



IMPLEMENTACIÓN DE LA BLOCKCHAIN DEL PADRÓN FEDERAL EN LA AGENCIA DE RECAUDACIÓN TRIBUTARIA DE LA PROVINCIA DE RÍO NEGRO

NAHUEL EMILIANO PICHON

Directora: Sonia Alejandra Formia

**Trabajo Final de Carrera
NOVIEMBRE 2022
LICENCIATURA EN SISTEMAS**

IMPLEMENTACIÓN DE LA BLOCKCHAIN DEL PADRÓN FEDERAL EN LA AGENCIA DE RECAUDACIÓN TRIBUTARIA DE LA PROVINCIA DE RÍO NEGRO

Nahuel Emiliano Pichon

Trabajo Final de Carrera
Licenciatura en Sistemas
Universidad Nacional de Río Negro

nahuelpichon@gmail.com

Resumen. En los últimos años blockchain provocó un auge en lo que respecta a la seguridad, confiabilidad y transparencia de los datos. Varios gobiernos ven la utilización de esta tecnología como una mejora en la prestación de servicios. El presente trabajo propone analizar las características de blockchain que permiten obtener un mayor beneficio en cuanto a mecanismos de estado y sistemas de gobernanza. Se presenta Blockchain Federal Argentina (BFA), una plataforma multiservicios abierta y participativa pensada para integrar servicios y aplicaciones sobre la cadena de bloques. Como ejemplo, se analiza la implementación en la Agencia de Recaudación Tributaria de la provincia de Río Negro que utiliza la cadena de bloques para mejorar sus procesos de coordinación con la Administración Federal de Ingresos Públicos y la velocidad de respuesta de sus servicios públicos.

Palabras clave: blockchain, smart contracts, organismos públicos, mecanismo de consenso, confianza, DAOs

Agradecimientos

En primer lugar a mis padres, porque sin ellos, nada hubiera sido posible, por seguirme y guiarme, creyendo y aportando en cada idea, actividad, proyecto y objetivo propuesto.

A mi hermano, pilar fundamental en mi vida, mi amigo, quien me acompaña en cada momento, desde lo más simple hasta las ideas más extrañas. Siendo el primero en escuchar, opinar, conocer y aportar. Gracias Tomy, desde lo profundo de mi ser.

A mi familia y amigos, por estar siempre y sin pensar a mi lado, alegrándose de los logros y apoyándose en las derrotas.

A mi directora de trabajo final, Sonia Formia, por quien tengo un profundo respeto y admiración, como profesional en la materia y como docente en la carrera. Gracias por el tiempo ofrecido y los conocimientos aportados en todos estos años.

Y por ultimo, agradezco a la Universidad Nacional de Río Negro, al personal docente y no docente, su esfuerzo es recompensado cada vez que un alumno consigue el objetivo que ellos, la facultad y la misma carrera le han marcado, su trabajo es formar profesionales, y así lo hicieron.

Nahuel Emiliano Pichón.

Indice

1. Introducción	5
2. Antecedentes	6
3. El Dilema de la Confianza	8
4. Que es BlockChain	10
5. Protocolos de Consenso	11
5.1 Implementación del Protocolo de Consenso	12
6. Implementación de BlockChain en el Sector Público	15
6.1 BlockChain Federal Argentina	16
6.2 Archivo de documentos y registros	18
6.3 Títulos Académicos	18
6.4 Licitaciones Públicas	19
7. Smart Contracts	20
7.1 Utilización de los Smart Contracts para el desarrollo de aplicaciones	21
7.2 Utilización de los Smart Contracts para la implementación de propiedades inteligentes	22
8. Organizaciones Autónomas Descentralizadas (DAOs)	24
8.1 Ventajas y desventajas de las DAOs	25
8.1.1 Ventajas	25
8.1.2 Desventajas	26
9 Aplicación de la Cadena de Bloques en la Agencia de Recaudación Tributaria de Río Negro	27
9.1 HLF-PROXY	27
9.2 BLOCK-CONSUMER	29
9.3 Herramientas y Entornos de Trabajo	31
9.4 Sistema de control de versiones de código fuente	31
9.5 Herramientas de Desarrollo	31
9.6 Entorno de Trabajo (FrameWork)	32
9.7 Metodología de Trabajo	33
10. Módulo de Alta de Contribuyentes	34
10.1 Primera etapa - Alta de Contribuyente Físico	35
10.1.1 Análisis y Diseño	35
10.1.2 Implementación	36
10.1.3 Pruebas y Resultados	40
10.2 Segunda etapa - Actualización y cambios de cuits	42
10.2.1 Análisis y Diseño	42
10.2.2 Implementación	43

10.2.3 Pruebas y Resultados	44
10.3 Tercera etapa - Alta de Domicilio Fiscal Electrónico	45
10.3.1 Implementación	45
10.3.2 Pruebas y Resultados	46
10.4 Cuarta etapa - Alta de Contribuyente Persona Jurídica	48
10.4.1 Análisis y Diseño	48
10.4.2 Implementación	50
10.4.3 Pruebas y Resultados	53
11. Módulo de Actualización de Domicilios Fiscales	56
11.1 Análisis y Diseño	56
11.2 Implementación	57
11.3 Pruebas y Resultados	60
12. Módulo de Consulta del Estado del Contribuyente en BCPadFed	62
12.1 Análisis y Diseño	62
12.2 Implementación	63
13. Conclusiones	65
14. Referencias Bibliográficas y Otras Fuentes	67

1. Introducción

Bajo condiciones siempre dinámicas y cambiantes, el desarrollo del sector tecnológico y de las comunicaciones, ha sido, en el mundo y en la Argentina, muy significativo. Estas tecnologías, incorporadas en la rutina de las personas, están hoy presentes en todas las actividades económicas y sociales y están cambiando irreversiblemente nuestra manera de producir, comerciar y trabajar, y las formas de educarnos, entretenernos, comunicarnos e incluso relacionarnos.

Las herramientas de comunicación han evolucionado desde su aparición a través de varios hitos durante los últimos 50 años. Durante el transcurso de esta transformación, hemos pasado por diferentes etapas, comenzando con la búsqueda de productividad y eficiencia, y luego atravesando la era de Internet con su revolución en las comunicaciones y el acceso a la información. Ahora también se suma la relación mediante las plataformas de redes sociales, el entretenimiento y la movilidad y, actualmente, conceptos un poco más complejos, como Big Data, Internet de la Cosas y, una de las herramientas que parece ser el camino hacia una nueva ola de cambios y que se posiciona como troncal a las nombradas recientemente, las Cadenas de Bloques o Block Chains.

En un período relativamente breve, el uso de las nuevas tecnologías ha pasado de ser solo para una minoría, a ser masivo, transformando muchos de los paradigmas con los que nos manejamos en nuestra vida. Estamos viviendo una era de grandes cambios, no solo tecnológicos, sino en el hábito de los usuarios, en los modelos de negocio, formas de prestar servicios y en la intermediación comercial. También son de esperar cambios en las políticas y regulaciones del mercado y de los usos de estas mismas tecnologías, lo que implica cambios profundos en el ecosistema de internet y las nuevas tecnologías.

El desarrollo de este documento tiene como objetivo presentar una visión detallada sobre el funcionamiento de la tecnología BlockChain y lo revolucionario de esta idea basada en un mecanismo de consenso. Veremos cómo el Estado Federal Argentino está implementando BlockChain para el desarrollo de múltiples plataformas que ayudan a la gobernanza digital del país y cómo la tecnología cuenta con la suficiente robustez y confianza como para considerarse un recurso principal de seguridad y de resguardo de información. Además, se analizará cómo la misma herramienta tiene facultades para avanzar e involucrarse en los

mecanismos y protocolos de toma de decisión y acercamiento a la comunidad. Presentaremos casos de uso sobre la BlockChain Federal Argentina (BFA) y nos adentraremos en las nuevas estructuras organizativas denominadas Organizaciones Autónomas Descentralizadas (DAOs), basadas puramente sobre la cadena de bloques y los Smart Contract. Por último estudiaremos la implementación desarrollada en la Agencia de Recaudación Tributaria de Río Negro (ARTRN) utilizando la BlockChain del Padrón Federal (BCPadFed).

2. Antecedentes

En el año 1991, Stuart Haber y W. Scott Stornetta, tuvieron la visión de lo que muchas personas han llegado a conocer como blockchain. Su primer trabajo consistió en crear una cadena de bloques protegida criptográficamente en la que nadie podía manipular las marcas de tiempo de los documentos. En 1992, actualizaron su sistema para incorporar *árboles de Merkle*¹ utilizados para mejorar la eficiencia de la herramienta, lo que permitía la recopilación de más documentos en un solo bloque.

En 1993, el matemático estadounidense David Chaum creó un sistema de pagos digitales que permitía realizar transacciones en línea. La revista Next! describió este producto como *“un producto técnicamente perfecto que permitía pagar por internet de manera segura y anónima [...] Servía perfectamente para enviar peniques y céntimos electrónicos por internet.”*[1]. Sin embargo, para ese entonces los compradores no se preocupaban por la confidencialidad y la seguridad de los datos, por lo tanto, con el avance de la tecnología y la desconfianza en la privacidad y garantías de la red por parte de los usuarios, la empresa quebró en 1998.

Previamente, en 1997, uno de los socios de Chaum, llamado Nick Szabo, había escrito un artículo titulado “El protocolo de Dios”, parafraseando la expresión “la partícula de Dios” con la que el premio Nobel Leon Lederman se refería a la importancia del **Bosón de Higgs**² en la física moderna. En este artículo, Szabo planteaba la creación de un protocolo tecnológico ideal en el que Dios fuera el mediador fiable de todas las operaciones. Esta era

¹ **Árbol de Merkle** o árbol hash es una estructura de datos creada con el objetivo de facilitar la verificación de grandes cantidades de datos. Para lograr esto, cada nodo hoja se encripta y se suma a su padre que repite la acción hasta llegar al nodo raíz, este último se compone de la encriptación de todos los nodos hijos.

² **Bosón de Higgs:** nombre real de la denominada “partícula de Dios”. Recibe su nombre en honor a Peter Higgs y es utilizado, junto al “mecanismo de Higgs” para explicar el origen de la masa de las partículas elementales.

una idea muy interesante: realizar transacciones monetarias en internet requiere mucha confianza. En la actualidad, la robustez y solidez de las infraestructuras ha mejorado considerablemente, no obstante, hay casos en los que aún debemos tratar con los intermediarios como si fueran dioses. [2]



Figura 1. Línea de tiempo sobre los eventos que antecedieron a la creación de BlockChain

Sin embargo, es en 2008 que Blockchain vuelve a ser escuchado en los artículos científicos de la época, gracias al trabajo de una persona o grupo con el nombre de Satoshi Nakamoto, el cerebro detrás de la tecnología blockchain. Si bien no se conoce mucho acerca de este grupo de individuos o sobre la persona creadora, se cree que trabajaron en la primer aplicación de registro digital conocido como **Bitcoin**³. Para una tecnología conceptualizada por primera vez en el año 2008 y vinculada originalmente a una aplicación tan difícil de aprender como una criptomoneda digital, el nivel de atención y de relevancia resulta completamente inusual. En cierto sentido, la tarea más compleja que blockchain como tecnología debe superar, es la de ser capaz de desprenderse de la complicada herencia que la vincula con bitcoin y ver cómo puede ser rápidamente adoptada para aplicaciones de todo tipo [6].

En la Figura 1. puede verse la línea del tiempo de los eventos enunciados, que anteceden al surgimiento de Blockchain.

³ **Bitcoin:** primera criptomoneda presentada por Satoshi Nakamoto, en el año 2008, junto con el protocolo blockchain.

3. El Dilema de la Confianza

La confianza en esta era digital parece ser uno de los problemas a enfrentar día a día. Las transacciones monetarias, el registro de datos personales, los movimientos de contribuyentes del estado, el seguimiento de trámites importantes, son solo algunos de los ejemplos que se pueden mencionar y en donde necesitamos que la información brindada sea transparente y honorable. Sin embargo, estas características, aún vienen sujetas por la facultad de ciertos individuos, intermediarios u otras entidades que actúan con sinceridad y respaldo. Como muchas veces no podemos conocer a la otra parte, ni menos aún si es íntegra, tenemos que fiarnos de terceros no sólo para que respondan a favor de desconocidos, sino también para que registren las transacciones y operen con la lógica propia del comercio en línea. Estos poderosos intermediarios - bancos, gobiernos, tarjetas de crédito y otros grupos empresariales - se llevan gran parte del valor que significa generar un cambio de estado transparente en las transacciones.

Este es el inconveniente que se pretende resolver. Nick Szabo fantaseaba con un protocolo de confianza ideal, en donde ponderaba al mecanismo como una deidad. Sus palabras recitaban: *“imaginemos un escenario en donde todas las partes confiaran y enviaran la información a Dios. Dios manejaría esa información y devolvería el resultado. Como Dios es la suma de la discreción y la confidencialidad, ninguna de las partes sabría de las demás más de lo que sabe de sí misma”* [2].

Satoshi Nakamoto plantea una solución más terrenal al dilema de la confianza, lo que nos permitirá introducirnos de lleno en la temática a abordar. En su escrito el autor propone: *“Lo que se necesita es un sistema de pagos electrónicos basado en pruebas criptográficas en vez de confianza, permitiéndole a dos partes interesadas en realizar transacciones directamente sin la necesidad de un tercero confiable.”*

En este trabajo, Dios estará representado por la misma red de bloques que se encargará de validar y verificar que la información cargada sea realmente cierta. La idea de un ente central y a su vez distribuido es lo revolucionario de esta idea, como un Dios omnipresente velando por la seguridad de los datos. La utilización de las BlockChains permitiría evitar los terceros de confianza, ya que el funcionamiento de la misma red y sus

nodos, asegura esta propiedad, el registro mismo es un fundamento de la protección e integridad de los datos.

El concepto que envuelve a esta tecnología está plagado de singularidades y la herramienta puede programarse para asentar prácticamente todo lo que tenga valor e importancia para la humanidad, desde la información más irrelevante, hasta los mecanismos más complejos.

4. Que es BlockChain

Blockchain, como la traducción de su nombre lo dice, es una cadena de bloques que almacena datos sobre distintas transacciones y que utiliza protocolos de seguridad certificados para que esta información se vuelva inalterable. En otros términos, blockchain (BC) es un libro de registro distribuido y punto a punto que nos brinda confianza y cuya base de datos se encuentra replicada en todos los nodos pertenecientes a la red. [8]

El contenido del registro solo se puede actualizar agregando otro bloque vinculado al bloque anterior. La base de datos creada se comparte entre los participantes de la red de manera transparente, por lo que todos pueden acceder a sus contenidos y la gestión de la base de datos se realiza de forma autónoma utilizando redes punto a punto y un servidor de sellado de tiempo. La estructura general de esta herramienta es sencilla, cada bloque en una blockchain está organizado de tal manera que hace referencia al contenido del bloque anterior, como se muestra en la Figura 2.

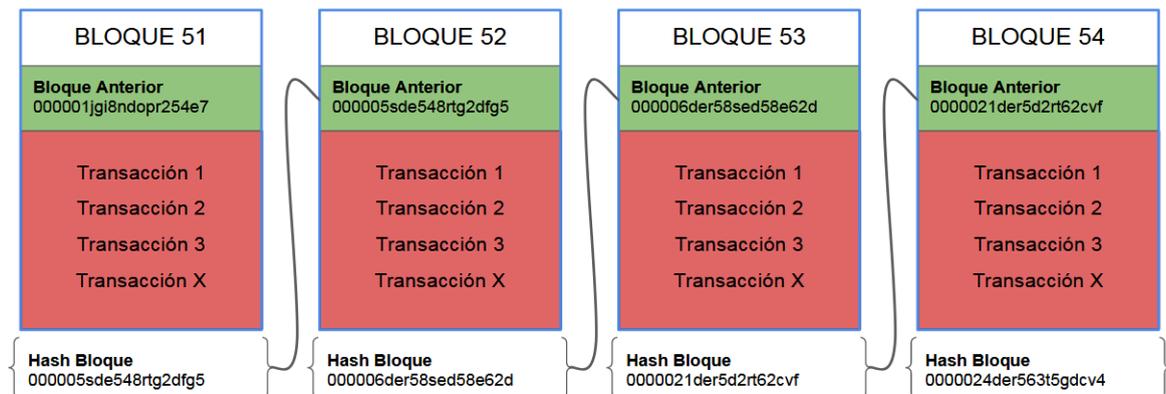


Figura 2. Ejemplo gráfico sobre el enlace de los bloques pertenecientes a una BlockChain.

