Ara — Estructura Descentralizada Global para Pagos, Identidad de Usuario, Alojamiento, Transmisión, y Propiedad de Archivos y Contenidos Digitales (Proyecto)

Eric Jiang, Charles Kelly, Joseph Werle, Tony Mugavero, Vanessa Kincaid

Actualización más reciente 14 de noviembre de 2018 (parcial**)

Resumen

Internet se ha convertido en una sombra de lo que fue, dominado por unas pocas compañías de gran envergadura que ejercen autoridad completa sobre toda la información al controlar la forma en que la misma fluye alrededor del mundo, cuánto cuesta y cómo y cuándo puede utilizarse. Ara es una plataforma descentralizada y un conjunto de protocolos que se diseñaron para corregir eso. Mediante la concesión de licencias y la venta de archivos y contenido digital utilizando un novedoso sistema de comprobante de titularidad, Ara maneja la entrega de datos globales y respalda la adquisición de aquellos activos mediante tokens nativos de Ara. Al hacer todo esto, la plataforma también utiliza un sistema generalizado y distribuido de identificación de usuario y billeteras, que permite que los usuarios conserven la titularidad de su información personal. En efecto, Ara es un nuevo modelo mental con respecto a la forma en que se aloja y entrega la información en Internet, y la forma en que los consumidores utilizan y pagan esa información; ofrece un nuevo paradigma no solamente para los negocios sino también para los consumidores en el aspecto en que los mismos pueden contribuir en el sistema a fin de obtener premios por alojar y participar en la red. La tecnología de cadena de bloques e intercambio de archivos entre pares (Peer-to-peer, P2P) para la concesión de licencias y la determinación de la titularidad, y la computación distribuida se combinan en un sistema único eficiente y descentralizado.

Una gran cantidad de integrantes se benefician de Ara. Los consumidores pueden utilizar su almacenamiento, ancho de banda y potencia de procesamiento ociosos para obtener tokens de Ara (similar a Airbnb para la computación) y pueden utilizar esos tokens para adquirir contenido. Los negocios obtienen ahorros entregando esta información a las personas a través de la tecnología P2P, la cual a su vez disminuye los costos para los consumidores y otros negocios. A fin de obtener premios, cualquiera puede participar en la red como un centro de datos. Los creadores de contenido digital, los desarrolladores de juegos y software, los estudios de películas y televisión, y los editores pueden utilizar los tokens de Ara para publicar contenido con licencia en la red, aumentando la rentabilidad de su trabajo debido a que generan un mayor nivel de ingresos y proporcionando el resto a sus simpatizantes por los costos de alojamiento. Es una situación beneficiosa para todos. Los consumidores reciben recompensas, los editores obtienen más ingresos y los negocios mejoran su utilidad neta. Todo esto se lleva a cabo de una forma descentralizada y neutral con respecto a la red, de forma que ninguna compañía intermediaria pueda restringir que un solo negocio le brinde información y contenido.

****** Esta es una actualización parcial de la documentación técnica de junio de 2018. En los próximos meses se publicará una actualización más minuciosa.

Nota: Ara se encuentra activamente bajo investigación y desarrollo. Esta documentación está sujeta a cambios. La versión más reciente estará disponible en https://ara.one. Dirija cualquier comentario o sugerencia a hello@ara.one.

Índice

1. Introducción

1.1 Antecedentes

El panorama de hipermedios de la actualidad es obsoleto. Los agregadores y las tiendas de aplicaciones están consolidando su influencia sobre los creadores de contenido; las redes tradicionales de entrega de contenido son ineficientes y costosas; la computación en la nube está centralizada en algunos pocos y selectos controladores de acceso; y los datos no los almacenan sus propietarios, sino aquellos que se benefician de los mismos. Los editores y creadores de contenido se ven obligados a aumentar los precios, transfiriendo los costos de este sistema lento y costoso a los consumidores, lo que resulta en una pérdida de valor para editores, consumidores y creadores por igual.

Considerando que el contenido en video representará más del 80 % de todo el tráfico de Internet para 2021 [2], el costo de contenido ha seguido aumentando a medida que el tamaño de los archivos y los costos de entrega de ese contenido aumentan. El contenido en 4K, la realidad virtual (RV) y los juegos AAA contribuyen en su totalidad con esta tendencia. Los consumidores no solo deben pagar más por contenido transaccional y suscripciones, sino que también deben hacer frente a un sistema complejo y abusivo de publicidad a fin de visualizar contenido gratuito. Estos factores empeoran el problema de la piratería, lo que genera miles de millones de dólares en pérdidas para los propietarios de contenido [14][11][3] y los mismos resultan en la aparición de herramientas para eliminar u omitir publicidad como los ad-blockers (bloqueadores de publicidad). Esto a su vez crea bloqueadores de bloqueadores de publicidad y servicios de suscripción más costosos a medida que los consumidores evitan los anuncios publicitarios y los propietarios de contenido tratan de recuperar los ingresos perdidos. Es un círculo vicioso.

La arquitectura de distribución de archivos entre pares (P2P) surgió como respuesta a estas deficiencias, cambiando gradualmente desde soluciones híbridas que incorporaban servidores centralizados como Napster hasta soluciones completamente descentralizadas como Gnutella y con el tiempo BitTorrent. Actualmente, la entrega de archivos P2P es tan rentable que compañías como Microsoft la utilizan (no en su propia estructura de Azure) para obtener ahorros en los costos de distribución de Windows 10.

No obstante, si bien es rentable, las redes de intercambio de archivos P2P históricamente han estado plagadas de mercados negros, oportunistas y piratería de todo tipo cuando se utilizan en entornos públicos. No había confianza de que la persona que subía el contenido tuviera derecho a hacerlo y tampoco había forma de verificar que el contenido propagado se entregaba de la forma prevista por el propietario de contenido. Tampoco había ningún tipo de premio por almacenar y compartir el contenido, por lo que los usuarios no tenían ningún incentivo de seguir como propagadores en el sistema de entrega. Los pares solo descargarían el contenido para sí mismos y no seguirían participando de la propagación de ese contenido con respecto a otros pares. Para combatir esto, las arquitecturas P2P empezaron a incorporar mecanismos de incentivo, como estrategias de canje, sistemas de reputación y divisas de propiedad exclusiva. Sin embargo, incluso estos mecanismos no están exentos de problemas y son objeto de ataques de tipo Sybil y de “blanqueamiento”.

1.2 Descripción general

En esta documentación técnica, presentamos a Ara: una plataforma de entrega de contenido y computación distribuida y descentralizada impulsada por la comunidad. Ara permite que cualquier dispositivo en el mundo se convierta en parte de una supercomputadora global, una base de datos y una red de entrega todo al mismo tiempo recurriendo a su capacidad de procesamiento, almacenamiento y ancho de banda sin utilizar. De forma conjunta, estos dispositivos forman la red Ara, un ecosistema donde cualquier persona puede participar y beneficiarse.

Básicamente, la red está compuesta de una comunidad superpuesta de consumidores, solicitantes de servicio, proveedores de servicios y desarrolladores de software, cada uno de los mismos con sus propios incentivos por la adopción. Con Ara, los solicitantes de servicio pueden tener a su alcance una vasta reserva de recursos informáticos junto con una biblioteca cada vez más amplia de servicios distribuidos. Los proveedores de servicio, quienes ya pagaron por sus dispositivos, p. ej. un teléfono inteligente, un computador portátil o una consola de juegos, pueden empezar a generar ingresos mediante el arriendo de los recursos sin utilizar. El único requisito para empezar a obtener premios es realizar un pequeño depósito que actúa como una retención sobre la cuenta. Esta retención se puede retirar en cualquier momento, pero es un requisito para obtener y canjear los premios. Los desarrolladores de software pueden aprovechar la escala incomparable del ecosistema de Ara con el fin de llevar a cabo tareas informáticas de gran volumen y crear servicios distribuidos novedosos para que los solicitantes y proveedores participen. Entretanto, los consumidores pueden seguir con sus actividades cotidianas mientras reciben premios por mirar los programas y escuchar la música que les encanta.

Por lo tanto, cualquier persona que tenga recursos informáticos adicionales puede actuar de inmediato como alguien que suministra servicios a fin de obtener premios por ayudar a distribuir contenido, mientras que cualquier persona que esté buscando recursos remotos puede solicitar los servicios descentralizados de Ara y disfrutar de una seguridad mejorada, la disponibilidad del archivo y la velocidad de la entrega por una fracción del costo en comparación con los proveedores de servicios de computación en la nube tradicionales. Gracias a que Ara elimina la carga que supone adquirir y gestionar una infraestructura, los creadores de contenido de todo tipo se beneficiarán, desde el artista independiente que ahora puede publicar por sí mismo y libremente su nuevo álbum sin necesidad de recurrir a un sello discográfico, hasta los grandes conglomerados de medios de comunicación que ya no necesitan recurrir a agregadores para llegar hasta su audiencia. Ara depende de los recursos que los miembros de la red proporcionan; cuanto mayor es el crecimiento de la red, la misma se vuelve más eficiente y robusta.

