Smart Bitcoin Cash: una Sidechain de Bitcoin Cash con Compatibilidad EVM y Web3.

Resumen

Mientras que Bitcoin Cash pretende proporcionar una infraestructura descentralizada, de alto rendimiento, de bajo coste y fácil de usar para la criptomoneda; cualquier cambio en la red principal requiere un alto nivel de consenso, lo que dificulta el proceso de prueba y error.

Así que decidimos desarrollar Smart Bitcoin Cash - una cadena lateral para Bitcoin Cash con el objetivo de explorar nuevas ideas y ofrecer un vasto universo de posibilidades. Será compatible con el EVM de Ethereum y la API Web3, ya que son los estándares de facto para las DApps de blockchain hoy en día.

Ethereum está resolviendo los problemas relacionados con el bajo rendimiento y el alto coste mediante el cambio a ETH2.0, que, como todos sabemos, aún requiere años de desarrollo para terminar. Smart Bitcoin Cash intenta abordar estos problemas de manera diferente, optimizando la implementación de EVM y Web3 a bajo nivel para aprovechar al máximo el potencial del hardware, especialmente su paralelismo inherente. Creemos que Smart Bitcoin Cash proporcionará los mismos beneficios de ETH2.0 en un tiempo mucho menor.

Motivación

Muchos usuarios desean nuevas y diversas características en Bitcoin Cash.

El intervalo de bloques de Bitcoin Cash sigue siendo de 10 minutos,
lo que resulta demasiado largo en comparación a otras cadenas que ofrecen intervalos de solamente algunos segundos. A pesar de que Bitcoin Cash soporta transacciones seguras con cero confirmaciones, casos más complejos que van más allá del simple procesamiento de pagos requieren plazos de confirmación muy cortos para una mejor experiencia de usuario, como es en el caso de DeFi.

Bitcoin Cash cuenta con un sistema de secuencia de comandos (script) limitado que no es Turing-completo, lo que hace que sea más difícil de utilizar que la EVM de Ethereum. Además, es menos versátil que la EVM junto a Solidity, ya que estos ofrecen el mejor ecosistema y acaparan a la mayoría de los programadores disponibles entre todas las plataformas de contratos inteligentes. Es una lástima que no estén pudiendo ser aprovechados por el ecosistema de Bitcoin Cash.

Actualmente, se ha demostrado que la capacidad de la red de Bitcoin Cash alcanza los 14 MB. A pesar de que en este momento su tamaño de bloque es de solamente 0,8 MB en promedio, se

encuentra creciendo rápidamente: más del triple desde comienzos de 2021. De continuar creciendo a ese ritmo, es posible que los 14 MB no sean suficientes en un futuro próximo. Dado que los problemas que puedan llegar a surgir relacionados con tamaños de bloque superiores a 14 MB siguen siendo aún desconocidos y no han sido probados en la práctica, sería mejor empezar a prepararse para un mayor escalado de Bitcoin Cash.

¿Qué puede aportar Smart Bitcoin Cash al ecosistema de Bitcoin Cash? Pues bien, brinda la infraestructura para poner a prueba nuevas características: Posee un tiempo de confirmación corto, soporta EVM y Web3 con novedosas optimizaciones que permiten proporcionar un mayor rendimiento, y también aporta nuevos canales para que nuevos usuarios puedan ser invitados a unirse al ecosistema de Bitcoin Cash.

A medida que Smart Bitcoin Cash vaya madurando, las bibliotecas que se vayan desarrollando y las experiencias adquiridas ayudarán a mejorar también a la red principal de Bitcoin Cash.

Antecedentes

Han pasado 8 años desde que Vitalik Buterin propuso Ethereum en 2013. Los contratos inteligentes nacieron, se desarrollaron y se han destacado desde entonces. Ethereum es ahora la plataforma de contratos inteligentes más exitosa y es posible examinar su ecosistema para realizar algunas observaciones a partir de ello:

La experiencia de usuario sobre una única cadena resulta difícil de abordar a través del sharding o de soluciones de capa 2. La latencia cero y la interoperabilidad atómica entre contratos inteligentes solamente es posible dentro de la misma cadena. La interoperabilidad cruzada desde las cadenas de capa 2 hacia la capa 1 debe ser sometida a un proceso de comunicación entre ambas, lo que propicia latencias similares a las de los depósitos y retiros en exchanges centralizados. Algunos mecanismos populares, como los flash loans y los flash swaps, no estarían funcionando de manera cruzada entre ambas cadenas.

