los bloques forman la cadena de bloques y almacenan un conjunto de transacciones. Estos poseen la siguiente estructura :

Tamaño	Campo
4 B	Número mágico 0xD9B4BEF9
4 B	Tamaño de bloque
80 B	Cabecera de bloque
1-9 B	Contador de transacciones
Variable	Transacciones

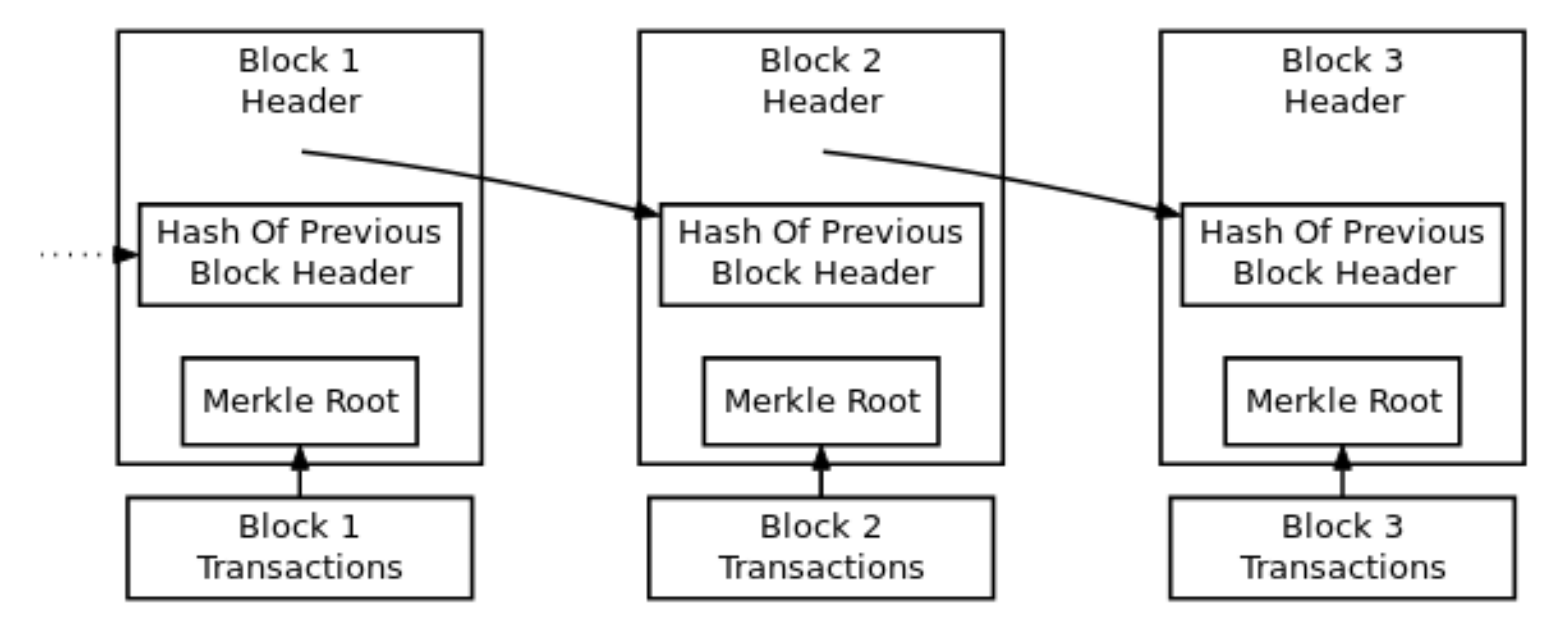
Bloques

La cabecera de cada bloque posee la siguiente estructura :

Tamaño	Campo
4 B	Versión
32 B	*Hash* de la cabecera del bloque anterior
32 B	*Merkle root*
4 B	Marca temporal tipo UNIX
4 B	Dificultad minera del bloque
4 B	Nonce

