



19



OFICINA ESPAÑOLA DE
PATENTES Y MARCAS

ESPAÑA

11 Número de publicación: **2 315 337**

51 Int. Cl.:

G06F 9/44 (2006.01)

G05B 19/05 (2006.01)

G06Q 10/00 (2006.01)

12

TRADUCCIÓN DE PATENTE EUROPEA

T3

96 Número de solicitud europea: **02078612 .5**

96 Fecha de presentación : **03.09.2002**

97 Número de publicación de la solicitud: **1296232**

97 Fecha de publicación de la solicitud: **26.03.2003**

54 Título: **Método y aparato para la programación de controladores programables y para generar datos de configuración desde un servidor centralizado.**

30 Prioridad: **24.09.2001 US 961931**

45 Fecha de publicación de la mención BOPI:
01.04.2009

45 Fecha de la publicación del folleto de la patente:
01.04.2009

73 Titular/es: **Siemens Energy & Automation, Inc.**
3333 Old Milton Parkway
Alpharetta, Georgia 30005-4437, US

72 Inventor/es: **Lo, George**

74 Agente: **Carvajal y Urquijo, Isabel**

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

Método y aparato para la programación de controladores programables y para generar datos de configuración desde un servidor centralizado.

Campo de la invención

La presente invención trata en general del campo de los controladores programables. En particular, la presente invención presenta un aparato y un método para escribir código de programación y/o generar datos de configuración para estos controladores programables sin la necesidad de adquirir ni instalar una herramienta de ingeniería en un dispositivo de programación.

Antecedentes de la invención

Los controladores programables, incluidos los Controladores Lógicos Programables (PLC, por sus siglas en inglés), se utilizan en muchas aplicaciones comerciales e industriales. Los controladores programables normalmente monitorizan entradas, toman decisiones basándose en cómo están programados y controlan las salidas de los procesos automatizados o de las máquinas. Uno de los controladores programables más comunes en uso es el PLC. Los PLC consisten en módulos o puntos de entrada, una unidad de procesamiento central (CPU) y puntos o módulos de salida. Una entrada acepta una variedad de señales digitales o analógicas de varios dispositivos de campo, como sensores, y los convierte en una señal lógica que puede utilizar el CPU. El CPU toma decisiones y ejecuta instrucciones de control basándose en instrucciones de programación almacenadas en una memoria. Estas instrucciones de programación determinan qué hará el PLC para una entrada específica. Los módulos de salida convierten las instrucciones de control desde el CPU en una señal digital o analógica que se puede utilizar para controlar varios dispositivos de campo, tales como actuadores o válvulas.

Ya que la mayoría de los controladores programables, incluidos los PLC, son en esencia ordenadores, almacenan información en forma de estados off y on (es decir, unos o ceros), a las que se conoce como dígitos binarios (es decir, bits). Un programa para un controlador programable consta de una o más instrucciones que realizan una tarea. Programar un PLC u otro controlador es cuestión de construir una serie de instrucciones. Programar implica muchas veces generar datos de configuración. Configurar un controlador programable implica asignar el área de entrada/salida (I/O) del controlador a la I/O física. Los editores de configuración son generalmente gráficos.

Existen varias maneras de abordar un programa, como por ejemplo, mediante la programación por diagrama de flujo, lenguaje Ladder (o diagrama de escalera), listas de instrucciones o diagramas de bloque de funciones. El lenguaje Ladder ("LD") es un lenguaje de programación utilizado con los PLC. Como se muestra en la figura 1, el código de lenguaje Ladder (10) utiliza símbolos gráficos que se asemejan a elementos electromecánicos utilizados en un formato de diagrama lógico de relé para describir un control cableado. La línea vertical izquierda de un diagrama típico en lenguaje Ladder generalmente representa un conductor activo o de potencia. Una línea vertical derecha representa la trayectoria de regreso de un diagrama de línea de un control cableado y puede ser omitida. Los diagramas Ladder (o diagramas de escalera) se leen de izquierda a derecha, de arriba abajo. A menudo, los peldaños se conocen como redes. Una red puede tener varias instrucciones de entrada y salida. Las instrucciones de entrada representadas por una serie de contactos en una o más ramas paralelas, ejecutan comparaciones o pruebas y están normalmente justificadas a la izquierda en el peldaño. Las instrucciones de salida representadas por espirales para las que puede haber sólo una en cada rama de salida, ejecutan algunas operaciones o funciones, y normalmente están justificadas a la derecha en el peldaño. Como se muestra en el código Ladder del ejemplo (10) representado en la figura 1, 10.0, 10.1 y Q0.0 representan la primera combinación de instrucción. Si se activan las entradas 10.0 y 10.1, se activa la salida Q0.0. Las entradas pueden ser conmutadores, botones pulsadores o cierres de contacto. Las salidas pueden ser, por ejemplo, un solenoide o una bombilla. 10.4, 10.5 y Q0.1 representan una segunda combinación de instrucción. Si las entradas 10.4 o la 10.5 están activas, la salida Q0.1 produce una activación, esto es, activa.

Una lista de instrucción (IL) proporciona otra visión de una serie de instrucciones y se ejemplifica (20) en la figura 1. La operación, es decir, lo que se va a hacer, se muestra a la izquierda. El operando, el elemento al que se aplica la operación, se muestra a la izquierda. El LD y la IL poseen una estructura similar. La serie de instrucciones en la IL (20) de la figura 1 ejecuta las mismas tareas que el LD de (10).

Los diagramas de funciones (FBD, por sus siglas en inglés) proporcionan otra visión de una serie de instrucciones y se ejemplifican (30) en la figura 1. Cada función tiene un nombre para designar su tarea específica. Las funciones se indican por medio de un rectángulo. Las entradas se muestran en el lado izquierdo del rectángulo y las salidas se muestran en el lado derecho. El diagrama de funciones (30) mostrado en la figura 3 realiza las mismas tareas que muestra la LD (10) de la figura 1 y la IL (20) de la figura 1.

Los controladores programables en general, y los PLC en particular, ejecutan un código de programa en un proceso repetitivo al que se conoce como un escaneo o exploración. Un escaneo o exploración puede empezar con la lectura de un estado de entradas por parte del CPU. El programa de aplicación se ejecuta utilizando el estado de entradas. Una vez que el programa se completa, la CPU realiza diagnósticos internos y tareas de comunicación. Los ciclos de escaneo o exploración finalizan con la actualización de las salidas y entonces vuelven a empezar. El tiempo del ciclo depende del programa, del número de entradas y salidas y de la cantidad de comunicación necesaria.

Para poder escribir instrucciones de programación y generar datos de configuración y descargar el código y los datos de configuración en un controlador programable, se necesitan varias herramientas. Como se muestra en la figura 1, un dispositivo de programación, como por ejemplo un ordenador personal (1), se interconecta con un PLC (7). Normalmente, un cable propietario (5), como por ejemplo Siemens® PC/PPI, conecta el puerto RS 232 del ordenador con el PLC 7. Previamente a la presente invención, las herramientas de software de ingeniería (3), como Siemens STEP7®, tenían que instalarse en el PC de manera que éste se pudiera utilizar para escribir instrucciones de programación para el PLC. Normalmente, las herramientas de ingeniería se venden en CDs o en otro medio de lectura informático.

La figura 2 muestra una idea general de los pasos normales y de los déficits que resultan de la adquisición de una copia de una herramienta de ingeniería. El cliente de una herramienta de ingeniería compra normalmente una copia del software y obtiene una licencia para utilizar la herramienta, paso (20). El cliente tiene entonces que instalar el software en un solo ordenador del cliente, paso (25). El cliente sólo puede desarrollar el software de aplicación, es decir, el código de programación para un controlador programable en ordenadores que incluyen la herramienta de ingeniería, paso (27). Los acuerdos de licencia que acompañan normalmente a las herramientas de ingeniería limitan la capacidad del cliente para instalar la herramienta en más de un ordenador sin pagar una licencia para ordenadores adicionales.

El software o las herramientas de ingeniería, tales como Siemens STEP7® o MicroWin®, son a menudo herramientas propietarias desarrolladas por el fabricante del controlador. Normalmente, el desarrollo de estas herramientas de ingeniería cuesta miles de horas-persona. Las herramientas se diseñan y se prueban con frecuencia para trabajar en un sistema operativo informático específico, como por ejemplo Microsoft Windows®98. Cuando cambia el sistema operativo para el ordenador en el que se está utilizando la herramienta, se tiene que volver a verificar la herramienta. A menudo, los vendedores de PC entregan sólo la última versión del sistema operativo de Microsoft en sus PC. Esto fuerza a los vendedores de herramientas de ingeniería a que también ofrezcan soporte técnico para el nuevo sistema operativo, lo cual supone normalmente una inversión de cientos o miles de horas-persona adicionales. En muchas organizaciones, se actualiza el sistema operativo del PC sin tomar en consideración el software, tales como las herramientas de ingeniería, que se encuentran en los PC.

Las herramientas de ingeniería se actualizan con mucha frecuencia con el paso del tiempo. En consecuencia, pueden existir al mismo tiempo versiones diferentes de las herramientas. En una instalación de fabricación grande es muy probable que no todas las herramientas de programación utilicen la misma versión. Esto no sólo aumenta los costes de propiedad, sino que, además, puede provocar problemas al utilizar diferentes dispositivos de programación para escribir el código de programación para la misma aplicación de PLC. Con frecuencia, se asigna un equipo de ingenieros para programar un PLC. Cada ingeniero del equipo puede trabajar independientemente en un aspecto específico de la aplicación. Más tarde, estos aspectos específicos se combinan en un programa que se ejecuta en el PLC. Si todos los ingenieros no están utilizando la misma versión de la herramienta, es posible que el código generado desde una versión de la herramienta pueda no ser compatible con una versión anterior de esa herramienta.

Además de los problemas asociados con la ejecución de sistemas operativos diferentes en los dispositivos de programación y versiones diferentes de las herramientas de ingeniería en los sistemas operativos, el código de programación para los controladores programables a menudo no se archiva de manera centralizada. Con frecuencia, el código para un PLC en una fábrica puede almacenarse en un portátil o en un ordenador personal de sobremesa y el código para otro PLC se puede almacenar en cualquier otro sitio. Si un PLC fue programado inicialmente con un primer PC y ese PC es reemplazado más tarde por un segundo PC, y si el PLC es reprogramado con un segundo PC, a menudo no habría forma de restablecer el programa original en el caso de que el nuevo programa fuera insuficiente.

WO 01/14963 describe el uso de un programa gráfico en un servidor central para superar los problemas de usuarios informáticos que no están familiarizados con la programación basada en texto.

WO 01/69335 describe la programación de un controlador lógico programable (PLC) utilizando un paquete de software que se encuentra en el PLC, pero al que se accede mediante Internet.

US6151609 describe la edición remota de un archivo almacenado en un servidor remoto utilizando un navegador Web.

Resumen de la invención

De acuerdo con un primer aspecto de la presente invención, un sistema consta de un servidor, una red, al menos un dispositivo de cliente que posee una aplicación de interfaz de red, en donde el servidor está en comunicación con al menos un dispositivo de cliente a través de la red; una herramienta de ingeniería que reside en el servidor y que es accesible por al menos un dispositivo de cliente mediante la red, en donde la herramienta de ingeniería se ejecuta en la aplicación de interfaz de red en al menos un dispositivo de cliente, una memoria asociada al servidor; y al menos un controlador programable; caracterizado porque el servidor es un servidor central; caracterizado porque la herramienta de ingeniería es una herramienta para generar código de programación para controladores programables que utilicen un lenguaje de programación; y caracterizado porque el código de programación y los datos de configuración generados con la herramienta de ingeniería se almacenan en la memoria asociada al servidor central y se descargan a un controlador programable de destino.