El bajo rendimiento disuade al común de los usuarios de interactuar de manera directa con las DApps. El límite máximo de gas para cada bloque es fijo y los mineros incluyen primero las transacciones que pagan altas tarifas de gas. Mientras tanto, las transacciones que ofrecen pagar bajas tarifas de gas deben esperar mucho tiempo para ser incluidas en algún bloque si es que alguna vez eso llega a suceder. Naturalmente, se justifica pagar una alta tarifa de gas para aquellas transacciones de alto valor que así lo requieran. Pero aquellos usuarios que no pueden afrontar el pago de altas tarifas de gas solamente pueden depositar sus fondos en organizaciones centralizadas para luego delegar en ellas la administración de los mismos. Solamente un pequeño número de cuentas son las que envían transacciones en el universo DeFi. Estas cuentas disponen de mucho capital, ya sea porque les pertenece o porque recaudan el capital de muchos usuarios comunes. Paradójicamente, las finanzas descentralizadas actualmente no se encuentran tan descentralizadas.

El almacenamiento demanda más recursos que el cálculo computacional cuando se ejecutan contratos inteligentes. A lo largo de la historia de Ethereum el costo del gas de almacenamiento ha sido incrementado por dos EIP (Ethereum Improvement Proposals): EIP-1884 y EIP-2200. Pero eso no es todo, ya que algunas investigaciones demuestran que Ethereum continúa subestimando algunas operaciones de almacenamiento, lo que lo hace susceptible a padecer ataques de DoS (Denial of Service). Es por eso que otro EIP, el EIP-2929, está a punto de aumentar el gas de las operaciones de alto almacenamiento en el próximo hard fork denominado Berlín. Al mismo tiempo, la aritmética de 256 bits de EVM es acelerada sustancialmente gracias a las nuevas bibliotecas provistas por Martin Holst Swende y Paweł Bylica.

El QPS (consultas por segundo) fuera de la cadena es tan importante como el TPS (transacciones por segundo) dentro de la cadena. Una DApp funciona no sólo enviando una transacción para ser ejecutada, sino también consultando el último estado de la cadena de bloques y los eventos históricos que hayan sucedido en la misma. A pesar de que una transacción se ejecuta por única vez, sus eventos paralelos y los cambios de estado pueden ser consultados muchas veces. Por lo tanto, el total de consultas por segundo (QPS) son mucho mayores al total de transacciones por segundo (TPS). En el ecosistema de Ethereum, las consultas son realizadas por medio de una API Web3, cuyo principal proveedor es Infura. Infura mantiene una implementación optimizada de Web3 que resulta ser mejor que cualquier cliente de nodo completo estándar como go-ethereum, pero no es de código abierto. Muchos son los desarrolladores que optan por utilizar el servicio de bajo costo de Infura en lugar de ejecutar su propio nodo completo. Por consiguiente, si Infura experimenta una grave interrupción de su servicio, muchas DApps y exchanges dejarían de funcionar.

Los usuarios son bastante tolerantes con la latencia de las transacciones y no prestan mucha atención al correcto orden de las mismas. Habitualmente se utiliza un monedero como MetaMask para firmar las transacciones, para ello es establecido un precio adecuado para la tarifa de gas antes de realizar la transferencia. Frecuentemente se elige un precio bajo de gas con el objetivo de ahorrar tarifas especulando con la posibilidad de que la transacción se confirme en unos pocos minutos o incluso en unas horas, en lugar de hacerlo en el siguiente bloque.

En algunos casos, otras transacciones pueden ser incluidas en el bloque antes que la suya, ocasionándole algún tipo de pérdida como un gran deslizamiento (slippage) del precio en la operación que intenta realizar. A pesar de ello, la mayoría de los usuarios parece poder convivir con eso.