1.3 Servicios de la plataforma

La plataforma Ara consta de tres (3) servicios y sistemas centrales:

1. **AraID:** (Identificación de Ara) establece identidades globales seguras, descentralizadas y comprobables para todos los agentes y contenido presentes en la plataforma Ara, lo que devuelve el control de los datos a sus propietarios legítimos.

2. **Red de entrega de contenido descentralizada (Decentralized Content Delivery Network (DCDN)):** DCDN funciona como la red de sistemas de archivos distribuidos seguros y entre pares, y redes de almacenamiento (Ara Filesystem, AFS) subyacentes de Ara que respaldan la integridad del contenido, los incentivos, el desarrollo de versiones y las identidades descentralizadas.
3. **Conjunto de protocolos Ara:** Ara se conecta a través de un conjunto seguro de protocolos que posibilita una interoperabilidad autónoma (trustless) entre DCDN, AraID y la cadena de bloques de Ethereum.

2. Descripción general de la plataforma

Red del sistema de archivos sin conflictos (Conflict-Free File System Network (CFSNet)) es la red troncal del sistema de archivos distribuidos entre pares, AraID y red de entrega de contenido descentralizada (Decentralized Content Delivery Network (DCDN)) de Ara. Al aprovechar una estructura de árbol de Merkle subyacente y el formato de archivo protocolo sincronizable del libro mayor de eventos exactos (Syncable Ledger of Exact Events Protocol (SLEEP)), CFSNet aborda una gran cantidad de inquietudes con respecto al transporte de archivos tradicional (tanto para cliente-servidor como P2P) y mejora las tecnologías existentes como IPFS al brindar integridad de contenido que se protege de forma codificada además de un historial de desarrollos de versiones y revisiones. La red se compone de sistemas de archivos aislados que se denominan CFS. Además, cada instancia CFS implementa un subconjunto de estándar de jerarquía del sistema de archivos (Filesystem Hierarchy Standard (FHS)) [7], que es compatible con particiones y permite que cada directorio exista como un archivo CFS autónomo con sus propios niveles de acceso. De estas particiones, AFS utiliza las particiones `/home` y `/etc` para almacenar el contenido y los metadatos AFS, respectivamente. Cada partición CFS puede identificarse de manera pública a nivel de toda la red mediante una clave pública de 32 bits exclusiva `Ed25519` que se genera en el momento de la creación. Una clave pública de CFS otorga acceso solo de lectura al sistema de archivos, donde únicamente el titular de la clave privada puede actualizar y publicar el contenido que se incluye en el mismo.

2.1 AraID

AraID es responsable de crear y resolver representaciones descentralizadas seguras y comprobables para todos los usuarios y todo el contenido en la plataforma Ara. Cumpliendo plenamente con la especificación W3C identificador descentralizado (Decentralized Identifier (DID)) [13], AraID usa los objeto descriptor de DID (DID Descriptor Object), o DDO para abreviar, para representar a los usuarios y el contenido (consultar la *Figura 1*). Los DDO son documentos JSON-LD sencillos que definen métodos de autenticación y autorización además de otros atributos de identidad, entre ellos, los puntos finales de servicio y los canales de comunicación privados controlados por el propietario [13]. Debido a que los DDOs nunca almacenan información de identificación personal (Personally-Identifiable Information (PII)) [13], estos puntos finales de servicio y canales de comunicación identifican medios seguros para obtener esa información y, por lo tanto, permiten que las entidades tengan autodeterminación con respecto a sus datos privados e identidades en línea.

2.1.1 Identidad descentralizada

Para todos los usuarios y todo el contenido en la plataforma Ara, se genera una AraID en la forma de:

- `did:ara:ee93189c629cdaf94 9fd57bac5b005b916936d2a5c6806 40fd1aedc8315730a0`

AraID implementa un método (`method`) de resolutor universal, representado por el segundo componente del DID (`ara` anterior) como parte del sistema de base descentralizada de identidad [10]. El método, que se conoce también como controlador, define la forma en que los DID y DDO se resuelven dentro de la plataforma Ara. A diferencia de los URI de Internet, los DID no requieren de una autoridad central de registro o control y forman una correspondencia biyectiva con los DDO en lugar de una relación no inyectiva, no sobreyectiva que se encuentra en TCP/IP y DNS.

El punto crucial de la seguridad de AraID se mantiene de forma codificada mediante infraestructura de clave pública descentralizada (Decentralized Public Key Infrastructure (DPKI))[12], donde las claves públicas `Ed25519` se utilizan tanto para la parte del `id` como del

DID (`ee9318...` anterior) y la clave pública del CFS donde se almacena el DDO correspondiente. Estos documentos incluyen una propiedad `publicKey` (clave pública) que alberga varias claves para firmas digitales, cifrado y otras operaciones criptográficas. Cuando se crea una identidad, esta matriz se rellena con la clave de la identidad del propietario, además de la clave pública de la cuenta Ethereum correspondiente.

Para los AraID AFS, también se almacena la clave pública de la partición `/etc` que incluye los metadatos de contenido asociados. Debido a que la clave para esto se almacena en el DDO AFS, cualquier solicitante que tenga la DID AFS puede resolverla.

Cuando se genera una nueva identidad, se utiliza una frase mnemotécnica para propagar el par de claves. La nemotecnia se devuelve al propietario para su custodia y pro mantenimiento sencillo de la clave privada, lo que permite que las entidades validen con facilidad la titularidad de los DID y restablezcan las cuentas sin necesidad de la clave privada.

Archivado y resolución de identidad

Cuando se crea una identidad, al comienzo se escribe localmente de forma que cualquier resolución local pueda comprobar la memoria caché antes de recurrir a la red. No obstante, antes de que una identidad se resuelva de forma remota, la misma debe archivarse primero. Ara ejecuta nodos de archivo cuya tarea es almacenar estas identidades para una resolución futura.

De manera similar a los nodos del archivador, Ara también ejecuta los nodos del resolutor que son responsables de consultar con los archivadores para los DDO solicitados. El resolutor solicita una primera búsqueda a fin de resolver localmente las identidades que podrían estar almacenadas en el disco antes de recurrir a la red para los archivadores remotos que tienen registrados la AraID en cuestión.

```
{
    'ddo': {
        '@context': 'https://w3id.org/did/v1',
        'id': 'did:ara:ee9318...',
        'authentication;: [{
           'type': 'Ed25519SignatureAuthentication2018',
           'publicKey': 'did:ara:ee9318...#owner'
        }],
        'publicKey': [{
           'id': 'did:ara:ee9318...#eth',
           'type': 'Secp256k1VerificationKey2018',
           'owner': 'did:ara:ee9318...,
           'publicKeyBase58': 'H3C2AVvLMv6gmMNam...'
        }],
        'service': {
           'ens': 'https://etherscan.io/enslookup',
        }
        ...
    }
}
```

Figura 1: *Example DDO*

Cuenta Ethereum

Cada identidad se crea con una cuenta Ethereum y una billetera Ethereum asociada, que puede recuperarse mediante una nemotecnia que se genera de forma aleatoria cuando se crea la identidad. Debido a que la cuenta Ethereum y la identidad en sí se crean de forma determinista mediante esta mnemotécnica, un usuario puede recuperar su identidad completa, incluyendo su cuenta y billetera Ethereum con solo usar esta mnemotécnica.

AraID se diseñó para ser compatible con cualquier cuenta respaldada por criptografía de clave pública. Por lo tanto, es independiente con respecto a los tipos de cuentas de criptomonedas con las cuales es compatible y puede asociarse con facilidad a billeteras de criptomonedas de cualquier tipo.

2.2 Red de entrega de contenido descentralizada (DCDN)

La Red de entrega de contenido descentralizada (DCDN) es la solución de para la distribución escalable y descentralizada de hipermedios y activos digitales. De forma esencial, la DCDN consta de una red implementaciones de CFS, sistemas de archivos de Ara (Sistema de archivos de Ara, AFS) que albergan el contenido y los metadatos asociados.

2.2.1 Sistema de archivos de Ara (AFS)

El sistema de archivos de Ara (AFS) es una versión de CFS cuyo objetivo es satisfacer necesidades y metas específicas de Ara. AFS aprovecha dos particiones existentes que implementa CFS, las particiones `/home` y `/etc`. Estas particiones, que se implementan como un conjunto de FHS [7], son responsables de los datos binarios sin procesar y de los metadatos del contenido, respectivamente. Solo se puede acceder a la partición `/home` después de que un usuario haya adquirido o se le haya otorgado acceso al contenido del AFS, mientras que a la partición `/etc` que incluye los metadatos se puede acceder independientemente de la titularidad del contenido. El propietario del AFS puede definir el esquema de los metadatos de forma que el mismo pueda ser objeto de análisis por parte de un solicitante. El protocolo no aplica un estándar estricto en lo que respecta a la forma en que los metadatos se deben estructurar, pero, recomendamos Schema.org como referencia para los paradigmas existentes a fin de compatibilizar mejor la interoperabilidad entre los servicios descentralizados.