De acuerdo con un segundo aspecto de la presente invención, un método para crear código de programación de un controlador programable en un sistema que consta de un servidor, una red; al menos un dispositivo de cliente; una herramienta de ingeniería que reside en el servidor y que es accesible por al menos un dispositivo de cliente, dispositivo que posee una aplicación de interfaz de red; y el sistema que consta además de una memoria y de, al menos, un controlador programable; el método que consta de acceso a la herramienta de ingeniería con al menos un dispositivo de cliente; ejecutándose la herramienta de ingeniería en una aplicación de interfaz de red instalada en al menos un dispositivo de cliente, donde el servidor está en comunicación con al menos un dispositivo de cliente mediante la red, y donde la memoria está asociada al servidor; caracterizado porque el servidor es un servidor central y porque la herramienta de ingeniería es una herramienta para generar código de programación para controladores programables que utilizan un lenguaje de programación; método que consta además de la generación de código de programación y datos de configuración para los controladores programables que utilicen la herramienta de ingeniería; almacenando el código de programa generado en la memoria asociada al servidor central; y descargando el código de programación a un controlador programable de destino.

La presente invención proporciona un sistema para la programación de un controlador programable, tales como un PLC u otro controlador industrial. En una ejecución, un dispositivo de cliente que posee una aplicación de interfaz de red, tal como, por ejemplo, una aplicación de navegación, se comunica a través de la red con un servidor. En el servidor reside una herramienta de ingeniería para crear el código de programación y los datos de configuración para los controladores programables. La herramienta de ingeniería emplea un lenguaje de programación para los controladores programables de programación y es accesible por el dispositivo de cliente utilizando la aplicación de navegación que reside en ella. El servidor puede tener asociada una unidad de memoria para almacenar el código de programación generado por la herramienta de ingeniería en el dispositivo de cliente. En algunas ejecuciones, la herramienta de ingeniería emplea un lenguaje de programación gráfico como el lenguaje Ladder (o diagrama de escalera), programación por diagramas de flujo o programación por diagramas de funciones. En otras ejecuciones, puede utilizar lenguajes basados en texto, tales como listas de instrucciones. Se puede encontrar también en el servidor un compilador para la compilación del código de programación o puede ser accesible para el servidor.

En consecuencia, la presente invención proporciona métodos numerosos y robustos para desarrollar código de programación para controladores programables. En una ejecución, un usuario accede a un servidor a través de una red con un dispositivo de cliente que posee una aplicación de navegación. La aplicación de navegación invoca preferiblemente la herramienta de ingeniería. El usuario crea, entonces, un documento que contiene instrucciones de programación con la herramienta de ingeniería. El documento se transmite a través de la red al servidor. El documento puede ser almacenado en el servidor para su posterior acceso y edición. En una ejecución particular, el documento es un documento XML.

En una ejecución de la presente invención, el código de programación se almacena en un servidor central. Esto proporciona un método eficiente y rentable para la programación de controladores programables. El servidor y el controlador programable pueden interconectarse para comunicarse entre sí a través de la red. Tras interconectarlos, puede descargarse el código de programación compilado desde el servidor al controlador a través de la red.

La presente invención ofrece un nuevo paradigma en la manera en la que las entidades abordan la programación de sus controladores programables. Las entidades que deseen programar sus controladores programables no necesitan ni la adquisición ni el mantenimiento de herramientas de ingeniería de elevado coste. En lugar de ello, se puede habilitar la utilización en la web de una herramienta de ingeniería y su mantenimiento se puede realizar en un servidor central al que se accede a través de una red informática. Los usuarios que posean dispositivos de cliente están provistos de acceso al servidor a través de la red. El servidor recibe del dispositivo de cliente un documento que contiene el código de programación creado en el dispositivo de cliente con la herramienta de ingeniería. El código se puede compilar en el servidor y se puede descargar al controlador programable. Puede ser descargado directamente al controlador si el dispositivo es accesible a través de la red de informática, o se puede descargar al dispositivo a través de un dispositivo de cliente que resida en la red y que esté interconectado con el controlador programable.

Breve descripción de los dibujos

La figura 1 es una ilustración de las herramientas de la técnica anterior necesarias para programar controladores programables.

La figura 2 es un diagrama de bloque que ilustra los pasos y deficiencias de los métodos y herramientas para la programación de controladores programables de la técnica anterior.

La figura 3 es una ilustración de una ejecución de la presente invención, en donde un dispositivo de cliente que está conectado a un PLC está en comunicación a través de una red con un servidor que contiene una herramienta de ingeniería que habilita su utilización en la Web para escribir código de programación para el PLC y para configurar los dispositivos y redes de entrada/salida.

La figura 4 es una ilustración de la arquitectura general de una ejecución de la presente invención, en donde una herramienta de ingeniería para escribir código de programación para un controlador programable y para la configuración de los dispositivos y la red de entrada/salida se encuentran en un servidor al que se accede a través de una red mediante un dispositivo de cliente.

ES 2 315 337 T3

La figura 5 ilustra una ejecución alternativa de la presente invención, en donde el servidor también consta de un nivel de capa de servicio Web.

5 La figura 6 ilustra, tanto desde la perspectiva de un cliente como desde la de un servidor, los componentes de una herramienta de ingeniería habilitada para la utilización en la Web de acuerdo con la presente invención que funciona a través de Internet, en donde se usan técnicas de programación gráfica.

10 La figura 7 ilustra un método alternativo de una herramienta de ingeniería habilitada para la utilización en la Web utilizando formato XML de acuerdo con la presente invención.

La figura 8 ilustra la habilitación para la utilización en la Web de una herramienta de ingeniería heredada, de acuerdo con la presente invención, utilizando el interfaz de la aplicación de la herramienta heredada.

15 La figura 9 es una vista de la pantalla de una herramienta de ingeniería de programación por diagrama de flujo habilitada para su utilización en la Web de acuerdo con la presente invención.

La figura 10 es una vista de pantalla de una herramienta de ingeniería de programación en lenguaje Ladder habilitada para la utilización en la Web de acuerdo con la presente invención.

20 La figura 11 es un diagrama de flujo que muestra los pasos del método de una ejecución de la presente invención.

La figura 12 es un diagrama de flujo que muestra un método de acuerdo con la presente invención para la programación de un controlador programable.

25 La figura 13 ilustra una configuración de un sistema de acuerdo con la presente invención, en donde un dispositivo de cliente está conectado en red con un servidor que posee una herramienta de ingeniería habilitada para la utilización en la Web, y un segundo dispositivo de cliente conectado en red con el servidor está en una red de área local (LAN) con uno o más controladores programables.

30 La figura 14 ilustra una configuración alternativa de la presente invención, en donde los controladores programables están interconectados directamente con un servidor, que posee una herramienta de ingeniería habilitada para la utilización en la Web, que se comunica con un dispositivo de cliente.

35 La figura 15 ilustra aún otra configuración alternativa de la presente invención, en donde un servidor, que posee una herramienta de ingeniería habilitada para la utilización en la Web, se encuentra en Internet o en una intranet con un dispositivo de cliente, y uno o más controladores programables están conectados en red con el servidor a través de una intranet o de Internet.

40 La figura 16 es un diagrama de flujo que muestra los pasos para proporcionar servicios de herramienta de ingeniería de acuerdo con la presente invención.

La figura 17 es un diagrama de flujo que muestra los pasos para generar código de programación fuera de línea, de acuerdo con la presente invención.

45 La figura 18 ilustra un nuevo paradigma de negocio para la venta de servicios de herramienta de ingeniería, de acuerdo con la presente invención.

50 La figura 19 ilustra un nuevo paradigma de negocio alternativo para la venta de servicios de herramienta de ingeniería, de acuerdo con la presente invención.

La figura 20 ilustra una ejecución de la presente invención en donde un código de ingeniería de legibilidad humana es almacenado en un servidor y es enlazado al código compilado ejecutándose en un controlador programable y accesible a un servidor de automatización.

55 La figura 21 ilustra cómo la presente invención puede ser constituida para ser una sola entidad corporativa o comercial.

La figura 22 ilustra un sistema de ingeniería colaborativa.

60 La figura 23 es un diagrama de flujo que muestra un método para el desarrollo y la comprobación del código de programación para el controlador programable y/o los datos de configuración.

La figura 24 es un diagrama de flujo que muestra un método para generar e instalar código de programación en un controlador programable de acuerdo con la presente invención.

65 La figura 25 ilustra una ejecución de la presente invención en donde un conjunto de herramientas de ingeniería reside en un servidor, contiene un editor de configuración y es invocado por un dispositivo de cliente mediante una red.

Figura 26 es un diagrama de flujo que muestra un método colaborador de ingeniería habilitado mediante la presente invención.

Figura 27 es un diagrama de flujo que muestra otro método colaborador de ingeniería alternativo.

Descripción detallada de la invención

La presente invención, entre otras cosas, proporciona un sistema y método para escribir código de programación (el cual normalmente también incluye la generación de datos de configuración) en un servidor central y para la descarga del código de programación compilado junto con cualquier dato de configuración a un controlador programable, como por ejemplo un PLC. Como se muestra en la figura 3, un dispositivo de cliente (30), como por ejemplo un ordenador personal que incluya un navegador, como por ejemplo Microsoft Internet Explorer® o Netscape Navigator®, se encuentra en una red pública o privada (35). La red (35) puede ser Internet o una intranet. En una ejecución, un PLC (40), como puede ser el Siemens® S7-200 PLC, se conecta al dispositivo de cliente (30) por medio de un cable de interfaz (45), como por ejemplo el Siemens® PC/PPI. El cable puede contener conmutadores DIP (48) para controlar los ajustes, como por ejemplo la tasa de baudios. En esta ejecución, el dispositivo de cliente (30) actúa como un dispositivo de programación para la programación del PLC (40). Sin embargo, a diferencia de los enfoques anteriores de la técnica para la programación del PLC (40), en donde las herramientas de ingeniería para la programación del PLC están instaladas en el dispositivo de programación, las herramientas de ingeniería necesarias para escribir y compilar código de programación residen en un servidor (50) al que puede acceder el dispositivo de cliente (30) mediante Internet o mediante una intranet u otra red (35). Las herramientas de ingeniería son preferiblemente herramientas habilitadas para la utilización en la Web para funcionar en cualquier aplicación de navegación de cualquier dispositivo de cliente (30).

Un usuario (55) accede al servidor (50) mediante la red (35) con el dispositivo de cliente (30). El dispositivo de cliente (30) puede ser un cliente ligero, es decir, no necesita contener ningún software que no sea el navegador. Así, el dispositivo de cliente (30) puede tener muchas formas, incluida la de ordenadores personales, PDAs habilitadas para la utilización en la Web, teléfonos móviles, dispositivos de televisión preparados para Internet, aparatos de Internet, etc. La presente invención es escalable en tanto que puede funcionar en una variedad de modos en una variedad de dispositivos de cliente que poseen una variedad de funcionalidad. Por ejemplo, la presente invención puede funcionar con funcionalidad limitada en un navegador de teléfono móvil habilitado para la utilización en la Web que tiene funcionalidad limitada y capacidad informática, o puede funcionar con funciones robustas en un ordenador personal sofisticado que posea un procesador de alta velocidad y grandes cantidades de memoria y capacidades gráficas robustas.

De acuerdo con la presente invención, las herramientas de ingeniería, similares a herramientas como Siemens STEP7®, se encuentran en el servidor (50) y están habilitadas para la utilización en la Web. Normalmente, estas herramientas de ingeniería emplearán esquemas de programación gráficos como lenguaje Ladder, programación por diagrama de flujo, por gráfico secuencial de funciones o por diagramas de funciones (véase, por ejemplo, la figura 1). Preferentemente, están configuradas para manejar una variedad de lenguajes de programación, incluida la programación de lista de instrucción y todos los demás lenguajes estándar, tales como por ejemplo todos los lenguajes del estándar IEC-1131 (LD, FBD, SFC; IL y ST). Las herramientas están configuradas para funcionar en una aplicación de navegación, consiguiendo de este modo que funcionen independientemente del sistema operativo que utilice el cliente. Preferentemente, las herramientas de ingeniería se configuran para que funcionen con una interfaz HTML o XML, de manera que puedan funcionar en cualquier navegador disponible. Las herramientas pueden proporcionar, además, un editor de configuración para asignar una área de entrada/salida lógica de un controlador a una área de entrada/salida física.