También observamos que, desde 2013, la principal tendencia en los ordenadores es **añadir más núcleos a las CPU**. Comparemos un MacBook de 2013 con otro de 2021 y tratemos de predecir cómo serán en 2029:

	MacBook Pro in 2013	MacBook Pro in 2021	MacBook Pro in 2029
# CPU cores	2	8	32
Highest CPU Frequency	2.9GHz	3.1GHz	3.3GHz
Lithography for CPU	22nm	5nm	1nm

La tecnología de circuitos integrados tiene dificultades para aumentar la frecuencia una vez alcanzados los 28 nm; sin embargo, esto promueve un mayor presupuesto de transistores en las nuevas generaciones. Los diseñadores pueden utilizar estos transistores para implementar cada vez más núcleos de CPU. Entre todo el furor de los nuevos lenguajes de programación de la última década se encuentra el sencillo apalancamiento que facilita la implementación de una mayor cantidad de núcleos de CPU: channels y goroutines de Go, isolates de Dart y fearless concurrency de Rust.

Estas observaciones previas deben guiar el diseño y la implementación de Smart Bitcoin Cash.

Componentes principales de Smart Bitcoin Cash

La innovación de Smart Bitcoin Cash reside en sus librerías. En lugar de inventar sofisticados algoritmos criptográficos y de consenso, hemos decidido adoptar otra metodología: desarrollar librerías de bajo nivel con el objetivo de aprovechar al máximo el potencial del hardware, especialmente su paralelismo intrínseco.
El común de los usuarios y desarrolladores disponen de una capa de compatibilidad que soporta EVM y Web3, por lo que las librerías optimizadas de bajo nivel "close to metal" permanecen ocultas debido a esta capa de abstracción. Durante la implementación, utilizamos el nombre en clave "Moeing", el cual es añadido como prefijo al nombre de cada librería.

Con estos potentes componentes, Smart Bitcoin Cash tiene como objetivo incrementar a mil millones el consumo de gas cada 15 segundos en el mediano plazo. A largo plazo, el rendimiento general se potenciará aún más mediante la implementación de sharding y de rollups.

MoeingEVM

MoeingEVM es un motor de ejecución paralelo que actualmente administra múltiples contextos de EVM y ejecuta una gran cantidad de transacciones. Basado en una implementación de EVM optimizada por evmone, se puede observar que se han adoptado varias técnicas novedosas para maximizar el paralelismo de transacciones.

A medida que las CPUs de varios núcleos se vuelven cada vez más populares, la escalabilidad se dificultaría en el contexto semántico de ejecución de un solo hilo de Ethereum ya que debido a esa restricción el aumento de la velocidad de la red se torna muy difícil.
Pero si se utiliza la semántica de ejecución multihilo se conseguiría un mejor resultado global ya que las CPUs de múltiples núcleos pueden ser utilizadas más fácilmente y de forma más sencilla.

Así que MoeingEVM ha sido desarrollada siguiendo la semántica de ejecución multihilo.

Para aprovechar al máximo el paralelismo intrínseco del hardware moderno, tratamos de sacar provecho de dos tipos de paralelismo:

1. El paralelismo entre el motor de consenso y el motor de ejecución de transacciones.
2. El paralelismo entre las diferentes transacciones.

Para que el motor de consenso y el motor de ejecución de transacciones trabajen simultáneamente, MoeingEVM utiliza el siguiente esquema: cuando se confirma un bloque, las transacciones que contiene no se ejecutan inmediatamente sino que en su lugar, estas transacciones se guardan como parte del estado global de la red. Una vez guardadas, se calcula la raíz de Merkle del estado global y se propone y determina el siguiente bloque; mientras tanto, las transacciones guardadas son examinadas y ejecutadas.

Para lograr que las EVMs funcionen en paralelo, se permite que las transacciones provenientes de diferentes bloques sean mezcladas y reordenadas. En cada ronda un conjunto de transacciones independientes es elegido y ejecutado en paralelo para lograr un mayor grado de simultaneidad. Luego de varias rondas, una parte o la totalidad de las transacciones guardadas son ejecutadas, mientras que el resto de las transacciones que no han sido ejecutadas son guardadas nuevamente en el estado global de la red para su posterior ejecución. Este esquema se denomina "**enforced-bundle parallelism** (EBP)" debido a que las transacciones son siempre incluidas en paquetes y cada paquete es ejecutado en paralelo.

