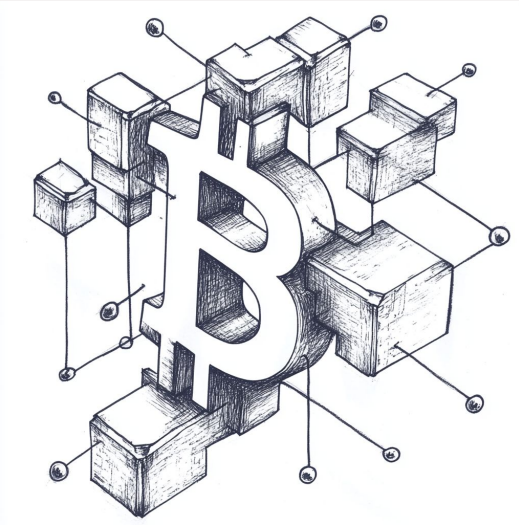
Algoritmos de consenso y Forks



Alejandro Narancio
ale.narancio@gmail.com
@anarancio

Algoritmos de consenso - Que intentan resolver?

En un sistema distribuido donde múltiples nodos operan independientemente

Algoritmos de consenso - Que intentan resolver?

En un sistema distribuido donde múltiples nodos operan independientemente

Cómo aseguramos que todos los nodos tengan la misma información?

Algoritmos de consenso - Que intentan resolver?

En un sistema distribuido donde múltiples nodos operan independientemente

Cómo aseguramos que todos los nodos tengan la misma información?

Cómo determinamos qué transacciones son válidas?

Algoritmos de consenso - Que intentan resolver?

En un sistema distribuido donde múltiples nodos operan independientemente

Cómo aseguramos que todos los nodos tengan la misma información?

Cómo determinamos qué transacciones son válidas?

Cómo evitamos el doble gasto sin una autoridad central?

Algoritmos de consenso - Que intentan resolver?

En un sistema distribuido donde múltiples nodos operan independientemente

Cómo aseguramos que todos los nodos tengan la misma información?

Cómo determinamos qué transacciones son válidas?

Cómo evitamos el doble gasto sin una autoridad central?

Cómo protegemos el sistema contra ataques?

Byzantine Generals' Problem

Quien lo definió?
Leslie Lamport, Robert Shostak y Marshall Pease en 1982

Byzantine Generals' Problem

Quien lo definió?
Leslie Lamport, Robert Shostak y Marshall Pease en 1982

Que ilustra?
Ilustra el problema al que se enfrenta un sistema distribuido para alcanzar un consenso cuando la comunicación no es confiable y algunos actores pueden tener errores o ser mal intencionados (maliciosos)

Byzantine Generals' Problem

Quien lo definió?
Leslie Lamport, Robert Shostak y Marshall Pease en 1982

Que ilustra?
Ilustra el problema al que se enfrenta un sistema distribuido para alcanzar un consenso cuando la comunicación no es confiable y algunos actores pueden tener errores o ser mal intencionados (maliciosos)

Como se ilustra el problema?
Varios generales deben coordinar un ataque a una ciudad, pero solo pueden comunicarse por mensajeros que podrían ser interceptados o corruptos. A su vez algunos generales pueden ser traidores que envian mensajes contradictorios para sabotear el plan.

Byzantine Generals' Problem

Quien lo definió?
Leslie Lamport, Robert Shostak y Marshall Pease en 1982

Que ilustra?
Ilustra el problema al que se enfrenta un sistema distribuido para alcanzar un consenso cuando la comunicación no es confiable y algunos actores pueden tener errores o ser mal intencionados (maliciosos)

Como se ilustra el problema?
Varios generales deben coordinar un ataque a una ciudad, pero solo pueden comunicarse por mensajeros que podrían ser interceptados o corruptos. A su vez algunos generales pueden ser traidores que envian mensajes contradictorios para sabotear el plan.

Cual es el desafío?
Cómo pueden los generales leales, decidir si atacan o no?

PBFT (Practical Byzantine Fault Tolerance)

Quien lo definió?
Miguel Castro y Barbara Liskov en 1999

PBFT (Practical Byzantine Fault Tolerance) - requisitos

Número mínimo de nodos
Para tolerar f nodos con errores o maliciosos se requieren 3f + 1 nodos. Es decir para tolerar 1 nodo malicioso se requieren 4 nodos, para tolerar 2 se requieren 7, etc.