La figura 4 representa una ejecución de la arquitectura general del sistema para herramientas de ingeniería habilitadas para la utilización en la Web. Un dispositivo de cliente Web (30) contiene un navegador Web estándar. Una herramienta de ingeniería para generar código de programación para un controlador programable reside en un servidor Web (50) y es capaz de funcionar en el cliente Web en una aplicación de navegación. La herramienta de ingeniería está habilitada para la utilización en la Web y, en su forma más básica, consta de un editor o editores que pueden funcionar en el navegador Web. De forma ventajosa, los editores son independientes con respecto al sistema operativo y pueden funcionar en las últimas versiones de navegadores, sistemas operativos y hardware. El servidor Web (50) puede contener también un editor para llevar a cabo configuración de hardware. Puede contener, asimismo, el software y hardware necesario para llevar a cabo la gestión de versión y para ejecutar simulaciones. El cliente Web (30) y el servidor Web (50) están los dos ubicados en una red (35), como por ejemplo una intranet o Internet. Un usuario crea código de programación y/o genera datos de configuración para un controlador programable en el cliente Web (30). El código y los datos de configuración se convierten en un documento XML y se transmiten a través de la red (35) al servidor Web (50). El código de programación y los datos de configuración pueden almacenarse en una base de datos de proyecto (58) asociada al servidor Web (50). La base de datos de proyecto (58) puede tener la forma de cualquier medio de almacenamiento estándar. El servidor (50) no necesita estar ejecutando la última la plataforma más actual (Sistema operativo y hardware).

La figura 5 representa una ejecución alternativa en la cual una capa de servicio Web (70) está incorporada con el servidor (50). Los servicios Web adicionales, tales como las herramientas y servicios de simulación, pueden estar en capas por encima de las herramientas de ingeniería existentes. Esto proporciona al desarrollador de herramientas de ingeniería una manera rápida y rentable de innovar y desarrollar nuevos productos. Un vendedor de herramienta

ES 2 315 337 T3

de ingeniería puede impulsar y aumentar la longevidad de su inversión existente desarrollando la herramienta de ingeniería. En esta ejecución, los usuarios de una herramienta de ingeniería básica tendrán a su disposición servicios de ingeniería robustos que están a sólo un clic de ratón.

- 5 XML es particularmente muy apropiado para su uso con la presente invención porque representaciones gráficas, del tipo que utiliza el lenguaje de programación común PLC, se convierten fácilmente a un documento XML. De este modo, la transmisión entre el cliente (30) y el servidor (50) es preferible en forma de documento XML.

- 10 La figura 6 ilustra la parte del cliente, la parte del servidor y los componentes de Internet necesarios para habilitar la utilización en la Web de una herramienta estándar de ingeniería, tal como la Siemens STEP7®, utilizando técnicas de programación gráficas. Aunque se pueden utilizar técnicas de programación gráficas para habilitar la utilización en la Web de herramientas de ingeniería estándar tales como Siemens STEP 7®, la tecnología Microsoft®.Net proporciona un método más eficiente para habilitar la utilización en la Web de herramientas de ingeniería. Como se muestra en la Figura 7, la tecnología .Net facilita la interfaz entre una aplicación de servicio Web (99) y un servidor Web utilizando un formato XML. También permite el intercambio de datos cliente-servidor en un formato XML.

- 20 Como se muestra en la figura 8, los nuevos servicios Web (50) se pueden construir encima de la Interfaz de Aplicación ("API") 59 de una herramienta de ingeniería 77 ya existente, como por ejemplo la Siemens STEP 7®. El servidor (50) y los dispositivos de cliente (30) intercambian entonces documentos XML utilizando protocolos estándar.

- 25 Las herramientas de ingeniería habilitadas para la utilización en la Web y ejecutables en un navegador poseen, preferentemente, funciones de fácil manejo para el usuario para las varias fases de un proyecto de automatización. Las vistas de pantalla que representan una herramienta típica de ingeniería habilitada para la utilización en la Web, tal y como se vería en un dispositivo de cliente con un navegador estándar, se muestran en las figuras 9 y 10. La figura 9 representa una herramienta de ingeniería de programación por diagrama de flujo habilitada para la utilización en la Web y la figura 10 representa una herramienta de ingeniería habilitada para la utilización en la Web para la programación en un esquema de lenguaje Ladder (o diagrama de escalera). Normalmente, estas herramientas de ingeniería incluyen funciones tales como la configuración y parametrización de hardware, definición de la comunicación, programación, prueba, arranque y servicio, documentación y archivo, y funcionamiento/diagnosis. Las herramientas pueden contener un editor de símbolos para definir designaciones simbólicas, tipos de datos y comentarios de etiqueta global. Las herramientas trabajan con todos los lenguajes de programación, incluidos aunque sin limitarse a ellos, la programación de organigrama, IL, ST, SFC y/o FBD, y todos los lenguajes de programación posteriores especificados por IEC.

- 35 Como se discutía arriba, las funciones proporcionadas por las herramientas de ingeniería pueden ser escalables dependiendo del tipo de dispositivo de cliente sobre los que se ejecuten. Por ejemplo, un dispositivo de cliente ligero, como por ejemplo un teléfono móvil habilitado para la utilización en la Web, puede tan sólo ser capaz de ejecutar una herramienta que utilice funciones de edición gráfica limitada o basadas en texto. Un dispositivo de cliente más sofisticado, como por ejemplo un IBM 600E Thinkpad®, puede tolerar más funciones y puede, a su vez, permitir a la herramienta de ingeniería llevar a cabo más funciones, tales como edición gráfica completa, durante el desarrollo del código. Así, de acuerdo con la presente invención, si las funciones de la herramienta de ingeniería se llevan a cabo en el dispositivo de cliente o si lo hacen en el servidor, puede variar dependiendo de las habilidades del dispositivo de cliente, la velocidad de la conexión de Internet o intranet y otros parámetros.

- 45 Como se muestra en la figura 3 y se muestra en este diagrama de flujo de la figura 11, en una ejecución de la presente invención, un usuario (o varios usuarios) puede acceder a la herramienta de ingeniería de programación de PLC (38) (o una herramienta de ingeniería para la programación de otros controladores programables) que residen en un servidor (50), paso (1000). En una ejecución, el servidor (50) reside en el área de la Web de Internet. En otra ejecución, el servidor reside en una red colectiva privada o en un intranet colectivo. El usuario (55) acciona la herramienta de ingeniería (38) en un navegador Web en el dispositivo de cliente (30), paso (1005), y crea el código de programación en el navegador para el controlador programable, paso (1010). Las medidas estándar de seguridad se pueden emplear para limitar el acceso tanto al servidor como a los programas creados en ellos. El usuario (55) puede crear un programa en una única sesión o en un periodo de varias sesiones. El usuario puede modificar, editar y depurar el código de programación en una aplicación de navegación, paso (1015). El código generado en el navegador puede estar en forma gráfica. El código se convierte entonces en un documento, paso (1020), y el documento se transmite mediante la red al servidor, paso (1025). Preferentemente, el código de programación se almacena en el servidor (50), paso (1030).

- 60 El software de edición, el software de depuración y un compilador pueden estar localizados en el servidor (50). Esto permite prácticamente a cualquier PC, PDA u otro dispositivo de cliente habilitado para la utilización en la Web (30) actuar como un dispositivo de programación. Después de que el usuario (55) cree y edite el código de programación, el código puede ser almacenado y compilado en el servidor (50), que tiene preferentemente un compilador. En algunas ejecuciones, un servidor adicional podría usarse para archivar el código. Más aún, el compilador puede residir también en un servidor diferente al que puede acceder el servidor que contiene la herramienta de ingeniería.

- 65 Después de que el código sea creado, depurado y compilado, es descargado a un controlador programable. Como se muestra en el diagrama de flujo de la figura 12, en una ejecución la programación de un controlador programable consta de dos pasos: establecer la comunicación entre el controlador y el servidor, paso (2000), y descargar el código de programación al controlador a través de la red, paso (2005). En una ejecución, como la mostrada en la figura 3, el

ES 2 315 337 T3

código puede ser descargado en el dispositivo de cliente (30) el cual, sucesivamente, puede entonces intercomunicarse con el controlador programable (40). En esta ejecución, una copia del código de programación compilado puede residir en el dispositivo de cliente o puede ser archivado en el dispositivo de cliente.

5 Normalmente, el dispositivo de cliente (30) se interconecta con el controlador programable (40) mediante un cable conector (45), como por ejemplo un Cable Conector Siemens PC/PPI (véase figura 3). Como se muestra en la figura 3, cuando un PC es utilizado como dispositivo de cliente (30), el cable conector permite a una interfaz en serie en el controlador programable comunicarse con una interfaz en serie RS-232 estándar. Cuando otro dispositivo, como por ejemplo una PDA, se utiliza como dispositivo de cliente, el cable conector se interconecta con un puerto de comunicación en el dispositivo. El cable, como se muestra en la figura 3, puede contener conmutadores DIP (48) que se utilizan para seleccionar una velocidad apropiada (tasa de baudios) a la que la información es transmitida entre el PLC (40) y el dispositivo de cliente (30). El dispositivo de cliente (30), como se muestra en la figura 13, puede interconectarse con una pluralidad de controladores programables (40). En algunas ejecuciones donde existe un número amplio de controladores programables interconectados con un dispositivo de cliente individual, puede ser necesario un repetidor (no mostrado) junto con el cable.

En una ejecución alternativa, como la mostrada en la figura 14, los controladores programables (40) se pueden conectar directa o indirectamente a una red, como por ejemplo una LAN (100), en la cual reside el servidor sin tener que pasar a través del dispositivo de cliente (30). En esta ejecución, los controladores programables (40), están en comunicación con el servidor (50) a través de las redes, (100) las cuales pueden constar de una red estándar que utilice protocolos de Ethernet TCP/IP, una red Profibus®, o prácticamente cualquier otra configuración de red. El código de programación compilado podrá entonces ser descargado directamente a un controlador programable (40).

En otra ejecución más, como la representada en la figura 15, los controladores programables (40) pueden comunicarse con el servidor (50) a través de una red (110), como por ejemplo una intranet o Internet. Los dispositivos programables, en algunas ejecuciones, pueden estar enlazados entre sí en una LAN (100) utilizando protocolos como TCP/IP o Profibus® o cualquier otro protocolo apropiado.

Sin tener en cuenta si el dispositivo programable está directamente interconectado con la red donde reside el servidor o si está en comunicación con el servidor a través de un dispositivo de cliente, se puede almacenar una copia de archivo del código de programación en una base de datos de proyecto de archivo (120) en el servidor (50) o en un dispositivo de memoria asociado al servidor (50) (véase figura 14). Esto proporciona, de forma ventajosa, un almacenamiento centralizado para copias de seguridad del código de programación compilado. La copia de archivo puede almacenarse también en un servidor diferente o en una base de datos disponible a través de la red.

En algunas ejecuciones puede ser deseable crear el código de programación para el controlador programable fuera de línea, es decir, después de que el dispositivo de cliente sea desconectado del servidor que contiene la herramienta de ingeniería. La programación fuera de línea permite a un usuario crear y editar el código de programa sin mantener una conexión con el servidor. Como se muestra en el diagrama de flujo de la figura 17, un usuario establece comunicación entre un dispositivo de cliente y un servidor que contenga una herramienta de ingeniería habilitada para la utilización en la Web y una base de datos de programación de proyectos, paso (3000). El usuario abre, entonces, la herramienta de ingeniería habilitada para la utilización en la Web en una aplicación de navegación en el cliente (3005). El usuario verifica entonces una copia del proyecto de la base de datos de proyecto, paso (3010), y la copia del proyecto se transmite al dispositivo de cliente, (3015). La comunicación entre el cliente y el servidor es entonces finalizada, paso (3020). El usuario es ahora libre de editar la copia verificada fuera de línea, paso (3025). Cuando el usuario termina la edición, el usuario restablece la comunicación entre el cliente y el servidor, paso (3030). El código de programación editado se vuelve a verificar en (3035), y otros usuarios que posean derechos de acceso podrán acceder a él. En algunas ejecuciones, mientras el proyecto es verificado, puede permitirse a otros usuarios leer una copia del proyecto verificado en la forma en la que existía antes de ser verificado. Dependiendo de la sofisticación y de los recursos del dispositivo de cliente, las técnicas y funciones de la programación fuera de línea pueden variar.

