

19



OFICINA ESPAÑOLA DE
PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **2 579 454**

51 Int. Cl.:

G06F 17/00 (2006.01)

G06F 9/44 (2006.01)

G06F 9/445 (2006.01)

12

TRADUCCIÓN DE PATENTE EUROPEA

T3

96 Fecha de presentación y número de la solicitud europea: **17.11.2006 E 06837917 (1)**

97 Fecha y número de publicación de la concesión europea: **04.05.2016 EP 1960899**

54 Título: **Ajuste de servicios web a un contrato actualizado**

30 Prioridad:

15.12.2005 US 304300

45 Fecha de publicación y mención en BOPI de la traducción de la patente:

11.08.2016

73 Titular/es:

**MICROSOFT TECHNOLOGY LICENSING, LLC
(100.0%)
One Microsoft Way
Redmond, WA 98052, US**

72 Inventor/es:

**PASHA, AZMAT, A. y
GIBSON, WILLIAM, E.**

74 Agente/Representante:

CARPINTERO LÓPEZ, Mario

ES 2 579 454 T3

Aviso: En el plazo de nueve meses a contar desde la fecha de publicación en el Boletín europeo de patentes, de la mención de concesión de la patente europea, cualquier persona podrá oponerse ante la Oficina Europea de Patentes a la patente concedida. La oposición deberá formularse por escrito y estar motivada; sólo se considerará como formulada una vez que se haya realizado el pago de la tasa de oposición (art. 99.1 del Convenio sobre concesión de Patentes Europeas).

DESCRIPCIÓN

Ajuste de servicios web a un contrato actualizado

Antecedentes

5 Los sistemas informáticos y la tecnología relacionada afectan a muchos aspectos de la sociedad. De hecho, la capacidad del sistema informático de procesar información ha transformado la forma en la que viven y trabajan los seres humanos. En la actualidad, los sistemas informáticos realizan comúnmente una multitud de tareas (por ejemplo, procesamiento de textos, planificación, gestión de bases de datos, etc.) que se realizaban de forma manual antes de la aparición de los sistemas informáticos. Más recientemente, los sistemas informáticos se han acoplado entre sí para formar unas redes informáticas a través de las cuales los sistemas informáticos se pueden comunicar por medios electrónicos para compartir datos. Los servicios web han sido una fuerza impulsora en el fomento de tales comunicaciones entre sistemas informáticos y están dando completamente la vuelta a la forma en la que se construye y se usa el soporte lógico.

15 Los servicios web dejan que las aplicaciones compartan datos, y - de forma más potente - invocan capacidades a partir de otras aplicaciones sin importar cómo se construyeron esas aplicaciones, en qué plataforma o sistema operativo se ejecutan estas y qué dispositivos se usan para acceder a las mismas. Los servicios web se pueden invocar a través de Internet por medio de protocolos según normas de la industria, incluyendo SOAP (*Simple Open Access Protocol*, Protocolo de Acceso Abierto Simple), XML (*eXtensible Markup Language*, Lenguaje de Mercado eXtensible), UDDI (*Universal Description Discovery Integration*, Integración de Descubrimiento de Descripciones Universal), WSDL (*Web Service Description Language*, Lenguaje de Descripción de Servicios Web), etc. A pesar de que los servicios web permanecen independientes unos de otros, estos se pueden vincular vagamente entre sí para dar un grupo colaborador que realiza una tarea particular.

25 A menudo, la comunicación electrónica en una red de servicio web incluye un sistema informático cliente (al que se hace referencia en lo sucesivo en el presente documento como "cliente") que solicita acceso a un servicio de red (por ejemplo, servicios web) en un sistema informático servidor (al que se hace referencia en lo sucesivo en el presente documento como "servidor", "servicio" o "servicio web"). Por consiguiente, el cliente envía una solicitud al servicio para un acceso particular a sus recursos de sistema, en la que el servicio responde con un mensaje de respuesta que proporciona la información deseada. Por supuesto, otros patrones de mensajería entre cliente y servicio se pueden encontrar disponibles e incluir mensajes solitarios simples así como intercambios de múltiples mensajes, más sofisticados, como, por ejemplo, notificaciones, petición - respuesta, patrones de publicador - abonado, sondeo, expulsión - inserción, puesta en cola y otros. Además, pueden existir otros procedimientos necesarios para acceder a los recursos de servicio, tales como mecanismos de autenticación y de validación.

35 Con independencia del tipo de procedimientos y o patrones de mensajes que se usen para acceder a un servicio, los servicios web proporcionan interoperabilidad de una forma agnóstica en cuanto a la plataforma, debido a que tanto clientes como servicios acuerdan un contrato básico. Representados por una mezcla de lenguajes humanos y legibles por máquina, tales contratos definen, entre otras cosas, las operaciones - unidades de funcionalidad - ofrecidas por el servicio web y/o el formato de los mensajes que se pasan a y/o que son devueltos por cada operación al invocarse. Los Lenguajes de Descripción de Protocolos de Red (por ejemplo, WSDL) proporcionan una especificación o contenedor global para describir contratos de servicio web en un lenguaje común o convencional. Tales especificaciones hacen que sea sencillo para los desarrolladores y las herramientas de los desarrolladores crear e interpretar contratos; y también tener conjuntos de herramientas exhaustivos, lo que explica en gran parte su popularidad.

45 Un contrato para un servicio web o bien se puede escribir independientemente de la implementación de servicio (por ejemplo, uno que incluye una lógica de negocio a partir de otra organización) o bien puede ser producido por algunos medios automatizados para reflejar el comportamiento de un servicio existente. Cada vez más, los desarrolladores necesitan de la capacidad de exponer con precisión el contrato independientemente de la implementación. Por ejemplo, los escenarios de interoperabilidad entre múltiples empresas, o grupos o equipos dentro de la misma empresa, usan a menudo una estrategia de desarrollo de tipo "contrato primero" o "regida por contrato". En tales casos, una vez que se ha acordado un contrato entre las partes participantes, se pueden crear una implementación o implementaciones específicas del servicio web para ajustarse a la lógica de negocio de los diversos puntos de extremo. Las herramientas existentes permiten la generación de una implementación de servicio web "de esqueleto" basándose en el contrato de servicio web, lo que permite entonces que un desarrollador complete los detalles de la lógica de negocio para cada operación que es proporcionada por el servicio. Por lo general, tales implementaciones de esqueleto comprenden una clase para el servicio con procedimientos para cada operación según sea definido por el contrato. Además, la firma de cada procedimiento describe los mensajes que se envían a y/o que son devueltos por la operación, con clases adicionales creadas en donde se requiera para describir estructuras de datos de mensaje a las que se hace referencia en la firma.

55 Muy a menudo, no obstante, el contrato de servicio web cambia después de que haya comenzado la implementación del servicio web. Por ejemplo, puede que el contrato necesite cambios para satisfacer requisitos de negocios nuevos o modificados o en el caso en el que los desarrolladores usen unas estrategias de desarrollo iterativas o por

incrementos. Por consiguiente, los cambios al contrato de servicio web requerirán, por lo general, unos cambios tanto al esqueleto (los procedimientos, firmas y clases o estructuras de datos) como a la lógica de negocio con el fin de ajustarse al contrato de servicio actualizado. Las herramientas actuales, no obstante, no soportan actualizar la implementación de servicio web para dar cabida a los cambios al contrato.

5 Por consiguiente, un desarrollador tiene pocas opciones disponibles para modificar una implementación de servicio web para ajustarse al contrato de servicio web actualizado. Por ejemplo, por lo general el desarrollador: o bien (1) recreará una nueva implementación de servicio a partir del contrato revisado usando herramientas existentes (por ejemplo, herramienta de esqueleto) y migrará de forma manual aquellos aspectos de la implementación de servicio anterior que este desee conservar; o bien (2) editará de forma manual el código de implementación existente para hacer los cambios necesarios para ajustarse al contrato revisado. Dependiendo del alcance del cambio, ambos enfoques requieren mucho esfuerzo y están sujetos a error humano. Además, estos enfoques pueden requerir que el desarrollador tenga un conocimiento profundo del lenguaje NPD (por ejemplo, WSDL) y/o el esquema que se usa (por ejemplo, un esquema de XML) de tal modo que este entienda la naturaleza del cambio y cómo asignar este a su implementación. Debido a las complejidades en tales lenguajes y esquemas, no obstante, tales modificaciones se encuentran más allá del típico conjunto de destrezas de la mayor parte de los desarrolladores.

El documento WO 02/46909 A se refiere a un procedimiento para implantar una aplicación de soporte lógico que incluye crear un documento estructurado que define la aplicación de soporte lógico, que se usa de forma automática para implantar la aplicación de soporte lógico. El documento estructurado puede comprender un documento en lenguaje de marcado, tal como un documento de XML. El documento estructurado puede describir múltiples niveles de la aplicación de soporte lógico (por ejemplo, un nivel de presentación, un nivel de lógica y un nivel de datos) y puede definir uno o más cualesquiera de datos, lógica e interfaces de la aplicación de soporte lógico.

Breve resumen

El objeto de la presente invención es superar las deficiencias y los inconvenientes que se han identificado en lo que antecede de los mecanismos actuales para modificar una implementación de servicio en respuesta a cambios en un contrato.

El presente objeto es solucionado por la materia objeto de las reivindicaciones independientes.

Se dan realizaciones en las reivindicaciones dependientes.

Obsérvese que el presente Sumario se proporciona para presentar, en una forma simplificada, una selección de conceptos que se describen adicionalmente en lo sucesivo en la descripción detallada. El presente Sumario no tiene por objeto identificar características clave o características esenciales de la materia objeto que se reivindica, ni este tiene por objeto usarse como una ayuda a la determinación del alcance de la materia objeto que se reivindica.