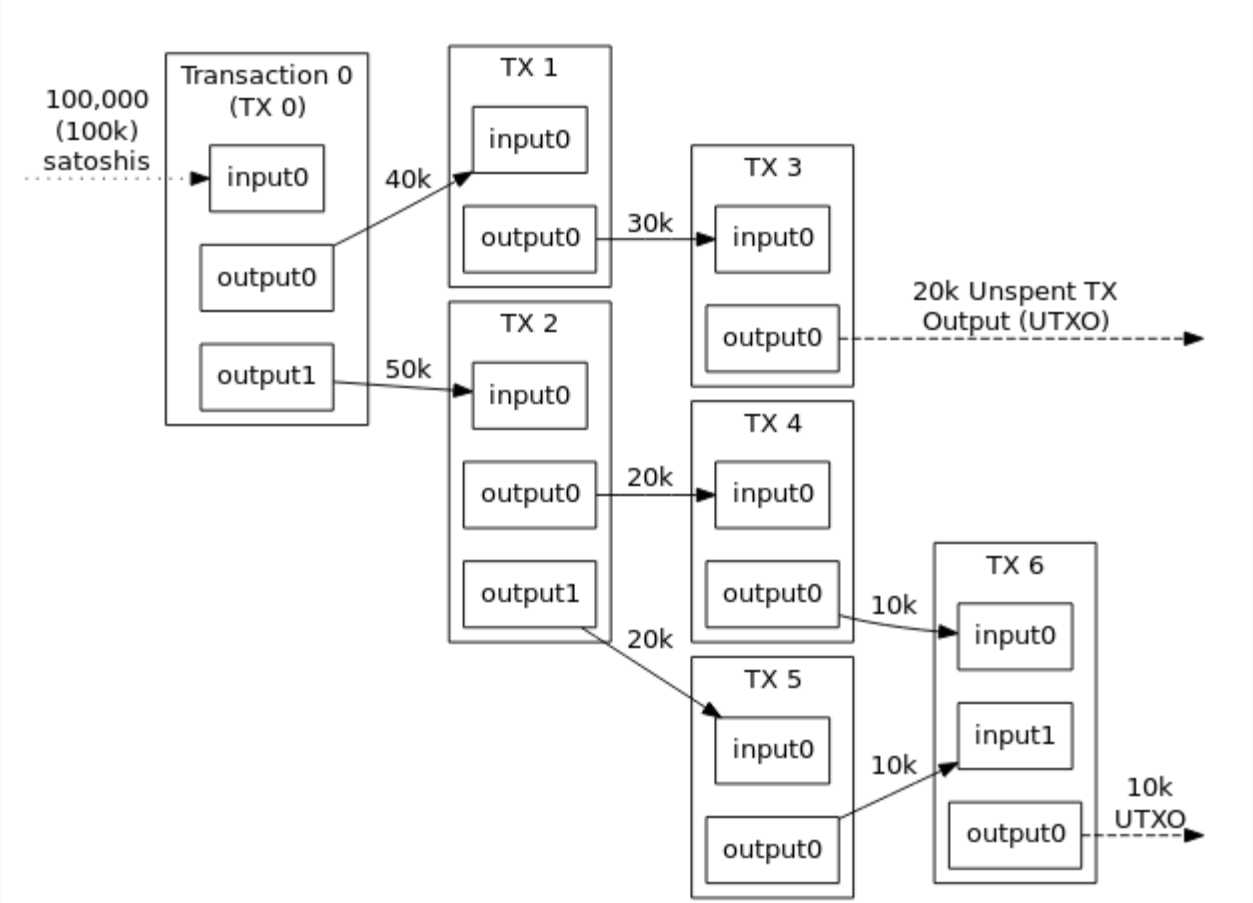
Ejemplo cadena de bloques

Una cadena de bloques tendría una estructura similar a la mostrada abajo, y es gracias a esta estructura que resulta realmente complicado tratar de modificar un bloque ya validado para obtener beneficio "ilegal", ya que habría que volver a validar todos lo bloques que "cuelguen de él"



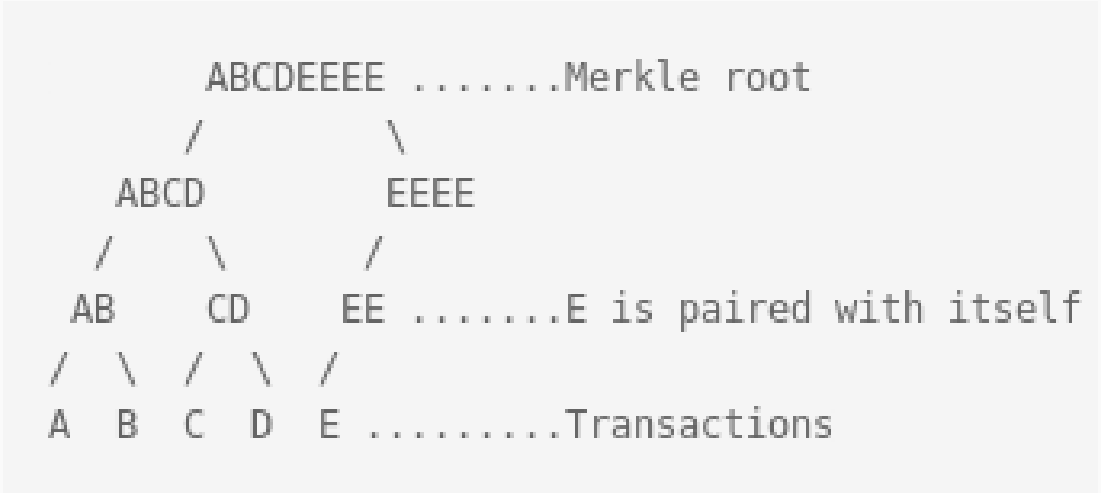
Transacciones

Aunque pensemos que en las transacciones los satoshis van de una cartera a otra cartera, esto no es así, ya que las transacciones tienen salidas y entradas, y así "viajan" entre estas.



Transacciones : merkle tree y merkle root

Buscar si una transacción se encuentra en un bloque descargando el arbol entero de transacciones sería increíblemente costoso, por eso se utilizan los llamados “merkle tree” y “merkle root”. Un ejemplo simplificado de un “merkle tree” sería el siguiente :



Prueba de trabajo

Validar un bloque en Bitcoin consiste a grandes rasgos en calcular un hash a partir de la cabecera de dicho bloque, y comprobar que el valor está por debajo de un umbral que es al que se llama "dificultad de validación". Esto es junto con la estructura de cadena es lo que realmente hace que sea difícil modificar las transacciones de bloques ya validados.

Prueba de trabajo

Esta dificultad de validación se va modificando cada 2016 bloques validados en función del tiempo que hayan tardado en validarse estos bloques :