Cuando el dispositivo de cliente carezca de sofisticación, se puede proporcionar una función simple de “sólo edición” basada en texto. Cuando el recurso del cliente contenga un nivel alto de sofisticación (por ejemplo, memoria, capacidad gráfica, velocidad de procesador, etc.) se pueden proporcionar funciones más sofisticadas de creación de código y edición gráfica. De este modo, la presente invención es escalable. Por ejemplo, en una ejecución que funcione en un dispositivo sencillo habilitado para la utilización en la Web, el usuario puede estar provisto con sólo una función de edición de texto, mientras que en otra ejecución que funcione en un cliente más sofisticado, el usuario puede estar provisto con una herramienta de ingeniería que funcione en el navegador Web que contenga todas las funciones de edición gráfica que existen cuando el usuario está conectado al servidor.

Además de depender de los recursos del cliente (por ejemplo, la memoria), otros factores, como el sistema operativo del cliente y cuestiones de seguridad, pueden influenciar la funcionalidad de la herramienta de ingeniería habilitada para la utilización en la Web. Para poder trabajar fuera de línea, el sistema operativo del dispositivo necesita admitir alguna forma de instalar la aplicación en el dispositivo de cliente. Normalmente, esto no es un problema para las PDA modernas, pero puede suponer un problema para aparatos más pequeños, como por ejemplo los teléfonos móviles. Por razones de seguridad, normalmente las aplicaciones de navegación Web tienen limitado el acceso a los recursos del sistema, tales como controladores y memoria de sistema. Microsoft Corp., Redmond Washington, proporciona un software que permite que las aplicaciones que se ejecuten en su Internet Explorer® accedan a los recursos del sistema

del cliente por medio de controles ActiveX incorporados en la aplicación. Para aquellos que deseen una mayor seguridad, Microsoft Corp. proporciona al usuario la capacidad de inhabilitar en su navegador la descarga de todos los controles ActiveX juntos, o sólo aceptar las descargas de sitios de confianza o solamente controles ActiveX contratados originalmente.

Las funciones específicas se proporcionan/suministran al dispositivo de cliente según sus necesidades. Por ejemplo, la verificación de la sintaxis no se proporciona hasta que el usuario del dispositivo de cliente necesite esa función. Esto permite al usuario descargar en el dispositivo de cliente funciones mínimas inicialmente y agregar funciones adicionales sólo cuando esas funciones se necesiten.

En general, no es necesario conectar el dispositivo programable al controlador programable cuando se está generando y compilando el código de programación. Sin embargo, puede resultar deseable que el controlador programable se encuentre en línea con el servidor y el dispositivo de cliente al mismo tiempo. En este caso, los cambios se descargan al controlador programable. Además, se puede monitorizar el estado de los elementos de entrada y de salida. El servidor puede estar equipado con software de diagnóstico o funciones que permitan la monitorización del controlador programable. El servidor puede simular entradas y salidas y comunicar éstas con el controlador programable. Esto permite crear y almacenar de manera centralizada algoritmos de prueba diversos y resistentes. Los algoritmos de prueba resistentes pueden ser accesibles sólo para usuarios autorizados y a los usuarios se les puede cobrar un precio para poder utilizarlos.

La presente invención proporciona un nuevo paradigma en la industria de herramientas de ingeniería. Previamente a la presente invención, las herramientas de ingeniería eran productos que se fabricaban y se vendían (véase figura 2). De acuerdo con la presente invención, las herramientas de ingeniería habilitadas para la utilización en la Web ofrecen un nuevo paradigma de negocio donde los fabricantes de una herramienta de ingeniería suministran un servicio (véase figuras 18 y 19). En una ejecución, como la mostrada por el diagrama de flujo de la figura 18, se proporciona a un usuario acceso al servidor que contiene la herramienta de ingeniería para desarrollar y compilar el código de programación para un controlador programable, paso (200). Se puede proporcionar al usuario con acceso gratis a la herramienta de ingeniería con el propósito de escribir código de programación. El usuario puede desarrollar el código en un dispositivo de cliente que se comunica con el servidor, paso (205). El usuario puede crear y modificar el código de programación sin que tenga que pagar por ello. Esto, de forma ventajosa, proporciona incluso a negocios de la pequeña empresa con acceso a herramientas de ingeniería sofisticadas y actualizadas. Sólo después de que el usuario complete el código y lo compile, se le aplicarán honorarios, paso (207). Los honorarios se pueden calcular en el momento en el que el código de programación es compilado o se cargan al usuario al descargar el código compilado al PLC (véase figura 19, paso (220)). Los honorarios por el código se pueden basar en el tamaño del código compilado, el tiempo necesario para escribir el código, el número de entradas y/o salidas que el código está diseñado para manejar, o una variedad de otros parámetros que pueden ser medidos. Puesto que la herramienta de ingeniería reside en un servidor central y no en un dispositivo de cliente, la limitación y el control del acceso a la herramienta es simplificado. Más aún, es tarea fácil para un representante de un servicio al cliente o para un vendedor suministrar a un cliente con el uso adicional de la herramienta de programación, cuando dicho uso adicional resultara ventajoso por un posible negocio. Por ejemplo, un vendedor podría proporcionar un periodo limitado de uso libre de coste de una herramienta de ingeniería habilitada para la utilización en la Web para programar y configurar un PLC cuando un cliente adquiere un PLC nuevo o un dispositivo de campo nuevo que se interconecte con un PLC ya existente.

Visto desde la perspectiva de la entidad que construye y/o distribuye las herramientas de ingeniería, el nuevo paradigma de negocio puede tomar la forma de los pasos mostrados en el diagrama de flujo de la figura 16. El fabricante o el distribuidor de la herramienta mantienen una herramienta de ingeniería habilitada para la utilización en la Web en un servidor al que acceden los clientes a través de una red, como por ejemplo Internet, paso (1601). A los clientes se les proporciona el acceso a la herramienta a través de la red, paso (1605). Se pueden utilizar contraseñas u otras medidas de seguridad para restringir el acceso sólo a usuarios autorizados. El público puede tener acceso a la herramienta, pero no acceder necesariamente al código creado con la herramienta. El distribuidor de la herramienta de ingeniería recibe un documento de un usuario - por ejemplo, a través de la red-, que contiene el código de programación que se ha generado con la herramienta de ingeniería habilitada para la utilización en la Web, paso (1610). El código de programación puede entonces ser compilado, paso (1615), o podría guardarse y hacerse accesible para la entidad que lo creó o para una entidad que tenga el acceso autorizado. Finalmente, el código se compila y luego se descarga al controlador programable, paso (1620).

Puesto que el código se crea en un servidor central, el código puede archivararse tanto en un formulario de legibilidad humana como en un formulario compilado (véase figura 20). Esto proporciona un método para reinstalar el software ya existente en controladores programables nuevos o restaurar versiones antiguas del código cuando una versión nueva no pueda funcionar de la forma deseada. Al usuario se le cobra una cantidad por obtener la descarga de una copia archivada. Esta cantidad puede ser menor que la cantidad inicial por la primera copia que ha descargado el usuario.

Además, si el formulario del código de legibilidad humana (conocido también como fuente) es archivado y enlazado con el código máquina de lectura, la monitorización habilitada para la utilización en la Web, la simulación y el control de servicios automatizados se facilitan. Por ejemplo, como se muestra en la figura 20, un PLC habilitado para la utilización en la Web (300) puede contener un servidor Web incorporado (310). El PLC puede ser conectado a una red (350), como por ejemplo Internet o una intranet. El PLC se puede programar con las herramientas de ingeniería habilitadas para la utilización en la Web de la presente invención. El código de programación para el PLC puede re-

residir en un servidor de ingeniería (400) tanto en formulario de legibilidad humana como en formulario compilado. El formulario de legibilidad humana se almacena en una memoria o en un dispositivo de almacenamiento de datos (410) y el código se enlaza con el formulario compilado que también reside en un dispositivo de memoria (420) asociado al servidor de ingeniería. Puesto que el servidor de ingeniería es accesible a través de Internet o de una intranet junto con el PLC, es posible mantener otro servidor (ya sea un servidor físicamente separado del servidor de ingeniería o un servidor virtual que sea parte física del servidor de ingeniería en el PLC), como por ejemplo un servidor de automatización en la misma red. El servidor de automatización puede incluso ser parte del PLC, como por ejemplo cuando el PLC contiene un servidor Web incorporado. La simulación robusta y sofisticada, la monitorización y el software de control pueden residir en el servidor de automatización y estar habilitados para la utilización en la Web para que un dispositivo de cliente acceda a ellos en Internet o en una intranet. Ya que la versión de legibilidad humana del código de programación para el software que se ejecuta en el PLC está disponible por medio del servidor de ingeniería, los paquetes de software de simulación robusta, de diagnóstico, de control y de monitorización pueden acceder de manera ventajosa al código de legibilidad humana y proporcionar información valiosa para ejecutar estos paquetes de software. Por ejemplo, ya que el código de legibilidad humana es fácilmente accesible, se facilita el acceso a información de etiquetas para dispositivos de campos diversos controlados por el PLC. La presente invención permite de este modo una comunicación sofisticada y robusta así como la transferencia de datos desde una planta de producción hasta una sala de juntas.

Como se muestra en la figura 5, mantener un servidor con herramientas de ingeniería centralizadas habilitadas para la utilización en la Web proporciona una plataforma para el desarrollador de la herramienta de ingeniería para lanzar nuevos servicios y productos. Por ejemplo, previamente a la presente invención, el coste de la adquisición de herramientas de simulación para optimizar el código de programación impedía a muchos clientes invertir en estas herramientas. Puesto que, en una ejecución, el código de programación es almacenado y generado en un servidor central al que accede un gran número de clientes, el coste de la instalación del software de simulación en el servidor puede ser distribuido entre este gran número de clientes. Es más, en una ejecución, los clientes sólo pagan por el uso de las herramientas de simulación. Además, las herramientas de diagnóstico podrían ser instaladas en el servidor y compartidas de forma similar. El propietario del servidor y de las herramientas podría obtener una compensación basándose en el uso, en el ahorro de costes al cliente o en cualquier otra variedad de esquemas comerciales.

Puesto que las herramientas están localizadas de manera central, la concesión de licencia de las herramientas se simplifica. Las licencias para las herramientas pueden concederse a empresas, a ciertos individuos dentro de una empresa o a los propios individuos. El acceso a las herramientas puede ser realizado utilizando técnicas de encriptación y/o protección con contraseña y puede estar condicionada al mantenimiento de una licencia válida. Puede ser deseable suministrar un periodo limitado de acceso libre de coste a la herramienta. En algunos casos, la cantidad de acceso libre de coste puede estar unido a la adquisición de controladores programables. Suministrar un servidor en modo local libre de coste durante un periodo limitado de tiempo puede, además, ser útil en la venta de licencias a largo plazo de servidores a clientes. Al usuario se le proporciona el servidor con un periodo limitado de prueba gratuito.

Como se muestra en la figura 21, una empresa puede elegir arrendar o ser propietario de su propio servidor. Para algunas empresas grandes manufactureras/de fabricación y/u otras entidades donde la seguridad es una preocupación importante, puede ser ventajoso limitar el acceso al servidor a los empleados de la empresa únicamente. Debido a que la herramienta de ingeniería reside en el servidor y no en cada PC de usuario, la gestión de versión, la actualización y otras tareas de mantenimiento del software son simplificadas. Es más, la herramienta de ingeniería va a funcionar independientemente del sistema operativo en los dispositivos de programación del usuario. Debido a que el servidor es accesible sólo para los empleados de la empresa, la información propietaria puede ser almacenada en el servidor y puede acceder a ella un equipo de programación que trabaja en localizaciones diferentes.