Una realización a modo de ejemplo prevé cambiar de forma automática al menos una porción de una implementación de servicio en respuesta a cambios en un documento de contrato de servicio web. En la presente realización, se recibe una implementación de servicio web que se ha construido de acuerdo con un primer documento de contrato de servicio web. El primer documento de contrato de servicio web describe cómo la implementación de servicio web se va a comunicar con un servicio o servicios. Además, también se recibe un segundo documento de contrato de servicio web el cual difiere del primer documento de contrato de servicio web. Los uno o más cambios afectan al comportamiento de la implementación de servicio web de una forma tal que la implementación de servicio web no se puede comunicar con puntos de extremo que esperan que se implemente el segundo contrato. A continuación de lo anterior, se identifican los uno o más cambios entre la implementación de servicio web y el segundo documento de contrato. Basándose en esta identificación, una porción de la implementación de servicio web se modifica de forma automática de tal modo que la implementación de servicio web se ajusta al menos parcialmente al segundo documento de contrato de servicio web.

En la descripción que sigue se expondrán características y ventajas adicionales de las realizaciones que se divulgan en el presente documento, y en parte estas serán obvias a partir de la descripción, o se pueden aprender mediante la práctica de la invención. Las características y ventajas de las realizaciones que se divulgan en el presente documento se pueden realizar y obtener por medio de los instrumentos y combinaciones particularmente señaladas en las reivindicaciones adjuntas. Estas y otras características de las realizaciones que se divulgan en el presente documento se harán más completamente evidentes a partir de la siguiente descripción y las reivindicaciones adjuntas, o se pueden aprender mediante la práctica de las realizaciones que se divulgan en el presente documento tal como se expone en lo sucesivo en el presente documento.

Breve descripción de los dibujos

Con el fin de describir la forma en la cual se pueden obtener las ventajas y características que se han enunciado en lo que antecede y otras ventajas y características de la invención, se presentará una descripción más particular de la invención que se ha descrito brevemente en lo que antecede por referencia a unas realizaciones específicas de la misma, las cuales se ilustran en los dibujos adjuntos. Entendiendo que estos dibujos muestran solo realizaciones típicas de la invención y, por lo tanto, no se ha de considerar que sean limitantes de su alcance, la invención se

describirá y se explicará con precisión y detalle adicional a través del uso de los dibujos adjuntos en los que:

la figura 1A ilustra un sistema informático distribuido para modificar de forma automática una implementación de servicio web en respuesta a cambios a un documento de contrato de servicio web de acuerdo con unas realizaciones a modo de ejemplo;

5 la figura 1B ilustra un ejemplo de una implementación de servicio;

la figura 2 ilustra un diagrama de flujo de un procedimiento para modificar de forma automática una implementación de servicio en respuesta a cambios a un documento de contrato de servicio web de acuerdo con unas realizaciones a modo de ejemplo; y

10 la figura 3 ilustra un diagrama de flujo detallado para un sistema específico que modifica de forma automática una implementación de servicio en respuesta a cambios a un documento de contrato de servicio web de acuerdo con unas realizaciones a modo de ejemplo.

Descripción detallada

15 La presente invención se extiende a procedimientos, sistemas y productos de programa informático para modificar o actualizar de forma automática una implementación de servicio de acuerdo con cambios en un documento de contrato de servicio web. Las realizaciones de la presente invención pueden comprender un ordenador de propósito especial o de propósito general que incluye diversos módulos o soporte físico informático, tal como se analiza con mayor detalle en lo sucesivo.

20 Unas realizaciones a modo de ejemplo abordan los problemas que están asociados con cambios en contratos de servicio mediante la provisión de mecanismos automatizados para actualizar porciones de una implementación de servicio existente. Por ejemplo, un mecanismo modifica el esqueleto de una implementación (por ejemplo, procedimientos, firmas de procedimiento, estructuras de datos, etc.) con el fin de ajustarse a cambios a un contrato de NPD (por ejemplo, de WSDL). En una realización de este tipo, por lo general el desarrollador solo necesitará realizar los cambios necesarios, de haber alguno, en la lógica de negocio, lo que por lo general no será conocido a partir de la definición de contrato. Por consiguiente, tales modificaciones automáticas a la implementación facilitan la adopción por parte de un desarrollador de un enfoque basado en contrato al desarrollo de servicios web.

30 Por ejemplo, una realización a modo de ejemplo toma como entradas: una implementación de servicio; el tipo de clase o clases que son usadas por el servicio web; un contrato de servicio web; y/u otra información de entrada (por ejemplo, información de asignación para enlaces). Usando un contrato de servicio web especificado que es diferente del contrato implementado en la actualidad por el servicio web, la implementación actual se actualiza de tal modo que su clase, atributos de clase, procedimientos, atributos de procedimiento, firmas de procedimiento y diversas otras propiedades o estructuras de datos se ajustan al contrato especificado. Obsérvese, no obstante, que la implementación de servicio web actualizada se podría no compilar o ejecutar después de haberse ajustado. Por consiguiente, para completar el procedimiento puede que el desarrollador necesite inspeccionar los cambios realizados y hacer uno o más de lo siguiente: (1) proporcionar un código de cuerpo de procedimiento y/o lógica de negocio apropiados para procedimientos añadidos que se corresponden con operaciones presentes en el nuevo contrato pero ausentes en el contrato anterior; (2) modificar el código de cuerpo de procedimiento y/o lógica de negocio según se requiera para procedimientos cuya firma ha cambiado y que requieren un comportamiento nuevo o de cambio y/o hacen referencia a tipos nuevos y/o cambiados en la firma; o (3) opcionalmente, retirar procedimientos que ya no están asignados a operaciones como resultado de que se retiren del contrato unas operaciones previamente presentes.

45 Obsérvese que unas realizaciones en el presente documento prevén la preservación de porciones de los procedimientos (por ejemplo, la lógica de negocio escrita por el usuario) en la implementación actualizada, lo que puede incluir información valiosa o representar el resultado de un esfuerzo significativo que el usuario puede desear usar en algún otro contexto. Por ejemplo, cuando el cambio en la implementación resulta de que se renombre una operación, puede que simplemente sea necesario que el código de cuerpo de procedimiento preservado se mueva al procedimiento recién nombrado. Además, la preservación del procedimiento puede ser alguna forma de comentar el procedimiento o procedimientos, atributo o atributos de procedimiento, firma de procedimiento, etc. Además, otras realizaciones también prevén la generación de comentarios durante el procedimiento de ajustar la implementación al contrato actualizado, haciendo de este modo incluso más sencillo para el desarrollador identificar el trabajo que se requiere para completar la actividad de actualización de servicio global.

50 A pesar de que una referencia más específica a características ventajosas se describe con mayor detalle en lo sucesivo con respecto a las figuras, unas realizaciones dentro del alcance de la presente invención también incluyen medios legibles por ordenador para portar o tener instrucciones ejecutables por ordenador o estructuras de datos almacenadas en los mismos. Tales medios legibles por ordenador pueden ser cualquier medio disponible al que pueda acceder un ordenador de propósito general o de propósito especial. A modo de ejemplo y no de limitación, tales medios legibles por ordenador pueden comprender RAM, ROM, EEPROM, CD-ROM u otro almacenamiento en disco óptico, almacenamiento en disco magnético u otros dispositivos de almacenamiento magnético, o cualquier otro medio el cual se pueda usar para portar o almacenar unos medios de código de programa deseados en forma de instrucciones ejecutables por ordenador o estructuras de datos y a las cuales puede acceder un ordenador de propósito general o de propósito especial. Cuando se transfiere o se proporciona información a través de una red u

otra conexión de comunicaciones (o bien cableada, o bien inalámbrica o bien una combinación de cableada o inalámbrica) a un ordenador, el ordenador ve apropiadamente la conexión como un medio legible por ordenador. Por lo tanto, cualquier conexión de este tipo se denomina, apropiadamente, medio legible por ordenador. También se deberían incluir combinaciones de lo anterior dentro del alcance de los medios legibles por ordenador.

5 Las instrucciones ejecutables por ordenador comprenden, por ejemplo, instrucciones y datos que dan lugar a que un ordenador de propósito general, un ordenador de propósito especial o un dispositivo de procesamiento de propósito especial realice una determinada función o grupo de funciones. A pesar de que la materia objeto se ha descrito en un lenguaje específico de características estructurales y/o actos metodológicos, se ha de entender que la materia objeto que se define en las reivindicaciones adjuntas no se limita necesariamente a las características o actos
10 específicos que se han descrito en lo que antecede. Más bien, las características y actos específicos que se han descrito en lo que antecede se divulgan como formas a modo de ejemplo de implementar las reivindicaciones.

Tal como se usa en el presente documento, la expresión “módulo” o “componente” puede hacer referencia a objetos o rutinas de soporte lógico que se ejecutan en el sistema informático. Los diferentes componentes, módulos, motores y servicios que se describen en el presente documento se pueden implementar como objetos o procedimientos que se ejecutan en el sistema informático (por ejemplo, como subprocesos separados). A pesar de que el sistema y los procedimientos que se describen en el presente documento se implementan preferentemente en soporte lógico, también son posibles y se contemplan implementaciones en soporte físico o una combinación de soporte lógico y soporte físico. En la presente descripción, una “entidad informática” puede ser cualquier sistema informático tal como se ha definido previamente en el presente documento, o cualquier módulo o combinación de
15 módulos que se ejecuten en un sistema informático.

La figura 1A ilustra un sistema informático 100 para modificar de forma automática al menos una porción de una implementación de servicio web para ajustarse a un documento de contrato de servicio web actualizado. El sistema informático 100 muestra diversos módulos y componentes que se pueden usar cuando se implementa la realización. Estos diversos módulos y componentes se pueden combinar en un único módulo e implementarse usando soporte lógico, soporte físico o una combinación de los mismos. Además, estos módulos/componentes se pueden incluir en el mismo ordenador o los mismos se pueden distribuir en cualquier número de sistemas. Por supuesto, también se encuentran disponibles para las presentes realizaciones otras configuraciones y mecanismos para realizar la funcionalidad que es proporcionada por las realizaciones que se describen en el presente documento. Por
20 consiguiente, el sistema informático 100 se usa solo por razones ilustrativas y no tiene por objeto limitar o reducir de otro modo el alcance de las realizaciones en el presente documento.