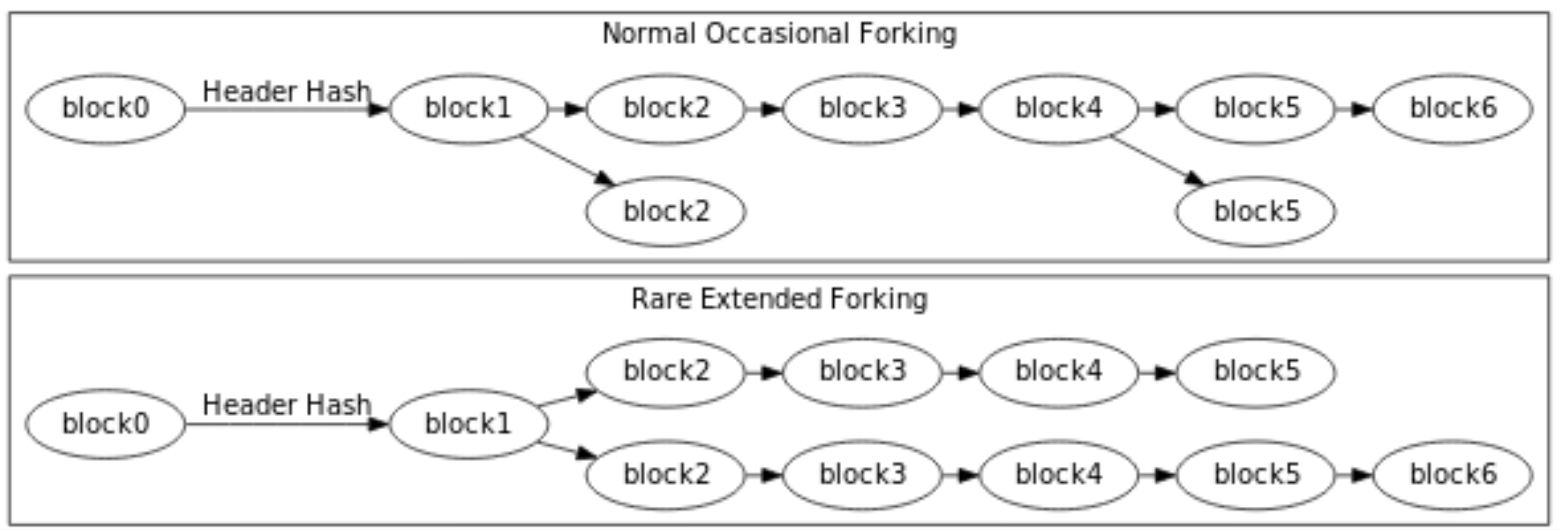
- Si ha tardado más de dos semanas, el valor que marca la dificultad se incrementa.
- Si ha tardado menos de dos semanas, el valor se decrementa.

Altura de bloque y forking

Llamamos altura de un bloque al número de bloques en la cadena entre este y el bloque génesis o bloque 0. Podría dar el caso en el que dos nodos validaran un bloque simultáneamente, lo que provocaría que hubiese dos bloques con la misma altura, lo que provocaría un aparente “fork”.

Altura de bloque y forking

Hay varios tipos de fork :



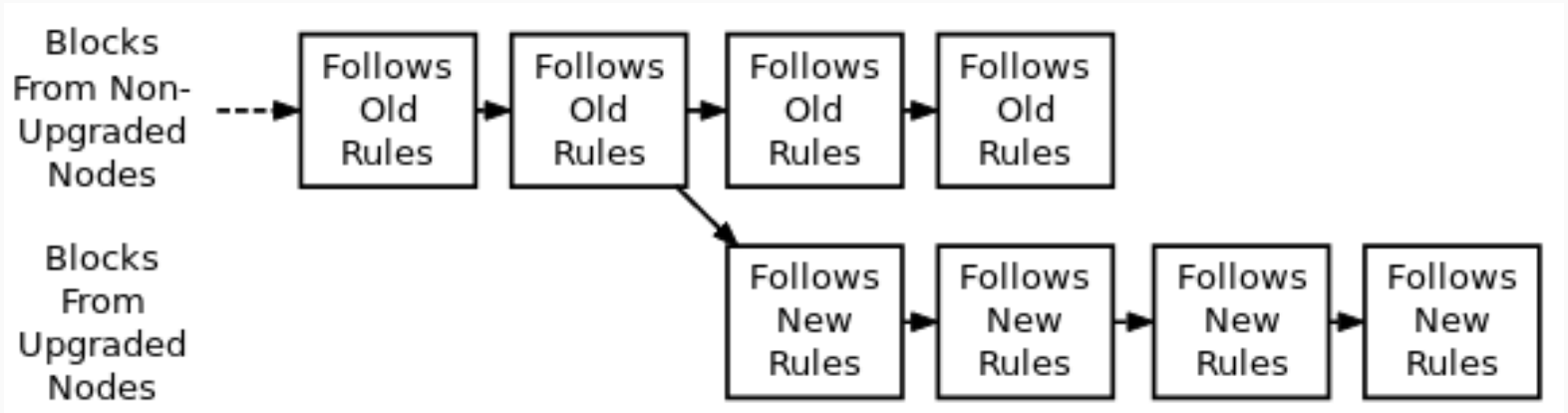
El primero se da comúnmente, mientras que el segundo es raro que ocurra, pudiendo ocurrir a causa de un ataque del 51%.

Reglas de consenso

Para conseguir que todos los nodos tengan los mismos bloques en su mejor cadena existen las llamadas reglas de consenso. Estas son modificadas cada cierto tiempo, por lo que existen lapsos de tiempo en los que algunos nodos siguen las antiguas reglas y otros las nuevas, lo que puede ocasionar los siguientes problemas :