Como se mencionó anteriormente, la blockchain es pública: esto quiere decir que todo el mundo puede verla cuando quiera, porque la información reside en la red, no en una determinada institución que se encargue de auditar las transacciones y llevar registros. Y además está encriptada: usa una encriptación que incluye claves públicas y privadas (en lugar de los

sistemas de dos claves de las cajas fuertes) que garantizan una total seguridad y confiabilidad.

La característica principal reside en que cada bloque debe referirse al bloque anterior para ser válido. Esta estructura registra exactamente el momento de las transacciones y las almacena, evitando que alguien pueda alterar el registro, esto se hace a través de protocolos de consenso que son aceptados e implementados por todos los nodos [8]. Si se quisiera modificar la información de alguno de los registros de un bloque, se tendría que reescribir toda la cadena de bloques a la vista de todos, lo que es prácticamente imposible [4]. Por eso las BlockChains son un registro distribuido y suponen la conformidad de la red con todas las transacciones que se han realizado.

5. Protocolos de Consenso

Los protocolos de consenso, son una parte fundamental de la tecnología y su naturaleza descentralizada. Las pruebas de consenso son las que permiten a la blockchain trabajar de forma automática. Esto gracias a que permiten establecer parámetros y reglas claras para que cada miembro de la red pueda aceptar y verificar los datos manejados por la cadena de bloques sin la necesidad de dar a conocer los recursos utilizados para tal acción.

En Blockchain, el hecho de completar un bloque con transacciones y sumarlo a la cadena, se denomina sellado. Cuando un bloque queda sellado, la información que contiene pasa a formar parte de la cadena de forma permanente, no puede ser modificado sin que se alteren todos los bloques que están enlazados con él, una operación que además necesitaría que la mayoría de los nodos la validaran.[9] Un “Protocolo de Consenso” es el mecanismo que se utilizará para regular la forma en que los nodos que sellan bloques lleguen a un acuerdo entre sí para poder hacerlo.

Existen dos métodos posibles para llevar a cabo un protocolo de consenso, por un lado se encuentra el mecanismo denominado Prueba de Trabajo, y por el otro, el denominado Prueba de Autoridad.

En el mecanismo de Prueba de Trabajo, todos los nodos son pares iguales en la red, y todos compiten para sellar un bloque antes que el resto y poder conseguir una recompensa a cambio. Para realizar esto, deben resolver un algoritmo complejo. El que primero logre hacerlo y pueda

agregar un bloque a la cadena es quien obtendrá la **criptomoneda**. Pero para realizar ese trabajo se necesita un alto nivel de procesamiento, lo que se termina traduciendo en un mayor costo energético.

En el mecanismo de Prueba de Autoridad, solo hay una cantidad determinada de nodos autorizados a resolver el sellado de bloques. Como esta tarea no percibe ningún tipo de recompensa por la acción de sellado, este protocolo no está basado en la competencia, sino en el hecho de que ese grupo reducido que tiene permisos para agregar bloques a la cadena se turne para hacerlo. Como aquí no hay necesidad de resolver algoritmos complejos, la cantidad de procesamiento es mínima. Por eso se considera a estos modelos como livianos y más eficientes en relación a consumo energético.

5.1 Implementación del Protocolo de Consenso

Cualquiera sea la opción elegida para implementar el Protocolo de Consenso, es necesario saber cómo este funciona, la diferencia circunstancial entre ambas es la recompensa obtenida al lograr sellar un bloque nuevo.

A partir de la figura 3 se puede ver gráficamente el sistema de enlace de los bloques de una red. Para implementar estos enlaces se utilizará el mecanismo de consenso deseado, para ello, las autoridades que inician la red de nodos deben elegir un sistema de encriptación con el que se crearán las claves únicas. Una vez elegido esto, el nodo minador se envuelve en la exploración de un valor nuevo, combinando el contenido del bloque a sellar (sería la suma de las transacciones del bloque), el hash o llave del bloque anterior y un número o valor aleatorio denominado **nonce**⁴, esto dará como resultado una nueva clave que será única e identificará al bloque que se está sellando.

⁴ **Nonce:** En criptografía, un nonce es un número arbitrario que se puede usar una única vez en una comunicación criptográfica. En una red blockchain basada en Prueba de Trabajo, el nonce funciona en combinación con el hash como un elemento de control para evitar la manipulación de la información de los bloques. Este número aleatorio garantiza que los hash antiguos no se pueden volver a utilizar en lo que se denominan ataques de repetición.

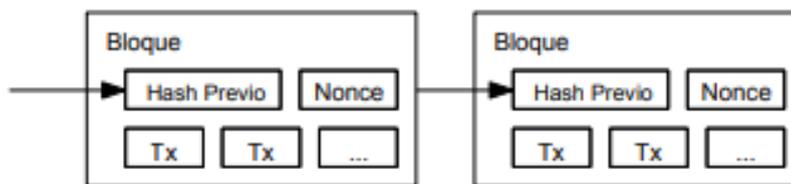


Figura 3. Nakamoto S. (2008). *Bitcoin: Un Sistema de Efectivo Electrónico Usuario-a-Usuario*. Recuperado de https://bitcoin.org/files/bitcoin-paper/bitcoin_es_latam.pdf

Cabe destacar que el nuevo valor debe cumplir con las condiciones determinadas por la lógica que lleva adelante a la red de nodos (por ejemplo, que el nuevo hash comience con una cierta cantidad de dígitos en cero). Para esto se utiliza el nonce como número aleatorio o valor comodín, que debe permutar indeterminadas veces hasta que el resultado de la encriptación coincida con las cláusulas definidas.

Cuando se ha gastado el esfuerzo de CPU para satisfacer el protocolo de consenso, ya no se puede cambiar el bloque sin rehacer todo el trabajo. Mientras más bloques se encadenen después de éste, para cambiar el bloque habría que rehacer todos los bloques siguientes con el esfuerzo que ello conlleve. [4]

El protocolo de consenso resuelve por naturaleza el problema que surge si dos nodos emiten versiones diferentes del próximo bloque simultáneamente, algunos nodos puede que reciban uno o el otro primero, esto se soluciona utilizando la decisión por mayoría. Los nodos siempre toman la cadena más larga como la correcta y realizan las tareas necesarias para extenderla. Cuando se da la situación antes mencionada, los nodos trabajan en el primer bloque recibido y mantienen la otra rama resguardada en caso de que ésta se vuelva más larga. Si la decisión de la mayoría es representada por la cadena más larga, la cual tiene el mayor esfuerzo en cuanto recursos y procesamiento invertido en ella, quiere decir que es controlada por la mayor cantidad de nodos honestos, la cadena honesta crecerá más rápido y pasará a cualquier otra cadena con la que esté compitiendo. La resolución llega cuando el próximo bloque es sellado y una rama se vuelve más larga; de esta manera los nodos que estaban en la rama más corta luego se cambian a la más larga. Esta forma de trabajar es una de las mayores características de las blockchains, ya que, si un atacante quisiera modificar un bloque en el pasado, tendría que rehacer toda la cadena y luego alcanzar y pasar el trabajo de los nodos honestos.

Las nuevas transacciones no necesariamente necesitan llegar a todos los nodos. Cuando llegan a muchos nodos, entrarán a un bloque antes de que pase mucho tiempo. Las emisiones de bloques toleran mensajes perdidos. Si un nodo no recibe un bloque, en el momento que reciba el próximo bloque se dará cuenta que se perdió uno y lo solicitará [4].

6. Implementación de BlockChain en el Sector Público

Como es de público conocimiento, y se nombró en párrafos anteriores, el principal problema de la blockchain es desprenderse de su aplicación primaria y ampliamente utilizada, las criptomonedas. Esta tecnología de registro digital de transacciones puede utilizarse para asentar cualquier tipo de trámite, procedimiento, título registral, informe económico o financiero, licitación pública, etc., de hecho cualquier cosa que pueda codificarse. Es necesario que el sector público deje atrás el papel. El uso de las blockchain como mecanismo de validación y resguardo de información da la posibilidad de entrar a un mundo más activo e inteligente, en el que es posible realizar un seguimiento en tiempo real de todas las acciones realizadas en línea. Además, los autores consultados resaltan lo siguiente:

“En muchas instancias la generación de información en el sector público se basa en una cadena de procesos compuesta por distintas personas o entidades. A través de la tecnología, la información puede registrarse de manera segura y confiable, convirtiendo a la red en una especie de notariado digital de datos y transacciones. Potencialmente, el incluir estos procesos en una cadena de blockchain permitirá prescindir de algunos de estos intermediarios, aumentar la trazabilidad de cada etapa del proceso de manera confiable y reducir costos tanto en tiempo como en recursos.”[6]

Esta tecnología puede crear las condiciones necesarias de custodia y trazabilidad de recursos en programas sociales del sector público, de tal manera que la distribución de esos recursos se realice de manera automática, sin desvíos innecesarios y sin altos costos administrativos. Esto es posible a partir del uso de contratos inteligentes (o Smart Contracts⁵), lo que permite llevar a cabo tareas preestablecidas utilizando la confiabilidad y transparencia que nos ofrece la cadena de bloques.

En este punto, el uso de la herramienta se vuelve realmente importante ya que nos permite llevar a cabo tareas que deben ser monitoreadas durante todo el flujo de trabajo, y que además significan la utilización de un gran caudal de recursos humanos y materiales que fácilmente pueden ser

⁵ **Smart Contracts:** flujos de tareas programables dentro de Blockchain, que abren la posibilidad de desarrollar aplicaciones. <https://bfa.ar/blockchain/smart-contracts>

corrompidos o simplemente no tenerlos disponibles al momento de realizar la labor.

Sin embargo, el desarrollo de aplicaciones, basadas en la cadena de bloques, que ayuden a los procesos administrativos no es una tarea sencilla. Los autores Serale, Redl y Muent-Kunigami describen ciertos retos que se deben enfrentar para llevar a cabo una puesta en marcha exitosa de sistemas basados en blockchain: “...la implementación de una solución exitosa – y sobre todo sostenible– utilizando tecnología blockchain presenta algunos retos: (i) organizacionales y de gobernanza, (ii) tecnológicos, (iii) regulatorios, (iv) de recursos, y (v) de uso y generación del ecosistema.”[6]

Estos retos pueden tomarse como premisa si se quiere llevar a cabo un desarrollo de esta índole. Los inconvenientes presentados son el resultado de procesos analógicos que se utilizan hace mucho tiempo, no obstante, para avanzar en la implementación de un mejor sistema de gobernanza electrónica, es necesario comenzar por resolver los casos estudiados.

6.1 BlockChain Federal Argentina

La tecnología blockchain también está revolucionando los mecanismos de gobernanza, permitiendo gobiernos más abiertos, con mejores posibilidades para la toma de decisiones de manera independiente. Las cadenas de bloques pueden contribuir en las funciones del ciudadano, tanto en los procesos políticos como en servicios sociales, resolviendo varios de los problemas actuales de la sociedad.

Un claro ejemplo de la implementación de esta nueva tecnología es BlockChain Federal Argentina (BFA), una plataforma multiservicios abierta y participativa pensada para integrar servicios y aplicaciones sobre blockchain. A través de **BFA** se consolida una iniciativa confiable y completamente auditable, construida en un marco de la colaboración entre los sectores públicos, privados, técnicos y académicos y que a su vez funciona como herramienta de crecimiento para toda la comunidad.

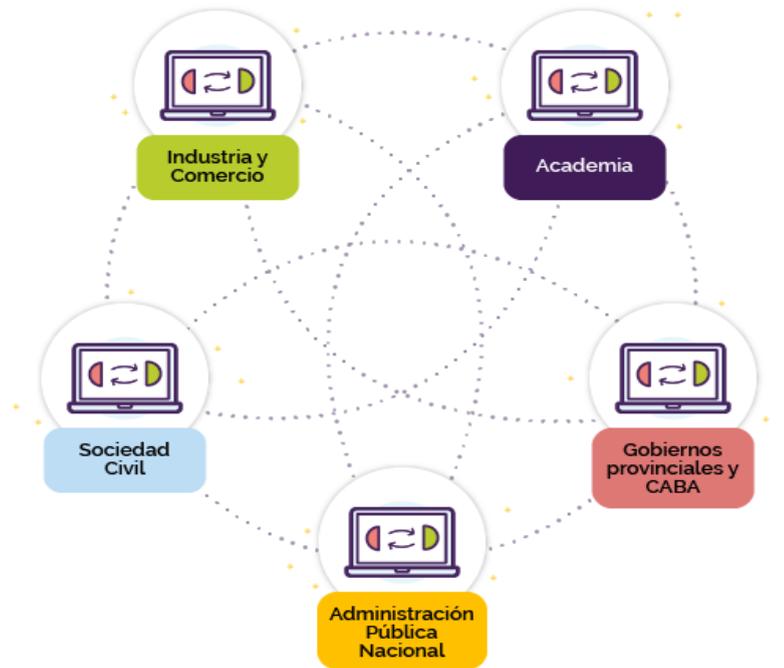


Figura 4. Blockchain Federal Argentina. Extraída de <https://bfa.ar/>

La Figura 4 muestra el modelo de Múltiples Partes Interesadas que es la base de una gobernanza en la que estén representados todos los sectores para la toma de decisiones. Blockchain Federal Argentina, al ser una plataforma pública, garantiza la libertad, visibilidad y auditabilidad de toda la información, no solo para las organizaciones que participen del proyecto, sino para toda la comunidad, que encuentra las puertas abiertas para participar en BFA. Aquellos individuos, organismos, instituciones o empresas que pretendan desarrollar aplicaciones que sirvan como base para el crecimiento de toda la comunidad, pueden hacerlo, aprovechando las características de la plataforma, o simplemente contribuyendo sobre el primer desarrollo de esta índole en el país.[9]

Repasemos algunos ejemplos que darán cuenta sobre los avances sobre esta tecnología.

6.2 Archivo de documentos y registros

Como se ha explicado, los activos materiales pueden convertirse en activos digitales. Toda la documentación relativa a un objeto en particular puede digitalizarse y subirse a una blockchain, como, por ejemplo, patentes, títulos de propiedad, garantías, certificados de inspección, de procedencia, seguros, fechas de repuesto, vistos buenos, etc., lo que aumentaría notablemente la disponibilidad e integridad de la información, reduciría el papeleo administrativo, facilitaría el archivo y evitaría pérdidas, esto mejoraría el procesamiento de esa documentación [1].

6.3 Títulos Académicos

Los actores involucrados en este proceso son aquellos alumnos que finalizan su carrera y solicitan su título académico, la institución educativa que valida la aprobación de las materias y emite el certificado dejando constancia de la culminación de la carrera, y el Ministerio de Educación que certifica el título en cuestión.

En la actualidad, los modelos de emisión y digitalización de títulos, no proveen un mecanismo para asegurar la inmutabilidad y transparencia de los datos utilizados y los certificados emitidos, sobre todo cuando en el proceso se encuentran incluidas diversas entidades durante toda la tramitación. Esto, además, está sujeto a la necesidad de legalizar copias del título cada vez que es necesario, durante lo que dure el procedimiento de emisión o luego de entregado el diploma.[9]

Gracias a los “sellos de tiempo” utilizados en las blockchains, es posible garantizar que los documentos involucrados en cada una de las etapas del trámite, no puedan ser modificados. Los sellos de tiempo generan claves criptográficas de la historia académica y de los certificados emitidos, estas claves quedan almacenadas en la red de bloques, a través de ellas se puede garantizar que los documentos no han sido corrompidos en ninguna etapa del proceso.

Esto se explica más en detalle en el apartado de Casos de Uso de BlockChain Federal Argentina. ([Títulos Académicos - Blockchain Federal Argentina \(bfa.ar\)](https://www.bfa.ar/))

6.4 Licitaciones Públicas

Supongamos un proceso de licitación pública, este tipo de concurso abierto es una de las principales vías de relación y comunicación entre el Estado y el sector privado. También, la sociedad interviene como parte interesada en los mecanismos de auditoría. Los recursos públicos se han ido modernizando en los últimos años, pero el seguimiento de las tareas que se realizan en cada una de las etapas y los procesos de control administrativos resultan ajenos a la mayoría de los ciudadanos. Por este motivo, resulta necesario que los concursos públicos se realicen en un marco que contribuya a favorecer las buenas prácticas y que sea transparente a los ciudadanos.

Al incorporar Blockchain a un proceso de licitación, encontramos nuevas formas de facilitar el proceso de auditoría, tanto a los oferentes como a la sociedad en general. Esta tecnología posibilita el desarrollo de una plataforma para la compra de bienes y la contratación de servicios por parte del Estado, que garantice transparencia e impida cualquier tipo de fraude. [9]

Este desarrollo fue implementado utilizando BlockChain Federal Argentina y puede conocerse más detalles del mismo en el apartado siguiente: [Licitaciones - Blockchain Federal Argentina \(bfa.ar\)](https://bfa.ar/).