El EIP-2930 realiza explícitamente una transacción que contiene un listado de direcciones y claves de almacenamiento a las que planea acceder. Este listado contribuye a analizar la interdependencia de las transacciones. MoeingEVM lo implementará en el futuro para potenciar aún más el paralelismo de la ejecución de las transacciones.

MoeingADS

¿Por qué son tan caras las operaciones de almacenamiento? Pues bien, el motor MPT de almacenamiento de Ethereum es definitivamente el motivo principal.

¿Es posible prescindir de MPT? Sí, muchas cadenas de bloques (incluida la Bitcoin Cash) funcionan bien sin él. Sin embargo, como se trata de una genuina estructura de datos capaz de

comprobar cuáles son las condiciones del estado global, no deja de ser muy importante para la confiabilidad de la red y es la piedra angular para correr clientes ligeros y para operaciones cruzadas entre cadenas.

De todos modos hemos avanzado en el desarrollo de MoeingADS - otra estructura de datos genuina capaz de sustituir a MPT.

El motor de almacenamiento de Ethereum tiene una arquitectura de dos capas. La primera capa es LevelDB y la segunda es MPT. Por otro lado, blockchains como Bitcoin Cash adoptan una arquitectura de una sola capa para almacenamiento, utilizando LevelDB para almacenar directamente las UTXO. MPT trabaja por encima de LevelDB funcionando como una genuina estructura de datos a costa de un menor rendimiento de lectura y de escritura. Cada vez que la EVM lee o escribe el estado global de la red, MPT debe realizar varias operaciones de LevelDB, lo que genera múltiples operaciones en el SSD, lo que finalmente produce como resultado la lentitud de MPT.

MoeingADS utiliza una arquitectura de única capa, pudiendo acceder directamente al sistema de archivos sin necesidad de tener que utilizar ninguna otra base de datos. Es una base de datos KV que puede proporcionar pruebas de existencia y de no existencia. Con MoeingADS, la lectura de un par KV requiere de una única lectura en el disco, la sobrescritura de un par KV requiere una única lectura y una única escritura, las inserciones requieren de dos lecturas y de dos escrituras, y el borrado requiere de dos lecturas y una única escritura. Además, las escrituras se anexan, lo que resulta muy amigable para los SSD.

Los ensayos demuestran que MoeingADS es incluso más rápido que LevelDB. El costo es un mayor consumo de DRAM: cada par key-value demanda unos 16 bytes.

MoeingDB

Además de soportar MPT, LevelDB también es utilizado habitualmente para almacenar datos históricos como bloques, recibos de transacciones y registros. Sin embargo, no está optimizado para sobrecargas de trabajo de la blockchain que presenten las siguientes características:

1. Volumen de lectura que resulte mucho mayor al de escritura.
2. No hay necesidad de admitir transacciones atómicas de lectura-modificación-escritura.
3. Las restricciones simples de lectura/escritura son mejores que MVCC, ya que las modificaciones consisten en grandes escrituras agrupadas por bloques.
4. Pobre disponibilidad espacial para un eficiente almacenamiento en caché, como resultado de que la mayoría de las claves son IDs de hashes.
5. Susceptibilidad a ataques DDoS, a menos que la latencia de lectura de los datos en frío tenga un límite superior razonable.

MoeingDB es una base de datos específica para aplicaciones que almacena el historial de la blockchain y que ha sido desarrollada teninedo en cuenta las características anteriormente mencionadas para que se pueda adaptar mejor a la carga de trabajo de blockchain. En función de sus características, es posible construir una API Web3 de código abierto de alto QPS que beneficie tanto a Smart Bitcoin Cash como a Ethereum. Esperamos que esto pueda facilitar que el mercado de proveedores de API Web3 sea más descentralizado.

MoeingKV

MoeingKV es un almacenamiento de KV mucho más rápido que LevelDB en términos de lectura y de escritura, a costa de eliminar el soporte de iteración.