PBFT (Practical Byzantine Fault Tolerance) - requisitos

Número mínimo de nodos
Para tolerar f nodos con errores o maliciosos se requieren 3f + 1 nodos. Es decir para tolerar 1 nodo malicioso se requieren 4 nodos, para tolerar 2 se requieren 7, etc.

Autenticación
Todos los mensajes deben estar firmados digitalmente para garantizar su autenticidad.

PBFT (Practical Byzantine Fault Tolerance) - requisitos

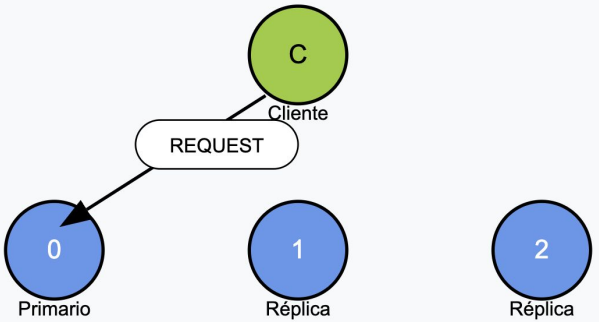
Número mínimo de nodos
Para tolerar f nodos con errores o maliciosos se requieren 3f + 1 nodos. Es decir para tolerar 1 nodo malicioso se requieren 4 nodos, para tolerar 2 se requieren 7, etc.

Autenticación
Todos los mensajes deben estar firmados digitalmente para garantizar su autenticidad.

Red asíncrona
El algoritmo puede manejar retrasos en la red, pero asume que los mensajes eventualmente llegarán.

PBFT (Practical Byzantine Fault Tolerance) - Funcionamiento

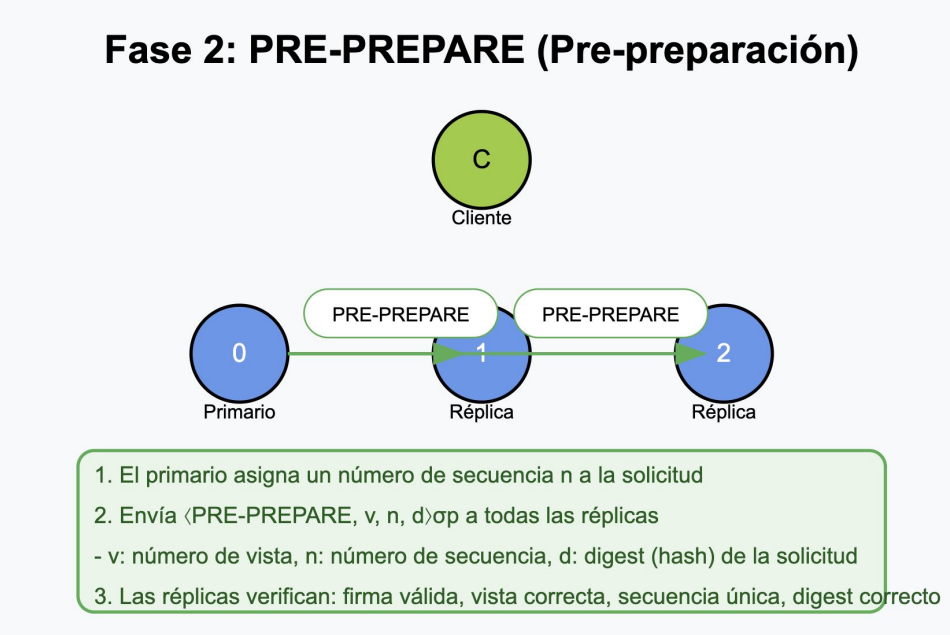
REQUEST
Cliente envía una solicitud al nodo primario



1. El cliente envía una solicitud con el formato: ⟨REQUEST, o, t, c⟩σc
- o: operación solicitada, t: timestamp, c: ID del cliente, σc: firma digital
2. La solicitud se envía al nodo primario (líder actual del sistema)