La ingeniería colaborativa es posible. Múltiples usuarios pueden acceder y desarrollar en conjunto el código de programación para uno o más controladores programables. Como se muestra en la figura 22, una pluralidad de usuarios puede acceder a un servidor. Una herramienta de ingeniería habilitada para la utilización en la Web (2200) se localiza en el servidor (50) y es capaz de ejecutarse en una aplicación de navegación en los dispositivos de cliente (30). El servidor puede contener una memoria para el almacenamiento de proyectos de ingeniería en una base de datos (2210), así como un administrador de configuración (2220). Los proyectos almacenados en la base de datos de proyecto (2210) pueden ser proyectos individuales o pueden estar relacionados. Normalmente, al usuario que creó un proyecto particular o una pieza de código se le otorgan derechos totales de lectura y escritura para ese proyecto. Como se muestra en la figura 22, un primer usuario (5000) puede acceder a un primer proyecto (2241). Al acceder al primer proyecto (2241), que normalmente es un documento que almacena código de programación en un formulario de legibilidad humana, se crea en el servidor (50) una copia activa (2250) del primer proyecto. Con la ayuda de la herramienta de ingeniería habilitada para la utilización en la Web (2200), el primer usuario (5000) puede modificar la copia activa del primer proyecto (2250). Otros usuarios (5001), (5002), (5003) del sistema pueden necesitar ver una copia del proyecto 1. Este es muy a menudo el caso cuando un segundo y tercer usuarios trabajan en proyectos relacionados. Puesto que una copia activa es un trabajo en proceso, es posible que el primer usuario (5000) nunca pueda guardar los cambios que el primer usuario (5000) está realizando sobre la copia activa, por lo que no sería ventajoso normalmente no otorgar al segundo y tercer usuarios derechos de lectura de la copia activa. Por el contrario, a menudo es preferible otorgar al segundo y tercer usuarios (5002), (5003) derechos de lectura para la copia archivada que reside en la base de datos del proyecto. En algunas ejecuciones, puede ser ventajoso proporcionar un cuarto usuario (5001) con derechos tanto de lectura como de lectura-escritura sobre la copia activa. Esto puede resultar especialmente útil cuando el cuarto usuario (5001) es el representante de un servicio al cliente, un consultor o supervisor que supervisa a los usuarios. El

acceso a las copias archivadas de los proyectos almacenados en la base de datos, así como el acceso a la copia activa, se controlan generalmente por un administrador de configuración. El creador de un proyecto otorga derechos a otros usuarios mediante el gestor de configuración (2220). A menudo, al menos un usuario distinto, como por ejemplo un supervisor o un administrador de IT, tiene la habilidad de otorgar derechos.

Como se representa en la figura 22, una tarea amplia de programación se divide en varios proyectos más pequeños. Cada proyecto se almacena como un documento en la base de datos de proyectos. Cada usuario es principalmente responsable de un proyecto diferente. El primer usuario (5000), por ejemplo, está a cargo del proyecto 1 (2241). Así, al primer usuario (5000), por medio del gestor de configuración, se le otorgan derechos de lectura y escritura sobre el primer proyecto (2241). Otros usuarios, como el segundo y el tercer usuarios (5002), (5003), pueden estar a cargo de proyectos diferentes o proyectores relacionados y, por tanto, se les otorgan derechos menores sobre el proyecto 1, tales como solamente derechos de lectura. Al segundo usuario (5002) se le ha asignado la tarea de crear el proyecto 2 (2242) y, por tanto, se le otorgan derechos totales sobre el proyecto 2, pero, a los usuarios 1 y 3 sólo se les dan derechos de lectura sobre una copia archivada del proyecto 2. Al usuario 3 se le ha asignado la tarea de crear el proyecto 3 (2243) y se le otorgan, por tanto, derechos de lectura-escritura sobre el proyecto 3, pero sólo derechos de lectura para las copias archivadas de los proyectos 1 y 2. Asimismo, al primer usuario (5000) se le otorgan sólo derechos de lectura sobre las copias archivadas de los proyectos 2 y 3. A un supervisor de proyecto, como el cuarto usuario (5001), se le pueden dar derechos de supervisión, lo que puede incluir el derecho de lectura y escritura de copias activas, así como de copias archivadas, el derecho de combinar varios proyectos (tanto las copias activas como las archivadas) en una pieza de código de programación, el derecho a compilar el código de programación y cualquier otro derecho necesario para supervisar un proyecto de programación.

La presente invención permite, por tanto, numerosos métodos para la programación colaborativa de controladores programables y la generación de datos de configuración para redes de entrada/salida y dispositivos, lo cual es normalmente parte del proceso de programación. Las figuras 26-27 representan algunos ejemplos de algunos de los muchos métodos de ingeniería colaborativa y técnicas posibilitadas por la presente invención. Mientras que los ejemplos específicos se muestran en estos dibujos, los pasos del método que están representados aquí están destinados a ser ilustrativos y no exhaustivos de los amplios y robustos métodos de ingeniería colaborativa posibilitados por la presente invención.

La presente invención proporciona el archivo idóneo del código de programación y de la gestión de versión. Un gestor de versión o sistema de gestión de versión puede ser incorporado al administrador de configuración o puede ser controlado por el administrador de configuración. Antes de que el nuevo código de programación compilado sea descargado a un controlador programable (bien directamente desde el servidor al controlador, o desde el servidor al controlador mediante el dispositivo de cliente), una copia del código ya existente es archivada en el servidor. El código de programación se archiva en un formulario de legibilidad humana antes de compilarlo. El código compilado y el código de legibilidad humana se pueden enlazar entre ellos de manera que se pueda acceder al código de legibilidad humana basándose en el código compilado (véase, por ejemplo, la figura 20).

Una herramienta de gestión de versión que reside en el servidor proporcionaría diversas ventajas y proporciona un nuevo modelo de negocio para proveedores de herramientas de ingeniería. Una de las muchas ventajas que ofrece es que un cliente puede buscar un cambio específico para un programa o puede comparar dos versiones de un programa. Esto es especialmente deseable en industrias reguladas por el Organismo para el Control de los Alimentos y los Medicamentos (FDA en sus siglas en inglés, *Food and Drugs Administration*). Por ejemplo, el FDA exige que se documenten todos los cambios en un programa PLC en una planta alimenticia. Una herramienta de gestión de versión puede proporcionar esta funcionalidad. En algunas ejecuciones, la herramienta de gestión de versión es parte de un gestor de configuración y reside en un servidor junto con una herramienta de ingeniería. En un escenario típico, un usuario realiza un cambio al programa de PLC (por ejemplo, versión 1.0), por ejemplo para aumentar la cantidad de azúcar en una mezcla. El programa modificado, junto con la documentación, se guarda como una nueva versión (por ejemplo, versión 1.1) en el servidor. Si en algún momento posterior en el tiempo se determina que el producto es demasiado dulce, un usuario subsiguiente puede corregirlo buscando los cambios del PLC que afectaron a la dulzura o a la cantidad de azúcar añadido al producto. El gestor de versión soporta preferiblemente esta funcionalidad. A menudo, no es posible volver a la versión original (por ejemplo, 1.0) porque ha habido cambios posteriores, pero el usuario puede crear una nueva versión realizando cambios a la versión existente que se está ejecutando en el PLC.

Además de las funciones de búsqueda que proporciona, la herramienta de gestión de versión puede también proporcionar funciones de comparación que permite a un usuario comparar una versión con otra para mostrar los cambios. La función de comparación puede funcionar en lenguajes de programación tanto basados en texto como en gráficos y preferentemente destaca los cambios.

Se puede poner a prueba el código de programación desarrollado en un servidor central. Como se muestra y se muestra en la figura 23, se accede con un dispositivo de cliente a un servidor que contiene una herramienta de ingeniería, paso (9900). La herramienta de ingeniería es accionada en el dispositivo de cliente, paso (9905), preferentemente en una aplicación de navegación en el dispositivo de cliente, y el código de programación es entonces creado en el servidor, paso (9910). El código de programación se puede poner a prueba ejecutando un software de simulación en el servidor, paso (9915). El software de simulación puede imitar un controlador programable real. El servidor se puede interconectar con un controlador programable de pruebas, paso (9920). Preferentemente, el controlador programable de pruebas se configura para poner a prueba el código de programación y/o los datos de configuración generados

con la herramienta de ingeniería en el servidor. Después de poner a prueba el código de programación, éste puede ser descargado a un controlador programable real, paso (9925), el cual es identificado a menudo como controlador programable objetivo.

5 En la figura 24 se muestra un método para descargar el código generado por la presente invención. El código de programación y/o los datos de configuración son generados en un servidor, paso (9000). El código es entonces compilado, paso (9005). Después, es descargado en un dispositivo de cliente, paso (9010) y copiado en un dispositivo de almacenaje que está asociado al dispositivo de cliente, paso (9015). El dispositivo de almacenaje puede tener muchas formas y puede ser un dispositivo extraíble de almacenaje, tales como una unidad de disco o una unidad de memoria Flash. El dispositivo de memoria puede estar interconectado o estar acoplado comunicativamente con un dispositivo electrónico, paso (9020). El código puede copiarse en el dispositivo electrónico, paso (9025). El código puede ser descargado entonces desde el dispositivo electrónico a un controlador programable objetivo, paso (9030).

15 Como se muestra en la figura 25, la presente invención también puede constar de un editor de configuración (9100) para generar código o datos para configurar redes de entrada/salida y dispositivos que residen en un servidor (9200). Un editor de configuración que es normalmente parte de la herramienta de ingeniería (9300) o parte de una serie de herramientas de ingeniería (9350). Aquellos que sean expertos en la materia, reconocerán que en muchos casos las herramientas de ingeniería descritas aquí pueden contener un editor de configuración, y la generación del código de programación puede constar de generación de datos de configuración. El editor de configuración puede tomar muchas formas, pero normalmente es gráfica y está adaptada preferiblemente para ser invocada por un dispositivo de cliente (9500) a través de una red (9600), como puede ser Internet. Un editor de configuración gráfico puede mostrar la topología de una red de entrada/salida y del dispositivo en la red. Los datos de configuración para redes de entrada/salida y dispositivos se descargan normalmente a un controlador programable (9700) en la red. En algunos casos, los datos se pueden descargar a dispositivos inteligentes en la red. Los datos de configuración constan a menudo de información como por ejemplo las tasas de baudío e información relativa a la asignación de entradas/salidas lógicas a físicas.

30 La presente invención, aunque descrita arriba en referencia a ejecuciones específicas, puede ser ejecutada de numerosas formas y no debería estar limitada por los ejemplos de ejecuciones descritos arriba, que son ilustrativos de la presente invención, pero no exhaustivos. Por ejemplo, aquellos que son expertos en la materia reconocerán también que, aunque los pasos del método de la presente invención son analizados arriba en un orden específico, a menudo se da el caso de que el orden de los pasos no es crítico y la invención se puede poner en práctica ejecutando los pasos en diferente orden. Por ejemplo, resulta a menudo irrelevante para la práctica de la presente invención si los pasos del método se ejecutan en un orden particular. Por ejemplo, no existe diferencia alguna si un cliente transmite un valor antes o después de descargar el código compilado y los datos de configuración a un PLC desde un servidor central.

40

45

50

55

60

65

REIVINDICACIONES

1. Un sistema que comprende un servidor (50), una red (35), al menos un dispositivo de cliente (30) que posee una aplicación de interfaz de red, en donde el servidor está en comunicación con al menos un dispositivo de cliente (30) a través de la red (35); una herramienta de ingeniería (77) que reside en el servidor (50) y que es accesible por al menos un dispositivo de cliente (30) a través de la red, en donde la herramienta de ingeniería funciona en la aplicación de interfaz de red en al menos un dispositivo de cliente (30); una memoria (58) asociada al servidor (50) y al menos un controlador programable (40); **caracterizado** porque el servidor (50) es un servidor central; porque la herramienta de ingeniería (77) es una herramienta para generar código de programación para controladores programables (40) empleando un lenguaje de programación; y porque el código de programación y los datos de configuración generados con la herramienta de ingeniería son almacenados en la memoria (58) asociada al servidor central y son descargados a un controlador programable objetivo (40).