Además, la utilización de “smart contract” permite establecer parámetros facilitando la automatización del proceso, de manera transparente y permitiendo, entre otras cosas, aceptar propuestas o rechazar aquellas que no cumplan con los requerimientos, notificar automáticamente al ganador o ejecutar los procesos administrativos vinculados. De esta manera, la evaluación puede realizarse de forma virtual de acuerdo a las condiciones de la convocatoria, pero es completamente pública y transparente y se pueden fiscalizar las órdenes de compras y las etapas de cumplimiento hasta el fin del proceso, incluyendo la ejecución del contrato.

7. Smart Contracts

Los contratos inteligentes son códigos informáticos que se encargan de ejecutar los servicios ofrecidos de un contrato de manera automática si se verifica que se han cumplido las condiciones a las que los servicios estaban ligados. Esto quiere decir que, un smart contract es un mecanismo que refleja el acuerdo de las partes interesadas sobre cómo se ejecutarán una o varias de las prestaciones asumidas, de manera automática, sin intervención de otros actores y, siempre y cuando, se den las condiciones antes mencionadas.

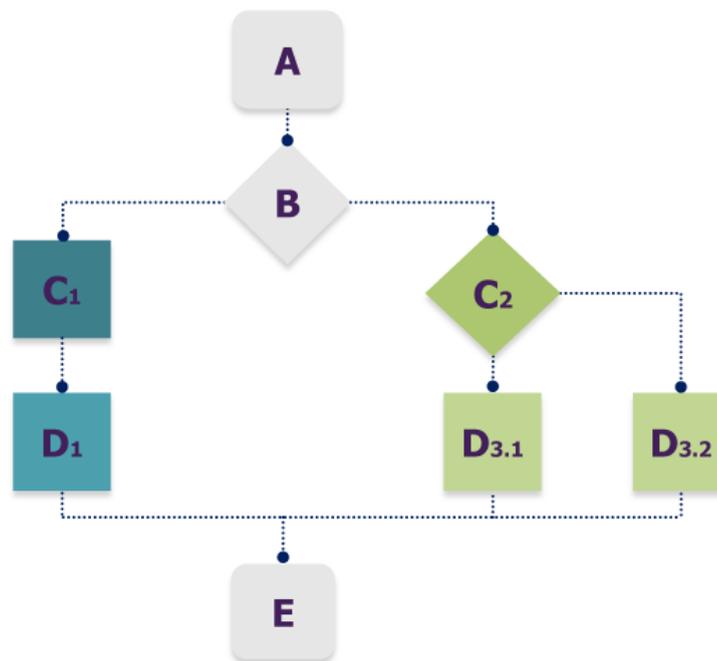


Figura 6. Blockchain Federal Argentina. Extraída de <https://bfa.ar/>

Este concepto fue nombrado por Nick Szabo en 1994 quien definió a los contratos inteligentes como “*Un protocolo transaccional computarizado que ejecuta los términos de un contrato. Los objetivos generales [de su diseño] son satisfacer las condiciones contractuales comunes, minimizar las excepciones temerarias y fortuitas, y minimizar la necesidad de terceros intermediarios fiables*”[16]. A partir de esta definición podemos decir que un contrato inteligente no es en realidad un contrato propiamente dicho, sino que se los percibe como una serie de algoritmos y tareas programadas y almacenadas que pueden utilizarse para cumplir objetivos precisos y

funcionalidades automatizadas, y sobre todas las cosas, no son controlados por ninguna de las partes involucradas.

La figura 6 simboliza un diagrama de flujo de tareas a realizar según se avance sobre algún tipo de proceso (como se explicó en el ejemplo de las licitaciones públicas). Este diagrama de flujo se convierte en un programa almacenado en la misma blockchain como si de una transacción más se tratara, pero a diferencia del resto, son transacciones activas, esto quiere decir que son aplicaciones corriendo dentro de la cadena. Si alguna de las condiciones determinadas en el proceso almacenado se cumple, se ejecutarán distintos eventos aceptados por las partes interesadas.

Esto quiere decir que podemos hacer uso de estas herramientas para controlar procesos mucho más grandes, como concursos de licitación pública, sin la necesidad de tener que controlar las condiciones para la resolución de un problema o para la toma de una decisión crítica. Esto permite ganar tiempos en momentos donde se deberían realizar trámites burocráticos, envíos de documentos, sellado de las ofertas, firmar pliegos, etc.

Los smart contracts se apoyan en la misma red para ser resguardados, como ya se ha mencionado, esto nos da la posibilidad de saber que esas tareas son inalterables y no necesitan de la intervención de un desarrollador o soporte técnico, ya que es posible programar un flujo de tareas preestablecido entre las partes interesadas, apoyado en todas las garantías de confianza y transparencia que nos da una red de cadena de bloques [8]. De la misma forma, la información y antecedentes que se utilizan para la ejecución de los contratos, también se encuentra almacenada y validada dentro de la red, por lo tanto, es correcto suponer que el resultado de la tarea que se va a realizar o del problema que se va a resolver esta dentro de los parámetros esperados y es una consecuencia aceptada por las partes intervinientes.

7.1 Utilización de los Smart Contracts para el desarrollo de aplicaciones

Gracias a la capacidad que nos brindan los contratos inteligentes, podemos desarrollar aplicaciones que se alimenten de la misma red para completar sus funcionalidades.

Siguiendo el ejemplo de las licitaciones públicas, imaginemos una página web desarrollada por el estado argentino, en esta aplicación los oferentes tendrán la posibilidad de presentar la documentación solicitada, adjuntando documentos necesarios y completando formularios con los datos requeridos para la participación en la licitación, esta información será cargada y validada dentro de la red. Una vez cerrado el periodo de presentación de las ofertas, la misma aplicación, haciendo uso de los contratos inteligentes, podría resolver automáticamente quienes fueron los mejores postores y se encuentran en condiciones de seguir en el concurso. Esta tarea puede ser aplicada la cantidad de veces que sea necesaria, incluso esto también puede ser parte del smart contract, y así hasta conseguir un resultado final y filtrado de oferentes válidos. De la misma forma se podría evaluar cada oferta sabiendo que la documentación presentada es completamente fiable aunque no esté resguardada dentro de la red [8].

El resultado final de este proceso quedaría, al igual que todo, resguardado en la red, con las ofertas presentadas siendo de acceso público y los ganadores publicados en la web inicial para ser consultados por cualquier ciudadano. Si se logra llegar a este punto, podríamos dejar de pensar a las blockchain como un simple registro distribuido de información y comenzar a pensar en procesos más complejos que se resguarden en la misma red y se alimentan también de ella.

7.2 Utilización de los Smart Contracts para la implementación de propiedades inteligentes

Una propiedad inteligente (o en inglés Smart Property) es un derecho de propiedad controlado utilizando las facultades de los Smart Contracts. Es la representación digital de los derechos de propiedad de un activo a través de su entrada en una blockchain, usando la clave privada del propietario. El propietario del activo puede probar a otros individuos la propiedad con la correspondiente clave pública. Cuando el propietario decide vender o transferir el activo, el contrato inteligente completa el proceso entregando la clave privada del propietario al comprador. Algunos tipos de propiedades como marcas, derechos de autor y patentes pueden ser propiedades inteligentes ya que pueden ser fácilmente codificados y procesados como documentos digitales. Sin embargo, con los activos físicos, es donde los propietarios están más expuestos a fraudes. Si se quisiera registrar en la cadena de bloques algo físico, como un automóvil,

necesitamos asignarle una etiqueta única o chip. Si la información contenida en la etiqueta o chip puede ser alterada, o puede ser arrancada, la propiedad inteligente puede no estar garantizada, esto llevaría a que los contratos no validen esa propiedad.

8. Organizaciones Autónomas Descentralizadas (DAOs)

El siguiente instrumento a tratar, puede entenderse como una de las herramientas a desarrollarse utilizando la máxima capacidad de la BlockChains, los SmartContracts y los protocolos de consenso, para brindar transparencia, inmutabilidad, autonomía y seguridad a la misma. Se trata de una nueva estructura organizativa basada plenamente en el poder de las tecnologías explicada en este trabajo.

Las siglas DAO provienen del inglés Decentralized Autonomous Organization, que quiere decir Organización Autónoma Descentralizada. Esto puede interpretarse como un tipo de organización controlada completamente por algoritmos informáticos. Este nuevo concepto fue descrito inicialmente por Werner Dilger, profesor de informática alemán, muy reconocido en el ambiente. Fue él quien en 1997 desarrolló su trabajo “Organización Autónoma Descentralizada del hogar inteligente según el principio del sistema inmune”. En esta investigación, Dilger definía las bases de las DAO como sistema autónomo y autosustentable, un trabajo muy adelantado para su tiempo, pero que era impracticable para la época. Esta tarea puede realizarse solamente con la ayuda de las blockchains.[15]

Sin embargo, en el año 2013 el concepto DAO volvió a escucharse, En esta ocasión, Daniel Larimer, fundador de BitShares y Steem, hablaba de ellas como Compañías Autónomas Descentralizadas (DAC), otra de las formas comunes de llamar a este tipo de organizaciones.[15]

En 2015, Vitalik Buterin, programador y escritor ruso, volvió a poner a la luz el concepto, gracias a la puesta en funcionamiento de Ethereum, que permitía crear códigos avanzados transparentes e inalterables, características que facilitaban en gran medida la creación y la interacción con las DAOs.[15]

Como el nombre lo menciona, las Empresas Autónomas Descentralizadas no se vinculan a ninguna ley o regulación en particular debido, justamente, a su naturaleza descentralizada. Para obtener este beneficio, es necesario utilizar blockchain y smart contracts almacenados, que permitan a la entidad ser regulada por la herramienta misma. Los contratos inteligentes utilizados para realizar esta tarea, pueden ser tan simples o complejos como se haya decidido programar, pero, sin distinción alguna, los mismos quedarán transparentes e inmutables en el momento que sean publicados en la cadena de bloques. Esto hará que todo el mundo pueda revisar su funcionamiento y las reglas que se han programado dentro, teniendo la seguridad de que no podrán ser modificados en un futuro sin que el resto de la red se entere.

Al ser coordinadas por piezas de código, es decir, por algoritmos puramente informáticos, se hace efectiva la colaboración de una manera poco conocida hasta el momento, pudiendo gestionar la toma de decisiones entre personas, sistemas informáticos y otros smart contracts (los que a su vez, pueden ser conducidos por otros humanos, máquinas y contratos inteligentes).

Es por estas razones, que las DAOs son un poderoso descubrimiento en la implementación de estas nuevas tecnologías. Esto gracias a que permiten la constitución de organizaciones autónomas, mucho más transparentes y sin un orden jerárquico preestablecido. Esta última es una característica fundamental para el funcionamiento de este tipo de entidades, ya que permite establecer nuevos modelos en las formas en que cooperan los integrantes de dicha institución.

8.1 Ventajas y desventajas de las DAOs

Como todo tipo de tecnología, una DAO puede ofrecer a sus participantes diferentes ventajas y desventajas. Sobre todo sabiendo que las mismas dependen de la programación o normativa de la DAO. Sin embargo, podemos generalizar algunos puntos que vale la pena conocer.

8.1.1 Ventajas

- Es posible crear organizaciones con un nivel de jerarquización mínimo y hasta casi nulo. Todos los que forman parte de la organización pueden ser creadores de ideas y votar por las mismas u otras presentadas por miembros de la red, esta es la principal causa de la descentralización de dicha entidad.

- Las DAOs heredan la capacidad de registrar todas las acciones realizadas en la red y hacer público dicho registro, por ello, su nivel de transparencia es muy alto.
- Es posible ofrecer servicios que, de otras formas, no estarían disponibles para muchas personas. Esto es posible debido a la naturaleza descentralizada de las blockchains.
- Facilitan de muchas formas la creación de organizaciones, la creación de una institución de estas características solo debe programar sus bases en una blockchain y desde ese momento comienza a constituirse como tal. Con ello se ahorra dinero, tiempo y tareas burocráticas, algo necesario para el registro tradicional de organizaciones o empresas.

8.1.2 Desventajas

- Si bien, la constitución de una DAO es sencilla, ya que la misma comienza a funcionar luego de su programación, realizar esta automatización de tareas, no es una tarea simple y se tienen que tener muchos factores en cuenta antes de plasmarlos en el código.
- Existen problemas para que las DAO puedan cumplir con las regulaciones en muchos países. Esto se debe principalmente a que la tecnología blockchain y las criptomonedas aún no tienen una clara regulación.

9 Aplicación de la Cadena de Bloques en la Agencia de Recaudación Tributaria de Río Negro

La provincia de Río Negro cuenta con una gran base de datos de contribuyentes gestionada por la Agencia de Recaudación Tributaria (ARTRN). Esta base de datos se encuentra sincronizada con los repositorios nacionales pertenecientes a la Administración Federal de Ingresos Públicos (AFIP). Sin embargo, los mecanismos utilizados para compartir información entre ambos organismos suelen ser poco eficientes y la automatización de los procesos es muy tediosa. Para enfrentar esta situación, se pretende acceder a la BlockChain del Padrón Federal (BCPADFED) impulsada por AFIP, la Comisión Arbitral del Convenio Multilateral y las Agencias de Recaudación Provinciales.

Los objetivos de esta red nacional son: evitar la carga múltiple en distintos portales, mejorar la calidad de los datos de los contribuyentes y lograr coherencia en la inscripción de impuestos (IVA, IIBB, Convenios Multilaterales, Regímenes Simplificados, etc.). Para llevar a cabo estos objetivos, la red propone dos tipos de aplicaciones de integración que se pueden utilizar para acceder a la información: HLF-PROXY y Block Consumer [10].

9.1 HLF-PROXY

La HLF-PROXY, permite interactuar con los Smart Contract y ofrece una API REST⁶ al cliente, que se puede utilizar para actualizar datos en su base o para leer un dato puntual de la blockchain. En vez de aprender cómo consultar la cadena de bloques o cómo enviar una transacción a la red, se utiliza este medio para conocer el estado actual de un contribuyente [17].

⁶ **API REST:** Una API de REST, o API de RESTful, es una interfaz de programación de aplicaciones (API o API web) que se ajusta a los límites de la arquitectura REST y permite la interacción con los servicios web de REST ful. <https://www.redhat.com/es/topics/api/what-is-a-rest-api>

Una organización autorizada a acceder a la Blockchain del Padrón Federal que no se constituye como nodo peer de la red, puede utilizar la API HLF-Proxy para invocar vía servicios REST las funciones publicadas en la interfaz de la herramienta. HLF-Proxy se conecta directamente a los nodos de la red utilizando internet. Sin embargo, una organización que está constituida como nodo sellador de la Blockchain, puede utilizar la aplicación conectándose a su propio contexto para tener mejor tiempo de respuesta en consultas.[17] El esquema de este funcionamiento se puede ver en el siguiente gráfico.

HLF-Proxy accediendo a nodos locales y remotos

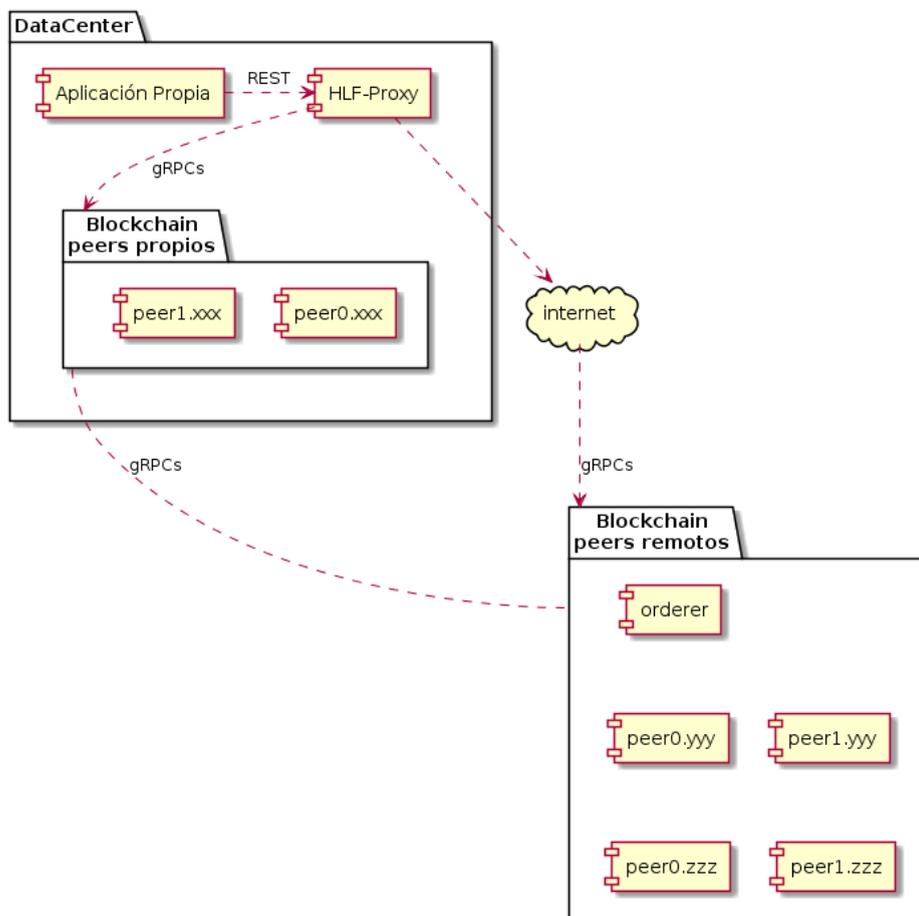


Gráfico obtenido de la documentación oficial de Blockchain Padrón Federal.
<https://github.com/padfed/padfed-doc/blob/master/applications/hlf-proxy/README.md>

En cuanto a los métodos de acceso y utilización de la API RestFul, nos centraremos en la utilización del punto de acceso denominado QUERY. Los mecanismos de uso se encuentran detallados en el siguiente cuadro extraído de la documentación oficial.