Con independencia del tipo o configuración del sistema informático 100, el sistema informático 100 recibe o accede de otro modo a un primer documento de contrato o de descripción (que se usan en el presente documento de forma intercambiable) 110, el cual define cómo una implementación de servicio web se va a comunicar con servicios o puntos de extremo que se proporcionan en el mismo (obsérvese que la expresión “punto de extremo” tal como se usa en el presente documento puede incluir un cliente, un servicio o cualquier otra entidad configurada para comunicarse con un servicio de acuerdo con un contrato). Tal como se ha mencionado previamente, en algunas realizaciones, el primer documento de contrato 110 puede ser acordado y generado por uno o más desarrolladores 170 antes de la construcción de la implementación de servicio web 130. Se puede considerar que el desarrollador 170 es un desarrollador 170 individual o un equipo de dos o más desarrolladores 170 o empresas, no siendo importante la constitución o el número exacto para las realizaciones que se describen en el presente documento. Además, una cualquiera de las entidades que se han mencionado en lo que antecede (de forma individual o combinadas) puede generar los diversos documentos que se describen como generados por un desarrollador 170.
25

Con independencia de quién crea o de cómo se crea la descripción de servicio web 110, por lo general el primer contrato 110 describirá el comportamiento de uno o más servicios proporcionados. Por consiguiente, la primera descripción 110 describe, entre otras cosas, la operación u operaciones que se proporcionan, el formato de los mensajes que se intercambian, así como otras propiedades. Obsérvese que esta lista no es inclusiva ni exhaustiva. Por ejemplo, una operación de un servicio puede recibir un mensaje de entrada pero no devolver un mensaje de salida. Además, pueden existir otros comportamientos y objetos de datos dentro de la primera (y otra) descripción 110 que no se enumeran en el presente caso. Por consiguiente, la lista de comportamientos y objetos de datos que se proporcionan en el presente documento es solo por razones ilustrativas y no tiene por objeto limitar o reducir de otro modo el alcance de las realizaciones a menos que se reivindique de forma explícita.
30

Obsérvese también que los documentos de contrato de servicio web de la figura 1 (por ejemplo, el primer documento de contrato 110 y el segundo documento de contrato 140) se pueden designar como documentos de contrato de WSDL debido al uso generalizado en la técnica de tal formato. Tal como se ha mencionado previamente, no obstante, WSDL es solo un formato a modo de ejemplo para un Lenguaje de Descripción de Protocolos de Red (NPD, *Network Protocol Description Language*). Por consiguiente, la designación de WSDL no se debería usar para limitar el alcance de las realizaciones divulgadas o las reivindicaciones adjuntas a menos que se reivindique de forma expresa de otro modo.
35

Sin embargo, el contrato de servicio web 110 se puede usar para crear una implementación de servicio web 130. Por ejemplo, el módulo de construcción 120 se puede usar para tomar datos que están incluidos dentro de la primera
40

descripción de servicio web 110 para generar una implementación 130 específica. Tal como se muestra en la figura 1B, una implementación de servicio web 130 puede incluir el tipo o tipos de clase de servicio web 132 que son usados por el servicio, el cual tendrá por lo general diversos atributo o atributos de clase 131 asociados con el mismo. La clase de servicio web 132 define la implementación real 130 del servicio web, incluyendo los diversos
 5 procedimiento o procedimientos 133 que se ofrecen. Tal como se ha mencionado previamente, los servicios web definen el tipo de mensajes que estos aceptan usando operaciones según sea definido dentro del contrato de servicio (por ejemplo, el primer documento de contrato 110). Estas operaciones se correlacionan con cada uno del procedimiento o procedimientos de servicio web 133 dentro de una implementación de servicio web 130.

Cada procedimiento 133 tiene asociados con el mismo diversos atributo o atributos de procedimiento 134 y otras propiedades y estructuras de datos 137. Además, por lo general un procedimiento 133 también incluye la firma de procedimiento 135 que define cosas tales como el nombre de procedimiento, tipo de procedimiento, parámetro o parámetros de entrada esperados y/o tipo o tipos de retorno del procedimiento 133. Además, tal como se describirá con mayor detalle en lo sucesivo, por lo general la firma de procedimiento incluirá una lógica de negocio 136 según sea definido por el desarrollador 170. Por supuesto, otras propiedades tales como formato de mensajería (por ejemplo, SOAP), protocolos de enlace, etc., también se pueden definir dentro del procedimiento 133, los atributos de
 10 procedimiento 134, la firma de procedimiento 135, y/u otras propiedades o estructuras de datos 137.

A modo de ejemplo y no de limitación, una implementación de servicio web 130 creada usando el módulo de construcción 120 puede ser de un tipo de clase 132 "aritmético", el cual tiene diversos atributos de clase 131 y procedimientos 133. Por ejemplo, en el presente caso los procedimientos 133 pueden incluir "adición", "sustracción",
 20 "multiplicación", "división", etc. Los atributos de procedimiento 134, la firma de procedimiento 135, y otras propiedades o estructuras de datos 137 pueden definir adicionalmente para cada procedimiento 133 el nombre de una operación específica (por ejemplo, dividir), el tipo de procedimiento (por ejemplo, división), el número de enteros cuya introducción se requiere (por ejemplo, dos), el tipo de salida que resultará (por ejemplo, un único tipo INT), el formato tanto de entrada como de salida (por ejemplo, mensajes de SOAP), así como cualesquiera otras
 25 propiedades apropiadas.

Obsérvese que, por lo general, el procedimiento o procedimientos 133 de un tipo de clase 132 que implementa un servicio no se asignan de forma automática a una operación en la definición de servicio web (por ejemplo, el primer documento de contrato 110). Por ejemplo, las herramientas típicas (por ejemplo, un módulo de esqueleto 112) crean un código de esqueleto 115 que define, entre otras cosas, el procedimiento o procedimientos 133, el atributo o atributos de procedimiento 134, las firmas de procedimiento 135, los tipos de clase 132, etc., sin la lógica de negocio 136 necesaria para una implementación completa. En consecuencia, por lo general el desarrollador 170 necesitará exponer de forma manual (usando, por ejemplo, el módulo de entrada de lógica de negocio 165) qué procedimientos 132 se deberían exponer como solicitudes y respuestas de servicio en la definición de servicio. Además, el desarrollador 170 necesita completar el detalle de la lógica de negocio 136 para cada operación según sea definido por el primer contrato 110 para el servicio. Obsérvese, no obstante, que en algunas realizaciones, la etapa intermedia de crear el código de esqueleto 115 y generar entonces la implementación de servicio web 130 es
 30 opcional.

Tal como se menciona, el módulo de construcción 120 produce la implementación de servicio web 130. La implementación de servicio web 130 puede incluir las clases, procedimientos, atributos de procedimiento, firmas web, tipos de mensaje, etc. según sea indicado por el primer contrato de NPDL (por ejemplo, de WSDL) 110 y también puede incluir el código de negocio específico que implementa los comportamientos de servicio web que va a ofrecer. Cuando se compila, la implementación de servicio web 130 debería realizar los comportamientos indicados por el contrato de NPDL.
 40

Con independencia de cómo se produce la implementación de servicio web 130, tal como se ha mencionado previamente el primer contrato de servicio 110 puede cambiar después de que se haya creado la implementación de servicio web 130. Por ejemplo, un desarrollador 170 puede producir un segundo contrato de servicio 140 que actualiza, modifica o sustituye completamente de otro modo el primer contrato de servicio 110. Estos cambios pueden incluir, entre otras cosas, cambios a la operación u operaciones que se proporcionan, el formato de los mensajes que se intercambian, así como otras propiedades. Por consiguiente, los puntos de extremo que usan la implementación de servicio web 130 que se genera usando el primer contrato 110 puede que ya no sean capaces de comunicarse con los servicios que se especifican dentro del segundo contrato 140 (obsérvese, no obstante, que en algunos casos los cambios al primer contrato 110 no quieren decir necesariamente que un punto de extremo que usa la implementación de servicio original 130 no se pueda comunicar con los servicios que se proporcionan en el segundo contrato 140).
 50

Por ejemplo, en el ejemplo anterior de un tipo de clase "aritmético", un desarrollador puede cambiar posteriormente el número y/o el tipo de las entradas. Por ejemplo, supóngase que el primer contrato 110 permitió que un procedimiento de "adición" tuviera una serie de entradas de extremos abiertos. Un desarrollador 170 puede desear limitar el número de entradas por razones tales como reducir un riesgo de seguridad potencial propenso a cosas tales como una denegación de ataques de servicio, o por cualquier otra razón. Por consiguiente, el desarrollador 170 puede cambiar el primer contrato 110 (es decir, crear un segundo contrato 140) para que permita solo una serie de entradas limitada, por ejemplo, cinco argumentos numéricos explícitos. Si la implementación de servicio web 130
 60

actual se creó para tener la serie de extremos abiertos, entonces puede que entradas de más de cinco ya no sean capaces de comunicarse con el servicio de acuerdo con el segundo contrato 140.