Para admitir iteradores, LevelDB incluye muchas compensaciones y optimizaciones. Pero en la mayoría de los casos los motores de almacenamiento de blockchain pueden funcionar sin iteradores, como sucede por ejemplo con el almacenamiento MPT de Ethereum y las UTXO de Bitcoin Cash.

En el diseño de la estructura de datos subyacente así como en la implementación del código, MoeingKV produce compensaciones y optimizaciones para acelerar las operaciones normales de lectura y escritura. Por lo tanto, es capaz de sustituir a LevelDB como una mejor "primera capa" para MPT.

Si bien MPT no puede ser sustituido por MoeingADS debido a cuestiones de compatibilidad, resulta posible utilizar MoeingKV para admitir a MPT. Es posible que MPT no sea tan rápido como MoeingADS al ser respaldado por MoeingKV, de todos modos sigue siendo mucho más rápido que uno respaldado por LevelDB.

Aunque MoeingKV no es utilizado en Smart Bitcoin Cash, sus principales ideas provienen de MoeingADS y de MoeingDB. Particularmente deseamos que otros proyectos puedan beneficiarse de MoeingKV.

MoeingAOT

MoeingAOT es un compilador ahead-of-time para EVM.

La EVM es adoptada más frecuentemente que otras VM como Web Assembly y se constituye como un estándar de facto para los contratos inteligentes.

Sin embargo, la EVM carece de ciertos procedimientos que resultan de importancia para su aceleración, como lo son los compiladores ahead-of-time (AOT) y los compiladores just-in-time (JIT). En la industria del software, prácticamente todas las máquinas virtuales importantes disponen de sus compiladores AOT y/o JIT, por ejemplo, JVM, ART VM, Javascript V8, DartVM, Web Assembly, LuaJIT y GraalVM.

Creemos que ya es hora de que la EVM tenga su compilador ya que la implementación de un compilador AOT sería razonable, dado que, a diferencia de Javascript y de Lua, este dispone de una semántica estática.

MoeingAOT es capaz de compilar el bytecode de EVM en código nativo y por consiguiente puede ser guardado como una librería vinculada dinámicamente. Cuando el intérprete de EVM comienza a ejecutar un contrato inteligente y encuentra su correspondiente archivo de librería compilado, la librería se cargará y se ejecutará, y la interpretación del bytecode ya no será necesaria.

Para los contratos de uso frecuente, como USDT y UniSwap, la compilación ahead-of-time es valiosa ya que el código nativo es mucho más rápido que la interpretación, lo que reduce drásticamente el tiempo de ejecución.

MoeingRollup

Los rollups son una metodología de segunda capa que permiten escalar el rendimiento de una cadena. Diversos proyectos, como optimistic rollup y arbitrum rollup, poseen diferentes implementaciones. En términos generales, los rollups consisten en agrupar un conjunto completo de estados dentro de un contrato, simplificándolos en un solo state root. Generalmente, la extensión de un rollup reside dentro de un contrato inteligente y un secuenciador es el encargado de mantener sus estados, empaquetando las transacciones de los usuarios dentro de los bloques y enviando los state roots respectivos al contrato inteligente. La transición entre los state roots de dos bloques adyacentes puede ser validada con alguna proof of data.

Un secuenciador honesto debe ser el encargado de mantener de forma confiable los estados y de empaquetar de manera neutral las transacciones de los usuarios, es decir, sin ningún tipo de censura. También debe proporcionar los estados y los bloques a cualquiera que los necesite; de lo contrario, los usuarios podrían expulsarlo para buscar un nuevo reemplazo utilizando algún mecanismo de staking predefinido en el contrato inteligente.

Además, la secuencia de state roots enviada debe estar vinculada a transiciones válidas entre roots adyacentes. Si se encuentra una inválida, el secuenciador podría ser desafiado a proporcionar la proof of data. Y si el secuenciador no lo hace, será interrumpido y expulsado.

Las reglas de ejecución son definidas detalladamente en los contratos inteligentes y pueden variar entre las diferentes extensiones. De todos modos, su propósito común es comprobar la validez de las transiciones de estado mediante proof of data. Desafortunadamente, la tarea de comprobación es bastante pesada y difícil de implementar en la EVM.