Cuando en un principio se crea el AFS, se crea una AraID Mediante una frase mnemotécnica `BIP39` [4] aleatoria de 12 palabras. El DID que se genera se utiliza como la clave pública del AFS la propiedad de autenticación (`authentication`) del DDO correspondiente se modifica para incluir la DID del propietario. De este modo, se puede determinar al propietario del AFS a partir de la resolución de su DID.

Se crea un AFS para cada pieza de contenido que se introduce en el sistema. Esto puede ser un archivo único como una película, o una recopilación de archivos como un juego. Para verificar la titularidad del contenido de forma codificada, se escriben dos conjuntos de memoria intermedia de 1 byte en la cadena de bloques de Ethereum. El primero son las entradas `metadata.tree` (árbol de metadatos), que representa el árbol de Merkle serializado de los datos que se incluyen en la capa de almacenamiento de datos. El segundo es el archivo `metadata.signatures` (firmas de metadatos), que incluye las firmas de los nodos de la raíz del árbol serializado.

2.2.2 Uso de tokens

En un principio, ser titular de los token de Ara con respecto a la DCDN otorga las siguientes capacidades:

1. La capacidad de comprar y descargar contenido dentro de la red.
2. La capacidad de participar en la entrega de archivos P2P y obtener premios con respecto a cualquier contenido mediante un depósito en Ara que actúa como una retención.

2.3 Conjunto de protocolos Ara

Las siguientes secciones secundarias definen los protocolos fundamentales de la plataforma, describen detalladamente cada parte del sistema y explican la interoperabilidad entre los mismos.

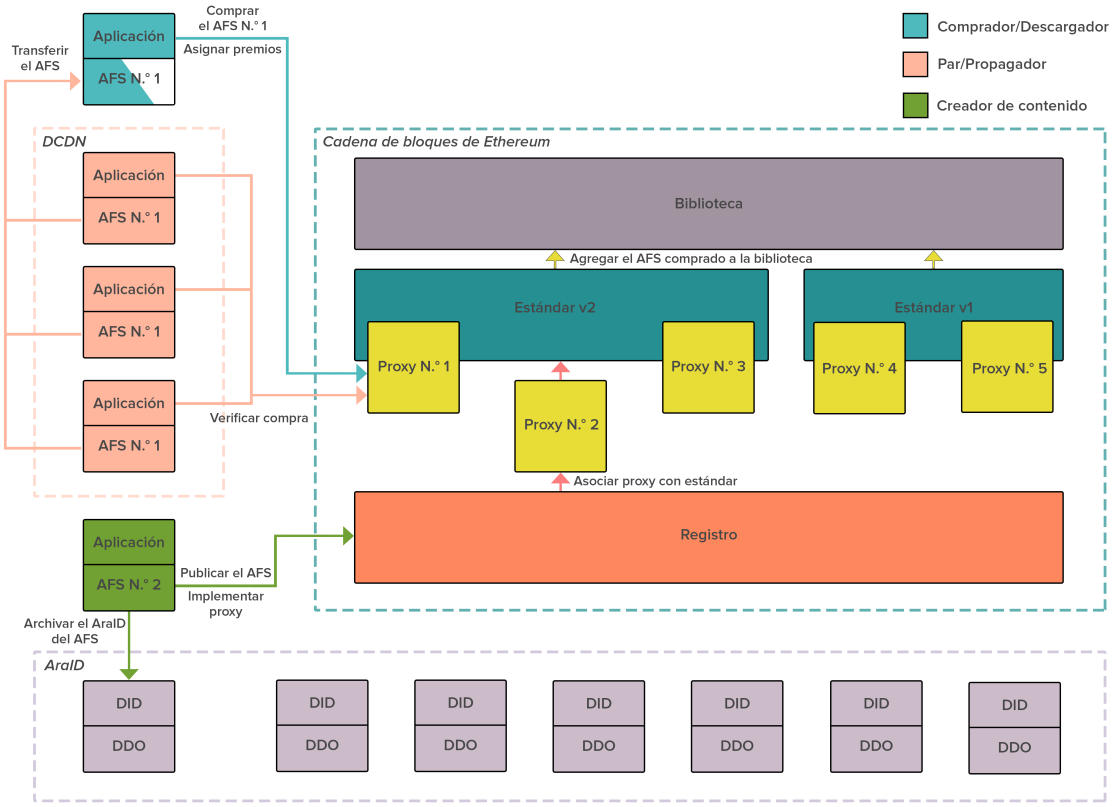

Figura 2: *Ilustración del protocolo Ara*

2.3.1 Premios e incentivos

Los sistemas de intercambio de archivos entre pares tradicionales, como BitTorrent, dependen del comportamiento altruista y carece de incentivos eficaces [6] para que los pares suban a la red la misma cantidad de contenido que descargan, lo que crea un desequilibrio donde los lícher (*leecher*) (usuarios que descargan un archivo distribuido) pueden dominar fácilmente un conjunto (*swarm*) (todos los pares que descargan o suben un archivo distribuido). Este desequilibrio, una forma del problema de oportunistas (Free Rider [5]), es perjudicial en una red saludable debido a que los leechers a menudo se benefician de la red a expensas de los demás sin ofrecer nada a cambio. Ara implementa un sistema de premios en calidad de mecanismo de incentivos a fin de mitigar esta ineficiencia de la red y aumentar la disponibilidad del archivo aplicando un pequeño costo a cada descarga y al distribuir ese costo en la forma de un premio a cada par que propaga la descarga. Dependiendo del modelo de pago, este costo podría incluirse en el costo total del contenido en la forma de una asignación de premios; en el caso del contenido "gratuito", este costo puede reemplazar a los anuncios publicitarios tradicionales o la recopilación de datos (la forma en que los usuarios pagan actualmente por el contenido "gratuito").

Debido a que cada descarga se convierte en una fuente de premios, con el tiempo, los participantes de la red podrán obtener mucho más de los premios que de lo que pagan por anticipado por la descarga.

2.3.2 Entrega de archivos

entrega de archivos (File Delivery) es el mecanismo principal que respalda la red de entrega de contenido de Ara y a los participantes para que obtengan premios. El protocolo de entrega de archivos de Ara empieza con un intercambio de señales (handshake) de cuatro pasos.

1. Alice, la solicitante de contenido, emite una solicitud de descarga para una pieza de contenido a través del protocolo de descubrimiento de una red (CFSNet implementa varias estrategias, como por ejemplo, mDNS y BitTorrent).
2. Bob, un validador de licencias y portador de contenido, recibe la emisión de la solicitud y responde con su disponibilidad del archivo.
3. Alice selecciona los pares a partir del grupo (el conjunto) de respuestas y envía un mensaje con su clave pública intermedia junto con su DID.
4. Bob verifica de forma codificada el mensaje de Alice y que ella adquirió una licencia para el AFS subyacente.

Una vez que se completa el intercambio de señales (handshake), empieza la transferencia del archivo.

2.3.3 Contratos inteligentes

En un principio, Ara se lanzará en la red principal de Ethereum, donde los contratos inteligentes servirán como el componente central del conjunto de protocolos de Ara. Estos contratos inteligentes realizan la mediación y facilitan la interoperabilidad entre DCDN, AraID y la capa de aplicaciones, lo que garantiza que todas las propiedades no transitorias y entidades en la plataforma estén registradas en la cadena de bloques de Ethereum, como por ejemplo:

- Contenido publicado
- Compras
- Premios
- Saldo de Ara

La arquitectura de contratos inteligentes de Ara se diseñó teniendo en cuenta la seguridad y la capacidad de modificación. A fin de respaldar las concepciones en constante desarrollo de la forma en que se deben vender y adquirir las AFS, la forma en que se deben manejar y distribuir los premios y la forma en que se deben procesar y canalizar los pagos dentro del sistema, en el caso de cada AFS que se publica, Ara implementa un contrato de proxy (`Proxy Contract`). Los proxy se implementan a través de un contrato de registro (`Registry Contract`), donde los mismos asocian con una versión específica de un estándar de AFS (`AFS Standard`), que define la lógica comercial de las AFS. Considerando que implementar el estándar de AFS completo para cada AFS sería algo costoso y difícil de actualizar, en esencia, la AFS estaría vinculada permanentemente con un estándar específico; la arquitectura de proxy permite que se despliegue un solo proxy durante la vida útil de una AFS y que se actualice al modificar a qué versión del estándar de AFS hace referencia. La arquitectura de proxy también garantiza que únicamente las direcciones proxy registradas (es decir, contenido AFS válido) pueden agregarse a las bibliotecas del usuario en el contrato de bibliotecas (`Library Contract`).