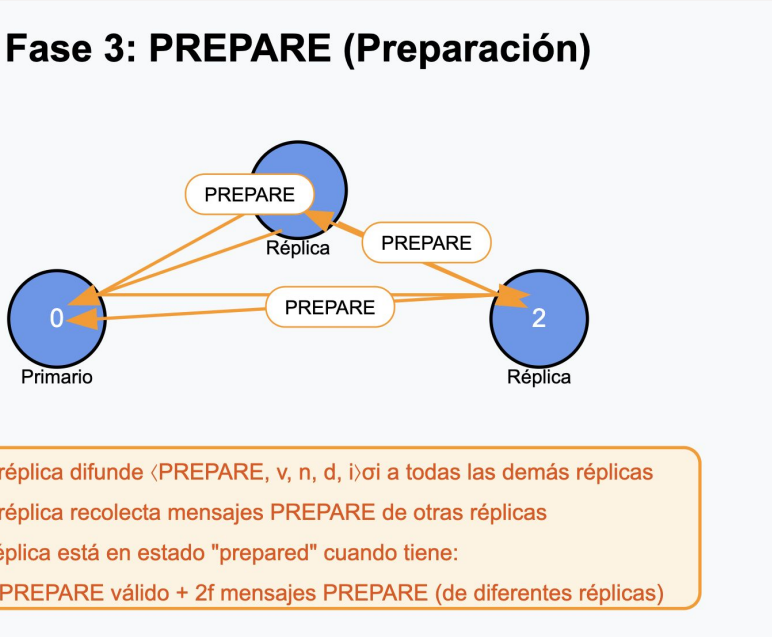
PBFT (Practical Byzantine Fault Tolerance) - Funcionamiento

PRE-PREPARE



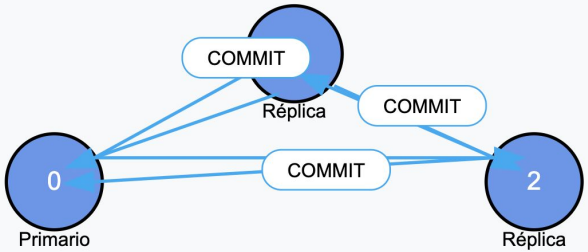
PBFT (Practical Byzantine Fault Tolerance) - Funcionamiento

PREPARE
Si la verificación es exitosa, cada réplica acepta el mensaje PRE-PREPARE y difunde un mensaje PREPARE a todas las demás réplicas.



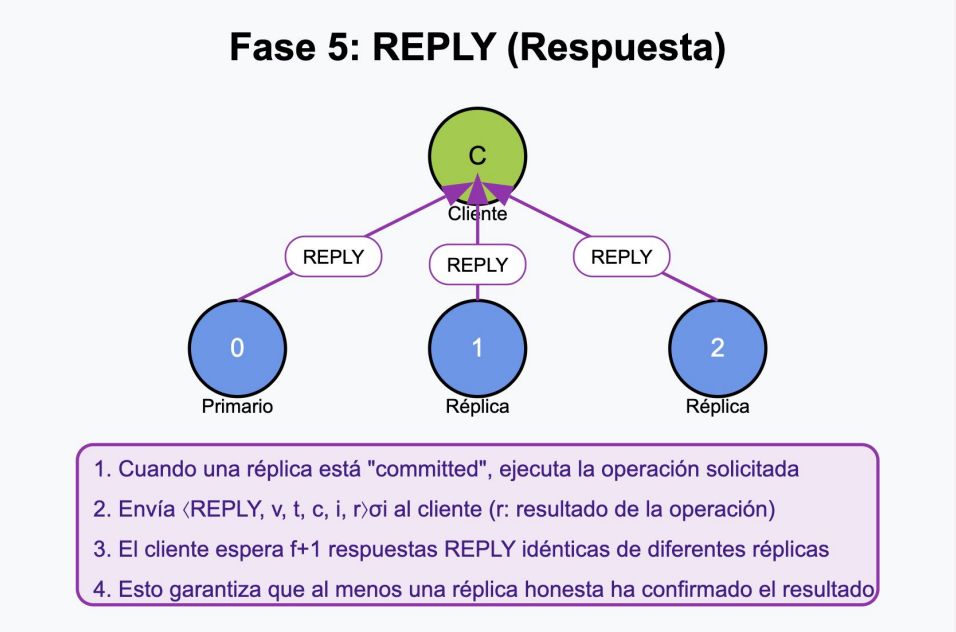
PBFT (Practical Byzantine Fault Tolerance) - Funcionamiento

COMMIT
Cuando una réplica alcanza el estado "prepared", difunde un mensaje COMMIT a todas las demás réplicas.



PBFT (Practical Byzantine Fault Tolerance) - Funcionamiento

REPLY
Cuando una réplica alcanza el estado "committed", ejecuta la operación solicitada por el cliente.