Endpoints	Método	Acción
<code>/hlfproxy/api/v1/query</code> <code>/[{channel}]/{cc}[/[{peer}]]</code> <code>[?{params}]</code>	POST	Invoca funciones del chaincode {cc} en el channel {channel} en el PEER {peer} o en los PEERs configurados en el <code>xxx.client.yaml</code> con <code>chaincodeQuery: true</code> . No envía la tx a los ORDERERS.

Cuadro obtenido de la documentación oficial de BlockChain Padrón Federal. <https://github.com/padfed/padfed-doc/blob/master/applications/hlf-proxy/README.md>

El método QUERY recibe como parámetro un objeto **JSON**⁷ que contiene la función que se pretende acceder y los datos del contribuyente a consultar. Es necesario aclarar, que la BCPadFed utiliza la CUIT como clave primaria, por lo tanto, todas las funciones de invocación de la HLF-Proxy, deberán contener este dato para interpretar y distinguir la información que se quiere obtener.

Un ejemplo de los datos a enviar como solicitud del método, es el siguiente:

```
{"function": "getPersona", "args": [20104249729]}
```

9.2 BLOCK-CONSUMER

Por otro lado, se puede acceder a la red como Block-Consumer, esta tarea es bastante más laboriosa, y se debe contar con una infraestructura más grande. Al momento de decidir qué aplicación utilizar hay que tener esto en cuenta, ya que la block-consumer no solo guarda datos de los contribuyentes por cada movimiento que se realiza, sino que también guarda información de todas las transacciones realizadas sobre esos contribuyentes. Es necesario aclarar que esta información se guarda en una base de datos relacional de entorno local, por lo tanto, hay que contar con los recursos necesarios para gestionar esa tecnología.

⁷ **JSON:** (JavaScript Object Notation - Notación de Objetos de JavaScript) es un formato ligero de intercambio de datos. Leerlo y escribirlo es simple para humanos, mientras que para las máquinas es simple interpretarlo y generarlo.

Al igual que con la api rest HLF-PROXY, una organización que corre nodos de la Blockchain del Padrón Federal, puede conectar el block-consumer a sus peers locales para lograr mejor performance de procesamiento.[17] El esquema de esta arquitectura se encuentra plasmado en el siguiente gráfico obtenido de la documentación oficial de BlockChain Padrón Federal:

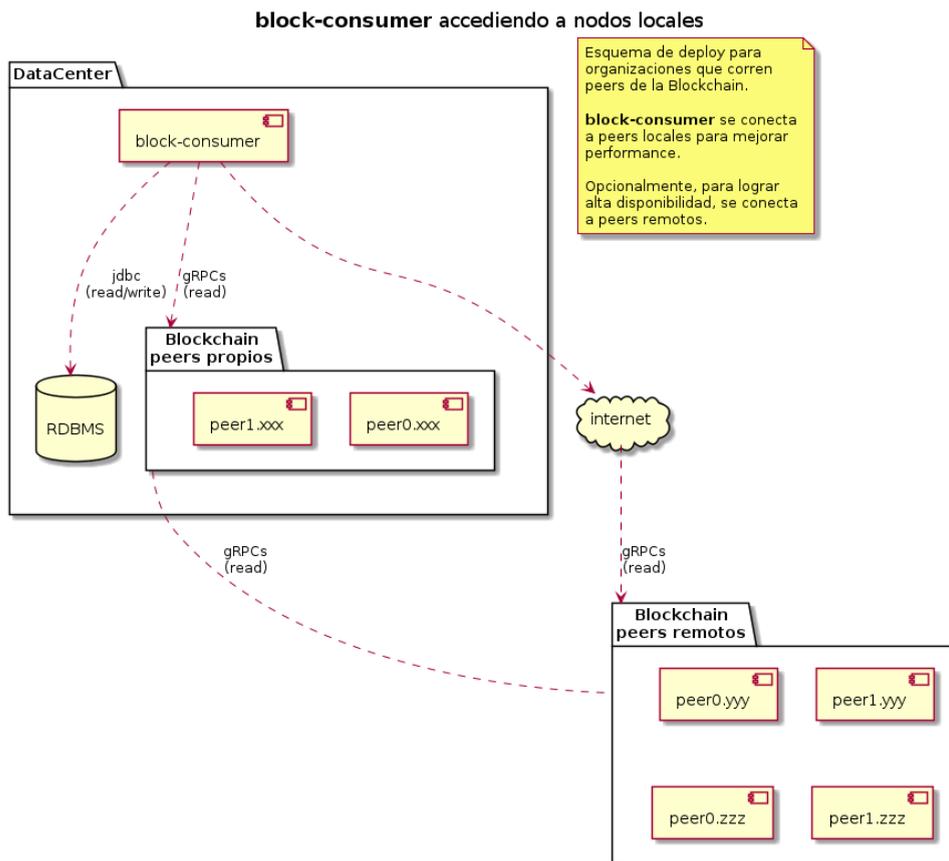


Gráfico obtenido de la documentación oficial de BlockChain Padrón Federal.
<https://github.com/padfed/padfed-doc/blob/master/applications/hlf-proxy/README.md>

9.3 Herramientas y Entornos de Trabajo

En la industria del software hay disponible una gran variedad de herramientas para implementar un proyecto web. Dada esta situación, se debió seleccionar la metodología de trabajo, el lenguaje de programación a utilizar, el tipo de base de datos para el servidor, editores de código, bibliotecas externas y el ambiente de desarrollo.

9.4 Sistema de control de versiones de código fuente

GitLab es un servicio web de control de versiones y desarrollo de software colaborativo basado en Git. Además de gestor de repositorios, el servicio ofrece también alojamiento de wikis y un sistema de seguimiento de errores, todo ello publicado bajo una licencia de código abierto. Contiene todo lo necesario para llevar a cabo el proyecto bajo el modelo iterativo incremental, desde la creación de la idea a la producción de la misma.

9.5 Herramientas de Desarrollo

Teniendo en cuenta las herramientas de trabajo utilizadas en la Agencia de Recaudación Tributaria, se decidió trabajar sobre una base de datos ORACLE versión 11g R2. Este sistema de base de datos cuenta con la capacidad suficiente para alojar la estructura de una BlockChain compleja, como lo es la BCPadFed. Además, cuenta con herramientas propias de desarrollo para realizar potentes aplicaciones y albergar la lógica de negocio necesaria para llevar a cabo los procesos planteados.

Para llevar a cabo el proyecto, fue necesario separarlo en 3 capas de desarrollo de software: Capa de Lógica de Negocios, capa de aplicación (BackEnd) y aplicación FrontEnd o Servicio RESTful según corresponda, esto debido a que los datos obtenidos se van a consultar desde una interfaz web, por un lado, y se van a ejecutar los servicios http, por el otro.

La capa de lógica de negocios se encuentra ubicada en la base de datos, Oracle nos permite trabajar con los datos de manera dinámica y eficaz, por lo que se convierte en una herramienta potente al momento de validar y llevar el hilo de tareas de manera coordinada.

Backend es el código que se escribe del lado del servidor, el cual contiene la arquitectura propuesta y que se conecta con la capa de lógica de negocios. Además, la capa de aplicación expone servicio REST para que el proyecto frontend (proyecto web) pueda interactuar e intercambiar datos. Para la gestión de todos los componentes del proyecto, módulos y manejo de dependencias se utilizó Composer. El lenguaje utilizado es PHP.

La aplicación web (FrontEnd) es la parte de un sitio web que interactúa con los usuarios, por eso decimos que está del lado del cliente. Para este caso, se desarrolló con el framework Laravel y la herramienta de templates Blade Template, ambas tecnologías cuentan con licencia de código abierto, por lo que el soporte técnico es amplio y muy conocido en la comunidad tecnológica.

9.6 Entorno de Trabajo (FrameWork)

Para el desarrollo de la capa de aplicación y la interfaz web se decidió utilizar el framework Laravel. Esta herramienta se utiliza en la ARTRN para los aplicativos de consulta de deudas y boletas publicadas en la web del organismo, por lo tanto seguir trabajando sobre ella, es la mejor opción. Laravel es un framework de código abierto para desarrollar aplicaciones y servicios web con PHP 5, PHP 7 y PHP 8. Su filosofía es desarrollar código PHP de forma elegante y simple. Fue creado en 2011 y tiene una gran influencia de frameworks como Ruby on Rails, Sinatra y ASP.NET MVC.2. Además, cuenta con los mecanismos necesarios para la publicación de servicios RESTful.

9.7 Metodología de Trabajo

El Modelo Iterativo/Incremental fue el seleccionado para la implementación del proyecto, este modelo fue propuesto por Harlan Mills en el año 1980. Surgió el enfoque incremental de desarrollo como una forma de reducir la repetición del trabajo en el proceso de implementación y dar oportunidad de retrasar la toma de decisiones en los requisitos hasta adquirir experiencia con el sistema. El Modelo Incremental combina elementos del Modelo Lineal Secuencial con la filosofía interactiva de Construcción de Prototipos. Como se muestra en la Figura 7, el modelo incremental aplica secuencias lineales de forma escalonada mientras progresa el tiempo en el calendario. Cada secuencia lineal produce un incremento del software. El primer incremento generalmente es un producto esencial denominado núcleo.

En una visión genérica, el proceso se divide en 4 partes:

- Análisis
- Diseño
- Implementación
- Prueba

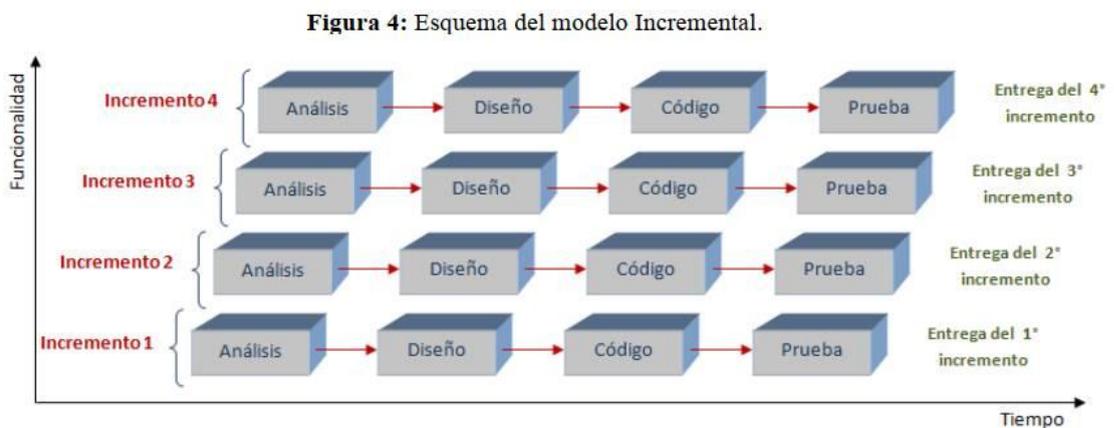


Figura 7. Representación gráfica del modelo Iterativo/incremental.

10. Módulo de Alta de Contribuyentes

La ARTRN cuenta con los recursos descritos anteriormente para poder llevar a cabo las tareas necesarias, por lo que se propone hacer uso de los mismos para identificar cuáles son aquellos contribuyentes nuevos que se dan de alta a diario en AFIP y cuyo domicilio fiscal pertenece a la provincia de Río Negro y aquellos otros que cambian su número de cuit, pero no el resto de sus datos como contribuyente en la provincia. De esta manera se podrán actualizar o dar de alta aquellas personas físicas o jurídicas en los registros provinciales sin la necesidad de que ellas mismas tengan que presentarse en las oficinas de la Agencia. Esta tarea no solo es beneficiosa para los contribuyentes, sino que también mejora la velocidad de respuesta de las oficinas del organismo, que en estos momentos reciben pedidos de alta de personas mucho antes que AFIP informe sobre los cambios y nuevos registros.

Hasta el momento, un contribuyente que se inscribe en AFIP en IIBB, también debe inscribirse en los impuestos locales de la Provincia de Río Negro. La ARTRN utiliza en su oficina virtual el método de acceso por clave fiscal de AFIP, teniendo de esta forma integrado el acceso seguro a los sistemas con dicho organismo nacional. El circuito que el contribuyente debía seguir hasta el momento era el siguiente:

1. Inscribirse y obtener su clave fiscal, ingresando al portal de AFIP
2. Solicitar el alta como contribuyente en la ARTRN: esta tarea sólo podía realizarse de manera presencial en alguna de las oficinas de la Agencia, dado que el requisito para acceder por la Oficina Virtual es ser contribuyente de la provincia, por lo cual no era posible solicitar el alta por ese medio.

En este contexto se hacía evidente que constituye una mejora sustancial para la atención al contribuyente posibilitar el alta automática en la base de datos de la Agencia desde la información de la BCPadFed, dado que permitirá el ingreso por la Oficina Virtual para solicitar el alta en el impuesto correspondiente, sin necesidad de trámite presencial alguno.

Mucho antes de la implementación de la Blockchain del Padrón Federal, la Agencia desarrolló interfaces de descarga por lotes que se encargan de actualizar de forma masiva la información de los contribuyentes, esto se realizó utilizando los servicios ofrecidos por AFIP. Dicha información es almacenada en un esquema de la base de datos provincial, estos registros alimentan todas las necesidades de sincronización y consulta de información compartida por ambos organismos. Este procesamiento manipula una gran cantidad de datos por lo que se programa mensualmente.

Posterior y paulatinamente, la integración entre ambos organismos mejoró la sincronización de información para algunos tributos, con la adhesión al Padrón Federal para contribuyentes de Convenio Multilateral y al Monotributo Unificado para contribuyentes del Régimen Simplificado. Esto requirió de la implementación de interfaces batch que realizan la coordinación de novedades de forma diaria.

En este punto, los contribuyentes del impuesto a los IIBB locales aún debían continuar con el circuito detallado anteriormente, dado que la Provincia aún no integra el Padrón Federal para dichos contribuyentes.

10.1 Primera etapa - Alta de Contribuyente Físico

Para el desarrollo del módulo de alta de contribuyentes provenientes de la red de bloques, se tuvieron en cuenta las herramientas disponibles en el organismo. Debido a la construcción de aplicaciones previas de uso interno desarrolladas en lenguaje de programación PHP, la decisión fue seguir esta línea de trabajo, haciendo uso, además, de los mecanismos de seguridad ya implementados en estas aplicaciones.

10.1.1 Análisis y Diseño

Para poder dar de alta un contribuyente físico en la base de datos, es necesario que se cumplan ciertos requisitos y validaciones de datos previas:

- La cuit de la persona que se encuentra en la blockchain, no debe estar dada de alta en la base de datos provincial.

- Los contribuyentes que no cuenten con domicilio fiscal cargado en AFIP, no son dados de alta, dado que la provincia del domicilio fiscal es el dato que se utiliza para considerar que es contribuyente de la provincia de Río Negro.
- Los contribuyentes cuyo código de nomenclador no existe o no está cargado en las tablas de localidades de la Agencia, no son dados de alta.

En la figura 8 podemos ver el diagrama de secuencia que refleja el proceso de alta de nuevos contribuyentes físicos, en la Agencia de Recaudación Tributaria.