5 Por lo tanto, unas realizaciones a modo de ejemplo proporcionan un módulo de ajuste 150 que se usa para evaluar de forma automática los cambios entre la implementación de servicio web 130 y el segundo contrato 140 y modificar al menos una porción de la implementación 130. Tal módulo 150 proporciona esencialmente un sistema de diferenciación que crea un nuevo conjunto de clase o clases a partir del segundo contrato 140 y compara la nueva clase o clases con la clase o clases existentes en la implementación 130, haciendo cambios según se requiera. Tales cambios aseguran, por ejemplo, que el esqueleto de implementación (por ejemplo, clase, atributos de clase, procedimientos, firmas de procedimiento, atributos de procedimiento, estructuras de datos, etc.) se ajusta al segundo contrato 140. Por ejemplo, el módulo de comparación puede tomar como entradas uno o más de: la implementación de servicio web 130; la clase o clases de tipo 135 que han sido usadas en la actualidad y/o previamente por el servicio; el contrato de servicio actualizado o segundo 140; y/o el contrato de servicio primero u original 110. Basándose en unos cambios identificados entre la implementación 130 (y/o el primer contrato 110 como puede ser el caso) y el segundo contrato 140, una o más porciones de la implementación de servicio web 130 se pueden modificar para generar una implementación de servicio web 160 en un intento de ajustar la implementación original 130 al segundo contrato 140.

20 Obsérvese que unas realizaciones a modo de ejemplo prevén que, por lo general, unas porciones de los procedimientos dentro de la implementación 130 (por ejemplo, la lógica de negocio) permanecerán sin cambios en la implementación modificada 160. Por ejemplo, tal como se describe con mayor detalle en lo sucesivo, la retirada de una operación puede dar lugar a que el procedimiento correspondiente se comente o se marque, se mueva, etc., de otro modo. Por consiguiente, la implementación modificada 160 puede no compilarse o ejecutarse. En ese sentido, con el fin de completar el procedimiento puede que el desarrollador 170 necesite usar el módulo de entrada de lógica de negocio 165 (o otro módulo) para hacer uno o más de lo siguiente: (1) proporcionar un código de cuerpo de procedimiento y/o lógica de negocio apropiados para procedimientos añadidos que se corresponden con operaciones presentes en el nuevo contrato pero ausentes en el contrato anterior; (2) modificar el código de cuerpo de procedimiento y/o lógica de negocio según se requiera para procedimientos cuya firma ha cambiado y que requieren un comportamiento nuevo o de cambio y/o hacen referencia a tipos nuevos y/o cambiados en la firma; o (3) opcionalmente, retirar procedimientos que ya no están asignados a operaciones como resultado de que se retiren del contrato unas operaciones previamente presentes.

30 Con independencia de si se necesita o no la lógica de negocio para la implementación modificada 160, las entradas que se usan para modificar la implementación de servicio 130 para ajustarse al segundo contrato 140 pueden variar. Por ejemplo, por lo general el módulo de ajuste 150 necesitará tomar como una entrada la implementación de servicio web 130. Las otras entradas, no obstante, pueden variar dependiendo de cómo se identifican los cambios. Por ejemplo, la clase o clases de tipo 135 a partir del segundo contrato de servicio 140 se pueden identificar y compararse directamente con la clase o clases dentro de la implementación de servicio web 130. Como alternativa, o además, el código de esqueleto 145 del segundo contrato de servicio 140 se puede generar usando un módulo de esqueleto 142, que se puede comparar entonces con el código de esqueleto 115 para el primer contrato 110 para identificar cambios al mismo.

40 Por supuesto, también se encuentran disponibles y se contemplan en el presente documento otras entradas para identificar los cambios entre la implementación de servicio web 130 y el segundo contrato de servicio 140. Por ejemplo, tal como se describe con mayor detalle en lo sucesivo, pueden existir casos en los que se necesita una entrada de usuario u otros parámetros para determinar diversas especificaciones que son necesarias para completar la implementación de servicio modificada 160. Por consiguiente, tal como se muestra en la figura 1A, unas realizaciones a modo de ejemplo proporcionan un módulo de entrada 147 que permite que un desarrollador 170 (o algún mecanismo automatizado dentro del módulo de entrada 147) defina otros parámetros (por ejemplo, especificando la asignación de un enlace particular) que son usados por el módulo de ajuste 150 para generar la implementación de servicio modificada 160. Por consiguiente, cualesquiera entradas particulares que son usadas por el módulo de ajuste 150 para identificar cambios entre la implementación 130 y el segundo contrato 140 se usan en el presente documento solo por razones ilustrativas.

50 Obsérvese también que prácticamente cualquier porción de la implementación 130 se puede modificar o actualizar de forma automática para ajustarse al segundo contrato de servicio 140. Por ejemplo, puede que los atributos de clase no hayan cambiado, no obstante, puede que el procedimiento, firmas de procedimiento, atributos de procedimiento u otras propiedades sí lo hayan hecho. Por consiguiente, unas realizaciones a modo de ejemplo están configuradas para identificar prácticamente cualquier cambio que tenga lugar en el esqueleto de implementación (es decir, tipos de clase, atributos de clase, procedimientos, atributos de procedimiento, firmas de procedimiento (incluyendo nombre, tipos, parámetros, lógica de negocio, etc.), estructuras de datos u otras propiedades). En ese sentido, cualquier porción específica de la implementación 130 modificada se usa en el presente documento solo por razones ilustrativas y no tiene por objeto limitar o reducir de otro modo el alcance de las realizaciones en el presente documento a menos que se reivindique de forma explícita.

60 Tal como se ha mencionado en lo que antecede, unas realizaciones en el presente documento prevén la preservación de porciones del procedimiento o procedimientos tales como la lógica de negocio escrita por el usuario

en la implementación actualizada o modificada 160, lo que puede incluir información valiosa o representar el resultado de un esfuerzo significativo que el usuario puede desear usar en algún otro contexto. Por ejemplo, cuando el cambio en la implementación 130 resulta de que se renombre una operación, simplemente es necesario que el código de cuerpo de procedimiento preservado se mueva al procedimiento recién nombrado. Obsérvese que la preservación de la lógica de negocio u otras porciones de un procedimiento puede tener lugar en cualquier número de formas. Por ejemplo, puede que el nombre de operación que cambió no dé lugar al renombramiento del procedimiento correspondiente. Además, la cancelación de una operación a partir del primer contrato 110 puede dar lugar al comentario, retirada de atributos, u otras marcas que indican que el procedimiento no se va a exponer como una operación del servicio web, o el marcado de otro modo del procedimiento anterior y su lógica de negocio contenida de tal modo que no se retira este.

Además, otras realizaciones también prevén la inserción de comentarios durante el procedimiento de generación de la implementación modificada 160. Por ejemplo, los comentarios pueden incluir una información tal como en qué fecha y hora tuvo lugar el cambio y/o qué dio lugar al cambio. Por supuesto, también se puede proporcionar otra información. Sin embargo, los comentarios adicionales ayudan adicionalmente al desarrollador 170 a identificar el trabajo que se requiere para completar la actividad de actualización de servicio global.

Con el fin de apreciar más completamente las realizaciones que se describen en el presente documento, la figura 3 proporciona un diagrama de flujo de un sistema específico que se puede usar en la puesta en práctica de las realizaciones que se describen en el presente documento. Obsérvese que la determinación de preguntas específicas puede tener lugar en un orden diferente del que se muestra en el diagrama. Además, las preguntas específicas que se usan también son solo por razones ilustrativas y no tienen por objeto limitar o reducir de otro modo el alcance de las realizaciones que se describen en el presente documento.

La lógica específica que se muestra en la figura 3 es, fundamentalmente, un sistema de diferenciación; este crea un nuevo conjunto de clases a partir del contrato de servicio y entonces compara las nuevas clases con las clases existentes en la implementación, haciendo cambios según se requiera. El flujo aborda diversos aspectos de los cambios a la implementación de servicio por turnos. Por ejemplo, en primer lugar este modifica los atributos en la clase de servicio web de tal modo que esta se ajusta a las especificaciones de servicio web en el contrato de servicio actualizado. El diagrama aborda entonces cada operación a la que hace referencia cada enlace de mensaje (por ejemplo, SOAP) y asegura que existe un procedimiento coincidente presente con un atributo de procedimiento web y retira de todos los procedimientos el atributo de procedimiento web. Este aborda entonces los tipos de esquema (por ejemplo, XML) a los que se hace referencia en las definiciones de mensaje y también, como una opción seleccionable por el usuario, crea un conjunto de sustitución de clases de serialización de datos o actualiza las clases existentes.

Tal como se muestra, el sistema determina en primer lugar si el espacio de nombres de enlace y/o nombre de enlace de servicio web de la implementación de servicio web es diferente del contrato de servicio web segundo/actualizado (el bloque de decisión 302). Si el espacio de nombres de enlace y/o nombre de enlace de servicio web es diferente (Sí en el bloque de decisión 302), entonces el sistema actualizará de forma automática 304 el espacio de nombres de enlace y/o nombre de enlace de servicio web. Obsérvese, no obstante, que por lo general el espacio de nombres/nombre de servicio no se debería actualizar debido a que un desarrollador puede haber decidido ofrecer el comportamiento bajo un nombre de servicio diferente incluso a pesar de que este conserve el espacio de nombres y el nombre de enlace. El espacio de nombres y el nombre de enlace son lo que define el tipo de comportamiento y se usan para la puesta en coincidencia con el contrato de servicio web segundo/actualizado, no el nombre de servicio.

Por ejemplo, supóngase que el espacio de nombres de enlace y el nombre de enlace cambiaron de los valores por defecto de WebService1 y <http://tempuri.org> a GolfCatalog y <http://catalog.org>. La siguiente modificación de código puede tener lugar de forma automática en la implementación de servicio web:

Antes de ajustarse al segundo contrato de servicio web:

```
[System.Web.Services.WebServiceBinding(Name = "WebService1")]
public class WebService1 : System.Web.Services.WebService
```

Después de ajustarse de forma automática al segundo documento de contrato de servicio web:

```
[System.Web.Services.WebServiceBinding(Name = "GolfCatalog", Namespace = "http://catalog.com")]
public class WebService1 : System.Web.Services.WebService
```

Obsérvese que en el ejemplo anterior, el nombre de servicio no cambió con la actualización en el espacio de nombres y el nombre de enlace.