MoeingRollup implementa esta tarea de comprobación de forma nativa. Utilizando una primitiva para contratos inteligentes, todas las extensiones de rollup son capaces de realizar las tareas de

comprobación de manera fácil y eficiente. La proof of data es necesaria para integrar estas tres partes:

1. Un bloque de transacciones.
2. Las entradas de los pares KV de las transacciones y su prueba de existencia en contraposición al estado inicial del root.
3. Las salidas de los pares KV de las transacciones y la diversa información para calcular el estado final del root.

Al mismo tiempo, MoeingRollup también facilita el trabajo del secuenciador proporcionándole utilidades, incluyendo la generación del proof of data.

MoeingLink

Smart Bitcoin Cash comenzará a funcionar como una cadena de un solo shard. Pero en largo plazo, será posible incluir más shards y transformarla en una cadena multi-shards.

MoeingLink es un protocolo que permite que diferentes shards interactúen directamente sin necesidad de que ejecuten ninguna transacción en la red principal de Bitcoin Cash.

Actualmente, todas las soluciones de sharding importantes requieren una cadena intermedia. En ETH2.0 es la Beacon Chain y en Polkadot es la Relay Chain. A medida que se vayan creando cada vez más shards, las transacciones cruzadas entre shards ocasionarán una gran presión sobre la ya saturada capa 1.

Para evitarlo, MoeingLink permite a los shards comprobar su propio estado con los demás por medio de la utilización de los root states de MoeingADS asignados en la capa 1.

El algoritmo de consenso

Smart Bitcoin Cash adopta tendermint como su motor de consenso. El quórum de validadores es elegido tanto por su poder de hash como por su tenencia de BCHs y ejercen sus funciones por épocas.

Cada época contiene 2016 bloques (abarca aproximadamente dos semanas). Durante cada época, los tenedores de BCH demuestran su propiedad sobre las UTXOs bloqueadas por ese período de tiempo y utilizan los valores de esas UTXO para votar por un validador; mientras que los pools de minería utilizan las transacciones coinbase para ejercer su voto. Éste es un modelo de consenso híbrido: Proof of Hashpower y de Stake. El proceso de votación es realizado en la cadena principal de Bitcoin Cash y totalmente libre de permisos ya que cada nuevo validador solamente necesita el aval de los mineros y/o de los tenedores de BCHs.

El momento de finalización de una época lo determina el timestamp más alto de sus bloques, y su tiempo de duración queda establecido por la diferencia entre los tiempos finales de épocas

adyacentes. El quórum elegido durante una época permanecerá en estado de espera durante aproximadamente el 5% del tiempo de duración de la época. Luego tomará su turno para prestar sus servicios hasta que el próximo quórum abandone su estado de espera, esto es necesario ya que cualquier reorganización en la red de Bitcoin Cash puede alterar los bloques de una época.

Cada validador debe prendar algunos BCH como garantía, los cuales se reducirían drásticamente si se comporta mal durante el cumplimiento de su tarea.

Durante la primera fase, luego del lanzamiento de Smart Bitcoin Cash solamente el hashpower será usado para elegir a los validadores. El bloqueo de BCH en la cadena principal para el staking será implementado posteriormente y entrará en vigor en un futuro hard fork.

Token y gas

Smart Bitcoin Cash no emitirá nuevos tokens. Su token nativo es BCH, y sus tarifas de gas son pagadas en BCH.

Al final del mandato de cada quórum, la mitad de las tarifas de gas recaudadas serán recompensadas a los validadores y la otra mitad será quemada. De este modo, BCH se convertirá en una moneda deflacionaria. Cada validador debe prendar garantías suficientes para obtener su recompensa de las tarifas de gas y esa recompensa debe pasar por un período de bloqueo antes de poder ser gastada.

El mecanismo de recompra y quema es utilizado muy frecuentemente para los tokens de exchanges (BNB, HT, FTT, OKB, etc) y para los tokens de gobernanza de DeFi. Filecoin también dispone de un mecanismo similar para quemar parte de las tarifas de gas, que será implementado por Ethereum en el EIP-1559. Este mecanismo ha dado pruebas de ser efectivo, por lo cual hemos decidido también implementarlo.