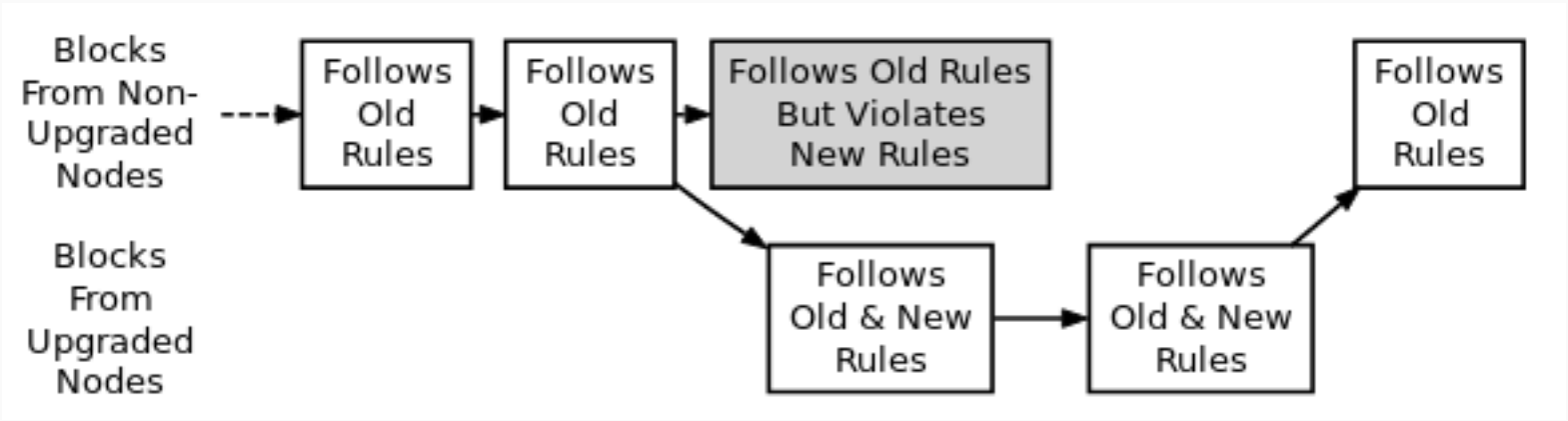
Reglas de consenso

- Un bloque que sea aceptado por las nuevas reglas no puede ser aceptado por las antiguas reglas. Esto produce un "hardfork" que tiene la siguiente forma :



Reglas de consenso

- Un bloque no aceptado por las nuevas reglas es aceptado por las antiguas reglas(supuesto que los bloques aceptados por las nuevas reglas son aceptados por las antiguas). Esto produce un “softfork” que tiene la siguiente forma :



Bitcoin

Bitcoin como red P2P

El protocolo de red Bitcoin permite que los nodos mantengan de forma colaborativa una red punto a punto para el intercambio de bloques y transacciones.

- Los nodos completos descargan y verifican cada bloque y transacción antes de retransmitirlos a otros nodos.
- Los nodos de archivo almacenan toda la cadena de bloques y pueden servir bloques históricos a otros nodos.
- Los nodos podados son nodos completos que no almacenan toda la cadena de bloques.

Descubriendo los nodos

- Cuando se inician por primera vez, los programas no conocen las direcciones IP de ninguno de los nodos completos activos.
- Para descubrir algunas direcciones IP, consultan uno o más nombres DNS que se encuentran en el código de los nodos.
- La respuesta a la búsqueda debe incluir uno o más registros DNS con direcciones IP de nodos completos que pueden aceptar nuevas conexiones entrantes.
- Las **semillas (seeds) DNS** son mantenidas por los miembros de la comunidad Bitcoin.

Descubriendo los nodos

- Los programas no deben depender exclusivamente de semillas DNS. Un atacante malicioso en el medio puede devolver solo las direcciones IP de los nodos controlados por él, aislandolo.
- Una vez conectado a la red, sus nodos pueden comenzar a enviarle mensajes addr con direcciones IP y números de puerto -> Método completamente descentralizado de descubrimiento entre iguales.
- Los nodos a menudo abandonan la red o cambian sus direcciones IP, por lo que es los programas tienen que realizar varios intentos de conexión -> Retraso
- Para evitar este posible retraso, BitcoinJ siempre usa semillas dinámicas. Bitcoin Core trata de encontrar un equilibrio entre minimizar las demoras y evitar el uso innecesario de semillas DNS.

Conectarse a los nodos

- La conexión se realiza enviando un mensaje con el número de versión, bloque y hora actual al nodo remoto.
- El nodo remoto responde con su propio mensaje de versión.
- Mensaje "verack".
- Ahora el cliente puede enviar al nodo remoto mensajes getaddr y addr para reunir nodos adicionales.
- Después de 90 min. el cliente asumirá que la conexión se ha cerrado.

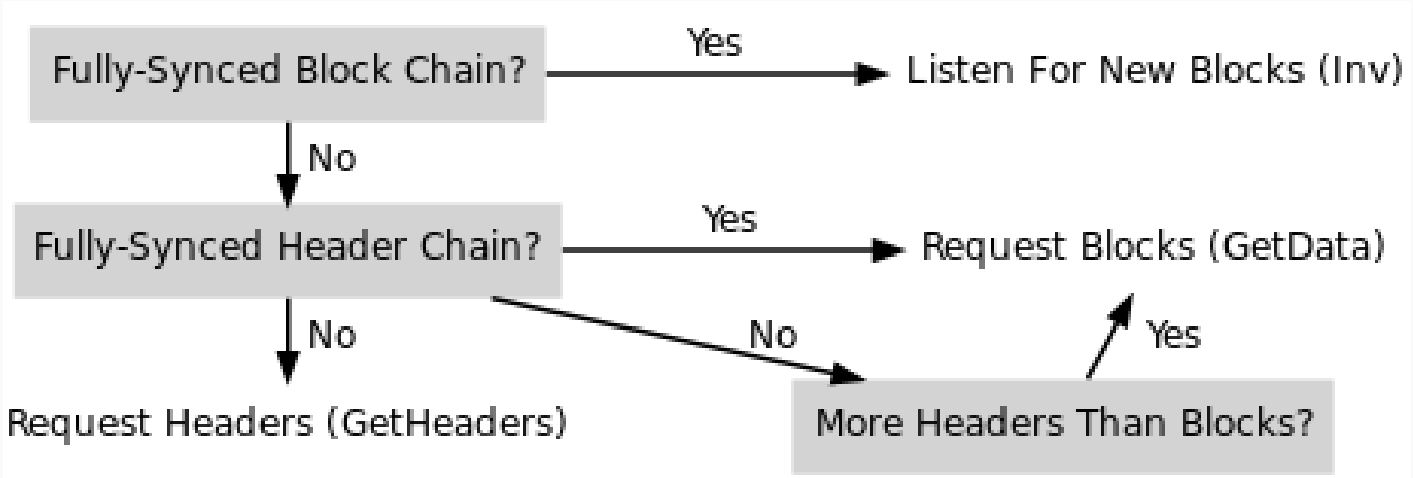
Descarga del bloque inicial

¿Qué es descargar el IBD ?