Estándar de AFS

El estándar de AFS permite que las AFS tengan una presencia definida, estructurada y autónoma en la cadena de bloques de Ethereum. El mismo incluye métodos para compras y premios, además de métodos para almacenar los archivos de árbol y firmas a partir del registro de metadatos SLEEP. El registro de metadatos SLEEP almacena en una AFS los metadatos del contenido, como por ejemplo, nombres de archivo, tamaño y permisos, mientras que el registro de contenido SLEEP almacena el contenido binario sin procesar de los archivos. Dentro del registro de metadatos, el archivo de árbol representa el

árbol de Merkle serializado que conforma los datos en el registro de contenido y el archivo de firmas almacena las raíces firmadas del árbol serializado. Considerando que con el CFS, estos archivos se almacenarían en un disco y serían leídos a partir de este, con AFS, los mismos se escriben en una cadena de bloques Ethereum y son leídos a partir de esta. Debido a que los AFS se comunican con este estándar a través de su propio proxy, pueden coexistir muchos estándares de AFS distintos, lo que permite que los creadores de contenido elijan el estándar que mejor se adapte a sus necesidades.

El uso de proxys separa la lógica del almacenamiento, donde el estándar de AFS sirve como la capa lógica para cualquier AFS que utilice esa versión del estándar y cada proxy sirve como la capa de almacenamiento para un solo AFS. Como mínimo, los estándares de AFS deben implementar una clase abstracta del estándar de AFS que haga cumplir la implementación de las funciones de precios, compras, premios y almacenamiento.

En el caso del estándar de AFS más básico (y predeterminado), solo el propietario de un AFS puede modificar los precios. Tras la compra, este precio se transfiere de la billetera Ara del comprador a la billetera Ara del propietario. El estándar de AFS base implementa la compatibilidad de los presupuestos de premios, que deben enviarse antes de la descarga. Una vez que se completa la descarga, el presupuesto se asigna entre los pares participantes, quienes, después, pueden canjear los premios que recibieron, en caso de que no hayan retirado el saldo de retención obligatorio para hacerlo.

A criterio de los creadores de contenido, los estándares de AFS también pueden ser compatibles con una serie de controles comerciales personalizables:

1. **Regalías**: Se pueden personalizar las compras para distribuir los beneficios entre muchas cuentas Ara diferentes mediante un desglose de porcentajes.

2. **Compras masivas**: Las compras se pueden estratificar en función de la cantidad comprada.

3. **Condiciones de reventa**: El contenido que se compró se puede revender varias veces por al menos el precio de reventa mínimo según lo especificado por el creador del contenido.

4. **Transferencias de titularidad**: El propietario de un AFS puede transferir rápidamente la titularidad a otra dirección de Ethereum.

5. **Pedidos anticipados**: El contenido se puede adquirir antes de que esté disponible para la descarga. Los compradores pueden enviar un presupuesto de premios de forma anticipada de modo que puedan empezar a descargar un AFS tan pronto como esté disponible.

6. **Desabastecimiento**: Los creadores de contenido pueden definir la cantidad máxima de ventas de un AFS, después de lo cual el AFS deja de figurar en la lista y no está disponible para la compra.

Estos controles de comercio proporcionan a los creadores de contenido todo tipo de facultad para definir sus propios modelos comerciales y de ingresos personalizados para que satisfagan sus necesidades de acuerdo con sus especificaciones exactas. Los controles pueden combinarse para formar modelos nuevos e interesantes que podrían ser difíciles de implementar a través de los medios tradicionales. Por ejemplo, una persona que disfruta remezclar música puede comprar las pistas del artista original mediante condiciones de reventa definidas y precios de reventa mínimos, puede remezclar las canciones y vender las versiones remezcladas mientras todavía paga al artista original. Considerando que anteriormente combinar estos tipos de controles hubieran requerido una exorbitante cantidad de tiempo, capital y participación legal, convirtiéndose en obstáculos importantes para el ingreso de creadores de contenido más pequeños, los mismos están disponibles actualmente para todos y de forma gratuita.

Registro

Como parte de la arquitectura de proxy, el contrato de registro tiene dos funciones principales:

1. Funciona como una fábrica de proxys
2. Hace un seguimiento de todas las versiones del estándar de AFS

Cuando un AFS se publica por primera vez, el registro implementa un proxy para ese AFS y establece una relación entre el mismo y un estándar de AFS específico. El proxy consulta el registro para obtener la dirección del estándar de AFS respectivo cuando es llamado y delega la llamada a esa dirección, donde la misma se procesa y devuelve al proxy.

Biblioteca

La red Ara aprovecha el contrato de biblioteca para crear una fuente canónica de información veraz para los AFS a los cuales tiene acceso el usuario, independientemente de que hayan sido comprados o accedidos de alguna otra manera. Cuando se compra contenido, la función compra en el estándar de AFS agrega automáticamente el DID AFS a la biblioteca del comprador en el contrato de biblioteca. Esto permite que cualquier servicio que necesite información acerca de la biblioteca de un usuario consulte el contrato para obtener esa información. El DID AFS que se almacena en la cadena de bloques permite que cualquier servicio resuelva el contenido subyacente. La biblioteca impone que únicamente los proxys registrados puedan agregar sus AFS correspondientes a la biblioteca de un usuario, lo que garantiza que nadie pueda alterar la biblioteca de otro usuario sin su consentimiento.

3. Desarrollo futuro

3.1 Módulos

La plataforma Ara es básicamente independiente con respecto a los tipos de servicios distribuidos que pueden ejecutarse en la misma. Los módulos son servicios distribuidos y/o descentralizados que implementan las API de los módulos y que se pueden utilizar indistintamente dentro de la plataforma. Del mismo modo en que el estándar de token ERC-20 permite que cualquier token en Ethereum pueda ser reutilizado por otras aplicaciones, las API de los módulos permiten que todos los servicios distribuidos implementen la interfaz que se utilizará a nivel de toda la plataforma. Esto facilita la formación de una comunidad de desarrolladores dedicada a construir un ecosistema de servicios distribuidos donde cada uno tiene la capacidad de utilizar los sistemas de premios, compras y pagos de Ara, lo que básicamente forma la economía de servicios distribuidos sujetos a premios. Se espera que los módulos que buscan utilizar el sistema de premios implementen una API de contrato inteligente adicional donde definan un mecanismo y una metodología propios para el otorgamiento de premios. Cada contrato inteligente de módulo almacena su propia asignación de premios respectiva que se acumula a través del uso del servicio distribuido y la distribuye en consecuencia.

3.2 Sistema de nombres de Ara (ANS)

El sistema de nombres de Ara (Ara Name System, ANS), al igual que el sistema de nombres de dominio (Domain Name System (DNS)) [8], es una forma descentralizada de registrar, consultar, invocar o revocar certificados en la red de Ara. Si bien el DNS forma parte de la capa de aplicaciones de Internet, resolviendo las URI legibles para humanos en direcciones IP subyacentes de forma que un agente de usuario (p. ej. un navegador web) pueda obtener y presentar el contenido solicitado, ANS resuelve los nombres legibles para humanos en DID para la resolución definitiva en los DDO. ANS utiliza el archivador de identidades y el resolutor en segundo plano para la segunda fase de resolución a fin de proporcionar los DDO a partir de las DID. ANS en esencia es un archivador y un resolutor en el caso de las URI legibles para humanos. Un ejemplo de aplicación de ANS podría ser el hecho de proporcionar los DDO a partir de los nombres del host, en el contexto de un navegador web desarrollado en Ara.

Para distinguir entre los diversos tipos de registros, cada registro almacena un registro de recurso `TYPE`, similar a la forma en que el DNS clasifica sus propios registros [15]. El campo `TYPE` se representa mediante un valor numérico, lo que permite almacenar otros tipos de registros en ANS en el futuro. La siguiente tabla describe el registro `TYPE`:

TYPE	Value	Description
USR	00	User
PCT	01	Published Content

Cada concentrador de supernodo que consta de ANS ejecuta una instancia HyperDB [1]. Una base de datos distribuida y altamente escalable como HyperDB brinda varias características que la convierten en ideal para un sistema como ANS. La primera es el uso de intentos por parte de HyperDB: busca árboles donde cada nodo es un prefijo de sus nodos secundarios. Al almacenar nombres con intentos, podemos garantizar que incluso con miles de entradas en la base de datos, las búsquedas son rápidas y baratas. Las búsquedas en los intentos son `O(n)` donde `n` es la longitud de la clave que se busca. HyperDB también utiliza los relojes de vectores, que hacen un seguimiento de la causalidad de los eventos dentro de un sistema distribuido a fin de evitar casos donde los nodos se desincronicen [9].