Algoritmos de consenso - Proof of Work (PoW)

Este es el algoritmo original implementado por Bitcoin

Algoritmos de consenso - Proof of Work (PoW)

Este es el algoritmo original implementado por Bitcoin

Mineros compiten para resolver un problema matemático complejo

Algoritmos de consenso - Proof of Work (PoW)

Este es el algoritmo original implementado por Bitcoin

Mineros compiten para resolver un problema matemático complejo

El primer minero en resolver el problema obtiene el derecho de añadir un nuevo bloque

Algoritmos de consenso - Proof of Work (PoW)

Este es el algoritmo original implementado por Bitcoin

Mineros compiten para resolver un problema matemático complejo

El primer minero en resolver el problema obtiene el derecho de añadir un nuevo bloque

La seguridad se basa en el poder computacional: para atacar la red, un adversario necesitaría más del 50% del poder de cómputo total

Algoritmos de consenso - Proof of Work (PoW)

Este es el algoritmo original implementado por Bitcoin

Mineros compiten para resolver un problema matemático complejo

El primer minero en resolver el problema obtiene el derecho de añadir un nuevo bloque

La seguridad se basa en el poder computacional: para atacar la red, un adversario necesitaría más del 50% del poder de cómputo total

Ventajas

Alta seguridad probada a lo largo del tiempo

Algoritmos de consenso - Proof of Work (PoW)

Este es el algoritmo original implementado por Bitcoin

Mineros compiten para resolver un problema matemático complejo

El primer minero en resolver el problema obtiene el derecho de añadir un nuevo bloque

La seguridad se basa en el poder computacional: para atacar la red, un adversario necesitaría más del 50% del poder de cómputo total

Ventajas

Alta seguridad probada a lo largo del tiempo

Sistema descentralizado sin puntos únicos de falla

Algoritmos de consenso - Proof of Work (PoW)

Este es el algoritmo original implementado por Bitcoin

Mineros compiten para resolver un problema matemático complejo

El primer minero en resolver el problema obtiene el derecho de añadir un nuevo bloque

La seguridad se basa en el poder computacional: para atacar la red, un adversario necesitaría más del 50% del poder de cómputo total

Ventajas

Alta seguridad probada a lo largo del tiempo

Sistema descentralizado sin puntos únicos de falla

Desventajas

Consumo energético alto

Algoritmos de consenso - Proof of Work (PoW)

Este es el algoritmo original implementado por Bitcoin

Mineros compiten para resolver un problema matemático complejo

El primer minero en resolver el problema obtiene el derecho de añadir un nuevo bloque

La seguridad se basa en el poder computacional: para atacar la red, un adversario necesitaría más del 50% del poder de cómputo total

Ventajas

Alta seguridad probada a lo largo del tiempo

Sistema descentralizado sin puntos únicos de falla

Desventajas

Consumo energético alto

Baja escalabilidad (10-20 transacciones por segundo en Bitcoin)

Algoritmos de consenso - Proof of Work (PoW)

Este es el algoritmo original implementado por Bitcoin

Mineros compiten para resolver un problema matemático complejo

El primer minero en resolver el problema obtiene el derecho de añadir un nuevo bloque

La seguridad se basa en el poder computacional: para atacar la red, un adversario necesitaría más del 50% del poder de cómputo total

Ventajas

Alta seguridad probada a lo largo del tiempo

Sistema descentralizado sin puntos únicos de falla

Desventajas

Consumo energético alto

Baja escalabilidad (10-20 transacciones por segundo en Bitcoin)

Tendencia a la centralización debido a economías de escala en minería

Algoritmos de consenso - Proof of Stake (PoS)

Los validadores apuestan ("stake") sus propias criptomonedas

Algoritmos de consenso - Proof of Stake (PoS)

Los validadores apuestan ("stake") sus propias criptomonedas

La probabilidad de ser elegido para validar un bloque es proporcional a la cantidad apostada

Algoritmos de consenso - Proof of Stake (PoS)

Los validadores apuestan ("stake") sus propias criptomonedas

La probabilidad de ser elegido para validar un bloque es proporcional a la cantidad apostada

Los validadores deshonestos pierden parte o la totalidad de sus fondos apostados

Algoritmos de consenso - Proof of Stake (PoS)

Los validadores apuestan ("stake") sus propias criptomonedas

La probabilidad de ser elegido para validar un bloque es proporcional a la cantidad apostada