- Antes de que un nodo completo pueda validar transacciones debe descargar y validar todos los bloques desde el 1 hasta la punta actual de la mejor cadena de bloques.
- Es reutilizable cada vez que se quiera hacer una descarga de un numero considerable de bloques.
- Bitcoin Core usa el método IBD cada vez que el último bloque en su cadena de tiene un tiempo de cabecera de bloque de más de 24 horas o si su cadena de bloque local está 144 bloques más baja que su cadena de encabezado local.

Headers-First

Bitcoin Core 0.10.0 usa un método de descarga de bloque inicial (IBD) llamado "Headers-first". El objetivo es descargar las cabeceras para la mejor cadena de cabeceras, validarlos parcialmente lo mejor posible y luego descargar los bloques correspondientes en paralelo.



Overview Of Headers-First Initial Blocks Download (IBD)

Headers-First

La primera vez que se inicia un nodo tiene un bloque en su cadena de mejor bloque local : el bloque génesis codificado (bloque 0). El nodo elige un nodo remoto, que llamaremos nodo de sincronización, y le envía el mensaje “getheader” :

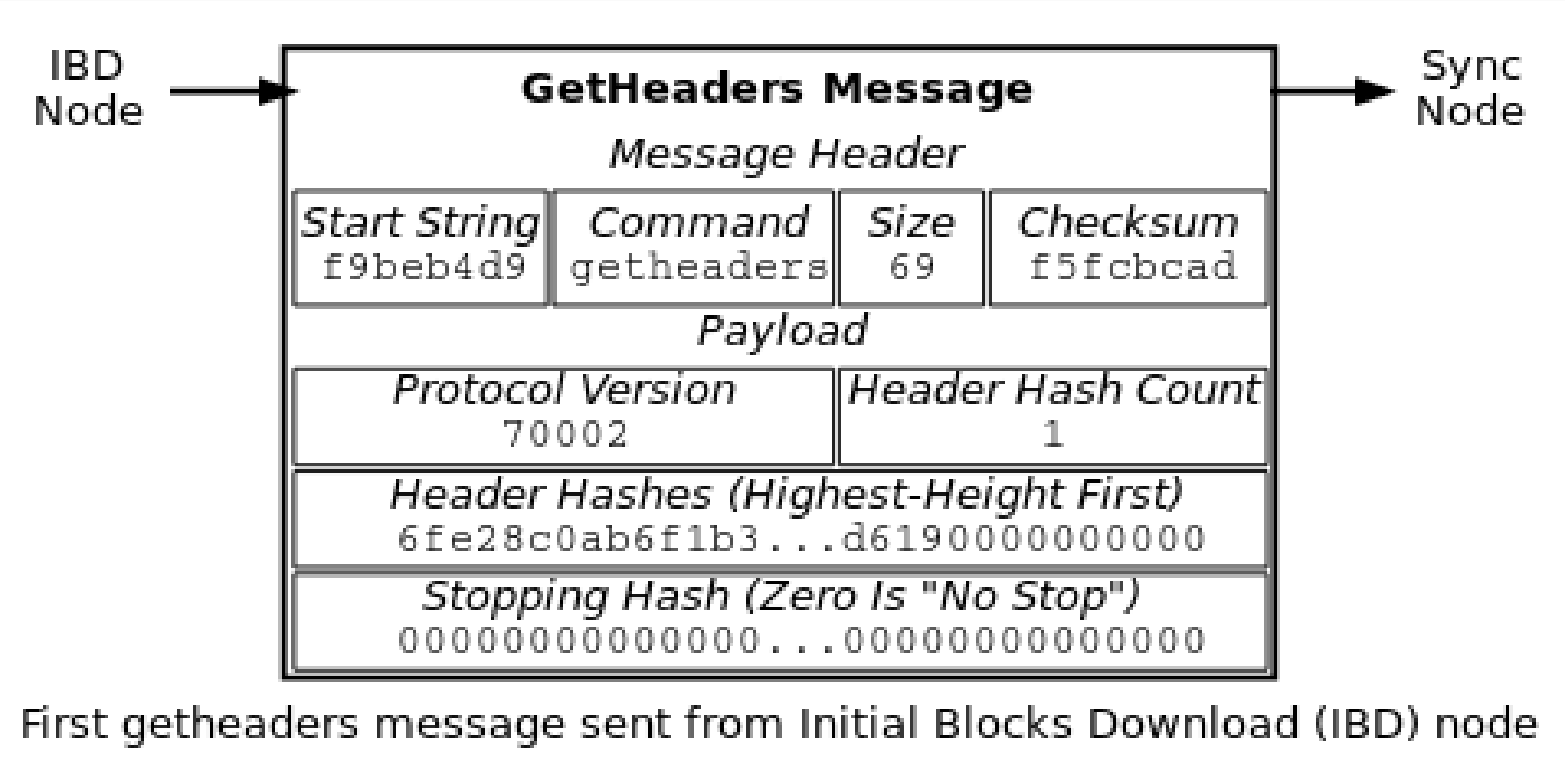


Figure 1 – GetHeaders Message

Headers-First

- En el campo de hash de cabecera del mensaje “getheaders”, el nuevo nodo envía el hash de encabezado del único bloque que tiene, el bloque de generación (6fe2 ... 0000). También establece el campo de parada de hash en todos los ceros para solicitar una respuesta de tamaño máximo.
- El nodo de sincronización toma el primer hash de encabezado y busca en su mejor cadena de bloque local un bloque con ese hash. Encuentra que el bloque 0 coincide, por lo que responde con un 2.000 cabeceras a partir del bloque 1.