Después de actualizar el espacio de nombres y el nombre de enlace de servicio web, entonces el sistema puede actualizar de forma automática 306 procedimientos de la implementación de servicio web que incluyen el antiguo nombre de enlace para incluir el nuevo nombre de enlace. Por ejemplo, la siguiente modificación de pseudo-código puede tener lugar de forma automática en la implementación de servicio web después de cambiar el nombre de

enlace “WebService1” a “GolfCatalog”:

Antes de ajustarse al segundo contrato de servicio web:

```
5 [System.Web.Services.WebMethod(...)]
  [System.Web.Services.Protocols.SoapDocumentMethod(Binding = WebService1")]
  public void getCatalog( )
```

Después de ajustarse de forma automática al segundo documento de contrato de servicio web:

```
10 [System.Web.Services.WebMethod(...)]
    [System.Web.Services.Protocols.SoapDocumentMethod(Binding = "GolfCatalog")]
    public void getCatalog( )
```

10 Después de actualizar los procedimientos con el nuevo nombre de enlace o si el espacio de nombres de enlace o nombre de enlace de servicio web no eran diferentes del segundo contrato de servicio (No en el bloque de decisión 302), el sistema actualiza atributos y firmas de procedimiento para ajustarse a las operaciones en el contrato de servicio segundo/actualizado para cada procedimiento en la implementación de servicio web que coincide con un nombre de operación en el contrato de servicio web. Para cada operación en el contrato segundo/actualizado, el sistema compara entonces 308 el procedimiento o procedimientos de implementación de servicio web con la operación u operaciones del contrato de servicio segundo/actualizado. Por ejemplo, los nombres de procedimiento se pueden comparar con los nombres de operación - suponiendo una correlación de enlaces biunívoca tal como se analiza con mayor detalle en lo sucesivo. Si existe una coincidencia (Sí en el bloque de decisión 316), el sistema determina entonces si el procedimiento está expuesto como una operación de servicio web (el bloque de decisión 310). Si existen uno o más procedimientos que no están expuestos como operaciones de servicio web (No en el bloque de decisión 310), los procedimientos se exponen de forma automática 312 mediante la adición de un atributo o atributos de procedimiento web al procedimiento no expuesto. Esto da cabida, por ejemplo, a un escenario de mejora para usuarios que han usado interfaces en las cuales se definen los atributos de procedimiento web y no han definido atributos de procedimiento web en la propia clase.

25 De lo contrario, si el procedimiento coincidente está expuesto (Sí en el bloque de decisión 310 y/o a partir de los procedimientos de servicio recién expuestos en el bloque 312), entonces se actualizan de forma automática 318 los atributos y firmas del procedimiento de implementación de los procedimientos web expuestos para ajustarse al contrato de servicio web segundo/actualizado. Obsérvese que el sistema puede no cambiar el nombre de mensaje/procedimiento y el atributo de procedimiento web.

30 Por ejemplo, si el contrato de servicio web segundo/actualizado expuso que la operación getCatalog tomó ahora ProductId como un parámetro, y devolvió un System.Xml.XmlDocument, las siguientes actualizaciones pueden tener lugar de forma automática:

Ejemplo 1

Antes de ajustarse al segundo contrato de servicio web:

```
35 [System.Web.Services.WebMethod(EnableSession = true)]
    [SoapDocumentMethod(...)]
    public void getCatalog( )
```

Después de ajustarse de forma automática al segundo documento de contrato de servicio web:

```
40 [System.Web.Services.WebMethod(EnableSession = true)]
    [SoapDocumentMethod(...)]
    public System.Xml.XmlDocument getCatalog(String ProductId)
```

Ejemplo 2

Antes de ajustarse al segundo contrato de servicio web:

```
45 [System.Web.Services.WebMethod(MessageName = "getCatalog", , EnableSession = true)]
    [SoapDocumentMethod(...)]
    public void getGolfCatalog( )
```

Después de ajustarse de forma automática al segundo documento de contrato de servicio web:

```
50 [System.Web.Services.WebMethod(MessageName = "getCatalog", , EnableSession = true)]
    [SoapDocumentMethod(...)]
    public System.Xml.XmlDocument getGolfCatalog(String ProductId)
```

El sistema también puede actualizar de forma automática atributos en la implementación de servicio web. Por

ejemplo, si el documento de contrato de servicio web segundo/actualizado expuso que el estilo del cuerpo debería ser en RPC y no Documento, entonces puede tener lugar la siguiente actualización:

Antes de ajustarse al segundo contrato de servicio web:

```
5 [System.Web.Services.WebMethod(EnableSession = true)]
[System.Web.Services.Protocols.SoapDocumentMethod(Binding = "WebService1")]
public void getCatalog( )
```

Después de ajustarse de forma automática al segundo documento de contrato de servicio web:

```
10 [System.Web.Services.WebMethod(EnableSession = true)]
[System.Web.Services.Protocols.SoapRpc-Method(Binding = "WebService1")]
public void getCatalog( )
```

15 Obsérvese que las firmas de procedimiento pueden incluir tipos de mensaje primitivos tales como cadena o entero, los cuales también están sujetos a cambios cuando se están modificando para ajustarse al documento de contrato de servicio web segundo/actualizado. Una forma de actualizar la implementación puede ser simplemente sustituir el tipo primitivo. Existen, no obstante, casos en los que el desarrollador puede desear preservar la singular funcionalidad del tipo primitivo. Por consiguiente, unas realizaciones permiten que el sistema añada un atributo al mensaje que no se incluirá como parte de la serialización de tal modo que este deja no expuesto el procedimiento. Esto ayuda a asegurar que la funcionalidad no se retira, sino que se encuentra disponible para su uso por parte de otro desarrollador mientras que no se implementa en una implementación de servicio web modificada.

20 Volviendo al bloque de decisión 316, si el nombre de operación en el contrato segundo/actualizado no coincide con un nombre de procedimiento (No en el bloque de decisión 316), el sistema expone de forma automática los procedimientos restantes (es decir, operaciones dentro del contrato y/o procedimientos actualizados dentro de la implementación original) mediante la adición de un procedimiento o procedimientos web con firmas a la implementación de servicio modificada (tal como se muestra en el bloque 326).

25 Una vez que se han expuesto todos los procedimientos dentro de la implementación modificada (es decir, que salen del bloque de decisión o bien 318 o bien 326), para cada procedimiento dentro de la implementación modificada, el sistema determina si el procedimiento o procedimientos expuestos existen en la implementación de servicio web modificada pero no en el documento de contrato de servicio web segundo/actualizado (el bloque de decisión 322). Si el procedimiento o procedimientos expuestos existen en la implementación de servicio web modificada y en el contrato segundo/actualizado, entonces el sistema termina 325 (No en el bloque de decisión 322). De lo contrario, si la operación expuesta sí existe en la implementación de servicio web modificada pero no en el documento de contrato de servicio web segundo/actualizado (Sí en el bloque de decisión 322), se preserva al menos una porción del procedimiento web. Por ejemplo, el procedimiento, atributo de procedimiento, firma de procedimiento u otra estructura de datos se puede comentar de forma automática de tal modo que el procedimiento ya no está expuesto como una operación de servicio web. Como alternativa, el procedimiento, atributo de procedimiento, firma de procedimiento, etc., se puede retirar y almacenar de otro modo en alguna otra parte para una referencia futura. Además, el sistema puede añadir un comentario adicional que expone cuándo y por qué el procedimiento ya no está expuesto. Esto asegura que el procedimiento web con las correspondientes firmas de procedimiento, parámetros y/o cualquier lógica de negocio específica no se retira de la implementación de servicio web, sino que simplemente no se compilará.

40 Por ejemplo, si la operación getCatalog se comentó de forma automática, puede tener lugar la siguiente modificación:

Antes de ajustarse al segundo contrato de servicio web:

```
45 [System.Web.Services.WebMethod(MessageName = "Catalog", EnableSession = true),
System.Web.Services.Protocols.SoapDocumentMethod(Binding = "WebService1")]
public void getCatalog( )
```

Después de ajustarse de forma automática al segundo documento de contrato de servicio web:

```
50 ///[7/2/2004 4:10 PM] El atributo WebMethod se retiró debido a que esta operación ya no se ajusta al lenguaje
WSDL.
///[System.Web.Services.WebMethod(MessageName = "Catalog", EnableSession = true),
[System.Web.Services.Protocols.SoapDocumentMethod(Binding = "WebService1")]
public void getCatalog( )
```

55 En algunas realizaciones, un desarrollador puede inspeccionar entonces la implementación de servicio web modificada y puede realizar modificaciones adicionales. Por ejemplo, cuando se compila o se ejecuta el servicio web modificado, pueden tener lugar uno o más errores. Para poner remedio a los errores, un desarrollador puede modificar la implementación tal como se ha descrito previamente.

Tal como se ha mencionado previamente, puede que el sistema anterior necesite parámetros de entrada adicionales o bien a partir de un usuario o bien a partir de otro mecanismo automatizado para definir adicionalmente qué procedimientos se deberían exponer en la implementación modificada. Por ejemplo, el sistema que se ha especificado en lo que antecede supuso un único par asignado específico de enlaces entre implementación y contrato - es decir, una correlación biunívoca. Los contratos (por ejemplo, documentos de WSDL), no obstante, pueden incluir múltiples enlaces (definiendo cada enlace un conjunto discreto de operaciones) cuando se asigna un contrato a una implementación. Por ejemplo, pueden existir casos en los que dentro de un contrato se soportan múltiples versiones; por lo tanto, el nombre o espacio de nombres podría incluir, por ejemplo, un identificador de versión (obsérvese que tales cambios al nombre y/o espacio de nombres son típicos cuando se hace cualquier cambio material al comportamiento o firma de cualquier operación en el enlace).