Los validadores deshonestos pierden parte o la totalidad de sus fondos apostados

Ventajas

Consumo energético mucho menor que PoW

Algoritmos de consenso - Proof of Stake (PoS)

Los validadores apuestan ("stake") sus propias criptomonedas

La probabilidad de ser elegido para validar un bloque es proporcional a la cantidad apostada

Los validadores deshonestos pierden parte o la totalidad de sus fondos apostados

Ventajas

Consumo energético mucho menor que PoW

Mayor escalabilidad potencial

Algoritmos de consenso - Proof of Stake (PoS)

Los validadores apuestan ("stake") sus propias criptomonedas

La probabilidad de ser elegido para validar un bloque es proporcional a la cantidad apostada

Los validadores deshonestos pierden parte o la totalidad de sus fondos apostados

Ventajas

Consumo energético mucho menor que PoW

Mayor escalabilidad potencial

Barrera de entrada más baja para participar en la validación

Algoritmos de consenso - Proof of Stake (PoS)

Los validadores apuestan ("stake") sus propias criptomonedas

La probabilidad de ser elegido para validar un bloque es proporcional a la cantidad apostada

Los validadores deshonestos pierden parte o la totalidad de sus fondos apostados

Ventajas

Consumo energético mucho menor que PoW

Mayor escalabilidad potencial

Barrera de entrada más baja para participar en la validación

Desventajas

Posible centralización por concentración de riqueza ("the rich get richer")

Algoritmos de consenso - Proof of Stake (PoS)

Los validadores apuestan ("stake") sus propias criptomonedas

La probabilidad de ser elegido para validar un bloque es proporcional a la cantidad apostada

Los validadores deshonestos pierden parte o la totalidad de sus fondos apostados

Ventajas

Consumo energético mucho menor que PoW

Mayor escalabilidad potencial

Barrera de entrada más baja para participar en la validación

Desventajas

Posible centralización por concentración de riqueza ("the rich get richer")

Menos probado en la práctica que PoW

Algoritmos de consenso - Delegated Proof of Stake (DPoS)

Los poseedores de tokens votan por un número limitado de delegados

Algoritmos de consenso - Delegated Proof of Stake (DPoS)

Los poseedores de tokens votan por un número limitado de delegados

Solo estos delegados pueden validar transacciones y crear bloques

Algoritmos de consenso - Delegated Proof of Stake (DPoS)

Los poseedores de tokens votan por un número limitado de delegados

Solo estos delegados pueden validar transacciones y crear bloques

Sistema democrático donde los delegados pueden ser reemplazados por votación

Algoritmos de consenso - Delegated Proof of Stake (DPoS)

Los poseedores de tokens votan por un número limitado de delegados

Solo estos delegados pueden validar transacciones y crear bloques

Sistema democrático donde los delegados pueden ser reemplazados por votación

Ventajas

Alta eficiencia y escalabilidad (miles de transacciones por segundo)

Algoritmos de consenso - Delegated Proof of Stake (DPoS)

Los poseedores de tokens votan por un número limitado de delegados

Solo estos delegados pueden validar transacciones y crear bloques

Sistema democrático donde los delegados pueden ser reemplazados por votación

Ventajas

Alta eficiencia y escalabilidad (miles de transacciones por segundo)

Gobierno más democrático

Algoritmos de consenso - Delegated Proof of Stake (DPoS)

Los poseedores de tokens votan por un número limitado de delegados

Solo estos delegados pueden validar transacciones y crear bloques

Sistema democrático donde los delegados pueden ser reemplazados por votación

Ventajas

Alta eficiencia y escalabilidad (miles de transacciones por segundo)

Gobierno más democrático

Bajo consumo energético

Algoritmos de consenso - Delegated Proof of Stake (DPoS)

Los poseedores de tokens votan por un número limitado de delegados

Solo estos delegados pueden validar transacciones y crear bloques

Sistema democrático donde los delegados pueden ser reemplazados por votación

Ventajas

Alta eficiencia y escalabilidad (miles de transacciones por segundo)

Gobierno más democrático

Bajo consumo energético

Desventajas

Mayor centralización (típicamente solo 21-100 validadores)

Algoritmos de consenso - Delegated Proof of Stake (DPoS)

Los poseedores de tokens votan por un número limitado de delegados

Solo estos delegados pueden validar transacciones y crear bloques

Sistema democrático donde los delegados pueden ser reemplazados por votación

Ventajas

Alta eficiencia y escalabilidad (miles de transacciones por segundo)