4. Reconocimientos

Esta documentación fue posible gracias a Littlstar y Token Foundry. Agradecimientos especiales para Logan Dwight, Andrew Grathwohl y Brandon Plaster por su contribución.

Acrónimos

AFS Ara File System. 4–7, 11

ANS Ara Name System. 12

CFS Conflict-Free File System. 5–7

CFSNet Conflict-Free File System Network. 5

DCDN Decentralized Content Delivery Network. 4–6

DDO DID Descriptor Object. 5–7, 12

DID Decentralized Identifier. 5–7, 11, 12

DNS Domain Name System. 5, 12

DPKI Decentralized Public Key Infrastructure. 5

FHS Filesystem Hierarchy Standard. 5, 7

PII Personally-Identifiable Information. 5

SLEEP Syncable Ledger of Exact Events Protocol. 5, 9

Referencias

[1] Mathias Buus. Hyperdb. https://github.com/mafintosh/hyperdb, Aug 2017.

[2] Cisco. Cisco visual networking index predicts global annual ip traffic to exceed three zettabytes by 2021. https://newsroom.cisco.com/press-release-content?type=webcontent&articleId=1853168, Jun 2017.

[3] Stewart Clarke. Piracy set to cost streaming players more than $50 billion, study says. http://variety.com/2017/tv/news/piracy-cost-streaming-players-over-50-billion-1202602184/, Oct 2017.

[4] Palatinus et al. Mnemonic code for generating deterministic keys. https://github.com/bitcoin/bips/blob/master/bip-0039.mediawiki, Sept 2013.

[5] Russell Hardin. The free rider problem. https://plato.stanford.edu/entries/free-rider, May 2003.

[6] Ahamad Jun. Incentives in bittorrent induce free riding. https://disco.ethz.ch/courses/ws0506/seminar/papers/freeriding_incentives.pdf, Aug 2005.

[7] The Linux Foundation LSB Workgroup. Filesystem hierarchy standard. https://refspecs.linuxfoundation.org/FHS_3.0/fhs-3.0.pdf, Mar 2015.

[8] Paul Mockapetris. Domain names - implementation and specification. https://tools.ietf.org/html/rfc1035, Nov 1987.

[9] Multiple. Vector clock. https://en.wikipedia.org/wiki/Vector_clock.

[10] Markus Sabadello. A universal resolver for self-sovereign identifiers. https://medium.com/decentralized-identity/a-universal-resolver-for-self-sovereign-identifiers-48e6b4a5cc3c, Nov 2017. Accessed on 2018-04-24.

[11] Stephen E. Siwek. The true cost of sound recording piracy in the us economy. https://www.riaa.com/wp-content/uploads/2015/09/20120515_SoundRecordingPiracy.pdf, Aug 2007.

[12] Rebooting the Web-of Trust. Decentralized public key infrastructure. http://www.weboftrust.info/downloads/dpki.pdf, Dec 2015. Accessed on 2018-04-19.

[13] W3C. Decentralized identifiers. https://w3c-ccg.github.io/did-spec, Apr 2018. Accessed on 2018-05-01.

[14] Music Business Worldwide. Why does the riaa hate torrent sites so much? https://www.musicbusinessworldwide.com/why-does-the-riaa-hate-torrent-sites-so-much/, Dec 2014.

[15] Inc ZyTrax. Dns resource records (rrs). http://www.zytrax.com/books/dns/ch8/, Oct 2015.

Apéndices

I. Aspectos económicos del token de la plataforma Ara consolidada (borrador)

Descripción general

Los servicios de nube tradicionales han ganado importancia debido a la flexibilidad, agilidad y ahorro en los costos que ofrecen con respecto a la compra y la gestión de la infraestructura interna. Muchos servicios en la nube son notoriamente complejos y enmascaran modelos de precios que implican compromisos y contratos a largo plazo (que a menudo el cliente negocia de manera individual) lo que obstaculiza directamente la flexibilidad y agilidad que se suponía debían proporcionar en primer lugar [1]. En años recientes, las CDN P2P empezaron a ganar impulso, donde el modelo emergente de software como servicio (Software as a Service, SaaS) pregona soluciones híbridas para la entrega de video. Si bien estos modelos de SaaS establecieron una solución ampliamente escalable con un modelo de precios más sencillo ya que aprovecha las máquinas de los espectadores de video, los mismos mantienen varias fuentes de descentralización, en concreto, sus modelos comerciales y sus atribuciones de usuarios como "ciudadano de segunda clase", donde la utilización de sus máquinas sucede sin el consentimiento de estos ni una compensación financiera. Ara lleva esto un paso más allá al premiar a los participantes de la red por la utilización de sus máquinas.

A fin de sustituir los costos existentes de la infraestructura de nube, la plataforma Ara implementa una utilidad de protocolo nativo: el token de Ara. Este token se puede utilizar en la plataforma Ara con el objetivo de crear incentivos criptoeconómicos para un comportamiento de red saludable y honesto, a fin de permitir una participación más directa entre los consumidores y los creadores de contenido, además de fomentar la adopción de la plataforma. El token de Ara puede ser visto como una encapsulación del valor que los miembros de la red proporcionan, donde cada token que se otorga representa un aumento marginal en la utilidad de la red.

El token de Ara

Los servicios distribuidos que se desarrollan utilizando los SDK de Ara, que se conocen como módulos, se pueden comprar, vender, solicitar y satisfacer totalmente en la red de Ara. Las tareas del módulo se tercerizan hacia los pares de la red que, tras una finalización exitosa de las mismas, pueden recibir una compensación mediante los tokens de Ara. A fin de propiciar un mercado abierto y competitivo, Ara permite que los solicitantes del servicio definan las asignaciones del premio, o recompensas, para los servicios que solicitan y que los proveedores del servicio definan una recompensa mínima para la aceptación. Al igual que Amazon Mechanical Turk, un mercado para tareas de participación colectiva (crowdsourcing) que requieren de inteligencia humana, Ara crea un mercado para tareas de desarrollo de redes o computación distribuida tercerizada. Los módulos pueden ser tareas únicas a petición, como una

transferencia de código distribuida, o pueden ser servicios periódicos continuos, como un servidor P2P para juego multijugador.

Función

El token de Ara se puede utilizar en toda la red de una variedad de formas.

- En el caso de los consumidores, los tokens de Ara se pueden utilizar para realizar cualquier tipo de compra, que va desde contenido digital para esparcimiento hasta nuevos módulos en los cuales participar.
- Los solicitantes del servicio pueden utilizar los tokens de Ara para iniciar solicitudes de trabajo y establecer recompensas por la finalización exitosa de esos trabajos.
- Los proveedores de servicio pueden depositar los tokens de Ara como un compromiso de cumplimiento de tareas a cambio de recompensas (los depósitos se pueden devolver a solicitud, pero el proveedor ya no podrá obtener las recompensas)
- Los desarrolladores podrán utilizar los tokens de Ara para implementar nuevos módulos en la red

Debido a que cada una de estas funciones se superponen en gran medida, un proveedor de servicio puede utilizar los mismos tokens de Ara que obtuvo como recompensa por el cumplimiento de una tarea para adquirir una nueva pieza de contenido, al igual que un desarrollador puede utilizar los tokens de Ara que obtuvo mediante las compras de módulos para iniciar una nueva solicitud de trabajo.

Dinámica del mercado

Debido a que los miembros de la red tendrán representación plena para decidir en qué tareas o servicios participarán, la red forma un mercado libre donde emerge un equilibrio económico. Es probable que cada módulo tenga sus propios aspectos económicos que se rigen por el comportamiento de los solicitantes y los proveedores para los trabajos de ese módulo específico. Por ejemplo, las transferencias de códigos de video distribuido puede ser un trabajo con característica de urgente en el caso de los productores de video, lo que resulta en una inelasticidad de los precios de la demanda de transferencias de códigos de video distribuido (es decir, los productores de video son relativamente indiferentes con respecto al nivel en que deberían recompensar a los proveedores de servicio). Por lo tanto, se forma un mercado del vendedor donde los proveedores de servicio de transferencia distribuida tienen la ventaja de determinar la asignación de recompensas, lo que resulta en un aumento de las mismas. Del mismo modo, es posible que un módulo de servidor de juegos P2P sea altamente solicitado pero tenga relativamente pocos proveedores de servicio. Nuevamente, se forma un mercado del vendedor y aumenta la recompensa. Por otro lado, es posible que pocas personas soliciten un módulo de aprendizaje automático distribuido pero tenga una gran cantidad de proveedores de servicios elegibles. Debido a la relativamente escasa demanda, muchos proveedores perderán oportunidades si los mismos establecen sus requisitos de recompensa mínimos en un nivel demasiado alto. Se forma un mercado del vendedor y disminuye la recompensa.