Por consiguiente, podrían tener lugar operaciones del mismo nombre en enlaces diferentes por lo que la asignación de nombres de operación sola puede ser insuficiente, debido a que un enlace está completamente identificado por propiedades de nombre y de espacio de nombres - las cuales se encuentran en el contrato y se registran como atributos en el código de implementación por lo que estas se pueden correlacionar. En el caso de que se definan en el contrato múltiples conjuntos de operaciones (enlaces), entonces el sistema puede requerir una entrada adicional si solo parte de los nombres (es decir, o bien el nombre o bien el espacio de nombres) de estos conjuntos han cambiado entre el primer y el segundo contrato. Tal entrada se podría proporcionar en forma de o bien entrada de usuario directa durante la ejecución de la modificación de implementación o bien como parámetros para la ejecución, los cuales pueden especificar la asignación.

Dicho de otra forma, puede que sea necesario que el sistema que se ha descrito en lo que antecede, así como las realizaciones que se describen en lo sucesivo, se ejecuten en el siguiente contexto: (1) un contrato y una implementación, cada uno con un único enlace (es decir, una correlación biunívoca), caso en el cual el enlace en el contrato y sus operaciones se usan para ajustar la implementación con independencia de los nombres y/o los espacios de nombres; (2) el contrato y/o la implementación tienen múltiples enlaces y el nombre y el espacio de nombres han de coincidir exactamente para que tenga lugar el ajuste (cualesquiera enlaces en el contrato pero no en la implementación se pueden añadir a la implementación o cualquier enlace en la implementación pero no en el contrato se debería retirar, por ejemplo, al comentar este); y (3) se han proporcionado instrucciones de asignación (a través de, por ejemplo, entrada de usuario o de otra forma automatizada), las cuales identifican de forma explícita qué enlaces en el contrato se deberían asignar a qué enlaces en la implementación (cualesquiera enlaces sin asignar se podrían ignorar o tratar como en (2)). Obsérvese, no obstante, que cómo se manejan los enlaces asignados y sin asignar se usa en el presente documento por razones ilustrativas y no debería limitar o reducir de otro modo el alcance de las realizaciones que se describen en el presente documento a menos que se reivindique de forma explícita.

Las realizaciones que se describen en el presente documento también se pueden describir en términos de procedimientos que comprenden etapas funcionales y/o actos no funcionales. Algunas de las siguientes secciones proporcionan descripciones de etapas y/o actos que se pueden realizar en la puesta en práctica de la presente invención. Por lo general, las etapas funcionales describen la invención en términos de los resultados que se logran, mientras que los actos no funcionales describen unas acciones más específicas para conseguir un resultado particular. A pesar de que las etapas funcionales y/o los actos no funcionales se pueden describir o reivindicar en un orden particular, la presente invención no se limita necesariamente a combinación u orden particular alguno de etapas y/o actos. Además, el uso de etapas y/o actos en la enumeración de las reivindicaciones - y en la siguiente descripción del diagrama de flujo para la figura 2 - se usa para indicar el uso específico deseado de tales términos.

Tal como se ha mencionado previamente, la figura 2 ilustra un diagrama de flujo para diversas realizaciones a modo de ejemplo de la presente invención. La siguiente descripción de la figura 2 hará referencia en ocasiones a elementos correspondientes a partir de las figuras 1A y B y 3. A pesar de que se puede hacer referencia a un elemento específico a partir de estas figuras, tales referencias se usan solo por razones ilustrativas y no tienen por objeto limitar o reducir de otro modo el alcance de las realizaciones descritas a menos que se reivindiquen de forma explícita.

Pasando a continuación a la figura 2, se ilustra un diagrama de flujo de un procedimiento 200 para modificar de forma automática una implementación de servicio web para ajustarse a cambios a un contrato de servicio web. El procedimiento 200 incluye una etapa para modificar de forma automática 210 una porción de una implementación de servicio web. Por ejemplo, el módulo de ajuste 150 se puede usar para modificar de forma automática una porción de la implementación de servicio web 130. Por lo general, tal modificación preservará una o más porciones de procedimientos (por ejemplo, la lógica de negocio) incluidos en la misma. Además, tal modificación puede incluir uno o más de lo siguiente: actualizar el espacio de nombres o el nombre de enlace de implementación de servicio web 304 si uno u otro son diferentes del segundo contrato de servicio web 140; preservar un procedimiento web 324 para una operación que no se incluye en el segundo documento de contrato de servicio web 140; añadir un procedimiento con atributos y firma de procedimiento para operaciones que se incluyen en el segundo contrato de servicio web pero no en la implementación de servicio web 326; modificar una firma de procedimiento para procedimientos web que coinciden con una operación 318; añadir un atributo a un procedimiento de tal modo que el procedimiento está expuesto como una operación; y/o retirar un atributo a un procedimiento de tal modo que el procedimiento ya no está expuesto como una operación, pero aún así se preserva el contenido del procedimiento.

5 La etapa para 210 incluye un acto de recibir 201 una implementación de servicio web que se ha construido de acuerdo con un primer documento de contrato de servicio web. Por ejemplo, la implementación de servicio web 130 puede ser recibida por el módulo de ajuste 150. La implementación de servicio web se puede construir de acuerdo con el primer documento de contrato de servicio 110. Tal como se ha descrito previamente, por lo general el primer contrato 110 describirá el comportamiento de uno o más servicios proporcionados. Por consiguiente, la primera descripción 110 describe, entre otras cosas, la operación u operaciones que se proporcionan, el formato de los mensajes que se intercambian, así como otras propiedades que definen cómo una implementación de servicio web 130 se va a comunicar con otros servicios del sistema informático 100.

10 La etapa para 210 incluye adicionalmente un acto de recibir 202 un segundo documento de contrato de servicio web que define un cambio o cambios al primer documento de contrato de servicio web. Por ejemplo, el módulo de ajuste 150 puede recibir el segundo contrato de servicio web 140, el cual puede incluir cambios a porciones del primer contrato de servicio 110. Tales cambios pueden cambiar la operación u operaciones que se proporcionan, el formato de los mensajes que se intercambian, así como otras propiedades que definen cómo una implementación de servicio web 130 se va a comunicar con otros servicios del sistema informático 100. Obsérvese que los cambios afectan al comportamiento para la implementación de servicio web 130 de una forma tal que la implementación de servicio web 130 probablemente no se puede comunicar con un punto de extremo (por ejemplo, un cliente) que espera un servicio que implementa el segundo contrato de servicio web 140. Además, obsérvese que tanto el primer 110 como el segundo 140 contrato pueden ser alguna forma de NPDL tal como WSDL.

20 La etapa para 210 incluye adicionalmente un acto de identificar 203 un cambio o cambios entre la implementación de servicio web y el segundo contrato de servicio web. Por ejemplo, el módulo de ajuste 150 o algún otro componente y/o módulo del sistema informático 100 puede identificar un cambio o cambios entre la implementación de servicio web 130 y el segundo contrato de servicio web 140. En algunas realizaciones, los cambios a una porción o porciones de la implementación de servicio web existente (por ejemplo, la implementación de servicio web 130) se identifican al comparar la implementación de servicio con una porción o porciones del segundo documento de contrato de servicio web (por ejemplo, el segundo documento de contrato de servicio 140). Por supuesto, en el presente documento se contemplan otras formas de identificar los cambios también tal como se ha descrito en lo que antecede. Por ejemplo, otras realizaciones pueden usar el código de esqueleto 145 que se genera basándose en el segundo contrato de servicio web 140 y comparar el mismo con la implementación de servicio web 130 o el primer contrato de servicio web 110 para identificar los cambios al mismo. En cualquier caso, una vez que se han identificado los cambios, el módulo de ajuste 150 puede generar la implementación de servicio web modificada 160 tal como se ha descrito previamente.

35 Otras realizaciones previeron adicionalmente exponer la porción modificada de la implementación de servicio web 160 usando el módulo o interfaz de entrada de lógica de negocio 165. Esto permite que un desarrollador 170 realice algo de lo siguiente: proporcionar un código de lógica de negocio apropiado para procedimientos añadidos a la implementación de servicio web 130 que se corresponden con operaciones presentes en el segundo documento de contrato de servicio web 140 pero ausentes en el primer documento de contrato de servicio web 110; modificar el código de lógica de negocio existente para procedimientos cuyas firmas o tipos de mensaje se han modificado para ajustarse al segundo documento de contrato de servicio web 140; y retirar procedimientos web que ya no están asignados a operaciones como resultado de las operaciones que se encuentran presentes en el primer documento de contrato de servicio web 110, pero ausentes en el segundo documento de contrato de servicio web 140.

40 Las realizaciones descritas se han de considerar a todos los efectos solo como ilustrativas y no restrictivas. Por lo tanto, el alcance de la invención es indicado por las reivindicaciones adjuntas en lugar de por la descripción anterior. Todos los cambios que entren dentro del significado y el rango de equivalencia de las reivindicaciones se han de incluir dentro de su alcance.