Headers-First

Los hashes de estas cabeceras los manda en el mensaje mostrado a continuación :

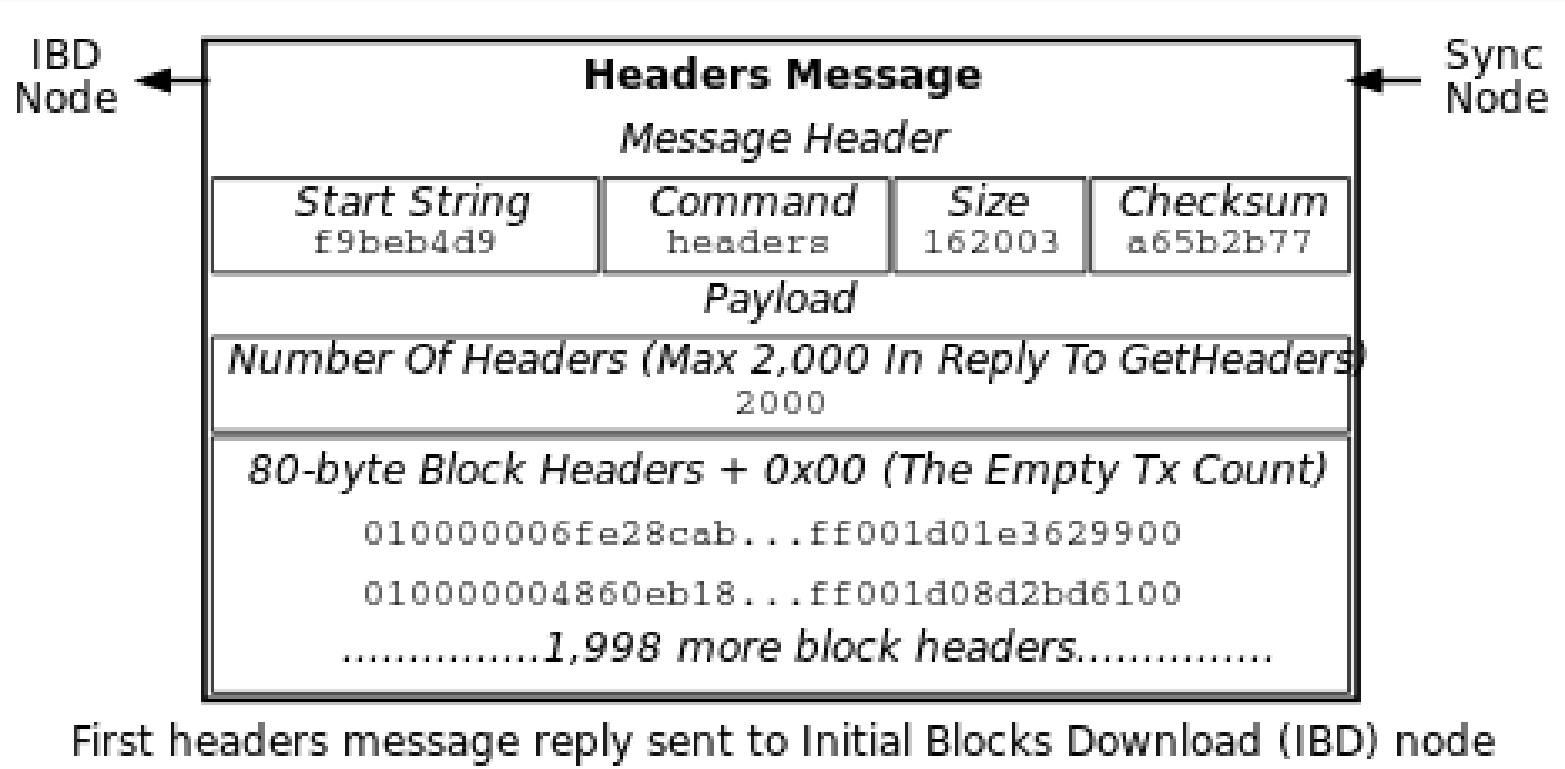


Figure 2 – Headers Message

Headers-First

El nodo IBD puede validar parcialmente estos encabezados.
Una vez validados parcialmente los encabezados se pueden hacer dos cosas en paralelo :

1. **Descargar más encabezados :** el nodo IBD puede enviar otro "getheaders".
2. **Descargar bloques :** mientras el nodo IBD continúa descargando cabeceras, y después de que terminen de descargarse, el nodo IBD solicitará y descargará cada bloque.

Bitcoin Core solicitará un máximo de 128 bloques simultáneamente.