Es importante tener en cuenta que no es un requisito que los módulos establezcan las recompensas y no existe un modelo estandarizado de la forma en que se deben establecer estas. Las metas son alinear los incentivos de los solicitantes del servicio y los proveedores de servicio de modo de respaldar todos los tipos de modelos de incentivo, además de adaptarse a los diferentes costos fijos en el caso de la infraestructura y las capacidades de red en todo el mundo.

Estructura de incentivos

Para comprender mejor la forma en que este modelo respalda una alineación de los incentivos entre los solicitantes del servicio y los proveedores de servicio, obtuvimos una descripción general de los intereses económicos de ambas partes. Los solicitantes del servicio desean que sus solicitudes se satisfagan al menor costo, mientras que los proveedores de servicio desean optimizarse para la mayor cantidad de servicios pagos de mayor nivel. En otras palabras, es en el mejor interés de los solicitantes y los proveedores el hecho de maximizar la utilidad de la red siempre y cuando ambos acuerden un precio. Por lo tanto, es más probable que se suministren los servicios que equilibren mejor la recompensa y el trabajo, lo que genera una presión de mercado que propicia las recompensas competitivas y la innovación en el diseño del servicio distribuido con el fin de mejorar la eficacia.

La variedad de servicios distribuidos que pueden ejecutarse en la plataforma necesita de flexibilidad al momento de determinar la unidad de trabajo recompensada ([Unit-of-Work Rewarded, *UWR*], que es la unidad base de trabajo demostrable por la cual se dividen y otorgan las recompensas) y modelo de recompensas (las condiciones que rigen la forma en que se paga la recompensa). Un servidor de juegos multijugador P2P podría utilizar una serie de solicitudes atendidas como su *UWR* y un desarrollador de juegos que invoca al módulo de servidor podría decidir que la acción más lógica es un modelo de recompensa recurrente basado en suscripciones. Por otro lado, un servicio de transferencia de código distribuido podría utilizar como su *UWR* una serie de bytes con transferencia de código por minuto y un productor de video podría pagar una recompensa por transferencia de código.

Los proveedores de servicio pueden poner a disposición sus tokens de Ara de modo de participar y obtener premios. Como en el caso de las recompensas, los solicitantes del servicio pueden determinar una participación mínima, si la hubiera, que los proveedores deben depositar a fin de contratar el servicio. El valor de la participación mínima indicaría el nivel de compromiso que requiere el servicio y se devuelve, junto con la recompensa, una vez que el servicio se completa de manera exitosa. Como parte de la determinación de una *UWR*, los servicios también deben definir un comprobante de verificación de su cumplimiento.

Opcionalmente, los proveedores de servicio pueden establecer una tarifa de suscripción para dedicar sus recursos a un servicio. Estos proveedores dedicados se conocen como supernodos y sus participaciones se depositan en garantía en un contrato inteligente hasta que finalice la suscripción. Debido a que los supernodos suelen ser más confiables que sus contrapartes no dedicadas, los mismos pueden controlar la dinámica del mercado en función de cómo establecen sus tarifas de suscripción. Los supernodos en Bogotá podrían cobrar más que los supernodos en Los Ángeles debido a los costos de hardware e Internet más altos.

Efectos en la red

La introducción de nuevo contenido en la red implica la invocación de supernodos DCDN, nodos de Ara dedicados a la redundancia y disponibilidad del contenido. Suponiendo que la recompensa para un archivo único es constante, el efecto del aumento marginal en la disponibilidad del archivo a partir de un par único que comparte ese archivo sobre las ganancias de premios potenciales a partir de ese archivo para cualquier par en la DCDN (y todos los sistemas de intercambio de archivos P2P basados en divisas con incentivos fijos [2]) se pueden modelar utilizando una función de conjunto submodular, que se puede considerar de manera intuitiva como una función que describe los retornos que disminuyen. Debido a esta propiedad, los supernodos de la DCDN pueden emplear un modelo de recompensas por suscripción para contrarrestar el costo de oportunidad cada vez mayor de alojamiento de contenido a medida que su disponibilidad aumenta.

Los editores de contenido pueden invocar y suscribirse a la cantidad de supernodos que prefieran en las ubicaciones geográficas de su elección sobre una base de contenido. Después, los editores pueden tener total libertad al momento de decidir la magnitud en la cual desean que su contenido esté fácilmente disponible a nivel global. Esto posibilita que se pueda brindar soporte al distribuidor de medios de gran envergadura que desea invocar todos los supernodos disponibles a nivel mundial con el objeto de respaldar a una audiencia global, mientras que también posibilita que se pueda brindar soporte al creador de contenido independiente que identifica a su audiencia principal como predominantemente europea y decide priorizar los supernodos europeos. Los editores de contenido también pueden determinar la asignación de premios por cada descarga de contenidos. Por lo tanto, existe una asignación de premios y distribución de supernodos óptimas de modo de alcanzar el nivel de participación deseado (es decir, disponibilidad del archivo).

Referencias

[1] Enterprise Strategy Group (2015, June), *Price Comparison: Google Cloud Platform vs. Amazon Web Services*, https://cloud.google.com/files/esg-whitepaper.pdf

[2] M. Salek, S. Shayandeh, and D. Kempe, *You Share, I Share: Network Effects and Economic Incentives in P2P File-Sharing Systems* https://arxiv.org/pdf/1107.5559.pdf

II. Análisis de costos de DCDN por Lester Kim

Introducción

Para calcular los costos de transmisión de un socio potencial, debemos conocer la cantidad de datos B (en bytes) que deben entregar a los consumidores por unidad de tiempo T (en segundos). Consideremos que tenemos N grupos de nodos cargadores donde $N \in \mathbb{N}$. $\forall n \in \{1, ..., N\}$, el ancho de banda promedio del grupo n es de b_n (en bytes/segundo por nodo). Consideremos que q_n es la cantidad del grupo de nodos n. Consideremos $\mathbf{b} = [b_1...b_N]^\intercal$ y $\mathbf{q} = [q_1...q_N]^\intercal$. Por lo tanto, la cantidad de bytes entregados por segundo limitados por $\frac{B}{T}$ es de

$$g(\mathbf{q}) = \mathbf{b} \cdot \mathbf{q} = \frac{B}{T}. \tag{1}$$

Queremos encontrar el $\mathbf{q}^*$ óptimo para minimizar el costo de distribución $C(\mathbf{q})$. Si $\mathbf{p} = [p_1...p_N]^\intercal$ donde p_n es el precio por nodo para el grupo n, tenemos

$$C(\mathbf{q}) = \mathbf{p} \cdot \mathbf{q}. \tag{2}$$

Maximización de las ganancias del cargador

Con el fin de determinar $\mathbf{p}$, consideremos el comportamiento de una empresa de maximización de ganancias. Consideremos que f es la función de producción con una entrada de energía de E (en kWh) y una salida de q (en nodos). Modelamos esta función de producción como

$$f(E) = AE^\alpha \tag{3}$$

donde A es el factor de producción (nodos/kWh$^\alpha$) y $\alpha \in [0, 1]$ es la elasticidad de la producción (aumento de porcentaje en la salida con respecto al aumento de porcentaje en la entrada) [1].

Consideremos que P (en kWh/s) es el aumento de potencia cuando un nodo empieza a cargar los datos. Esto incluye el envío de datos mediante su controlador de interfaz de red (Network Interface Controller, NIC) pero también puede incluir el encendido de la máquina (ya sea porque estaba apagada o en modo de espera). Si cada nodo tiene una potencia P, entonces, para algunos E, un nodo único puede funcionar durante $\frac{E}{P}$ segundos. Sin embargo, debido que existe una limitación de tiempo T para completar el trabajo, deben haber $\frac{E}{PT}$ nodos. Por lo tanto,

$$A = \frac{D}{(PT)^\alpha} \tag{4}$$

donde D es la productividad total del factor [2] (en nodos).

Consideremos que p es el precio de un nodo y que p_E es el precio de la energía (por kWh). La función de ganancias π de la empresa es

$$\pi(q, E) = pq - p_E E. \tag{5}$$

El costo del ancho de banda se ignora debido a que es un costo fijo a corto plazo cuando se considera en segundos en comparación con meses. Bandwidth cost is ignored because it is a fixed cost in the short-term when looking at seconds as opposed to months[1].