BCH puede ser transferido bidireccionalmente entre la cadena principal de Bitcoin Cash y la Smart Bitcoin Cash, lo que significa que es posible bloquear un determinado número de monedas en la cadena principal para desbloquear la misma cantidad de monedas en la Smart Bitcoin Cash y viceversa. Para iniciar Smart Bitcoin Cash estamos invitando a los principales actores del ecosistema de Bitcoin Cash a conformar un gateway vinculado bidireccional y federado que une la red principal a la Smart Bitcoin Cash y que sirve para transferir BCH, al igual que funcionan RSK y Liquid. Estos actores no necesariamente deben ser validadores.

Hoy en día, las diferentes comunidades de criptomonedas se han adaptado al esquema de mantener el "mismo símbolo en diferentes cadenas". Por ejemplo, USDT hace referencia tanto a un token en la capa Omni de Bitcoin, como a un token SLP de Bitcoin Cash o como a un token en la red de Ethereum, o de Tron, etc. Por lo tanto, para evitar malentendidos, no vamos a usar otro símbolo para referirnos a los BCH en Smart Bitcoin Cash.

Somos conscientes de que el lenguaje de scripting de Bitcon Cash es capaz de implementar un gateway no custodial y libre de confiabilidad mediante el uso de un script de bloqueo para rastrear el proceso de votación que se lleva a cabo dentro de transacciones coinbase. Sin embargo, este esquema no ha sido puesto a prueba en la práctica. Vamos a redactar propuestas específicas para describir este esquema que será implementado en CashScript una vez aprobado. Posteriormente, Smart Bitcoin Cash pasará a este nuevo esquema en un futuro hard fork.

Interoperabilidad con otras soluciones de segunda capa en Bitcoin Cash

Hay muchas extensiones de segunda capa en Bitcoin Cash que permiten a cualquier emisor crear tokens fungibles y no fungibles. Entre ellas, la más exitosa e importante es Simple Ledger Protocol (SLP). Dentro del ecosistema de un token SLP su emisor juega un papel preponderante para poder servir de ayuda en la transferencia cruzada de sus tokens hacia la red de Smart Bitcoin Cash.

Por ejemplo, si Alice quiere transferir 10 monedas XYZ de SLP a Smart Bitcoin Cash, puede enviar las monedas al emisor de XYZ utilizando SLP, luego el emisor le enviará 10 monedas en Smart Bitcoin Cash y viceversa. Para incrementar la seguridad de este proceso Alice puede usar un atomic swap para asegurarse de que las transacciones de cada lado ocurran ambas o ninguna.

Hoja de ruta

MoeingADS, MoeingEVM y MoeingDB están casi finalizados. De cualquier modo, es necesario realizar algunas pruebas de rendimiento antes de que Smart Bitcoin Cash pueda lanzarse oficialmente.

MoeingAOT estará listo y comenzará a funcionar luego de un hard fork planificado para finales de 2021. MoeingKV también se desarrollará en 2021 con la esperanza de satisfacer las posibles demandas de la red principal de Bitcoin Cash.

MoeingRollup y MoeingLink serán desarrolladas en 2022. Si para entonces el tráfico en Smart Bitcoin Cash se encuentra congestionado, serán implementados en un hard fork para una mayor escalabilidad.

Conclusión

Smart Bitcoin Cash proporciona una cadena lateral para Bitcoin Cash, compatible con EVM y Web3, con staking de su hashpower y utilizando BCH como gas. Además, al incorporar componentes compatibles con el hardware, su escalabilidad queda liberada. Creemos que proporcionará los mismos beneficios que ETH2.0 en un período de tiempo mucho más corto, alcanzando un límite de gas de mil millones para el bloque.

En gran medida, Smart Bitcoin Cash puede ser vista como una demostración y como un experimento de las técnicas novedosas y agresivas que hemos estado desarrollando, cuyo objetivo es optimizar los motores de almacenamiento y de ejecución para alcanzar rendimientos extremos. Al igual que otros proyectos de código abierto, será probable encontrar errores y vulnerabilidades en su diseño e implementación. Por lo tanto, tenga en cuenta los riesgos posibles y asegúrese de que las pérdidas potenciales sean tolerables al momento de transferir sus activos (incluyendo BCH) a Smart Bitcoin Cash.