Emisión de bloques

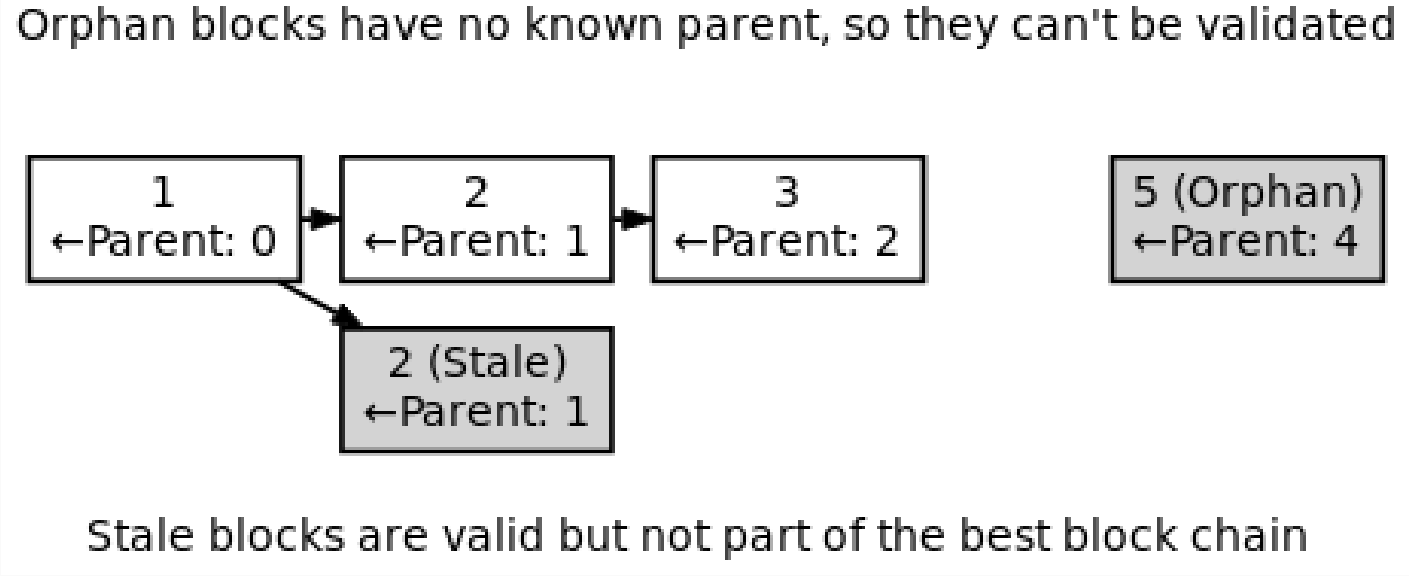
Cuando un minero descubre un nuevo bloque, transmite el nuevo bloque a sus nodos conectados

- **Unsolicited Block Push :** el minero envía un mensaje de bloque a cada uno de sus peers completos con el nuevo bloque.
- **Standard Block Relay :** el minero, que actúa como un nodo de retransmisor estándar, envía un mensaje "inv" a cada uno de sus nodos conectados con un inventario que hace referencia al nuevo bloque.
- **Direct Headers Announcement :** un nodo retransmisor puede omitir la sobrecarga de ida y vuelta de un mensaje "inv" seguido de encabezados obsoletos mediante el envío de un mensaje de encabezado.

Por defecto, Bitcoin Core difunde bloques con la 3ª política a todos los nodos que han enviado señales con "sendheaders" y la 2ª a los

Bloques huérfanos

Bloques cuyo campo hash de encabezado del bloque anterior hace referencia a un encabezado de bloque que este nodo aún no ha visto, es decir, no tienen padres conocidos.



Memory pool

- Debido a que las transacciones no confirmadas no tienen un estado permanente en Bitcoin, se almacenan en memoria no persistente, llamándolas “memory pool”.
- Cuando un par se apaga, su “memory pool” se pierde a excepción de cualquier transacción almacenada por su billetera ("wallet").
- Las transacciones que se extraen en bloques que luego se vuelven obsoletos se pueden volver a agregar a “memory pool”.
- Bitcoin Core elimina los bloques obsoletos de la cadena uno por uno, comenzando con la punta (bloque más alto).

Nodos con mal comportamiento

- Existen mecanismos para castigar a los nodos que toman ancho de banda y recursos informáticos mediante el envío de información falsa.
- Si un peer obtiene un banscore por encima del umbral ($-banscore =< n >$), estará prohibido durante el número de segundos definidos por $-bantime =< n >$, que es 86400s=24h.

Bitcoin

Minado

La minería agrega nuevos bloques a la cadena de bloques, por lo que es difícil modificar el historial de transacciones. Toma dos formas :

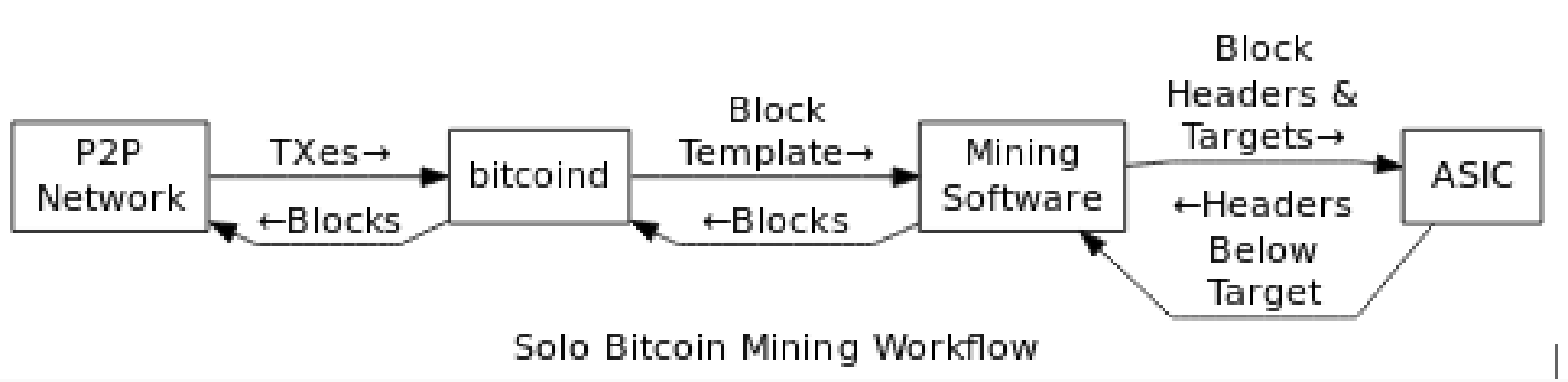
- Minería en solitario.
- Minería combinada.