Deseamos maximizar las ganancias de la firma brindando un requisito de salida de al menos q;

$$\max_{q,E} \ \pi(q, E) \quad \text{s.t. } f(E) \geq q. \tag{6}$$

Para resolver esto, asumimos que nuestro lagrangiano será

$$\mathcal{L}(q, E, \lambda) = pq - p_E E - \lambda(AE^{\alpha} - q). \tag{7}$$

Al tomar las derivadas parciales y establecerlas en cero da

$$\frac{\partial \mathcal{L}}{\partial q} = p + \lambda = 0 \tag{8}$$

$$\frac{\partial \mathcal{L}}{\partial E} = -p_E - \lambda A \alpha E^{\alpha - 1} = 0 \tag{9}$$

$$\frac{\partial \mathcal{L}}{\partial \lambda} = q - AE^{\alpha} = 0. \tag{10}$$

Resolver estas condiciones de primer orden da

$$q^* = \left(\frac{\alpha A^{\frac{1}{\alpha}} p}{p_E} \right)^{\frac{\alpha}{1-\alpha}}. \tag{11}$$

Volver a escribir esto (omitiendo el asterisco de q) da

$$p = \frac{p_E}{\alpha} \left(\frac{q^{1-\alpha}}{A} \right)^{\frac{1}{\alpha}}. \tag{12}$$

Esta fórmula permite que el creador conozca qué deber ser p de modo de obtener la cantidad de nodos deseada.

Desde (12), encontramos que los ingresos óptimos en términos de q son

[1] Aun cuando se incluya el ancho de banda, su costo por segundo tiene el mismo orden de magnitud que el de la energía. En la ciudad de Nueva York, 50 MBit/s cuestan 3 USD x 10^{-5}/segundo [3]

$$pq = \frac{p_E}{\alpha}\left(\frac{q}{A}\right)^{\frac{1}{\alpha}}. \tag{13}$$

Minimización de los costos del distribuidor

Debido a que los ingresos de la empresa se están gastando en el consumidor (el creador), podemos escribir (2) como

$$C(\mathbf{q}) = \frac{p_E}{\alpha}\sum_{n=1}^{N}\left(\frac{q_n}{A_n}\right)^{\frac{1}{\alpha}}. \tag{14}$$

El problema de minimización de los costos del creador es

$$\min_{\mathbf{q}} \; C(\mathbf{q}) \quad \text{s.t. } g(\mathbf{q}) \geq \frac{B}{T}. \tag{15}$$

El lagrangiano es

$$\mathcal{L}(\mathbf{q}, \lambda) = C(\mathbf{q}) - \lambda(g(\mathbf{q}) - \frac{B}{T}). \tag{16}$$

Las condiciones de primer orden son

$$\frac{\partial \mathcal{L}}{\partial \mathbf{q}} = \frac{\partial C}{\partial \mathbf{q}} - \lambda\frac{\partial g}{\partial \mathbf{q}} = 0 \tag{17}$$

$$\frac{\partial \mathcal{L}}{\partial \lambda} = \frac{B}{T} - g(\mathbf{q}) = 0. \tag{18}$$

Desde (14), (4) y (1),

$$\frac{\partial C}{\partial \mathbf{q}} = \frac{p_E T\mathbf{P}}{\alpha^2} \circ (\mathbf{q}^{\circ(1-\alpha)} \oslash \mathbf{D})^{\circ\frac{1}{\alpha}} \tag{19}$$

$$\frac{\partial g}{\partial \mathbf{q}} = \mathbf{b} \tag{20}$$

donde $(\mathbf{D}, \mathbf{P}) = ([D_1 ... D_N]^\intercal, [P_1 ... P_N]^\intercal)$. "$\circ$", "$^\circ$", "$\oslash$" son el producto, la potencia y la división de Hadamard (a nivel del ingreso), respectivamente [4]. Entonces $\forall m, n \in \{1, ..., N\}$,

$$\frac{P_m^\alpha q_m^{1-\alpha}}{b_m^\alpha D_m} = \frac{P_n^\alpha q_n^{1-\alpha}}{b_n^\alpha D_n}. \tag{21}$$

Por lo tanto,

$$b_m q_m = \left(\frac{b_m D_m P_n^\alpha}{P_m^\alpha b_n D_n} \right)^{\frac{1}{1-\alpha}} b_n q_n. \tag{22}$$

Combinar (22) con (18) da

$$\boxed{\mathbf{q}^* = \frac{B\mathbf{b}^{\circ -1}}{T\kappa} \circ (\mathbf{b} \circ \mathbf{D} \oslash \mathbf{P}^{\circ \alpha})^{\circ \frac{1}{1-\alpha}}} \tag{23}$$

$$\boxed{C^* = \frac{p_E T}{\alpha} \left(\frac{B}{T\kappa^{1-\alpha}} \right)^{\frac{1}{\alpha}}} \tag{24}$$

donde

$$\kappa \equiv \sum_{m=1}^{N} \left(\frac{b_m D_m}{P_m^\alpha} \right)^{\frac{1}{1-\alpha}}. \tag{25}$$

Caso: $\alpha = 1$

Cuando $\alpha = 1$, (23) y (24) se convierten en

$$q_n^* = \begin{cases} \frac{B}{|\Upsilon| T b_n} & n \in \Upsilon \\ 0 & n \notin \Upsilon \end{cases} \tag{26}$$

$$C^* = \frac{p_E B P_n}{b_n D_n} \qquad \text{any } n \in \Upsilon \tag{27}$$

donde

$$\Upsilon \equiv \left\{ n \in \{1, ..., N\} \;\middle|\; n = \arg\max_{1 \le m \le N} \frac{b_m D_m}{P_m} \right\}. \tag{28}$$

$\forall n \in \Upsilon$„ cada nodo en el grupo n debería entregar $\frac{B}{|\Upsilon| q_n^*} (= b_n T)$ de datos y recibir al menos $\frac{p_E P_n T}{D_n}$ en compensación. Sin embargo, existen varias soluciones para $\mathbf{q}^*$. Por ejemplo, para cualquier $n \in \Upsilon$, el grupo n puede tomar todo el trabajo empleando $\frac{B}{T b_n}$ nodos.

Ejemplo

Consideremos un ejemplo en la ciudad de Nueva York donde

$$\alpha = 1 \tag{29}$$
$$B = 1 \text{ GB} \tag{30}$$
$$N = 2 \tag{31}$$
$$p_E = \$0{,}2321/\text{kWh [5]} \tag{32}$$
$$T = 1 \text{ s} \tag{33}$$
$$\mathbf{b} = \begin{bmatrix} 100 \text{ MB/s} \\ 1 \text{ MB/s} \end{bmatrix} \text{ [6]} \tag{34}$$
$$\mathbf{D} = \begin{bmatrix} 1 \text{ node} \\ 1 \text{ node} \end{bmatrix} \tag{35}$$
$$\mathbf{P} = \begin{bmatrix} 200 \text{ W} \\ 2 \text{ W} \end{bmatrix} \text{ [7][8]} \tag{36}$$

para encontrar ejemplos de $\mathbf{q}^*$ y C^* para un creador. Entonces,

$$\mathbf{q}^* = \begin{bmatrix} 5 \text{ nodes} \\ 500 \text{ nodes} \end{bmatrix} \tag{37}$$
$$C^* \approx \$1{,}29 \text{ x } 10^{-4}. \tag{38}$$

Esto es entre 155 a 659 veces (99,35 % a 99,85 %) más barato que el precio a pedido de AWS Cloudfront (0,020 USD/GB a 0,085 USD/GB) [9]. Cada nodo del grupo 1 manejaría 100 MB mientras que cada nodo del grupo 2 manejaría 1 MB. Cada nodo en el grupo 1 y 2 necesitaría más de 1,29 USD x 10^{-5} y 1,29 USD x 10^{-7}, respectivamente.

Para poner esto en perspectiva, suponga que Netflix es un socio potencial. En 2017, Netflix tuvo como promedio más de 140 millones de horas de contenido visto por día. En promedio, un video de Netflix es un GB/hora [11]. En la plataforma Ara, el gasto anual de 51,1 exabyte [12] sería de solo 6,6 millones de USD/año (0,2106 USD/segundo). Si calculamos que el costo de transmisión de Netflix es de 0,03 USD/GB [13], obtenemos 1,5 mil millones/año (46,61 USD/segundo). Utilizar la red de Ara casi cuadruplicaría los ingresos netos de Netflix de 2017 que fueron 558,9 millones de USD [14]. (Manhattan tiene 1,66 millones de habitantes [15] donde 287.008 suscriptores de Netflix transmiten 321,45 TB/día, 3,72 GB/s. Eso requiere 3.721 grupos de 2 nodos a una tarifa de 41,47 USD/día).