45

REIVINDICACIONES

- 5 1. Dentro de un sistema informático que incluye una implementación de servicio de un documento de contrato de servicio web, el cual define cómo la implementación de servicio se va a comunicar con un servicio particular, un procedimiento de cambiar de forma automática al menos una porción de la implementación de servicio en respuesta a cambios en el documento de contrato de servicio web, comprendiendo el procedimiento:
- un acto de recibir (201) una implementación de servicio web que se ha construido de acuerdo con un primer documento de contrato de servicio web que describe cómo la implementación de servicio web se va a comunicar con uno o más servicios;
- 10 un acto de recibir (202) un segundo documento de contrato de servicio web, el cual define uno o más cambios al primer contrato de servicio web que afectan al comportamiento de la implementación de servicio web de una forma tal que la implementación de servicio web no se puede comunicar con uno o más puntos de extremo que esperan que se implemente el segundo contrato;
- 15 un acto de identificar (203) los uno o más cambios entre la implementación de servicio web y el segundo documento de contrato de servicio web, al comparar (308), para cada operación del segundo documento de contrato de servicio, un nombre de la operación con nombres de procedimientos de la implementación de servicio web y determinar (316) si el nombre de la operación coincide con un nombre de procedimiento en la implementación de servicio web; y
- 20 basándose en los uno o más cambios identificados, una etapa para modificar de forma automática al menos una porción de la implementación de servicio web para ajustarse al menos parcialmente al segundo documento de contrato de servicio web, que incluye, para cada operación del segundo documento de contrato de servicio:
- cuando el nombre de operación en el segundo contrato coincide con un nombre de procedimiento en la implementación de servicio web, determinar (310) entonces si el procedimiento está expuesto como una operación de servicio web; y
- 25 cuando el procedimiento no está expuesto como una operación de servicio web, exponer (312) entonces el procedimiento mediante la adición de un atributo de procedimiento web al procedimiento no expuesto.
2. El procedimiento de acuerdo con la reivindicación 1, en el que el acto de identificar los uno o más cambios a la implementación de servicio web comprende:
- comparar una o más porciones de la implementación de servicio web con una o más porciones del segundo documento de contrato de servicio web.
- 30 3. El procedimiento de acuerdo con la reivindicación 1, en el que la modificación de la al menos una porción de la implementación de servicio web preserva una o más porciones de uno o más procedimientos incluidos en la misma.
4. El procedimiento de acuerdo con la reivindicación 3, en el que las una o más porciones de uno o más procedimientos incluidos dentro de la implementación de servicio web se preservan al comentar estas de forma automática.
- 35 5. El procedimiento de acuerdo con la reivindicación 1, que comprende adicionalmente:
- generar un código de esqueleto basándose en el segundo documento de contrato de servicio web;
- comparar la implementación de servicio web con el código de esqueleto; y
- basándose en la comparación, modificar de forma automática al menos una porción de la implementación de servicio web para ajustarse al código de esqueleto.
- 40 6. El procedimiento de acuerdo con la reivindicación 1, en el que modificar de forma automática al menos una porción de la implementación de servicio web para ajustarse al segundo documento de contrato de servicio web incluye:
- actualizar un espacio de nombres de enlace o nombre de enlace de implementación de servicio web si uno u otro son diferentes del segundo contrato de servicio web.
- 45 7. El procedimiento de acuerdo con la reivindicación 1, en el que modificar de forma automática al menos una porción de la implementación de servicio web para ajustarse al segundo documento de contrato de servicio web incluye adicionalmente:
- añadir un procedimiento con atributos y firma de procedimiento para operaciones que se incluyen en el segundo contrato de servicio web pero no en la implementación de servicio web.
- 50 8. El procedimiento de acuerdo con la reivindicación 1, en el que modificar de forma automática al menos una porción de la implementación de servicio web para ajustarse al segundo documento de contrato de servicio web incluye:
- modificar una firma de procedimiento para procedimientos web que coinciden con una operación.

9. El procedimiento de acuerdo con la reivindicación 1, en el que el primer y el segundo documentos de contrato de servicio web son documentos de Lenguaje de Descripción de Servicios Web, WSDL.

10. El procedimiento de acuerdo con la reivindicación 1, que comprende adicionalmente:

5 exponer la al menos una porción modificada de la implementación de servicio web usando una interfaz de entrada de lógica de negocio con el fin de permitir que un desarrollador realice al menos uno de:

proporcionar una lógica de negocio apropiada para procedimientos web añadidos a la implementación de servicio web que se corresponden con operaciones presentes en el segundo documento de contrato de servicio web pero ausentes en el primer documento de contrato de servicio web;

10 modificar la lógica de negocio para procedimientos web cuyo tipo de procedimiento o firma se ha modificado para ajustarse al segundo documento de contrato de servicio web; y

retirar procedimientos web que ya no están asignados a operaciones como resultado de las operaciones que se encuentran presentes en el primer documento de contrato de servicio web pero ausentes en el segundo documento de contrato de servicio web.

11. El procedimiento de la reivindicación 1, en el que modificar de forma automática al menos una porción de la implementación de servicio web para ajustarse al segundo documento de contrato de servicio web incluye:

15 retirar un atributo a un procedimiento de tal modo que el procedimiento no se retira de la implementación de servicio web, pero el procedimiento tampoco está expuesto como una operación.

12. Un producto de programa informático para su uso en un sistema informático que incluye una implementación de servicio web de un documento de contrato de servicio web, el cual define cómo la implementación de servicio web se va a comunicar con un servicio particular, usado el producto de programa informático en la implementación de un procedimiento de cambiar de forma automática al menos una porción de la implementación de servicio web en respuesta a cambios en el documento de contrato de servicio web, comprendiendo el producto de programa informático uno o más medios legibles por ordenador que tienen instrucciones ejecutables por ordenador almacenadas en los mismos que, cuando se ejecutan por uno o más procesadores del sistema informático, dan lugar a que el sistema informático realice el procedimiento de una de las reivindicaciones 1 a 11.