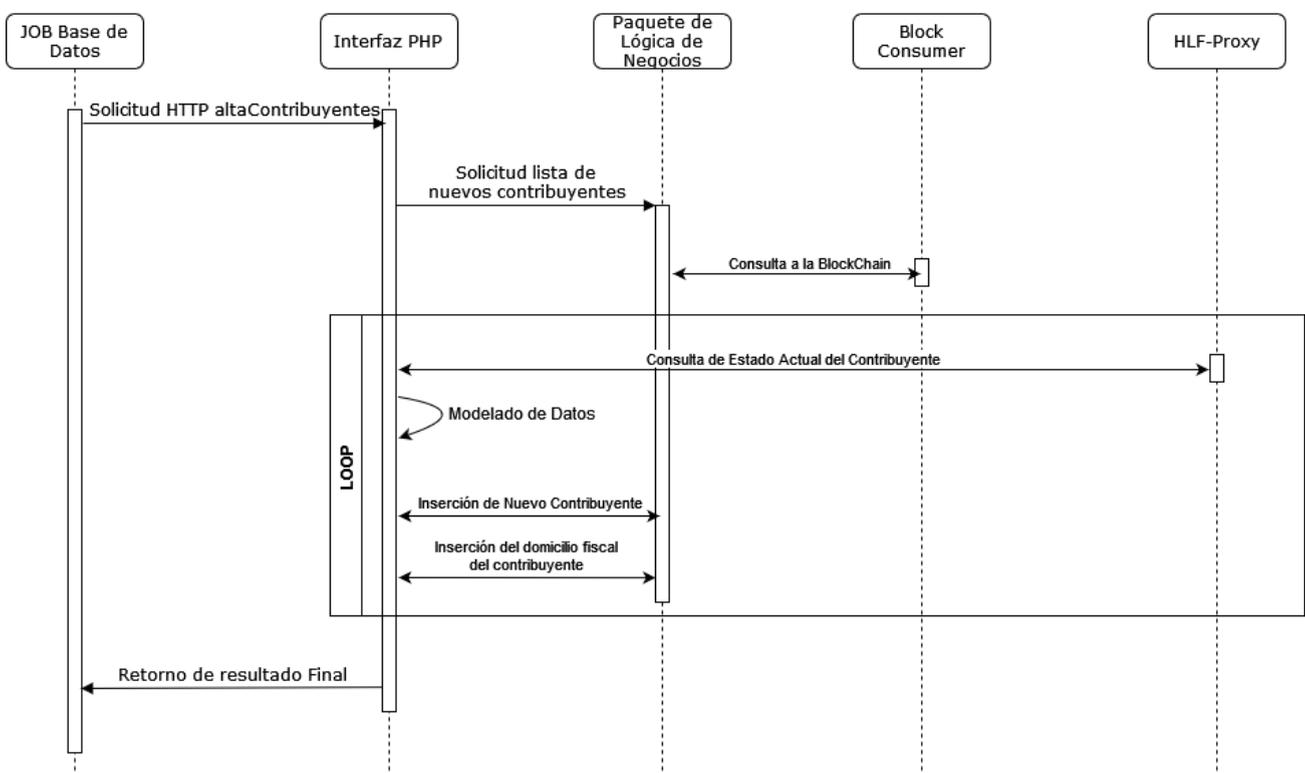


Figura 8. Diagrama de secuencia del alta de contribuyentes físicos.

10.1.2 Implementación

El proceso comienza a partir de un JOB almacenado en la base de datos ORACLE, esta tarea realiza una solicitud o **método HTTP**⁸ a la interfaz PHP, que cuenta con los módulos necesarios para realizar el trabajo

⁸ **Método HTTP:** indica la acción que se desea realizar para un recurso determinado. Utilizado mayormente en las aplicaciones de tipo API Restful.

requerido. El módulo denominado **altaContribuyentes**, recibe como parámetros la FECHA para realizar una inserción masiva o una CUIT particular para realizar una inserción particular, obtenidos estos datos, se solicita un set de datos provenientes de la capa de lógica de negocios, esta capa accede directamente a la *Block Consumer*, y retorna mediante consultas SQL la información de aquellos contribuyentes que no se encuentran cargados en la base de datos provincial.

Una vez que la *Interfaz PHP* cuenta con la lista de contribuyentes que se darán de alta en la base de la Agencia de Recaudación Tributaria, comienza un proceso iterativo sobre la misma. Cabe destacar que el conjunto de datos obtenidos cuenta con contribuyentes físicos y personas jurídicas, y en esta primera etapa solo se tendrán en cuenta aquellos contribuyentes físicos que cumplan las condiciones para la inserción en la base, por esto, el primer paso del proceso de iteración es verificar este tipo de referencias.

En segundo lugar, se consulta la API Rest, *HLF-Proxy*, para conocer el estado actual del contribuyente y poder modelar la información antes de ser enviada a la capa de lógica de negocios para su próxima inserción en la base de datos.

El modelado de datos verifica algunas de las condiciones mencionadas anteriormente, en este caso que la persona cuente con un domicilio fiscal relacionado.

```

/*
 * SE OBTIENE EL DOMICILIO CON CLAVE 1.1.1 (DOMICILIO FISCAL)
 * SOLO SE CARGA EL REGISTRO SI EL COD DE NOMENCLADOR ESTA SETEADO Y PERTENECE A LA PROVINCIA DE RIO NEGRO
 */
foreach ($ccResponse->domicilios as $dom) {

    if ($dom->orden == 1 && $dom->org == 1 && $dom->tipo == 1) {

        if (isset($dom->nomenclador) && $dom->provincia == $this->codProvincia) {

            $fecha = date($dom->ds);
            $nuevafecha = strtotime( time: "now", strtotime($fecha));
            $fechadesde = date( format: 'd/m/Y', $nuevafecha);

            //echo $fechadesde; exit();

            $domicilioContribuyente = new DomicilioContribuyente();
            $domicilioContribuyente->setTipoafip($dom->tipo);
            $domicilioContribuyente->setIdnomenclaafip($dom->nomenclador);
            $domicilioContribuyente->setCalle( calle: isset($dom->calle) ? $dom->calle : "S/D");
            $domicilioContribuyente->setNumero( numero: (isset($dom->numero)) ? $dom->numero : null);

            if (isset($dom->adicional)) {
                $domicilioContribuyente->setTipodatoadic($dom->adicional->tipo);
                $domicilioContribuyente->setDatoadic($dom->adicional->dato);
            }

            $domicilioContribuyente->setPostafip($dom->cp);
            $domicilioContribuyente->setFechadesde($fechadesde);

            $contribuyente->setDomicilios($domicilioContribuyente);

            break;
        }
    }
}
}

```

Figura 9. Extracto del código perteneciente a la interfaz PHP. Verifica que el contribuyente cuente con un domicilio fiscal.

La figura 9 es un extracto del código implementado en la interfaz PHP, esta recopilación comprueba que el domicilio del contribuyente sea de la organización 1, de tipo 1 y con número de orden 1 que, según la documentación expuesta para la HLF-Proxy (Figura 10), corresponde al domicilio fiscal.

persona.domicilios					
key	per:{id}#dom:{org}#{tipo}#{orden}				
item	desc	tipo	min	max	req
org		#organización			x
tipo	Una persona puede tener un único domicilio tipo 1 (Fiscal para AFIP), un único domicilio tipo 2 (Real para AFIP) y 0 a N domicilios tipo 3 . Los domicilios migrados por las jurisdicciones (org > 1) siempre tienen tipo 3 .	integer	1	3	x
orden	Número secuencial comenzando desde 1 para cada org y tipo	integer	1	9999	x

Figura 10. Fragmento de la documentación presentada para la HLF Proxy.

Solo si la persona cuenta con un domicilio fiscal asociado, es que se procederá con la carga del contribuyente en la base de datos.

Luego del modelado de datos necesario para la creación de un nuevo contribuyente, se envía la información nuevamente a la capa de lógica de negocios. En primera instancia, se procede a la persistencia del contribuyente en la base, los datos requeridos son los siguientes:

```
function alta_contribuyente_fisico(p_n_cuit in contribuyentes.n_cuit%type,
                                  p_f_actualiz in varchar2,
                                  p_d_denominacion in contribuyentes.d_denominacion%type,
                                  p_m_persona in contribuyentes.m_persona%type,
                                  p_c_tipo_documento in contribuyentes.c_tipo_documento%type,
                                  p_n_documento in varchar2,
                                  p_c_sexo in personas_fisicas.c_sexo%type,
                                  p_f_nacimiento in varchar2,
                                  p_impuesto in number,
                                  p_impuesto_estado in varchar2,
                                  p_email in varchar2,
                                  p_id_nomencla_afip in number) return t_cursor;
```

Aquí se verifica que el código del nomenclador provisto por AFIP no se encuentre definido con valores nulos o no válidos y se mapea el tipo de documento para que coincida con las referencias que se manejan dentro de la base de datos provincial.

Una vez dado de alta correctamente el nuevo contribuyente, se continúa con la carga de su domicilio fiscal, dato requerido para poder gestionar correctamente todos los movimientos, transacciones y beneficios con los que va a contar la persona. Para llevar adelante esta tarea, se necesitan los siguientes datos:

```

function alta_domicilio_fiscal(
    p_tipo_afip in number,
    p_id_nomenclatura_afip number,
    p_d_calle domicilios.d_calle$type,
    p_n_numero domicilios.n_numero$type,
    p_tipo_datos_adic number,
    p_datos_adic varchar2,
    p_c_post_afip varchar2,
    p_id_contribuyente in varchar2,
    p_f_vig_desde in varchar2) return t_cursor is

```

Dentro de la capa de lógica de negocios, se modelan y mapean los datos para que correspondan con las tablas y referencias utilizadas en todo el sistema.

10.1.3 Pruebas y Resultados

El resultado de cada una de las iteraciones es cargado en un arreglo de datos que va a ser utilizado para mostrar el estado final del módulo de **altaContribuyentes** en la pantalla del navegador, si es ejecutado por este medio, y para enviar un mail a aquellos usuarios encargados de controlar que la información almacenada sea la correcta.

En las figuras 11 y 12, se puede ver un ejemplo de los resultados obtenidos.

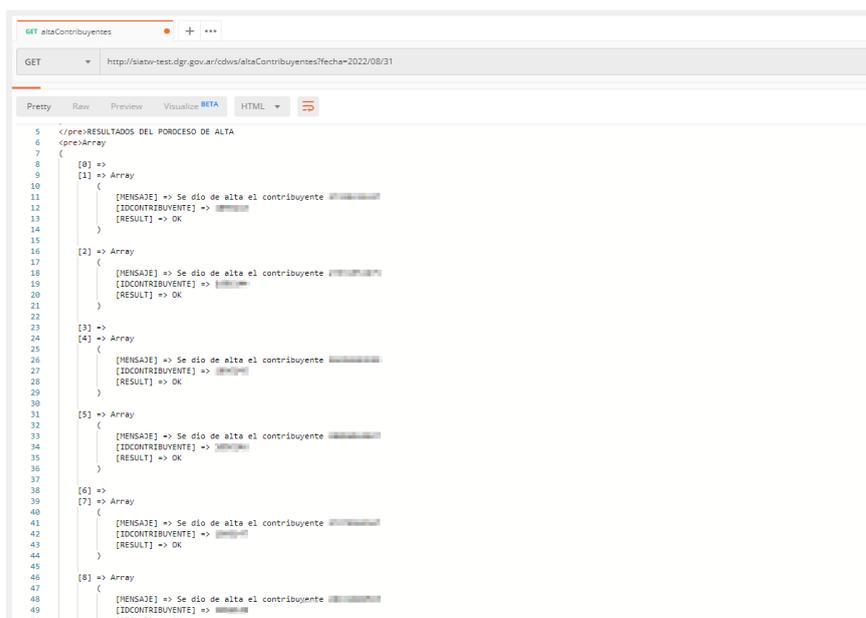


Figura 11. Resultados de la ejecución de carga realizada mediante Postman.

De Agencia de Recaudación Tributaria Río Negro [Ver perfil de usuario](#)
A Nahuel Emiliano Pichon [Ver perfil de usuario](#)
Asunto 31/08/2022 - Alta de Contribuyentes Nuevos - Ambiente de TEST

RESULTADOS DEL PROCESO DE ALTA

- Se dio de alta el contribuyente [REDACTED]

Figura 12. Resultados de la ejecución de carga enviados por mail.

10.2 Segunda etapa - Actualización y cambios de cuits

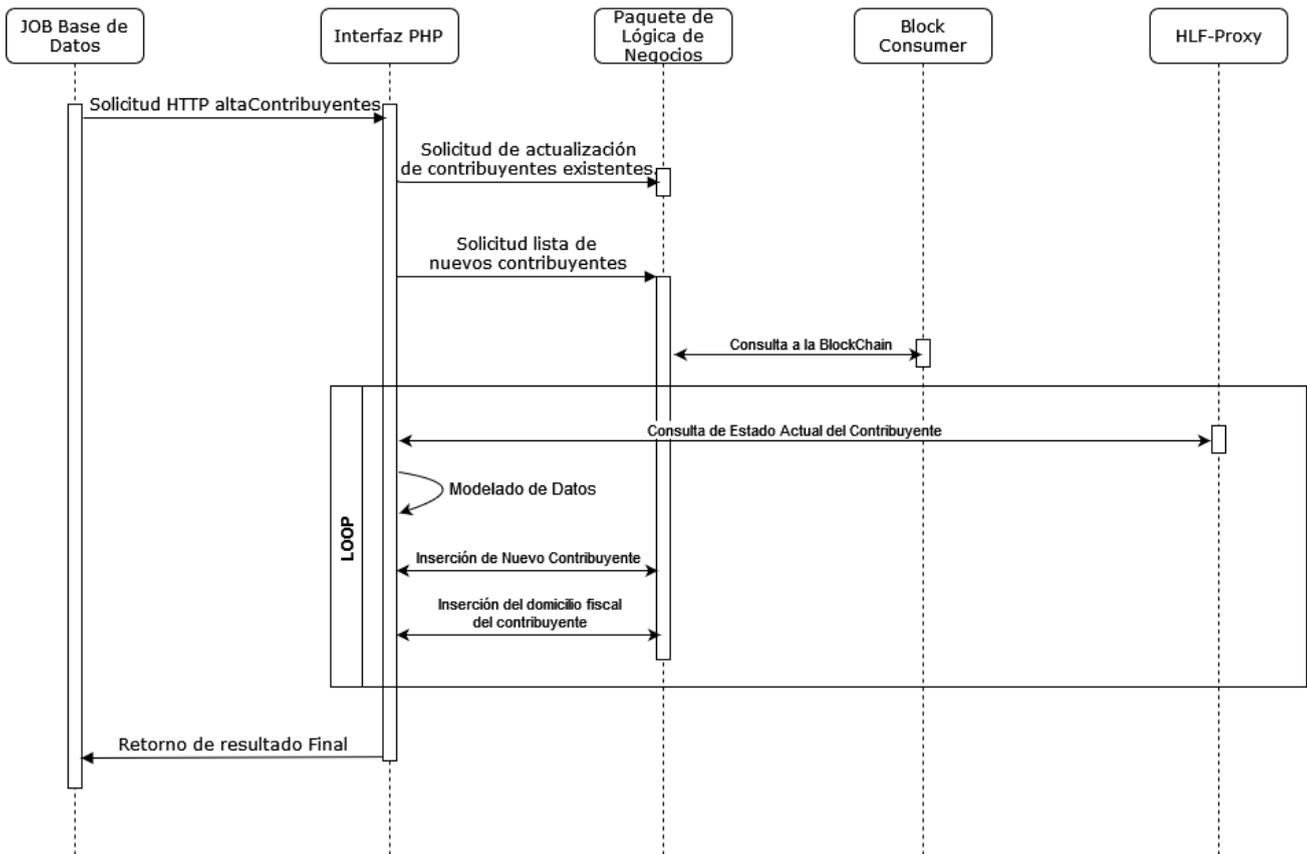
Debido a las modificaciones que suelen realizarse en los distintos organismos de Recaudación Tributaria y oficinas de AFIP, fue necesario modificar el proceso anterior para capturar aquellos cambios de cuits que suelen surgir por distintos motivos. La cuit es la clave del negocio que identifica a un contribuyente a nivel país. Si bien el modelo de base de datos utilizado por la ARTRN utiliza un **id_contribuyente** interno como clave primaria y respeta la unicidad de la cuit como clave candidata, es necesario detectar los cambios de cuit para mantener consistente la información de los contribuyentes.

10.2.1 Análisis y Diseño

Para llevar a cabo esta tarea, es necesario cumplir con los requisitos que se nombra a continuación:

- Antes de dar de alta el nuevo contribuyente se debe verificar si la cuit ingresante no tiene una cuit anterior asociada, que es la forma en que AFIP identifica los casos de cambio de cuit. Si esto sucediera, no se debe realizar el proceso de alta, sino que es necesario realizar un cambio de cuit en los registros asociados a la cuit vieja.
- Si la cuit no cuenta con otra cuit anterior asociada, se puede comenzar con el proceso de alta.

El diagrama de secuencia siguiente, es una réplica del diagrama del proceso original anexando la nueva “solicitud de actualización de contribuyentes existentes”.