2. Un sistema de acuerdo a la reivindicación 1, en donde el lenguaje de programación consta de un lenguaje de programación gráfico.

3. Un sistema de acuerdo a la reivindicación 2, en donde el lenguaje de programación consta de un lenguaje de programación Ladder (o diagrama de escalera).

4. Un sistema de acuerdo a la reivindicación 2, en donde el lenguaje de programación gráfico consta de un esquema de programación de bloque de funciones.

5. Un sistema de acuerdo a la reivindicación 2, en donde el lenguaje de programación gráfico consta de un esquema de programación de diagrama de flujo.

6. Un sistema de acuerdo a la reivindicación 1, en donde el lenguaje de programación consta de un lenguaje basado en texto.

7. Un sistema de acuerdo a la reivindicación 6, en donde el lenguaje de programación consta de un lenguaje de programación de lista de instrucción.

8. Un sistema de acuerdo a la reivindicación 6, en donde el lenguaje de programación consta de un lenguaje de texto estructurado.

9. Un sistema de acuerdo a la reivindicación 1, en donde la herramienta de ingeniería (77) está configurada para intercambiar información entre el servidor y el cliente en forma de documento XML.

10. Un sistema de acuerdo a la reivindicación 1, en donde la aplicación de interfaz de red consta de un navegador Web y en donde el sistema además consta de un compilador que reside en el servidor para la compilación del código de programación.

11. Un sistema de acuerdo a la reivindicación 1, en donde el dispositivo de cliente es una PDA.

12. Un sistema de acuerdo a la reivindicación 1, en donde el dispositivo de cliente es un ordenador personal.

13. Un sistema de acuerdo a la reivindicación 1, en donde el dispositivo de cliente es un teléfono móvil habilitado para la utilización en la Web.

14. Un método para la creación de código de programación de controlador programable en un sistema que consta de un servidor (50), una red (35), al menos un dispositivo de cliente (30); una herramienta de ingeniería (77) que reside en el servidor y es accesible por al menos un dispositivo de cliente (30), teniendo el dispositivo una aplicación de interfaz de red; un sistema que, además, consta de una memoria (58) y al menos un controlador programable (40); un método que consta de acceso a la herramienta de ingeniería con al menos un dispositivo de cliente; ejecutando la herramienta de ingeniería (77) en una aplicación de interfaz de red instalada en al menos un dispositivo de cliente; en donde el servidor está en comunicación con al menos un dispositivo de cliente (30) a través de la red; y en donde la memoria (58) está asociada al servidor; **caracterizado** porque el servidor (50) es un servidor central, y porque la herramienta de ingeniería (77) es una herramienta para generar código de programación para al menos un controlador programable (40) empleando un lenguaje de programación; un método que, además, consta de la generación de código de programación y datos de configuración para los controladores programables que utilizan la herramienta de ingeniería; almacenando el código de programación y los datos de configuración en la memoria (58) asociada al servidor central; y descargando el código de programación a un controlador programable objetivo (40).

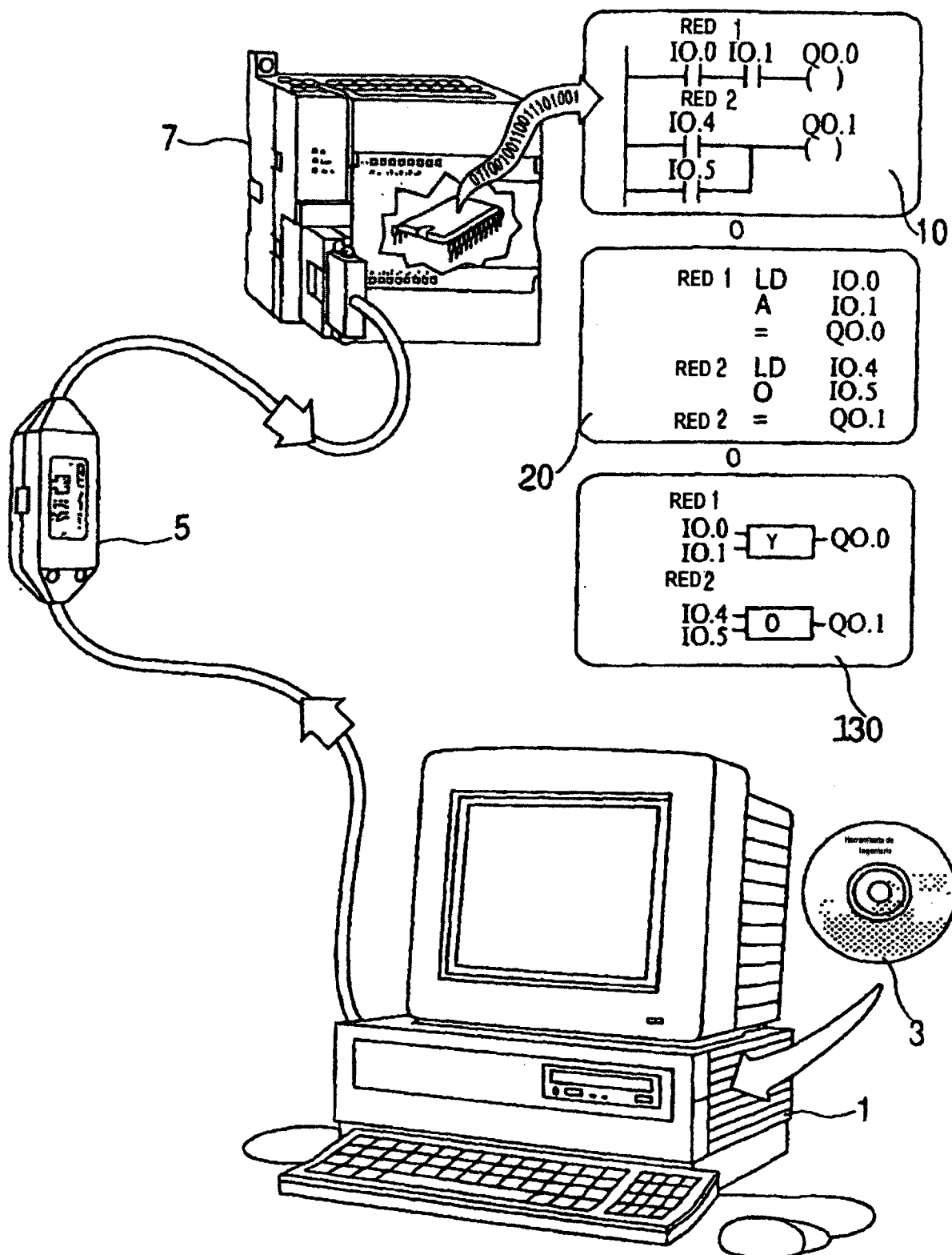


FIG. 1
(TÉCNICA ANTERIOR)

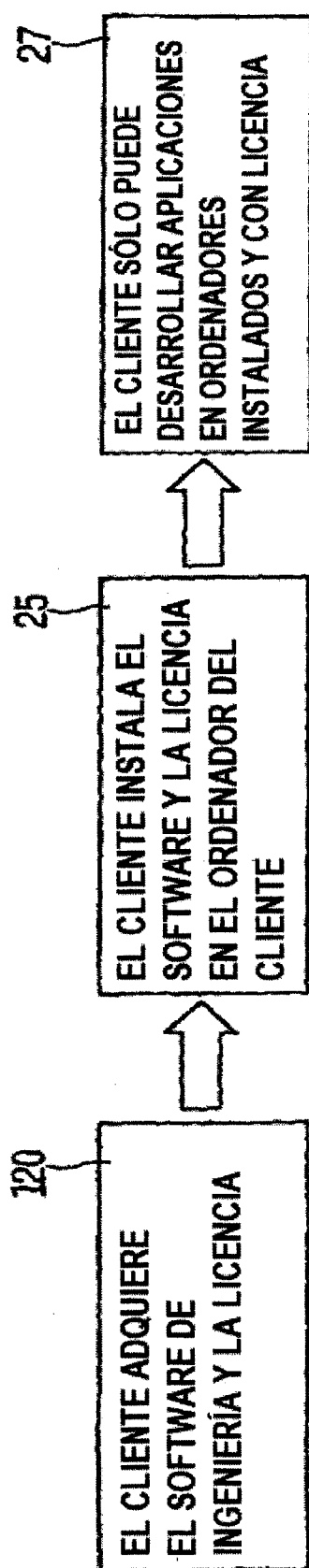


FIG. 2
(TÉCNICA ANTERIOR)