25

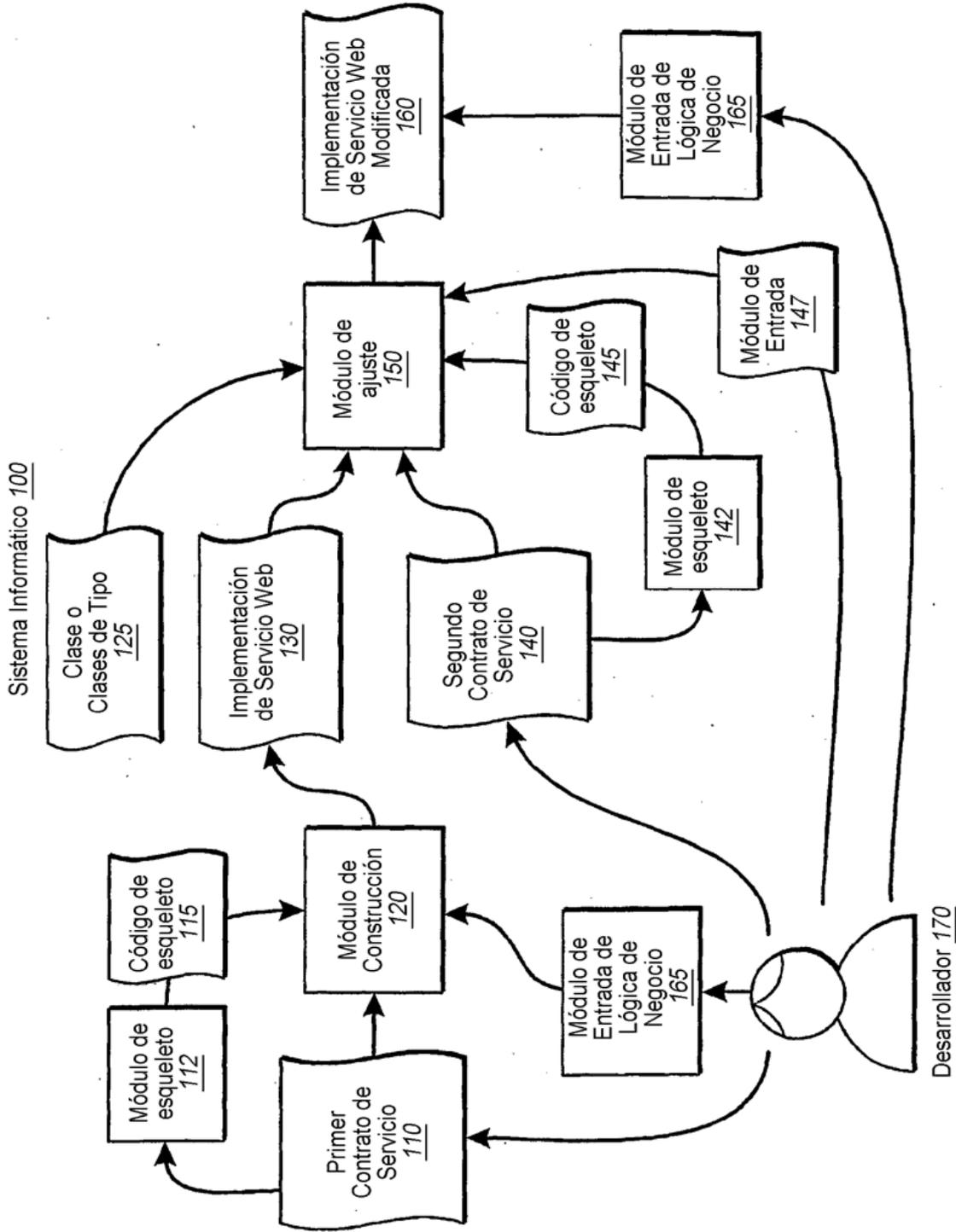


Fig. 1A

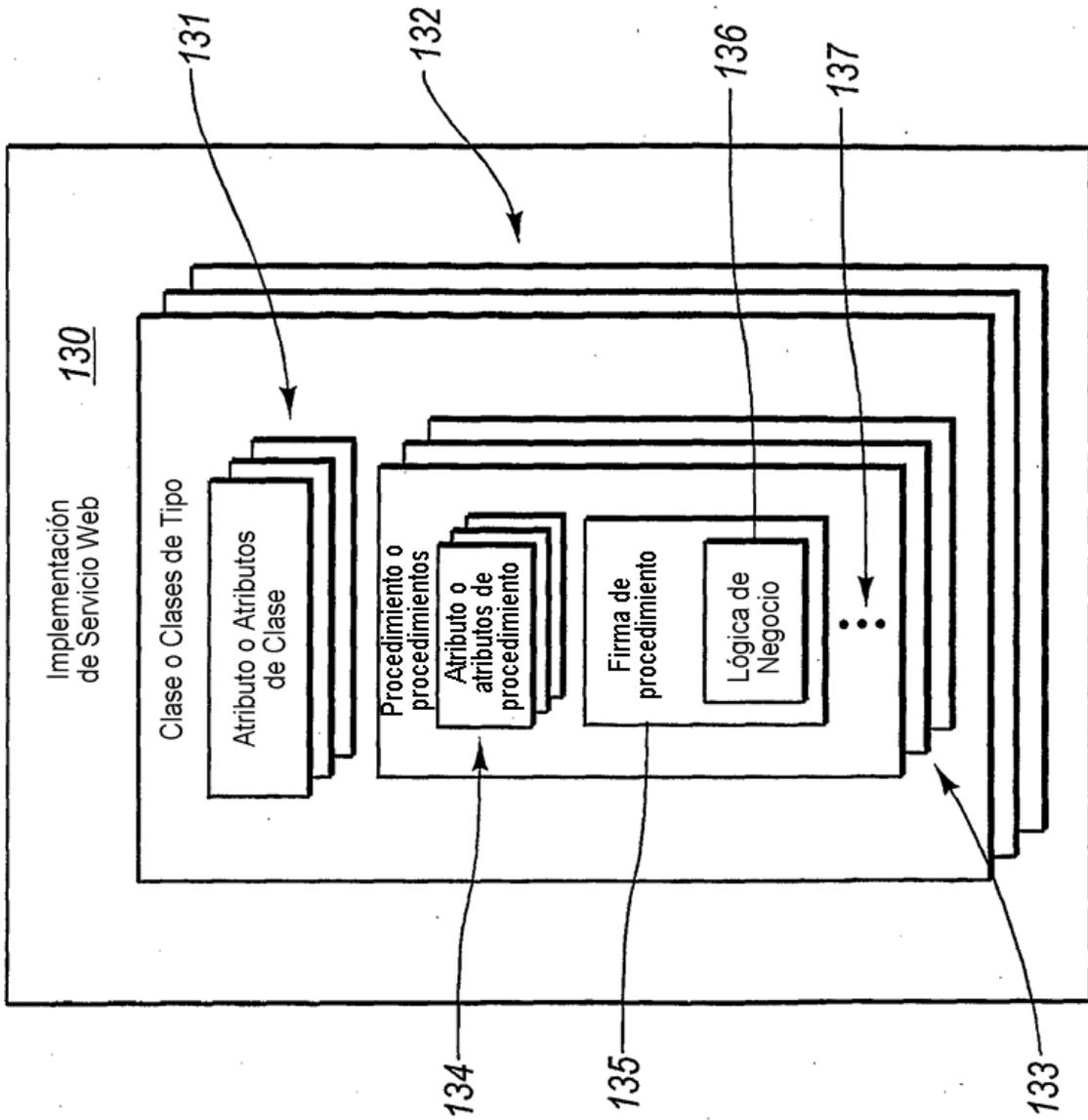


Fig. 1B

200

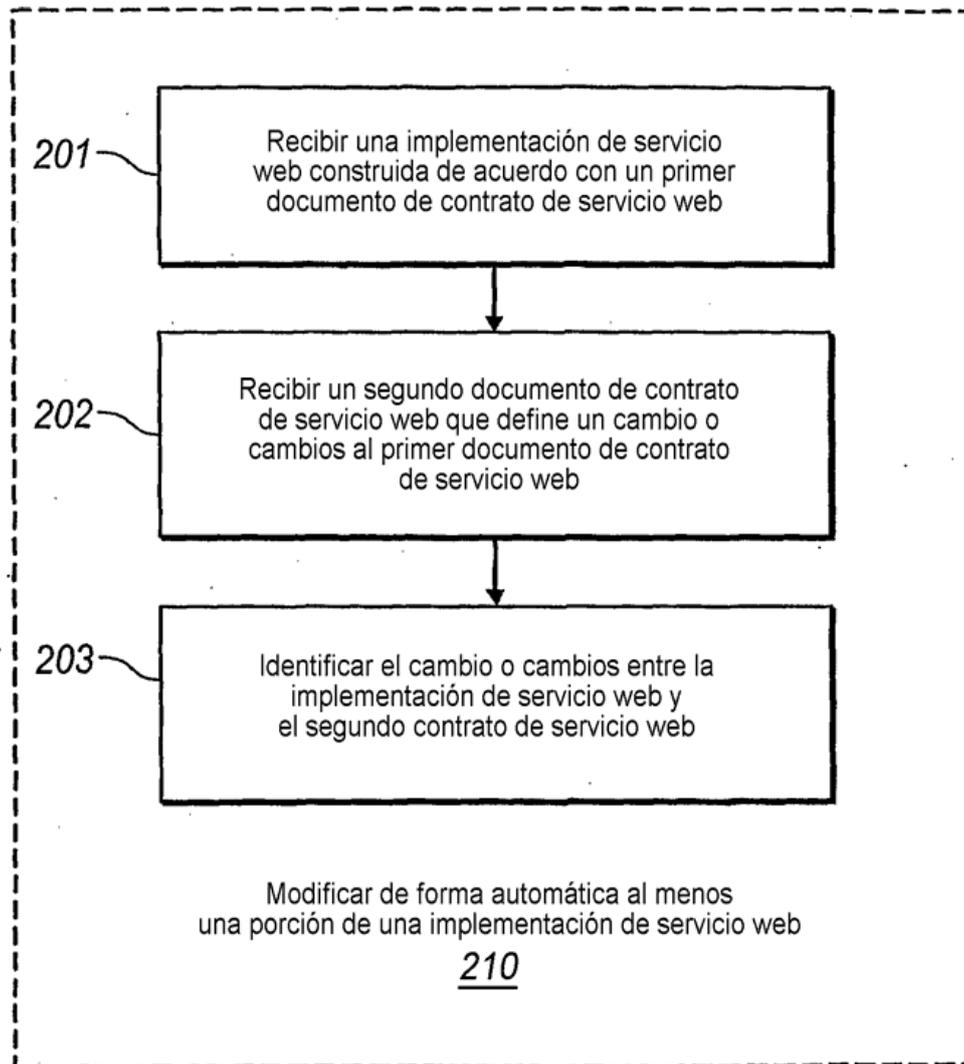


Fig. 2

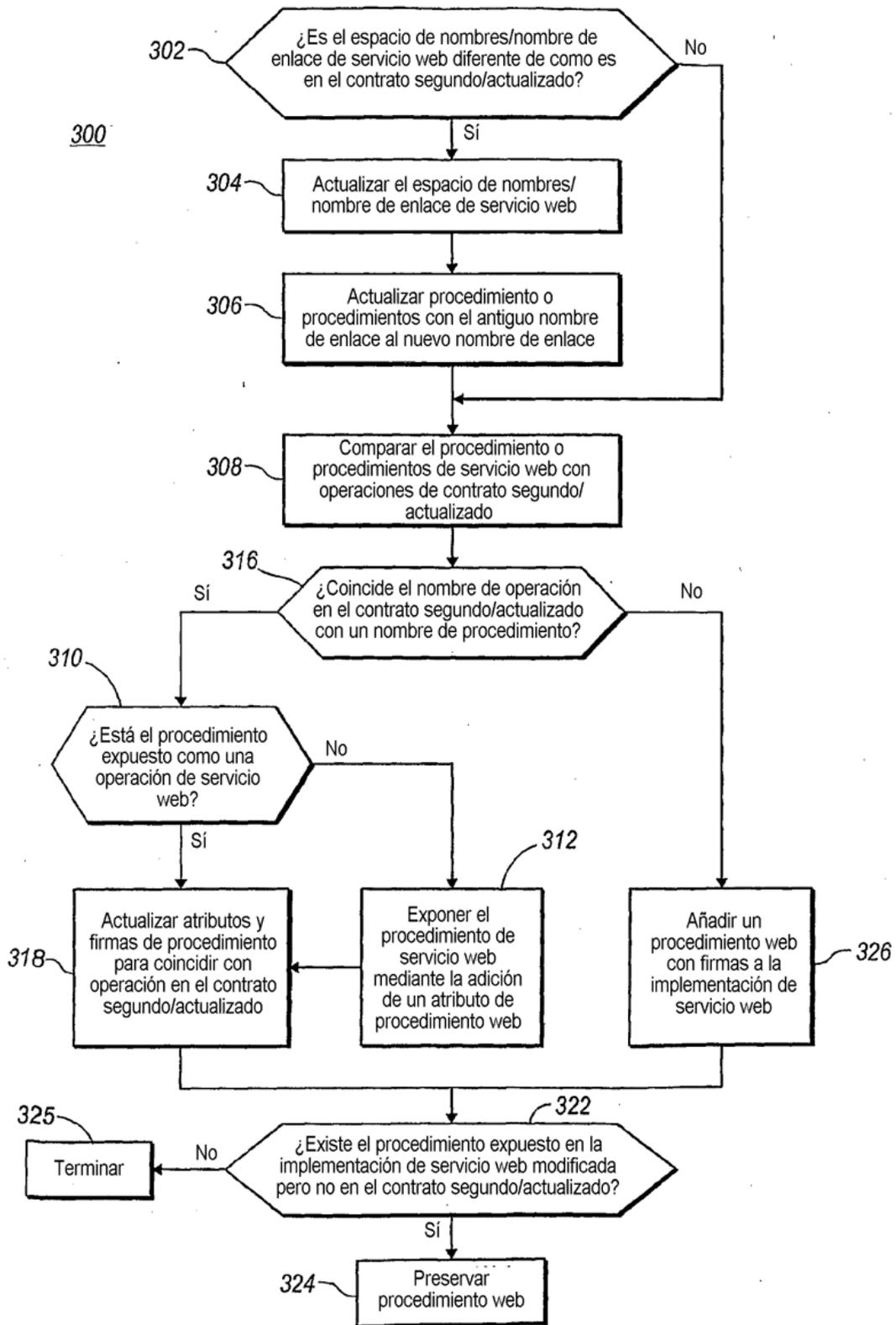


Fig. 3