Referencias

[1] Wikipedia (2018, April 22), *Cobb–Douglas production function*, https://en.wikipedia.org/wiki/Cobb%E2%80%93Douglas_production_function

[2] Wikipedia (2018, June 5), *Total factor productivity*, https://en.wikipedia.org/wiki/Total_factor_productivity

[3] Spectrum (2017, December 29), *Broadband Label Disclosure*, p.2, https://www.spectrum.com/content/dam/spectrum/residential/en/pdfs/policies/Broadband_Label_Disclosure_Charter_122917.pdf

[4] Wikipedia (2018, March 10), *Hadamard product (matrices)*, https://en.wikipedia.org/wiki/Hadamard_product_(matrices)

[5] Electricity Local (2018, June 19), https://www.electricitylocal.com/states/new-york/new-york

[6] Wikipedia (2018, June 19), *Bandwidth (computing)*, https://en.wikipedia.org/wiki/Bandwidth_(computing)

[7] Energuide.be (2018, June 19), *How much power does a computer use? And how much CO2 does that represent?*, https://www.energuide.be/en/questions-answers/how-much-power-does-a-computer-use-and-how-much-co2-does-that-represent/54/

[8] R. Sohan, A. Rice, A. W. Moore, and K. Mansley, "Characterizing 10 Gbps Network Interface Energy Consumption," *The 35th Annual IEEE Conference on Local Computer Networks (LCN) Short Papers*, University of Cambridge, Computer Laboratory, July 2010, https://www.cl.cam.ac.uk/ acr31/pubs/sohan-10gbpower.pdf.

[9] Amazon Web Services Pricing (2018, June 19), *Amazon Cloudfront Pricing*, https://aws.amazon.com/cloudfront/pricing

[10] L. Matney (2017, December 11), "Netflix users collectively watched 1 billion hours of content per week in 2017," *Techcrunch*, https://techcrunch.com/2017/12/11/netflix-users-collectively-watched-1-billion-hours-of-content-per-week-in-2017

[11] K. Hubby (2017, May 23), "The surprising amount of data Netflix uses," *The Daily Dot*, https://www.dailydot.com/debug/how-much-data-netflix-use/

[12] Wikipedia (2018, June 20), *Exabyte*, https://en.wikipedia.org/wiki/Exabyte

[13] D. Rayburn (2009, July), "Stream This!: Netflix's Streaming Costs," *Streaming Media (June/July 2009)*, http://www.streamingmedia.com/Articles/Editorial/Featured-Articles/Stream-This!-Netflixs-Streaming-Costs-65503.aspx

[14] Netflix (2018, January 1), p.40, https://ir.netflix.com/static-files/20c3228d-bf1f-4956-a169-c8b76911ecd5

[15] Wikipedia (2018, July 5), *Manhattan*, https://en.wikipedia.org/wiki/Manhattan

[16] Statistica (2018, July 5), *Number of Netflix streaming subscribers in the United States from 3rd quarter 2011 to 1st quarter 2018 (in millions)*, https://www.statista.com/statistics/250937/quarterly-number-of-netflix-streaming-subscribers-in-the-us/

[17] United States Census Bureau (2018, July 5), *U.S. and World Population Clock*, https://www.census.gov/popclock/

III. Análisis de los premios previstos en Ara por Lester Kim

Modelo de la red

Representemos la red de nodos de Ara como un gráfico completo [1] $G = (V, E)$ donde V contiene N vértices donde cada vértice representa un modo y E contienen $\frac{N(N-1)}{2}$ extremos donde cada extremo representa un canal de comunicación entre dos nodos. Consideremos a C como una recopilación de contenido (en nuestro caso, un conjunto de archivos de entretenimiento digital) que todos los nodos quieren tener y consideremos que el subconjunto $S \subseteq V$ contiene todos los nodos que tienen C (es decir, $S = \{v \in V : C \in v\}$).

Consideremos el tiempo como $t \in \mathbb{N}$. At $t = 0$, $|S| = 1$. En $t = 0$, $|S| = 1$, por lo que solo hay un $v_0 \in V$ que tiene el contenido C; por lo tanto, solo hay un vértice que puede entregar una copia de C a los demás nodos de la red. Supongamos que todos los demás nodos $N - 1$ desean C y v_0 tiene suficiente ancho de banda para entregar C a solo un nodo. Desde $t = 0$ a $t = 1$, $|S|$ aumenta de 1 a 2. En general, en el tiempo t,

$$|S| = \begin{cases} 2^t & 0 \leq t < \log_2 N \\ N & t \geq \log_2 N. \end{cases} \tag{39}$$

Se debe tener en cuenta que $|S| = N$ empieza en $t = \lceil \log_2 N \rceil$.

$\forall s \in S, s$ entregará C a algunos $v \in V \setminus S$ solo si v paga s un monto de p. Consideremos que M es el presupuesto total de la red para la entrega de entretenimiento. Dividir esto de manera uniforme entre N nodos da $p = M/N$.

En $t = 0$, el único $v_0 \in S$ recibe p de algunos $v_1 \in V \setminus S$. Entonces, en $t = 1$, cada $v_0, v_1 \in S$ recibe p de algunos $v_2, v_3 \in V \setminus S$. En cualquier $t < \lfloor \log_2 N \rfloor$, cada uno de los $|S| = 2^t$ nodos en S recibe p desde 2^t nodos en $V \setminus S$. En $t = \lfloor \log_2 N \rfloor$, $|S| > \frac{N}{2}$, por lo tanto, hay más proveedores que solicitantes de C. Cuando eso ocurre, los $N - 2^t$ nodos de S se seleccionan de manera aleatoria para entregar C. En $t = \lceil \log_2 N \rceil,\ S = V$.

En este modelo, v_0 obtiene al menos

$$\frac{M \lfloor \log_2 N \rfloor}{N}; \tag{40}$$

v_1 obtiene al menos $\frac{M(\lfloor \log_2 N \rfloor - 1)}{N}$; v_k obtiene al menos

$$\frac{M(\lfloor \log_2 N \rfloor - \lceil \log_2 (k+1) \rceil)}{N}. \tag{41}$$

La mayor k como v_k obtiene al menos $\frac{M}{N}$ es cuando

$$\lfloor \log_2 N \rfloor - \lceil \log_2 (k+1) \rceil \geq 1 \tag{42}$$

lo que implica

$$\log_2 (k+1) \leq \log_2 \frac{N}{2}. \tag{43}$$

Por lo tanto, el valor máximo de k para obtener al menos $\frac{M}{N}$ es $k = \lfloor \frac{N}{2} \rfloor - 1$. En promedio, cada uno obtiene

$$\frac{M - \frac{M}{N}}{2^{\lceil \log_2 N \rceil - 1}} = \frac{M(1 - \frac{1}{N})}{2^{\lceil \log_2 N \rceil - 1}}. \tag{44}$$

El numerador es $M - \frac{M}{N}$ to exclude v_0 a fin de excluir el presupuesto de entretenimiento de v_0 ya que el mismo tenía C en $t = 0$. El denominador es $2^{\lceil \log_2 N \rceil - 1}$ porque en $t = \lceil \log_2 N \rceil - 1$, $|S| = 2^{\lceil \log_2 N \rceil - 1}$ y en ese punto, S consistía de todos lo nodos que tenían la posibilidad de obtener premios durante todo el proceso. Esto significa que existen $N - 2^{\lceil \log_2 N \rceil - 1}$ nodos que no podrán obtener premios.

Ejemplo

Aproximadamente el 80 % de los estadounidenses tiene computador con acceso a Internet [2]. Teniendo en cuenta que existen 327 millones de habitantes que viven en los EE. UU. [3], existen (0,8)(327M) = 261,6M estadounidenses con dispositivos conectados a Internet. Suponiendo que cada uno tiene un dispositivo, entonces N = 261,6M. El consumo anual de entretenimiento en EE. UU. es de 734 mil millones de USD [4] [5]. Supongamos que la mayor parte de este gasto en los años venideros será contenido digital, pero incluyamos únicamente el presupuesto del 80 % de los estadounidenses que tiene acceso a Internet de forma que el nivel de gasto entre los mismos es de (0,8)(734B) = $587B. Consideremos que el 10 % del gasto será para cubrir los costos de distribución. Entonces, M = (0,1)($587B) = $58,7B. Entonces, $p = M/N$ = $58,7B/261,6M = $224,39. Por (44), las ganancias anuales promedio serán de por nodo. Por (40), lo máximo que el v_0 puede obtener es $6282,87. Por lo tanto, este ejemplo ilustra que los pares iniciales que comparten contenido obtendrán el mayor nivel de premios.

Referencias

[1] Wikipedia (2018, June 19), *Complete graph*, https://en.wikipedia.org/wiki/Complete_graph

[2] C. Ryan and J. M. Lewis, "Computer and Internet Use in the United States: 2015," *American Community Survey Reports* U.S. Census Bureau, September 2017 https://www.census.gov/content/dam/Census/library/publications/2017/acs/acs-37.pdf

[3] World Population Review (2018, June 18) *United States Population 2018* http://worldpopulationreview.com/countries/united-states-population/

[4] SelectUSA (2018, August 22), *MEDIA AND ENTERTAINMENT SPOTLIGHT*, https://www.selectusa.gov/media-entertainment-industry-united-states

[5] Bureau of Labor Statistics (2017, August 29), *CONSUMER EXPENDITURES–2016* https://www.bls.gov/news.release/cesan.nr0.htm