Gobierno más democrático

Bajo consumo energético

Desventajas

Mayor centralización (típicamente solo 21-100 validadores)

Posible formación de “carteles” entre delegados

Algoritmos de consenso - Proof of Authority (PoA)

Solo nodos previamente autorizados pueden validar transacciones

Algoritmos de consenso - Proof of Authority (PoA)

Solo nodos previamente autorizados pueden validar transacciones

La identidad de los validadores es conocida y verificada

Algoritmos de consenso - Proof of Authority (PoA)

Solo nodos previamente autorizados pueden validar transacciones

La identidad de los validadores es conocida y verificada

Utilizada en blockchains permisionadas o privadas

Algoritmos de consenso - Proof of Authority (PoA)

Solo nodos previamente autorizados pueden validar transacciones

La identidad de los validadores es conocida y verificada

Utilizada en blockchains permisionadas o privadas

Ventajas

Extremadamente eficiente y rápido

Algoritmos de consenso - Proof of Authority (PoA)

Solo nodos previamente autorizados pueden validar transacciones

La identidad de los validadores es conocida y verificada

Utilizada en blockchains permisionadas o privadas

Ventajas

Extremadamente eficiente y rápido

Sin necesidad de tokens o incentivos económicos

Algoritmos de consenso - Proof of Authority (PoA)

Solo nodos previamente autorizados pueden validar transacciones

La identidad de los validadores es conocida y verificada

Utilizada en blockchains permisionadas o privadas

Ventajas

Extremadamente eficiente y rápido

Sin necesidad de tokens o incentivos económicos

Adecuado para casos de uso empresariales o gubernamentales

Algoritmos de consenso - Proof of Authority (PoA)

Solo nodos previamente autorizados pueden validar transacciones

La identidad de los validadores es conocida y verificada

Utilizada en blockchains permisionadas o privadas

Ventajas

Extremadamente eficiente y rápido

Sin necesidad de tokens o incentivos económicos

Adecuado para casos de uso empresariales o gubernamentales

Desventajas

Altamente centralizado

Algoritmos de consenso - Proof of Authority (PoA)

Solo nodos previamente autorizados pueden validar transacciones

La identidad de los validadores es conocida y verificada

Utilizada en blockchains permisionadas o privadas

Ventajas

Extremadamente eficiente y rápido

Sin necesidad de tokens o incentivos económicos

Adecuado para casos de uso empresariales o gubernamentales

Desventajas

Altamente centralizado

Requiere confianza en los validadores

Forks - Conceptos

Qué son?
Son bifurcaciones de la cadena activa o válida

Forks - Conceptos

Qué son?
Son bifurcaciones de la cadena activa o válida

Cuándo ocurren?
Ocurren cuando los nodos se desincronizan, temporal o permanentemente

Forks - Conceptos

Qué son?
Son bifurcaciones de la cadena activa o válida

Cuándo ocurren?
Ocurren cuando los nodos se desincronizan, temporal o permanentemente

Con qué frecuencia ocurren?
Temporales (o accidentales) ocurren casi constantemente, el resto se da con cambios de reglas en las reglas de consenso.

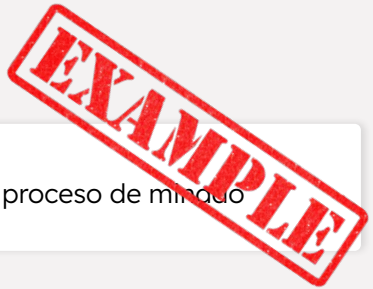
Forks - Conceptos

Qué son?
Son bifurcaciones de la cadena activa o válida

Cuándo ocurren?
Ocurren cuando los nodos se desincronizan, temporal o permanentemente

Con qué frecuencia ocurren?
Temporales (o accidentales) ocurren casi constantemente, el resto se da con cambios de reglas en las reglas de consenso.

Ejemplo
Ejemplo de forks temporales durante proceso de minado



Forks - Conceptos

Qué son?
Son bifurcaciones de la cadena activa o válida

Cuándo ocurren?
Ocurren cuando los nodos se desincronizan, temporal o permanentemente

Con qué frecuencia ocurren?
Temporales (o accidentales) ocurren casi constantemente, el resto se da con cambios de reglas en las reglas de consenso.

Tipos
Hard fork y Soft fork