10.2.2 Implementación

Las tareas de actualización y cambios de cuits se realizan exclusivamente en la capa de *Lógica de Negocios*. El procedimiento comienza obteniendo el set de datos que cumpla las condiciones mencionadas en la etapa de análisis, para ello, se consulta la *Block Consumer* ofrecida por la BCPadFed, cabe destacar, que esto se realiza todos los días, por lo tanto el resultado puede variar en la cantidad de casos, e incluso no encontrar ninguna coincidencia.

Una vez que se obtiene el resultado de la consulta, se comienza el bucle de acciones necesarias para la actualización o cambio de cuits, esta labor corresponde a procedimientos almacenados desarrollados previamente. Por esta razón, no fue preciso volver a implementar las herramientas necesarias para conseguir los objetivos de la etapa.

10.2.3 Pruebas y Resultados

El proceso original no contaba con la actualización de cuits, por lo que se modificó para evitar que se realicen altas de contribuyentes que ya están cargados en la base de datos. Al finalizar el alta de contribuyentes, se agregó al mail inicial, un apartado para informar el resultado de aquellos cambios de cuits realizados al comienzo del procedimiento.



Figura 13. Resultados de la ejecución de carga enviados por mail.

10.3 Tercera etapa - Alta de Domicilio Fiscal Electrónico

La Agencia de Recaudación Tributaria a través de la Resolución ART Nro 809/15 determina las condiciones que debe cumplir el domicilio fiscal electrónico: el mismo consta de un mail válido declarado por los contribuyentes a través de la página web de la Agencia y es de carácter obligatorio para los contribuyentes de Impuestos a los Ingresos Brutos y Agentes de Recaudación, siendo dicha obligatoriedad paulatinamente ampliada al resto de los contribuyentes. A pesar de esa obligatoriedad y de las muchas campañas realizadas por la Agencia, todavía cerca de la mitad de los contribuyentes no cuentan con un Domicilio Fiscal Electrónico (DFE) asociado en la base de datos provincial. Las ventajas de contar con un DFE son evidentes para la ARTRN, dado que le permiten emitir electrónicamente avisos, citaciones, notificaciones, intimaciones, envío de boletas para el pago de impuestos, liquidaciones administrativas, información de inicio y cierre de fiscalizaciones, resoluciones de multas y cualquier tipo de comunicación dirigida a contribuyentes y/o responsables en materia tributaria, catastral o de defensa del consumidor. Por estas razones, es que se decide agregar al alta de nuevos contribuyentes, la obtención de un mail válido informado por AFIP, de manera de comunicarse por ese medio con el contribuyente brindando la posibilidad de confirmarlo como DFE.

10.3.1 Implementación

En un principio, se utilizó la BCPadFed para actualizar la información de aquellos contribuyentes que ya se encontraban cargados en la base de datos y no poseían DFE asignado. Se creó un proceso de alta masiva consultando, uno por uno, el estado actual de las personas seleccionadas previamente, para determinar cuál era su última dirección de correo electrónico actualizada en la BC, de manera de enviar un mail al contribuyente que le permita confirmar esa dirección de correo como DFE y poder cargarla en la base de datos de la ARTRN.

Finalizado este trabajo, se agregó el módulo de carga/confirmación de domicilios fiscales electrónicos al proceso de alta del contribuyente, anticipadamente, se controla que el parámetro correspondiente al correo electrónico no se encuentre vacío:

```

----- ALTA DFE -----
-- carga el mail proveniente de afip
l_lugar := 'Insertando en rnw_transaccion, cuit: ' || p_n_cuit;
if(p_email is not null)then
.....
    pkg_contribuyentes_bc.alta_dfe(p_n_cuit, p_email);
.....
end if;

```

Al concluir todas las tareas de carga del DFE, se envía una notificación utilizando la nueva dirección de email, al contribuyente en cuestión, con el fin de que el mismo confirme el ingreso de esta información en el sistema, de esta manera el proceso queda finalizado.

```

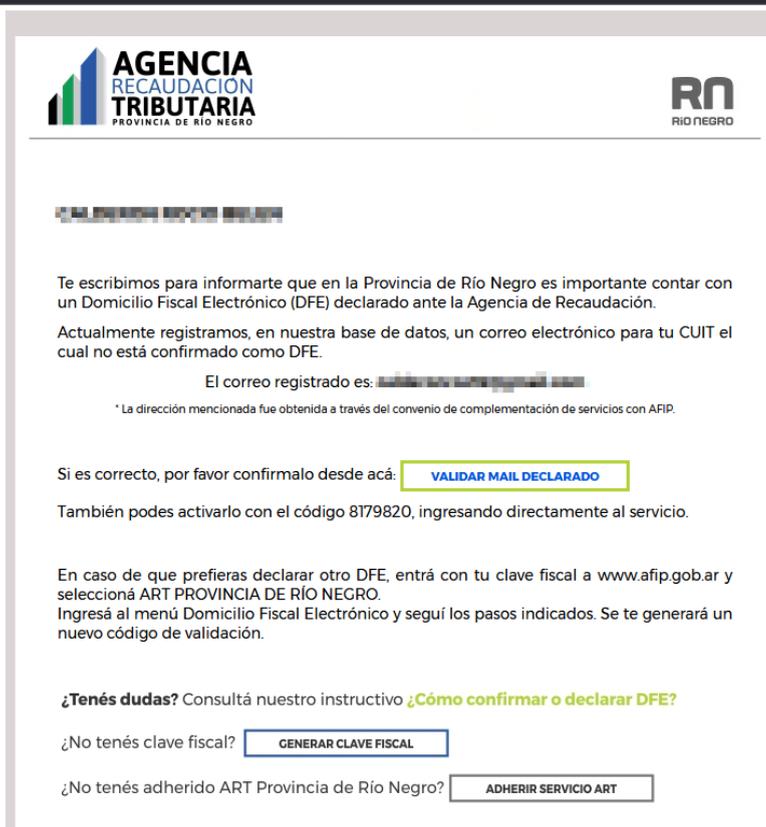
utl_mail.send('Agencia de Recaudación Tributaria Río Negro',
.....
    reg_email.d_email,
    null,
    null,
    'Confirmá tu DFE como contribuyente de Río Negro.',
    l_cuerpo_mail,
    'text/html; charset=AL32UTF8');
.....

```

10.3.2 Pruebas y Resultados

En la figura 14, puede verse el ejemplo del mail recibido por un contribuyente para solicitarle que se constituya como domicilio fiscal electrónico, el mail dado de alta durante el procedimiento de carga de contribuyentes.

De: Agencia de Recaudación Tributaria Río Negro
A: Nahuel Emiliano Pichon
Asunto: **Confirmá tu DFE como contribuyente de Río Negro.**



AGENCIA RECAUDACION TRIBUTARIA
PROVINCIA DE RÍO NEGRO

RN
RÍO NEGRO

Te escribimos para informarte que en la Provincia de Río Negro es importante contar con un Domicilio Fiscal Electrónico (DFE) declarado ante la Agencia de Recaudación.

Actualmente registramos, en nuestra base de datos, un correo electrónico para tu CUIT el cual no está confirmado como DFE.

El correo registrado es: [REDACTED]

* La dirección mencionada fue obtenida a través del convenio de complementación de servicios con AFIP.

Si es correcto, por favor confirmalo desde acá: [VALIDAR MAIL DECLARADO](#)

También puedes activarlo con el código 8179820, ingresando directamente al servicio.

En caso de que prefieras declarar otro DFE, entrá con tu clave fiscal a www.afip.gob.ar y seleccioná ART PROVINCIA DE RÍO NEGRO. Ingresá al menú Domicilio Fiscal Electrónico y seguí los pasos indicados. Se te generará un nuevo código de validación.

¿Tenés dudas? Consultá nuestro instructivo [¿Cómo confirmar o declarar DFE?](#)

¿No tenés clave fiscal? [GENERAR CLAVE FISCAL](#)

¿No tenés adherido ART Provincia de Río Negro? [ADHERIR SERVICIO ART](#)

Figura 14. Mail recibido por el contribuyente para constituir su DFE en el proceso de alta de contribuyentes.

10.4 Cuarta etapa - Alta de Contribuyente Persona Jurídica

Dada la naturaleza de este tipo de contribuyentes, el alta de personas jurídicas fue una tarea que se aplazó hasta el final, ya que, en principio, debían resolverse previamente las tareas ya descritas en etapas anteriores, como así también, contar con los recursos necesarios en materia de información y categorización de funciones y tipos de cargos.

10.4.1 Análisis y Diseño

Los contribuyentes personas jurídicas cuentan con un tratamiento distinto de los datos, los integrantes de una organización, sociedad anónima, empresa privada o del estado, asociación, fundación, etc. suelen cumplir distintos cargos dentro de la misma, por lo que era necesario contar con esta tipificación, ya que es importante para la Agencia de Recaudación Tributaria conocer a los integrantes de las formas jurídicas nombradas previamente. Para conseguir este objetivo fue necesario acudir a personal de AFIP y COMARB (Comisión Arbitral, Organismo de aplicación del Convenio Multilateral), de esta manera se pudieron mapear los tipos de datos de Forma Jurídica y Sub Tipo de Cargo, para que coincidan con la información persistente en la base de datos de la ARTRN.

Las figuras 15 y 16 son una muestra de los distintos tipos de formas jurídicas soportadas por la BCPadFed y la ART utilizados para el alta de los contribuyentes en cuestión.

CARGO_BC	DESCRIPCION
1	Otros cargos
2	Director Titular
3	Presidente
4	Socio
5	Representante
6	Administrador fiduciario
7	Condomino
8	Condomino Administrador
9	Sociedad Gerente (1)
10	Sociedad UTE (2)
11	Gerente Titular
12	Administrador Titular
13	Socio Comanditado
14	Socio Comanditario
15	Miembro del Consejo de Vigilancia

Figura 15. Extracto de los tipos de cargos, obtenidas de AFIP.

C_DATO	DESCRIPCION	ABREVIATURA
▶ 23	AGRUP. DE COLABORACION EMPRESARIA	AGRUP DE COLABORACION EMPR
13	ANONIMA C/PART. NACIONAL / EXTRANJERA	ANONIMA C/PART.NAC/EXT
9	ASOCIACIONES	ASOCIACIONES
6	CAPITAL E INDUSTRIA	CAPITAL E INDUSTRIA
5	COLECTIVA	COLECTIVA
18	CONDOMINIO	CONDOMINIO
11	COOPERATIVAS	COOPERATIVAS
8	DE HECHO	DE HECHO
10	ECONOMIA MIXTA	ECONOMIA MIXTA
0	EMPRESA UNIPERSONAL	UNIPERSONAL
12	EMPRESAS DEL ESTADO	EMPRESAS DEL ESTADO
4	EN COMANDITA POR ACCIONES	EN COMANDITA POR ACCIONES
7	EN COMANDITA SIMPLE	EN COMANDITA SIMPLE
16	ESTADO NACIONAL, PROVINCIAL O MUNICIPAL	ESTADO NAC.PROV.MUN
21	FIDEICOMISO POR CONTRATO	FIDEICOMISO POR CONTRATO
22	FIDEICOMISO POR HERENCIA	FIDEICOMISO POR HERENCIA
19	FUNDACION	FUNDACION
20	MUTUALES	MUTUALES
99	NO INFORMADA	NO INFORMADA
14	OTRAS	OTRAS
3	RESPONSABILIDAD LIMITADA	S.R.L.
2	SOCIEDAD ANONIMA	S.A.
26	SOCIEDAD ANONIMA UNIPERSONAL	SOCIEDAD ANONIMA UNIPERSONAL
24	SOCIEDAD COMERCIAL SIMPLE	SOC COMERCIAL SIMPLE
17	SOCIEDAD CONYUGAL	SOCIEDAD CONYUGAL
25	SOCIEDAD POR ACCIONES SIMPLIFICADA	SAS
15	UNION TRANSITORIA DE EMPRESAS	UTE

Figura 16. Información de las distintas Formas Jurídicas, obtenidas de AFIP.

10.4.2 Implementación

Para llevar a cabo el alta de contribuyentes personas jurídicas, fue necesario modificar estructuralmente la *Interfaz PHP* desarrollada y utilizada en las etapas anteriores.

Se decidió utilizar el patrón de diseño Factory Method para llevar a cabo las tareas de modelado de datos. Este patrón se incluye dentro de los 23 patrones de diseño publicados en 1994 en el libro *“Patrones de diseño: elementos de software orientado a objetos reutilizables”*, también conocidos como Gang of Four o **patrones de diseño GoF**⁹.

⁹ **Patrones de diseño GoF:** término que se refiere a los cuatro autores del libro: Erich Gamma, John Vlissides, Ralph Johnson y Richard Helm.

El cambio supuso una reingeniería de código y un cambio de estructura en la primera capa, que impactó positivamente en la forma de tratar la información correspondiente a un contribuyente físico o persona jurídica. Esta reestructuración mejoró notablemente el orden de la *Interfaz PHP* y permitió utilizar un patrón arquitectónico de capas dentro de la misma.

La primera capa está compuesta por el controlador y receptor de las solicitudes HTTP, este mismo recibió el nombre de **ContribuyentesBcController**, aquí dentro se encuentra alojado el método *altaContribuyentes*, nombrado en la etapa uno.

La segunda capa la componen las clases generadas a partir de la documentación propuesta por BCPadFed, y corresponden a cada uno de los objetos de datos que componen la totalidad de la información soportada por un contribuyente. Aquí tenemos: **ContribuyenteBC**, **DomiclioContribuyente**, **EmailContribuyente**, **RelacionContribuyente**, **TelefonoContribuyente**, **ImpuestoContribuyente**, **MotivoImpuesto** y clase patron *FactoryContribuyenteBC*.

También se desarrolló una tercera capa de servicios para alojar las clases encargadas de comunicarse con la API Restful provista por la Blockchain del Padrón Federal. En esta sección se encuentra la clase **ClienteBCPadronFederal**, que cuenta con los métodos necesarios para consultar el estado final completo de una persona y el historial y estado particular de las propiedades que componen a un contribuyente (categorías, impuestos, domicilios, email, teléfonos, relaciones, actividades, etc.)

Por último, se implementó una capa para identificar las clases encargadas de la conexión a la base de datos de la ARTRN. La clase **DaoContribuyenteBC** recibe los datos modelados para poder enviarlos a la capa de *Lógica de Negocios*, alojada en la base de datos ORACLE de la organización.

En la figura número 17 podemos ver como quedó estructurada la *Interfaz PHP* luego de los cambios realizados al comenzar la etapa de desarrollo del alta de contribuyentes personas jurídicas.

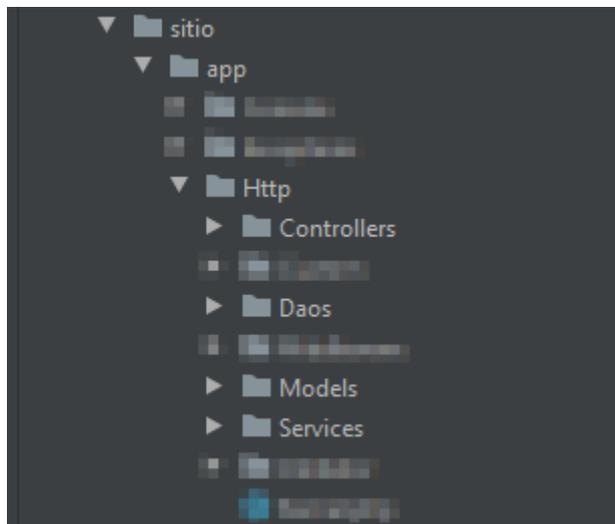


Figura 17. Estado de las capas pertenecientes a la Interfaz PHP..

El alta de contribuyentes personas jurídicas se suma al alta de contribuyentes físicos detallada en la primera etapa. Las condiciones para determinar si una persona debe ser dada de alta son similares, el contribuyente debe contar con un domicilio fiscal en la provincia y no existir en la base de datos provincial. Sin embargo, los integrantes de dicha institución pueden tener un domicilio fiscal perteneciente a otra jurisdicción, por lo que no debe ser un inconveniente cargarlo como contribuyente, para realizar esta labor, fue necesario crear el procedimiento **Alta de Integrante** alojado en la *Capa de Negocios*.

Este método hace uso de los tipos de cargos consultados a AFIP y la Comisión Arbitral, este dato es importante para la ART, ya que logra determinar la relación entre los distintos integrantes y la forma jurídica cargada, información que puede ser utilizada para cualquier tipo de comunicación dirigida a contribuyentes y/o responsables, si fuera el caso.