Minería en solitario

- El minero intenta generar nuevos bloques por su cuenta, donde genera ganancias a partir del "premio" por bloque y de las tasas por transacción.
- Usan “bitcoind" para obtener nuevas transacciones de la red. Su software de minería solicita periódicamente nuevas transacciones utilizando el RPC “getblocktemplate".

Minería en solitario

- El software de minería construye un bloque usando una plantilla y crea un encabezado de bloque.
- Se envía el encabezado del bloque a su hardware de minería (un ASIC). El hardware de minería itera a través de cada valor posible para el encabezado del bloque y genera el hash correspondiente.



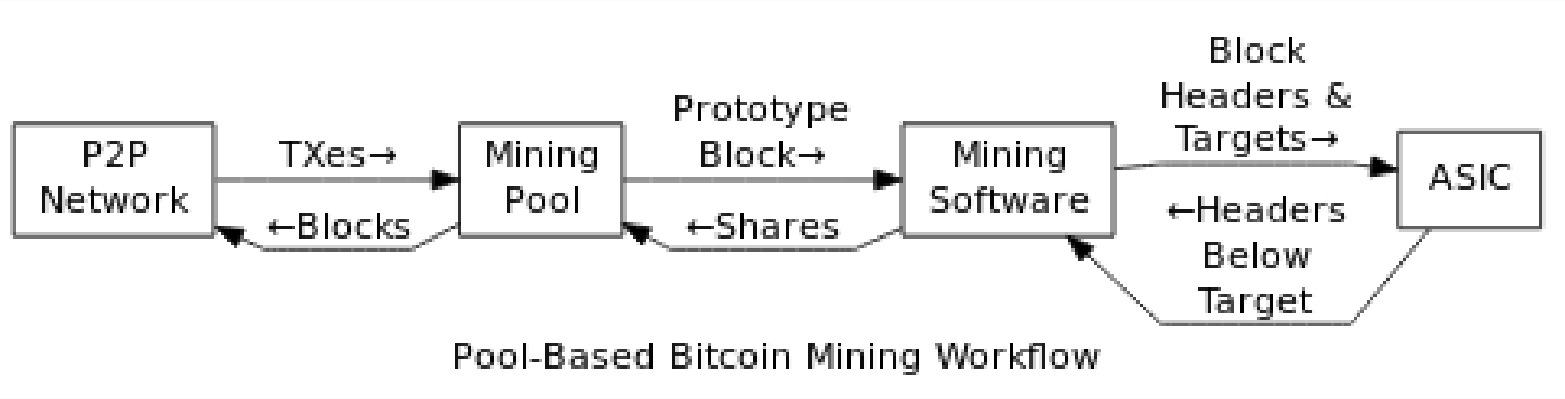
Solo Bitcoin Mining Workflow

Minería combinada

- El minero reúne recursos con otros mineros para encontrar bloques más a menudo, con los beneficios compartidos entre los mineros en una correlación aproximada con la cantidad de potencia de hashing que aportaron.
- El software de minería de cada minero se conecta al grupo y solicita la información que necesita para construir encabezados de bloque.
- El minero envía al grupo una copia de la información que el grupo necesita para validar que el encabezado tiene un hash debajo del objetivo y que el bloque de transacciones es válido.

Minería combinada

- La información que el minero envía al grupo se denomina participación.
- Las tarifas de recompensa y transacción del bloque se pagan al grupo de minería.
- El grupo minero paga una parte de estos ingresos a mineros.



Pool-Based Bitcoin Mining Workflow

Bitcoin

Demo de una cadena de bloques

Demo

El archivo fuente de la cadena de bloques se encuentra en https ://gi-thub.com/guillegalor/P2P_BITCOIN/blob/master/blockchain.py

Demo

Para interactuar con la cadena lo haremos a través de mensajes http, y en concreto usando curl. Los mensajes para interactuar con ella son los siguiente :

- curl -X POST -H "Content-Type : application/json" -d '{
  "sender" : "...",
  "recipient" : "...",
  "amount" : n
  }' "http ://localhost :5000/transactions/new"
- curl "http ://localhost :5000/mine"
- curl "http ://localhost :5000/chain"

Demo

- curl -X POST -H "Content-Type : application/json" -d '{ "nodes" : ["http ://localhost :5000/"] }' "http ://localhost :500/nodes/register"
- curl "http ://localhost :5000/nodes/resolve"

Agradecimientos

- Daniel van Flymen, @dvf
  https ://github.com/dvf/blockchain — Repositorio original con la cadena de bloques
  https ://blockchain.works-hub.com/blog/Learn-Blockchains-by-Building-One — Articulo dónde explica el código completo
- Daniel Pozo Escalona, @danipozo

Bibliografía

Bibliografía

- https ://es.wikipedia.org/wiki/Peer-to-peer
- https ://es.wikipedia.org/wiki/P2P_privado
- https ://es.wikipedia.org/wiki/Friend-to-friend
- https ://es.wikipedia.org/wiki/Bitcoin
- https ://en.wikipedia.org/wiki/Bitcoin_network
- www.deic.uab.cat/ cperez/papers/FC2014-donet-perez-herrera.pdf
- https ://bandaancha.eu/foros/todos-programas-p2p-aqui-839441
- https ://es.wikipedia.org/wiki/Bitcoin
- https ://en.wikipedia.org/wiki/Bitcoin
- https ://www.bitcoin.com/
- https ://en.bitcoin.it/wiki/