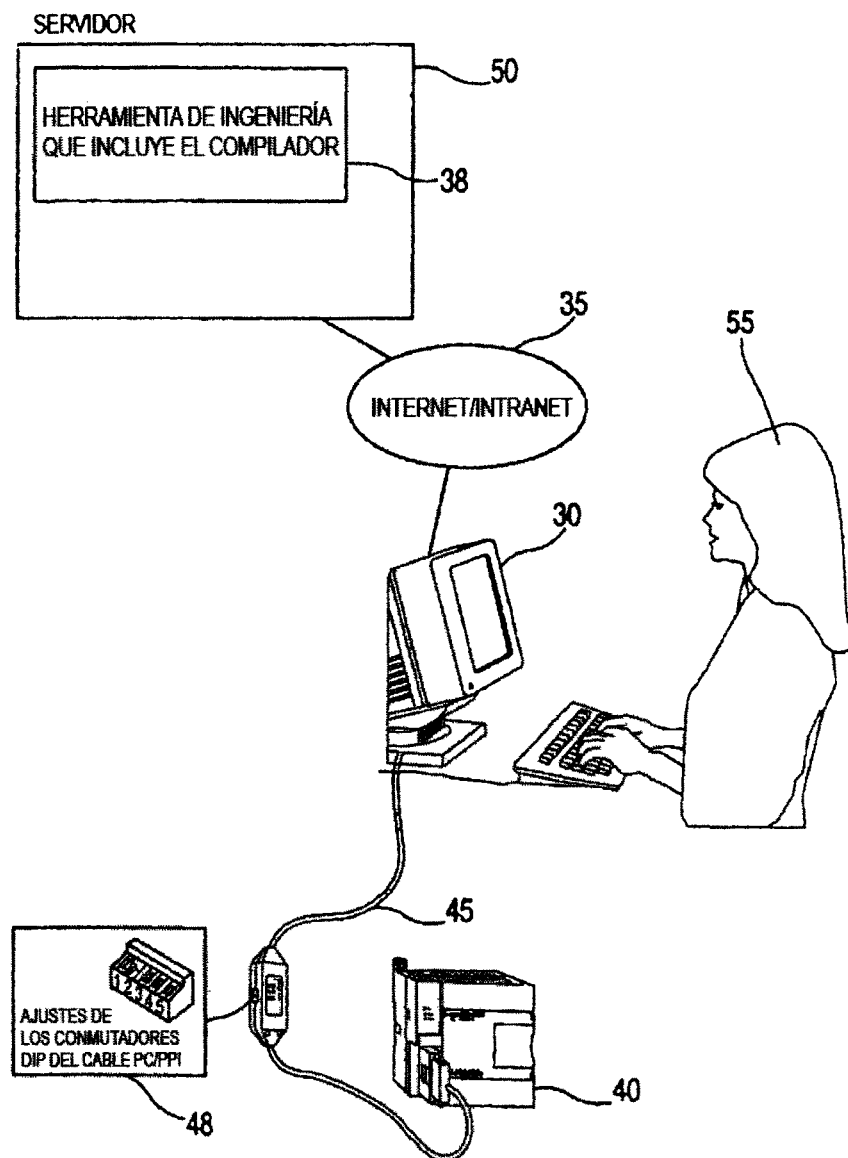


FIG. 3

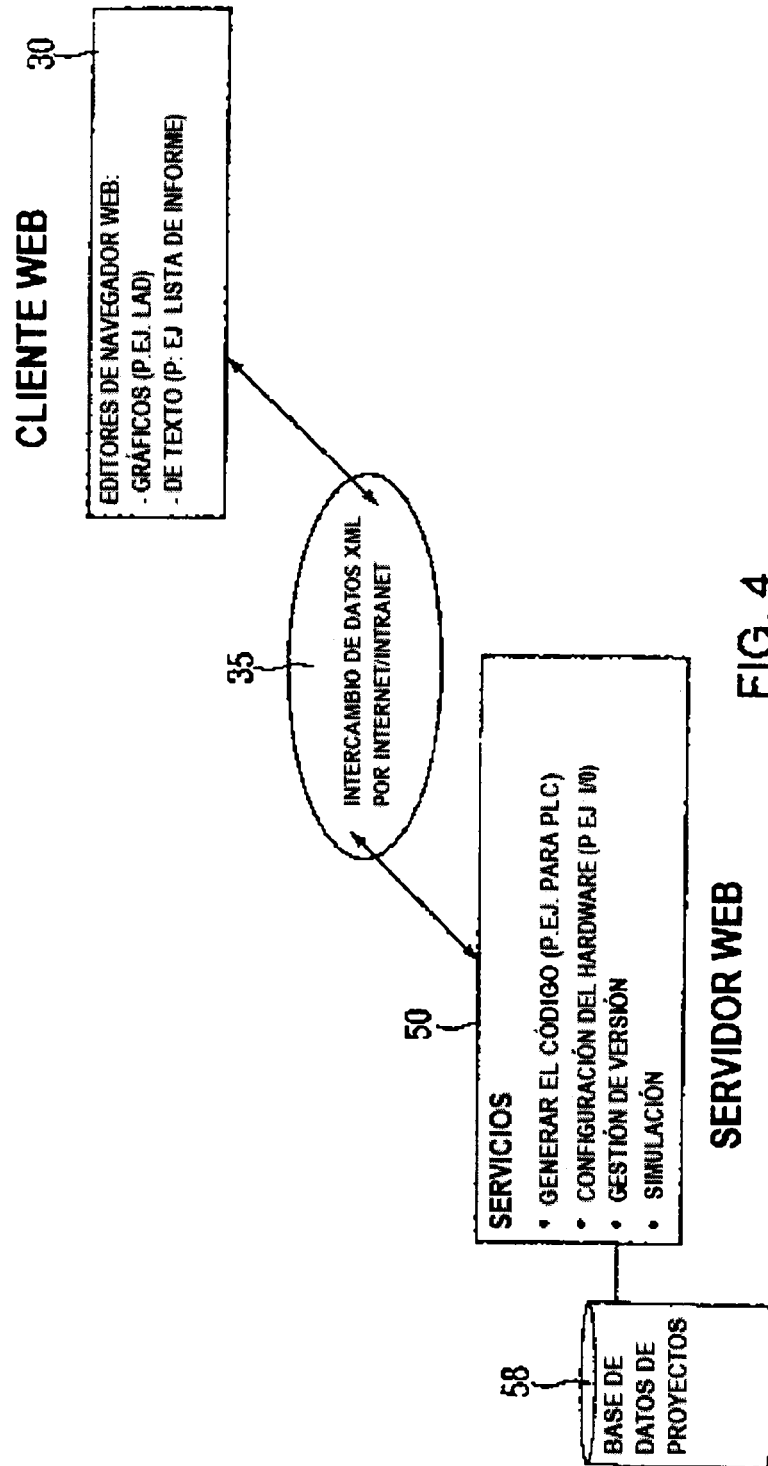


FIG. 4

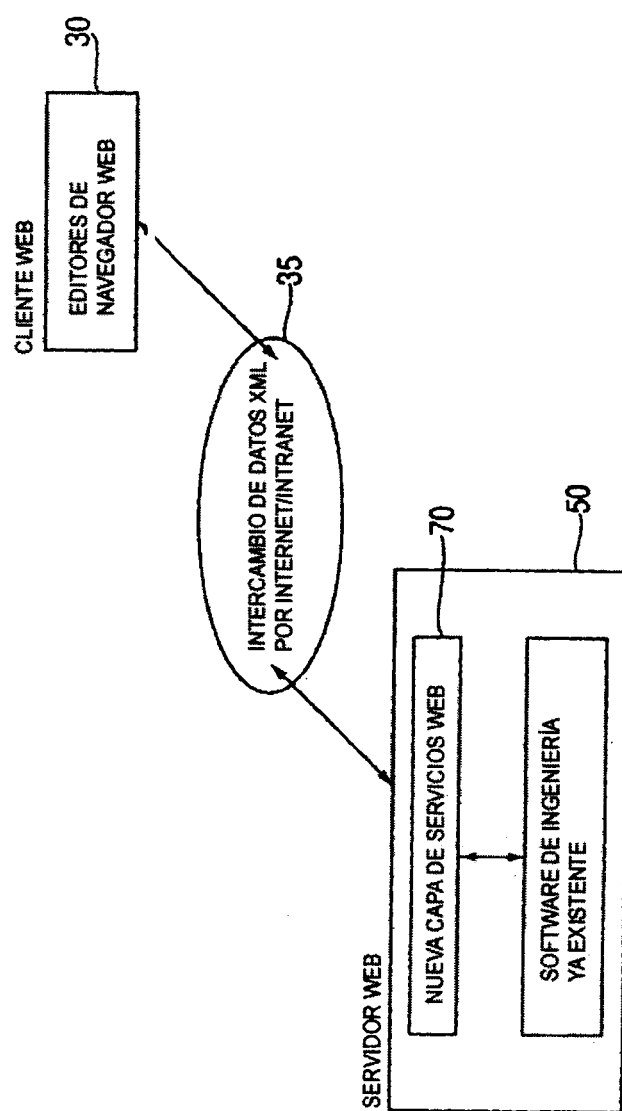
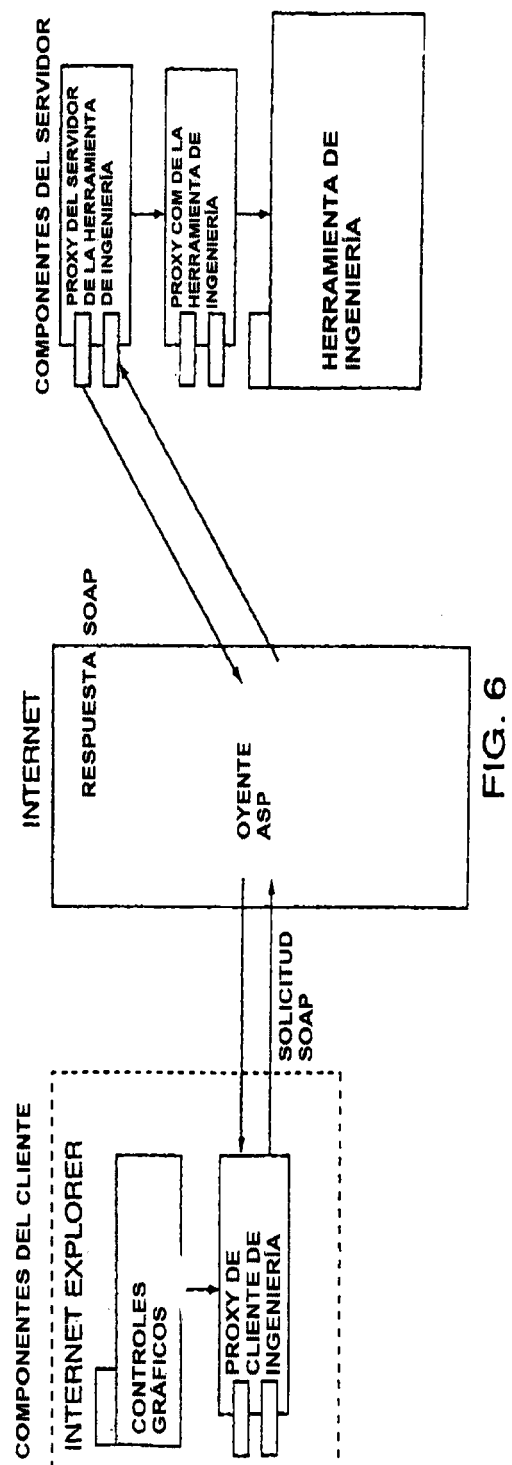