Para llevar adelante esta tarea, se necesitan los siguientes datos:

```
function alta_integrante(p_id_contribuyente integrantes.id_contribuyente%type,  
                        p_n_cuit integrantes.n_cuit%type,  
                        p_n_documento integrantes.n_documento%type,  
                        p_c_tipo_documento in integrantes.c_tipo_documento%type,  
                        p_d_denominacion integrantes.d_denominacion%type,  
                        p_c_sexo integrantes.c_sexo%type,  
                        p_cargo_bc afipinfo.bc_corresp_cod_cargo.cargo_bc%type,  
                        p_id_nomenclafip number,  
                        p_c_postal domicilios.c_postal%type,  
                        p_c_provincia number,  
                        p_descrip_afip varchar2,  
                        p_d_calle integrantes.d_calle%type,  
                        p_n_numero integrantes.n_numero%type,  
                        p_depto in out integrantes.d_depto%type,  
                        p_n_telefono integrantes.n_telefono%type  
                        )return t_cursor;
```

Los integrantes de una institución también son cargados como contribuyentes físicos de la ART, para efectuar esta tarea, se reutiliza el código implementado en la primera etapa. Finalmente, un integrante se asocia a sí mismo como contribuyente y forma parte de una persona jurídica.

10.4.3 Pruebas y Resultados

El resultado final de las altas realizadas es cargado en el mismo arreglo de datos que fue utilizado para mostrar el estado final de el alta de contribuyentes físicos, por lo tanto, el estado se muestra en la pantalla del navegador, si es ejecutado por este medio, y se envía por mail a aquellos usuarios encargados de controlar que la información almacenada sea la correcta.

A diferencia de la etapa anterior, el arreglo de datos contiene la información del contribuyente cargado, junto con el estado de los integrantes relacionados, si fuera una persona jurídica, el resultado de las cuits actualizadas, si se encontraron casos que cumplan las condiciones, y el estado final de los contribuyentes físicos, si los mismos cumplen con los requisitos necesarios.

En las figuras 18 y 19, se puede ver un ejemplo de los resultados obtenidos.

```
GET https://www.gub.uy/consulta/consultas/fecha=2022/10/24 Send Save

Pretty Raw Preview Visualize BETA

CUITS ACTUALIZADAS
Array
(
)

RESULTADOS DEL POROCESO DE ALTA
Array
(
  [0] => Array
  (
    [MENSAJE] => El contribuyente [REDACTED], cuenta con impuestos de monotributos, por lo que se dará de alta por otros medios.
  )
  [1] => Array
  (
    [MENSAJE] => Se dio de alta el contribuyente [REDACTED]
    [IDCONTRIBUYENTE] => [REDACTED]
    [INTEGRANTES] => Array
    (
      [0] => Se dio de alta el integrante [REDACTED].
      [1] => Se dio de alta el integrante [REDACTED].
      [2] => Se dio de alta el integrante [REDACTED].
      [3] => Se dio de alta el integrante [REDACTED].
    )
    [RESULT] => OK
  )
  [2] => Array
  (
    [MENSAJE] => El contribuyente [REDACTED], cuenta con impuestos de monotributos, por lo que se dará de alta por otros medios.
  )
  [3] => Array
  (
    [MENSAJE] => Se dio de alta el contribuyente [REDACTED]
    [IDCONTRIBUYENTE] => [REDACTED]
    [RESULT] => OK
  )
  [4] => Array
  (
    [MENSAJE] => Se dio de alta el contribuyente [REDACTED]
    [IDCONTRIBUYENTE] => [REDACTED]
    [RESULT] => OK
  )
)
```

Figura 18. Resultados de la ejecución de carga realizada mediante Postman.

CAMBIOS DE CUITs

RESULTADOS DEL PROCESO DE ALTA

El contribuyente [REDACTED], cuenta con impuestos de monotributos, por lo que se dará de alta por otros medios.

Se dio de alta el contribuyente [REDACTED]

Integrantes

Se dio de alta el integrante [REDACTED]

El contribuyente [REDACTED], cuenta con impuestos de monotributos, por lo que se dará de alta por otros medios.

Se dio de alta el contribuyente [REDACTED]

Se dio de alta el contribuyente [REDACTED]

Se dio de alta el contribuyente [REDACTED]

Integrantes

Se dio de alta el integrante [REDACTED]

Se dio de alta el integrante [REDACTED]

Se dio de alta el integrante [REDACTED]

Se dio de alta el contribuyente [REDACTED]

Figura 19. Resultados de la ejecución de carga enviados por mail.

11. Módulo de Actualización de Domicilios Fiscales

En los artículos 23 y 24 del Código Fiscal se reglamenta que los contribuyentes de la Provincia de Río Negro deben tener como domicilio fiscal en la provincia el mismo que registran en AFIP. Esto se implementó originalmente mediante un proceso mensual de actualización de domicilios fiscales desde la información actualizada de AFIP obtenida por el proceso de sincronización de padrones mensuales antes descritos.

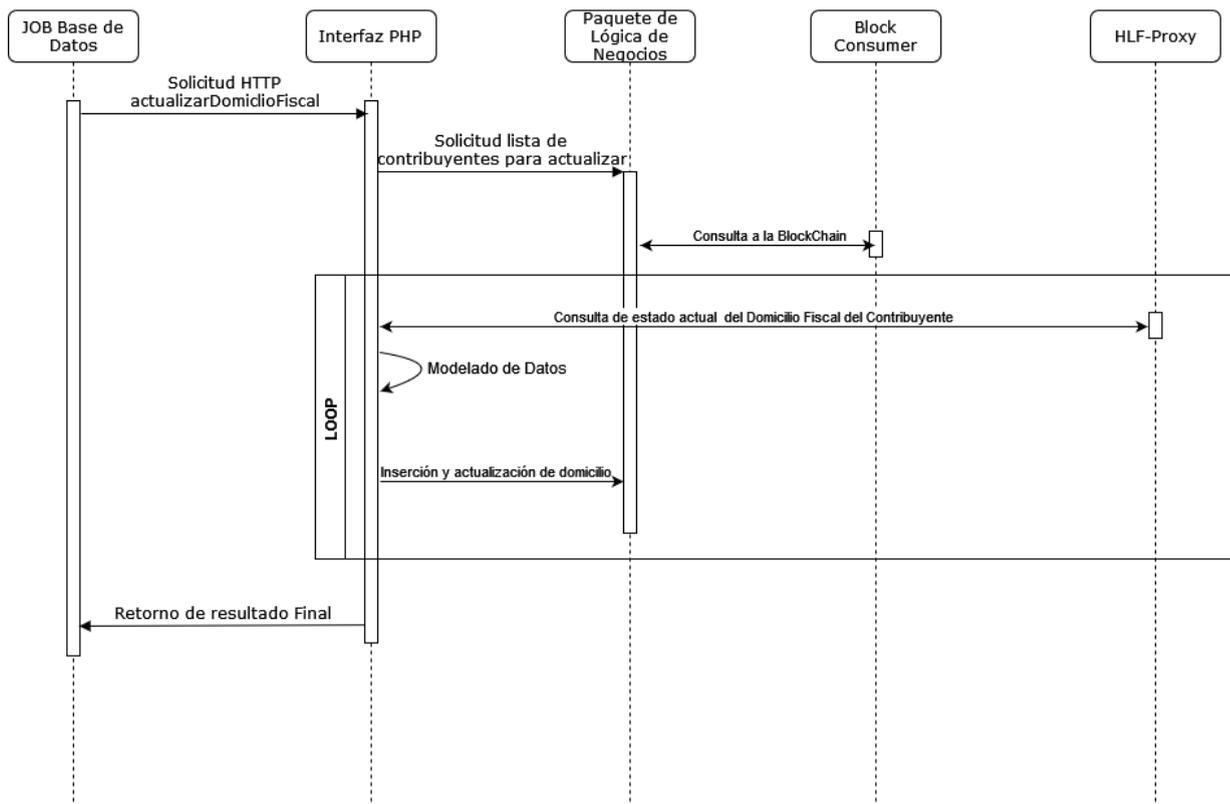
Los agentes de atención al público reportaron los reiterados reclamos de los contribuyentes que modifican su domicilio fiscal en AFIP y no ven reflejado el cambio hasta que el proceso mensual se ejecuta.

11.1 Análisis y Diseño

Para el desarrollo de este módulo se realizaron tareas de reingeniería sobre procesos existentes. Estas funciones, implementadas en la capa de lógica, tomaban la información desde tablas cargadas mensualmente, por lo que no recibían parámetros de entrada, y tampoco emitían resultados como salida.

Utilizando las herramientas mencionadas y dispuestas para el desarrollo del módulo anterior, se planteó una solución automática y diaria, aprovechando las facultades de la BCPadFed.

El siguiente diagrama de secuencia muestra el orden de las tareas realizadas para la solución planteada:



11.2 Implementación

Al igual que el módulo de Alta de Contribuyentes, el proceso comienza a partir de un JOB almacenado en la base de datos ORACLE, esta tarea realiza una solicitud o método HTTP a la interfaz PHP, que cuenta con los módulos necesarios para realizar el trabajo requerido. En esta ocasión, el encargado de las tareas de consultar los datos que coincidan con las condiciones necesarias, se denomina **actualizarDomicilioFiscal**, así mismo, recibe como parámetros la FECHA para realizar una actualización masiva o una CUIT particular para realizar una actualización individual. Obtenidos estos datos, se solicita a la capa de *Lógica de Negocios* la lista de contribuyentes que cuentan con algún tipo de cambio en su domicilio fiscal, esta capa accede directamente a la Block Consumer, y retorna mediante consultas SQL la información de aquellos contribuyentes que cumplan con estas condiciones.

Una vez que la *Interfaz PHP* cuenta con la lista de contribuyentes cuyo domicilio fiscal cumple las características para ser actualizado, comienza un proceso iterativo sobre la misma. Se consulta la API Rest, HLF Proxy, para conocer el nuevo domicilio del contribuyente y poder modelar la información antes de ser enviada a la capa de Lógica de Negocios para su próxima actualización en la base de datos de la ARTRN.

```

foreach ($domicilios as $domicilio) {

    $this->clienteBc->getDomicilioFiscal($domicilio['PERSONA']);
    $ccResponse = $this->clienteBc->getCcResponse();

    if ($ccResponse->status == 200 && $ccResponse->content->provincia == $this->codProvincia) {

        $fecha = date($ccResponse->content->ds);
        $nuevafecha = strtotime( time: "now", strtotime($fecha));
        $fechadesde = date( format: 'd/m/Y', $nuevafecha);

        $domicilioContribuyente = new DomicilioContribuyente();
        $domicilioContribuyente->setTipoafip($ccResponse->content->tipo);
        $domicilioContribuyente->setIdnomenclafip( idnomenclafip: isset($ccResponse->content->nomenclador) ? $ccResponse->content->nomenclador : null);
        $domicilioContribuyente->setCalle( calle: isset($ccResponse->content->calle) ? $ccResponse->content->calle : "S/D");
        $domicilioContribuyente->setNumero( numero: (isset($ccResponse->content->numero)) ? $ccResponse->content->numero : null);
        $domicilioContribuyente->setLocalidad( localidad: isset($ccResponse->content->localidad) ? $ccResponse->content->localidad : null);
        $domicilioContribuyente->setPiso( piso: isset($ccResponse->content->piso) ? $ccResponse->content->piso : null);
        $domicilioContribuyente->setSector( sector: isset($ccResponse->content->sector) ? $ccResponse->content->sector : null);
        $domicilioContribuyente->setManzana( manzana: isset($ccResponse->content->manzana) ? $ccResponse->content->manzana : null);
        $domicilioContribuyente->setTorre( torre: isset($ccResponse->content->torre) ? $ccResponse->content->torre : null);
        $domicilioContribuyente->setUnidad( unidad: isset($ccResponse->content->unidad) ? $ccResponse->content->unidad : null);

        if (isset($ccResponse->adicional)) {
            $domicilioContribuyente->setTipodatoadic($ccResponse->content->adicional->tipo);
            $domicilioContribuyente->setDatoadic($ccResponse->content->adicional->dato);
        }
        $domicilioContribuyente->setPostafip($ccResponse->content->cp);
        $domicilioContribuyente->setFechadesde($fechadesde);

        $response = $this->dao->actualizarDomicilioFiscal($domicilioContribuyente, $domicilio['ID_CONTRIBUYENTE']);
        if ($response['RESULT'] == 'OK') {
            array_push( &array: $result, ...vars: "Se actualizó el domicilio fiscal del contribuyente: " . $domicilio['PERSONA'] . ".");
        } else {
            array_push( &array: $result, $response['MENSAJE']);
        }
    } else {
        if($ccResponse->content->provincia != $this->codProvincia){
            array_push( &array: $result, ...vars: "El domicilio fiscal del contribuyente " . $domicilio['PERSONA'] . " no pertenece a RIO NEGRO.");
        }else{
            array_push( &array: $result, ...vars: "Error al consultar el domicilio fiscal del contribuyente: " . $domicilio['PERSONA']);
        }
    }
}

```

Figura 20. Extracto del código perteneciente a la interfaz PHP.

La figura 20 es un extracto del código implementado en la interfaz PHP, en esta recopilación podemos visualizar la construcción de un nuevo domicilio perteneciente al contribuyente, con los datos requeridos por la capa de *Lógica de Negocios*, la condición para constituir este tipo de

domicilios se refiere a que el mismo debe pertenecer a la provincia de Río Negro.

Luego del modelado de datos se procede a la actualización del domicilio fiscal, los parámetros necesarios para llevar adelante esta labor, son:

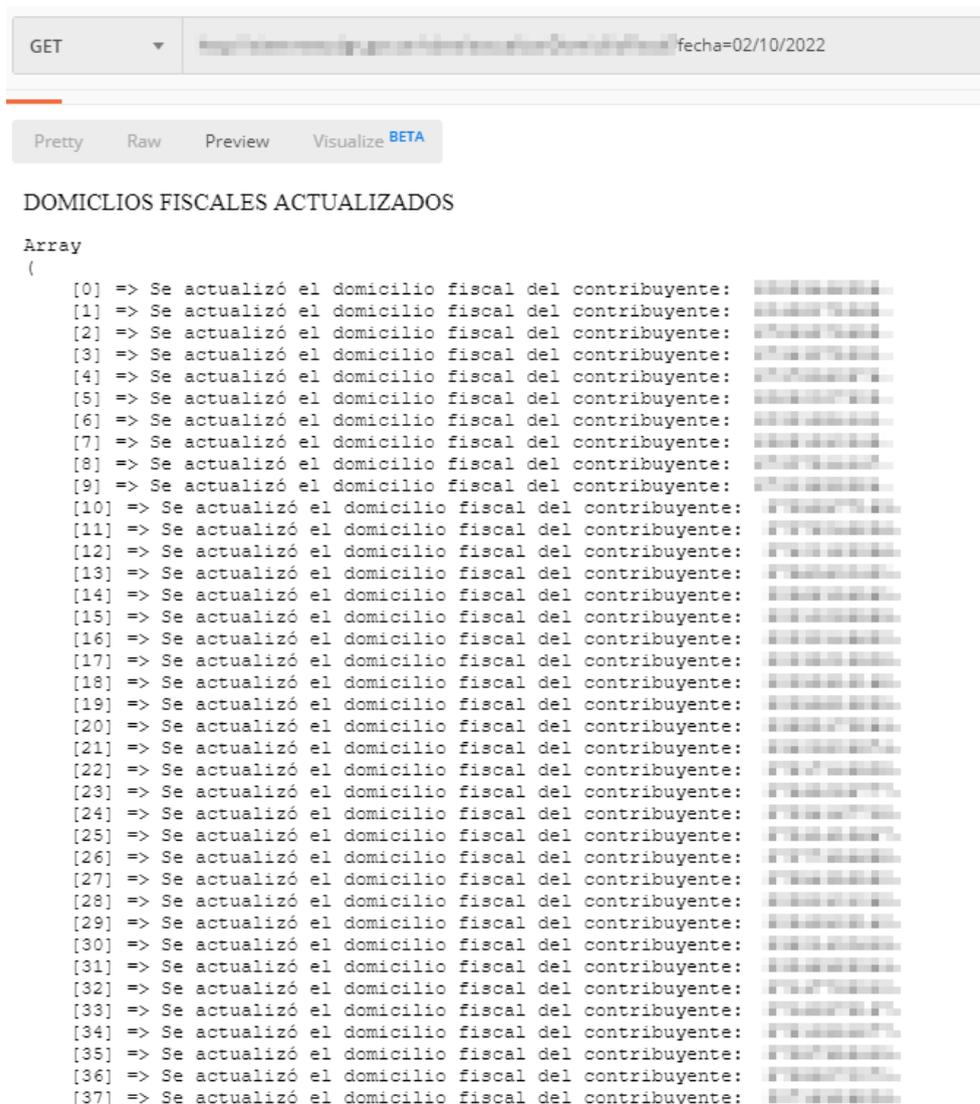
```
function act_part_dom_fiscal( p_id_contribuyente contribuyentes.id_contribuyente%type,  
                             p_vig_desde      varchar2,  
                             p_postal        domicilios.c_postal%type,  
                             p_id_nomencla_afip number,  
                             p_localidad     varchar2,  
                             p_calle        domicilios.d_calle%type,  
                             p_numero       domicilios.n_numero%type,  
                             p_piso         in out domicilios.d_piso%type,  
                             p_depto        in out domicilios.d_depto%type,  
                             p_manzana      in out domicilios.d_manzana%type,  
                             p_tipo_dataadic in out number,  
                             p_dato_adic    in out varchar2,  
                             p_vig_hasta    in domicilios.f_vig_hasta%type default null) return t_cursor;
```

Dentro de la capa de lógica de negocios, se modelan y mapean los datos para que correspondan con las tablas y referencias utilizadas en todo el sistema.

11.3 Pruebas y Resultados

El resultado de cada una de las iteraciones es cargado en un arreglo de datos que va a ser utilizado para mostrar el estado final del módulo en la pantalla del navegador, si es ejecutado por este medio, y para enviar un mail a aquellos usuarios encargados de controlar que la información almacenada sea la correcta.

En las figuras 21 y 22, se puede ver un ejemplo de los resultados obtenidos.



```
GET http://192.168.1.100:8080/api/actualizar-domicilio-fiscal?fecha=02/10/2022

Pretty Raw Preview Visualize BETA

DOMICLIOS FISCALES ACTUALIZADOS

Array
(
  [0] => Se actualizó el domicilio fiscal del contribuyente: 2022-10-02T10:00:00.000Z
  [1] => Se actualizó el domicilio fiscal del contribuyente: 2022-10-02T10:00:00.000Z
  [2] => Se actualizó el domicilio fiscal del contribuyente: 2022-10-02T10:00:00.000Z
  [3] => Se actualizó el domicilio fiscal del contribuyente: 2022-10-02T10:00:00.000Z
  [4] => Se actualizó el domicilio fiscal del contribuyente: 2022-10-02T10:00:00.000Z
  [5] => Se actualizó el domicilio fiscal del contribuyente: 2022-10-02T10:00:00.000Z
  [6] => Se actualizó el domicilio fiscal del contribuyente: 2022-10-02T10:00:00.000Z
  [7] => Se actualizó el domicilio fiscal del contribuyente: 2022-10-02T10:00:00.000Z
  [8] => Se actualizó el domicilio fiscal del contribuyente: 2022-10-02T10:00:00.000Z
  [9] => Se actualizó el domicilio fiscal del contribuyente: 2022-10-02T10:00:00.000Z
  [10] => Se actualizó el domicilio fiscal del contribuyente: 2022-10-02T10:00:00.000Z
  [11] => Se actualizó el domicilio fiscal del contribuyente: 2022-10-02T10:00:00.000Z
  [12] => Se actualizó el domicilio fiscal del contribuyente: 2022-10-02T10:00:00.000Z
  [13] => Se actualizó el domicilio fiscal del contribuyente: 2022-10-02T10:00:00.000Z
  [14] => Se actualizó el domicilio fiscal del contribuyente: 2022-10-02T10:00:00.000Z
  [15] => Se actualizó el domicilio fiscal del contribuyente: 2022-10-02T10:00:00.000Z
  [16] => Se actualizó el domicilio fiscal del contribuyente: 2022-10-02T10:00:00.000Z
  [17] => Se actualizó el domicilio fiscal del contribuyente: 2022-10-02T10:00:00.000Z
  [18] => Se actualizó el domicilio fiscal del contribuyente: 2022-10-02T10:00:00.000Z
  [19] => Se actualizó el domicilio fiscal del contribuyente: 2022-10-02T10:00:00.000Z
  [20] => Se actualizó el domicilio fiscal del contribuyente: 2022-10-02T10:00:00.000Z
  [21] => Se actualizó el domicilio fiscal del contribuyente: 2022-10-02T10:00:00.000Z
  [22] => Se actualizó el domicilio fiscal del contribuyente: 2022-10-02T10:00:00.000Z
  [23] => Se actualizó el domicilio fiscal del contribuyente: 2022-10-02T10:00:00.000Z
  [24] => Se actualizó el domicilio fiscal del contribuyente: 2022-10-02T10:00:00.000Z
  [25] => Se actualizó el domicilio fiscal del contribuyente: 2022-10-02T10:00:00.000Z
  [26] => Se actualizó el domicilio fiscal del contribuyente: 2022-10-02T10:00:00.000Z
  [27] => Se actualizó el domicilio fiscal del contribuyente: 2022-10-02T10:00:00.000Z
  [28] => Se actualizó el domicilio fiscal del contribuyente: 2022-10-02T10:00:00.000Z
  [29] => Se actualizó el domicilio fiscal del contribuyente: 2022-10-02T10:00:00.000Z
  [30] => Se actualizó el domicilio fiscal del contribuyente: 2022-10-02T10:00:00.000Z
  [31] => Se actualizó el domicilio fiscal del contribuyente: 2022-10-02T10:00:00.000Z
  [32] => Se actualizó el domicilio fiscal del contribuyente: 2022-10-02T10:00:00.000Z
  [33] => Se actualizó el domicilio fiscal del contribuyente: 2022-10-02T10:00:00.000Z
  [34] => Se actualizó el domicilio fiscal del contribuyente: 2022-10-02T10:00:00.000Z
  [35] => Se actualizó el domicilio fiscal del contribuyente: 2022-10-02T10:00:00.000Z
  [36] => Se actualizó el domicilio fiscal del contribuyente: 2022-10-02T10:00:00.000Z
  [37] => Se actualizó el domicilio fiscal del contribuyente: 2022-10-02T10:00:00.000Z
)
```

Figura 21. Resultados de la ejecución de carga realizada mediante Postman.

DOMICLIOS FISCALES ACTUALIZADOS

Se actualizó el domicilio fiscal del contribuyente: [REDACTED]

Figura 22. Resultados de la ejecución de carga enviados por mail.

12. Módulo de Consulta del Estado del Contribuyente en BCPadFed

Este módulo fue desarrollado utilizando la interfaz gráfica de la herramienta web interna (Intranet) de la Agencia de Recaudación Tributaria. Para ello, fue necesario adaptar la tecnología para poder conectarse a la API Restful, HLF-Proxy. La herramienta web está implementada utilizando el lenguaje de programación PHP, por lo que la adaptación no fue costosa, pero no fue posible utilizar la estructura de proyecto usada en la creación de los módulos anteriores.

12.1 Análisis y Diseño

Existe un gran número de oportunidades en que personal de la Agencia debe constatar datos contra la información de AFIP, ya sea en atención presencial de contribuyentes en las Oficinas o en la resolución de solicitudes de contribuyentes recibidas desde la Oficina Virtual, como así también para el desarrollo de tareas internas de Recaudación, Cobranzas, Análisis Tributario, Fiscalización, entre otras.

Estas necesidades eran resueltas mediante interfaces de consulta a tablas que contienen los padrones mensuales obtenidos de AFIP, información que en algunos casos resulta desactualizada al momento de utilizarla. Por este motivo se hizo necesaria la creación de esta herramienta para que los usuarios del sistema puedan acceder en línea a los datos de un contribuyente en AFIP para llevar a cabo sus tareas diarias.

La consulta de estado del contribuyente, muestra la información actual en la BlockChain del Padrón Federal, de cada una de las propiedades pertenecientes a un contribuyente físico o persona jurídica. Estos registros son: datos identificatorios de la persona jurídica, impuestos, domicilios, roles de domicilios, categorías, contribuciones municipales, actividades, etiquetas, teléfonos, emails, relaciones , jurisdicciones , sedes para el convenio multilateral, archivos, puntos de venta.

12.2 Implementación

El sistema web interno de la Agencia cuenta con una interfaz amigable aprovechando los estándares de desarrollo más utilizados en la actualidad, HTML5, CSS3 y el framework Bootstrap 5.

La interfaz visual del módulo de consulta, es la siguiente:

Datos Del Contribuyente

CUIT (*)
20370562483

ID Contribuyente
20370562483

Documento
37056248

Nombre
NAHUEL EMILIANO

Apellido
PICHON

Tipo de Persona
Física

Sexo
Masculino

Estado
Activo

Fecha de Alta
2017-04-01

Fecha de Nacimiento
1992-10-05

Actividades | Impuestos | **Domicilios** | Telefonos | Emails | Archivos | Categorías | Contribuciones Municipales

ORG	TIPO	ESTADO	CALLE	NÚMERO	PISO	SECTOR	MANZANA	TORRE	UNIDAD	PROVINCIA	LOCALIDAD	CP	NOMENCLADOR	NOMBRE	ADICIONAL	BAJA	PARTIDO	PARTIDA	DS
1	1	11	GUEMES	40	PB				3	22	VIEDMA	8500	6358						2022-06-08
1	2	11	GUEMES	40	PB				3	22	VIEDMA	8500	6358						2022-06-08

La estructura de carpetas del módulo respeta la utilizada en la *Intranet* del organismo. Esto se refiere a que existe un archivo principal que inicia las actividades sobre el instrumento e incluye los archivos necesarios para el funcionamiento e imagen del mismo. En las figuras 23 y 24, podemos vislumbrar el contenido del archivo inicial y la estructura de carpetas utilizadas para el funcionamiento de la consulta.

```
<?php
require_once(INTRANET."header.php");
require_once("contribuyentes_bc/php/ClienteBCAfip.php");
?>

<script type="text/javascript" src="contribuyentes_bc/js/grillas.js?no_cache=<?=date( format: 'dmy')?>"></script>
<script type="text/javascript" src="contribuyentes_bc/js/funciones.js?no_cache=<?=date( format: 'dmy')?>"></script>
<link rel="stylesheet" type="text/css" href="contribuyentes_bc/css/estilos.css">

?>php

include('contribuyentes_bc/html/principal.html');

require_once(INTRANET."footer.php");
?>
```

Figura 23. Contenido del archivo **contribuyentes_bc.php**.

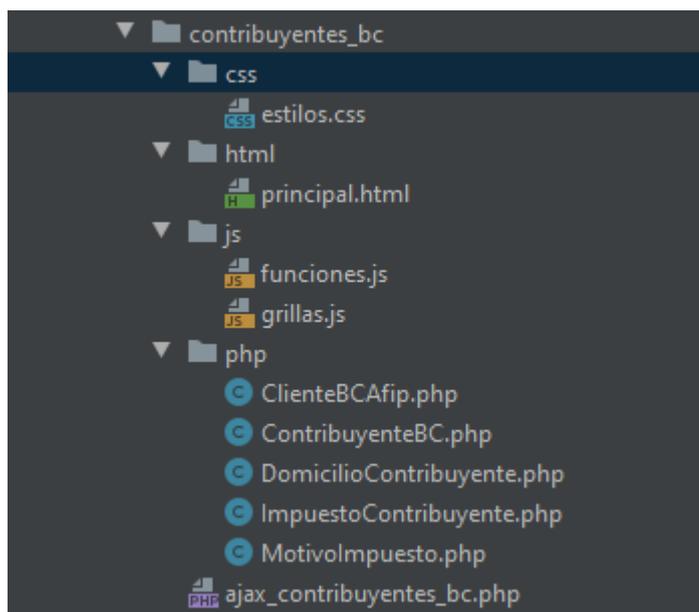


Figura 24. Estructura de carpetas del módulo de consultas del estado del contribuyente en la BCPadFed.

13. Conclusiones

A lo largo de este escrito se intentó demostrar el amplio abanico de posibilidades que nos brinda BlockChain, y todas las tecnologías asociadas, para la mejora de la seguridad y confiabilidad. Sin embargo, es una tecnología nueva, que aún se encuentra en etapa de investigación y en plan de crecimiento continuo, más allá de los grandes casos de éxito que se conocen. La sencillez de su funcionamiento es lo que la hizo tan popular y al mismo tiempo permitió que la base con la que fue presentada mute día a día. Como se dijo en un principio, el mayor de los obstáculos de esta herramienta es, y será durante un tiempo, el de desacoplarse de su aplicación primaria, lo que no supone en realidad un obstáculo en sí, pero confunde a aquellos que comienzan a involucrarse en la materia sin conocimiento previo.

Los autores consultados coinciden en que la aplicación de la cadena de bloques en distintos ámbitos es totalmente posible, pero para ello, es necesario entender muy bien su funcionamiento. En particular, Serale, Redl y Munte-Kunigami proponen pasos a seguir para la puesta en marcha de la cadena de bloques en los organismos gubernamentales. Además, hablan de ciertas problemáticas asociadas a los funcionarios encargados de tomar la decisión sobre el desarrollo con la tecnología propuesta.

Estamos frente a un nuevo cambio tecnológico, una evolución de la red. Sin embargo, tal vez no cueste tanto como los hitos anteriores, ya que la sociedad de hoy se encuentra inmersa en un mundo de telecomunicaciones y relaciones virtuales. La llamada generación de la tecnología deberá ser la precursora de este cambio, pero antes de eso, debemos entender hacia dónde vamos y cómo queremos dirigirnos a ello. Es posible que esta nueva tecnología se vea dificultada, detenida, poco utilizada o algo peor, si no comprendemos bien su funcionamiento o no le damos la relevancia que realmente tienen sus características. El sistema BlockChain se encuentra ahora en pleno auge, pero no podemos predecir si logrará triunfar, ni con la rapidez que lo hará, si fuera el caso. Como sugiere Satoshi Nakamoto, ***“la confianza tiene un costo elevado y depender de terceros para realizar transacciones de datos puede ser un gran problema”***. Por eso, la solución propuesta parece ser la más acertada, aunque esto signifique lidiar con personas en el poder cuyos intereses se vean afectados.

En particular, la utilización de esta tecnología por parte de la Agencia de Recaudación Tributaria no solo es una actualización de sistema, sino un cambio de perspectiva total en cuanto al manejo de procesos, mecanismos de automatización y servicios ofrecidos al contribuyente. Se espera seguir creciendo en el manejo de la herramienta, mejorando la calidad de los datos resguardados y compartiendo información con el resto de los nodos de la red, hasta lograr constituirse como *Nodo Peer* de la BlockChain del Padrón Federal.

14. Referencias Bibliográficas y Otras Fuentes

1. Tapscott D. y Tapscott A. (2018). *La Revolución Blockchain*, Penguin.
2. How DigiCash Blew Everything, NEXT magazine, enero 1999. [Cómo DigiCash sopló todo \(cryptome.org\)](http://www.cryptome.org)
3. Nick Szabo. (1997). *The God Protocols*.
<https://nakamotoinstitute.org/the-god-protocols/>
4. Nakamoto S. (2008). *Bitcoin: Un Sistema de Efectivo Electrónico Usuario-a-Usuario*.
https://bitcoin.org/files/bitcoin-paper/bitcoin_es_latam.pdf
5. Cicomra, *El impacto de las TIC en la economía y la sociedad, Opiniones de expertos y testimonios sectoriales*.
6. Serale F., Redl C., Muent-Kunigami A. (2019). *BLOCKCHAIN EN LA ADMINISTRACIÓN PÚBLICA: ¿Mucho ruido y pocos bloques?.* Banco Interamericano de Desarrollo.
7. Larimer, D. and Maloney, M. and Rubock, D. and Mackness, A. and Salatin, J. (2021). *More Equal Animals: The Subtle Art of True Democracy*. <https://books.google.com.ar/books?id=0rsszgEACAAJ>
8. Erich Gamma, John Vlissides, Ralph Johnson y Richard Helm. (1994). *Patrones de diseño: elementos de software orientado a objetos reutilizables*. Pearson Educación, S.A.
9. Blockchain Federal Argentina. <https://bfa.ar/>
10. Tapscott D.(2016, Junio). How the Blockchain changing money and business[VIDEO]. Conferencias TED.
https://www.ted.com/talks/don_tapscott_how_the_blockchain_is_changing_money_and_business#t-24051
11. Historia De La Tecnología Blockchain: Guía Definitiva. Historia de la tecnología Blockchain: Guía definitiva (101blockchains.com)
12. PADFED Docs. Padrón Federal. <https://github.com/padfed/padfed-doc>

13. Hacia un Registro Único Tributario a nivel federal.
<https://www.ca.gob.ar/hacia-un-registro-unico-tributario-a-nivel-federal>
14. W. Dilger, *Decentralized autonomous organization of the intelligent home according to the principle of the immune system*.
<https://ieeexplore.ieee.org/document/625775>
15. ¿Qué es una DAO?. <https://academy.bit2me.com/que-es-una-dao/>
16. Szabo, Nick (1994). *Smart Contracts*.
<https://www.fon.hum.uva.nl/rob/Courses/InformationInSpeech/CDROM/Literature/LOTwinterschool2006/szabo.best.vwh.net/smart.contracts.html>
17. Documentación oficial de la BlockChain del Padrón Federal.
<https://github.com/padfed/padfed-doc>
18. Jolias, Lucas (2017). Blockchain, transparencia y el futuro del Gobierno Abierto