FIG. 5



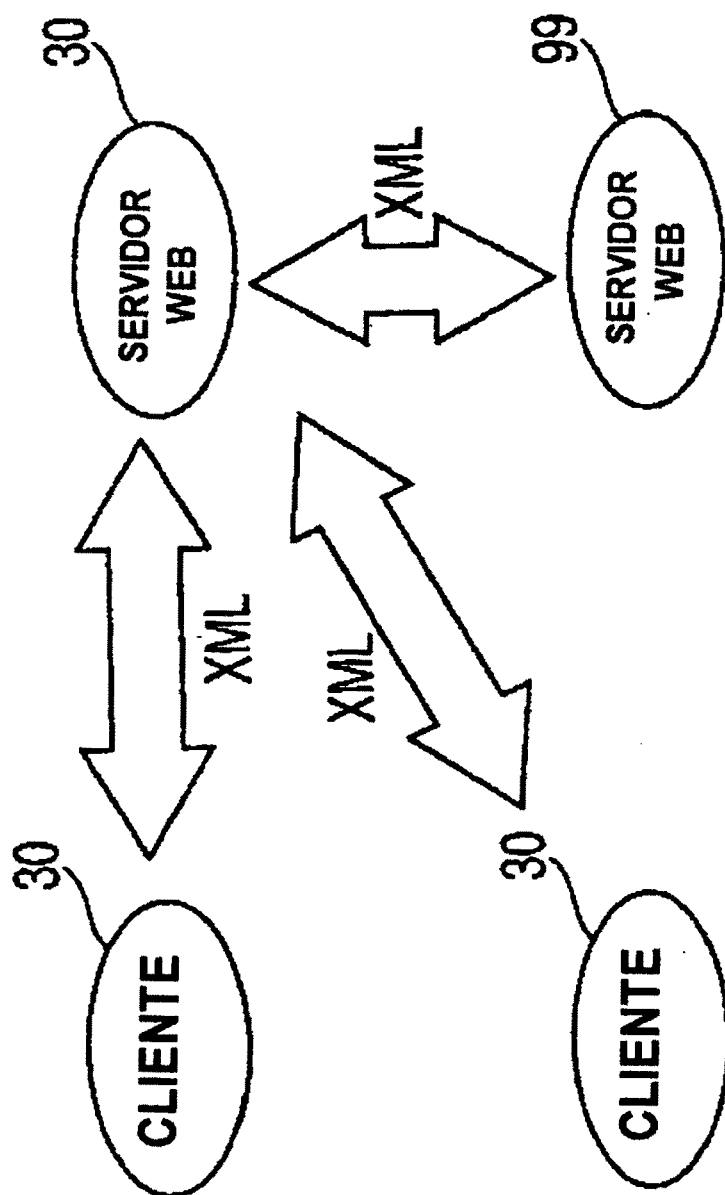


FIG. 7

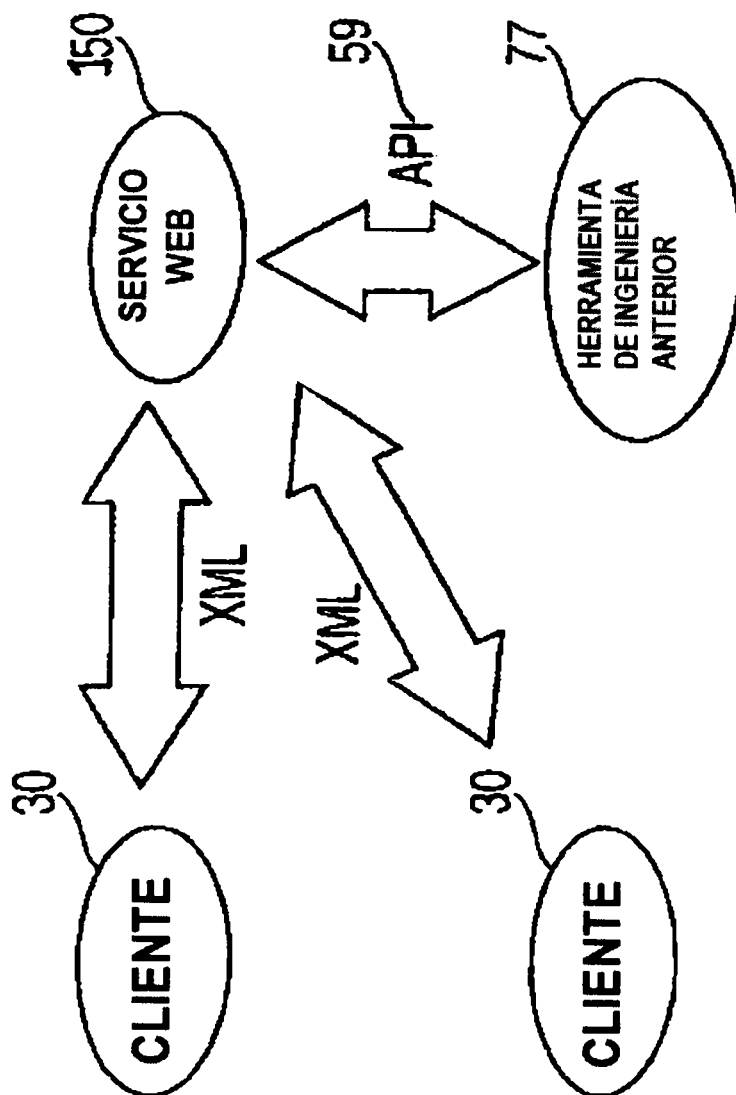


FIG. 8

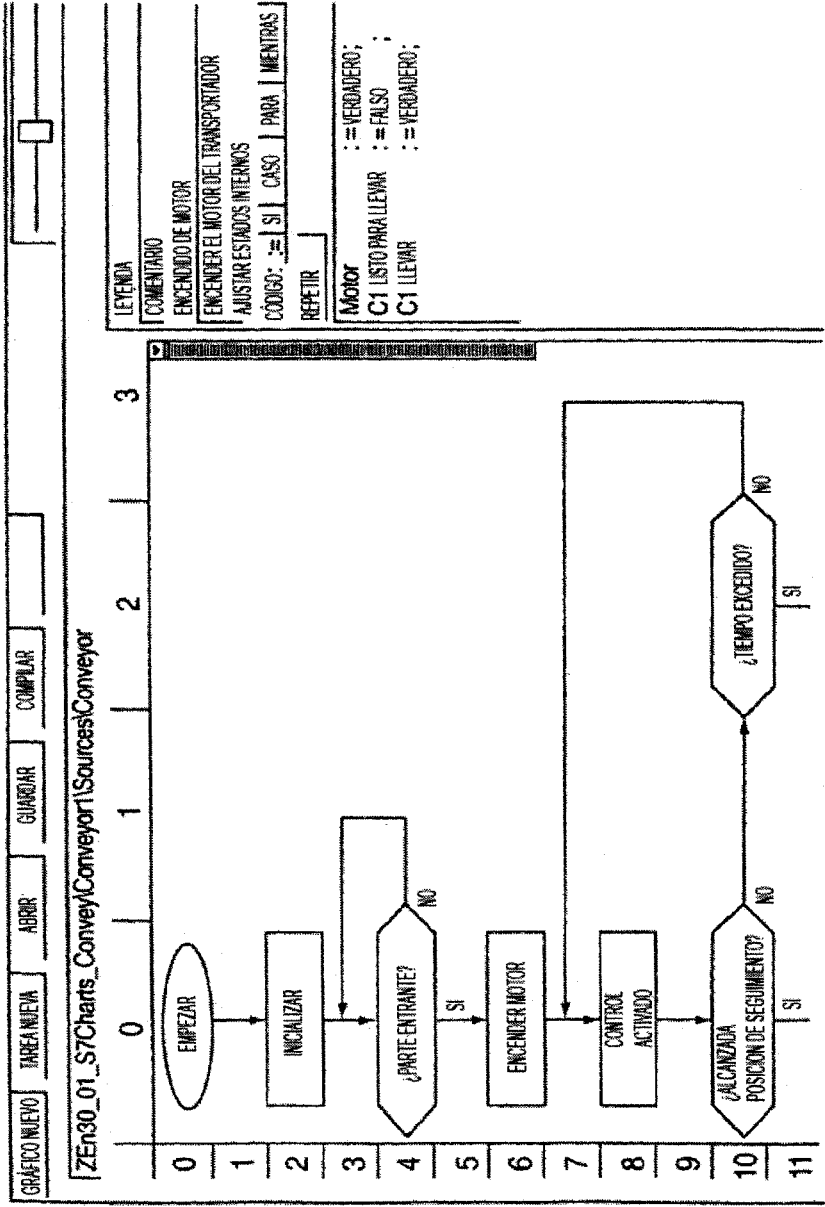


FIG. 9

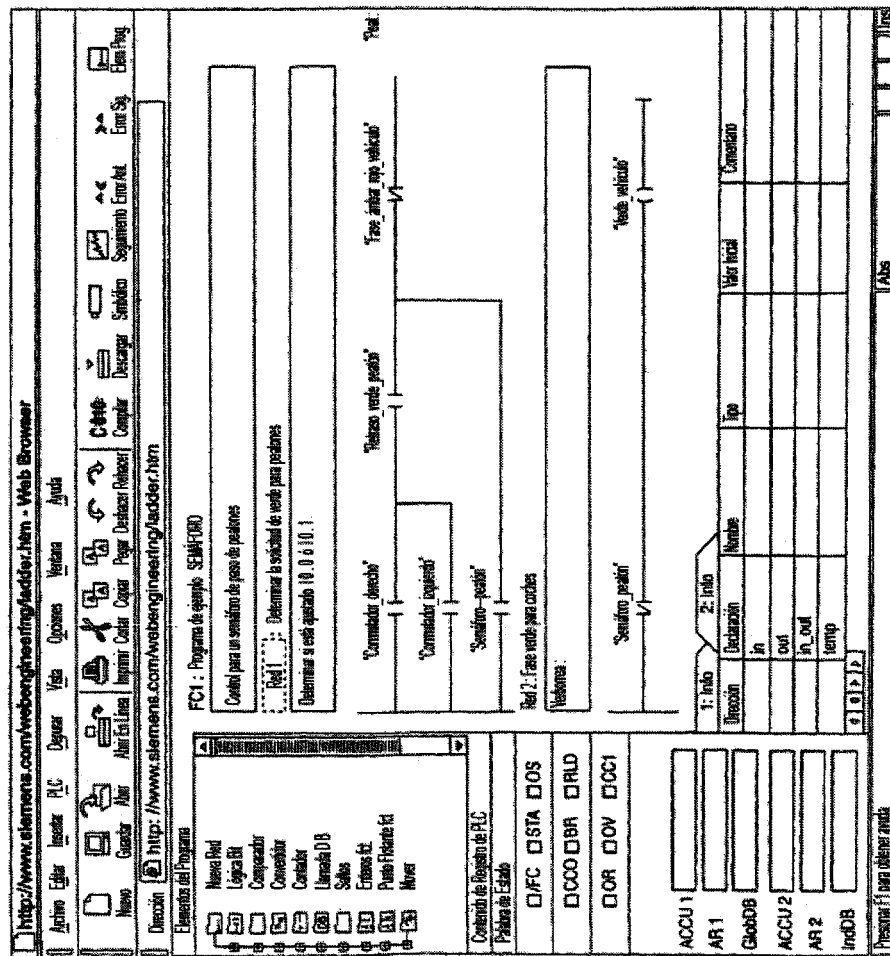


FIG. 10

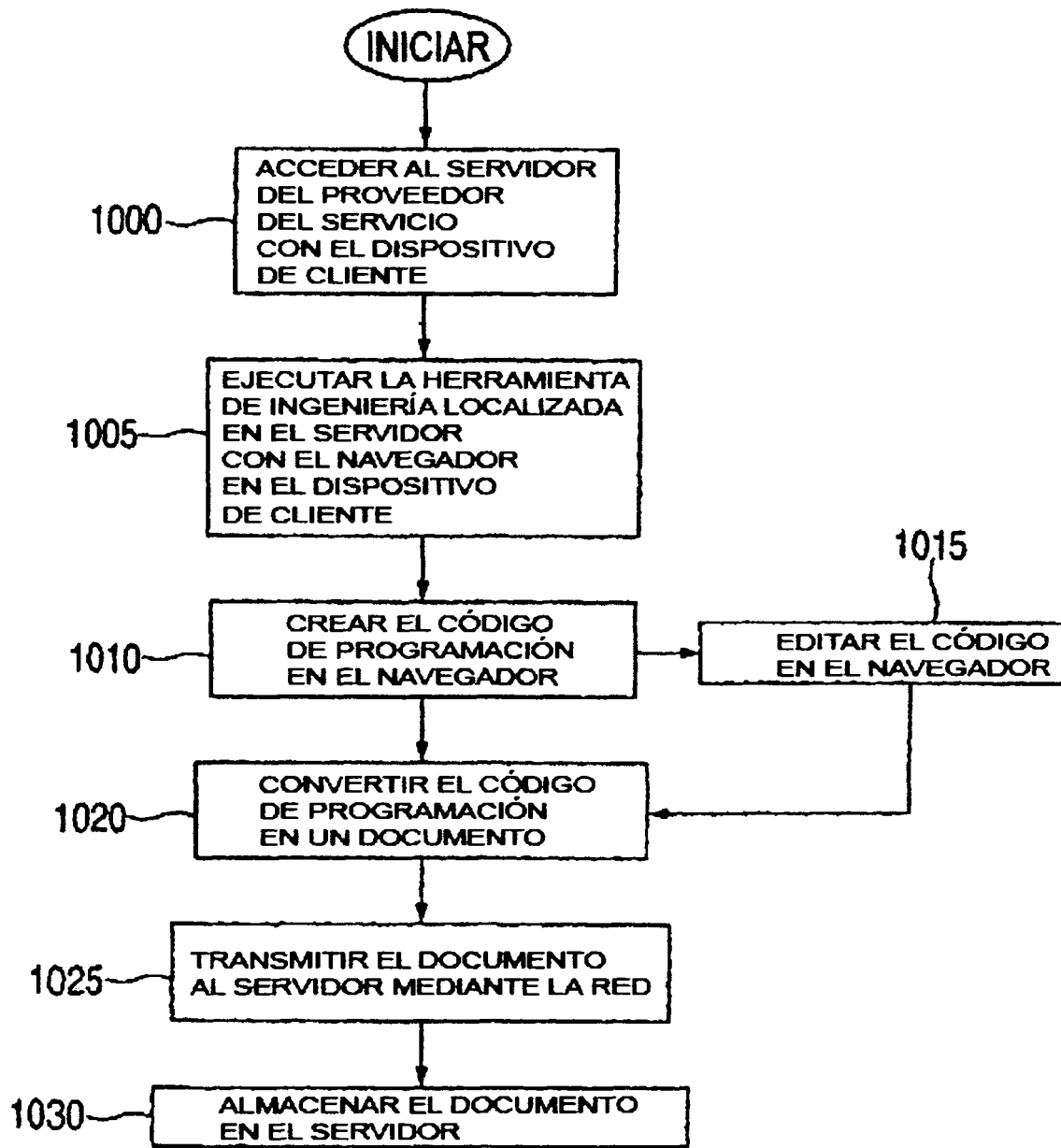


FIG. 11

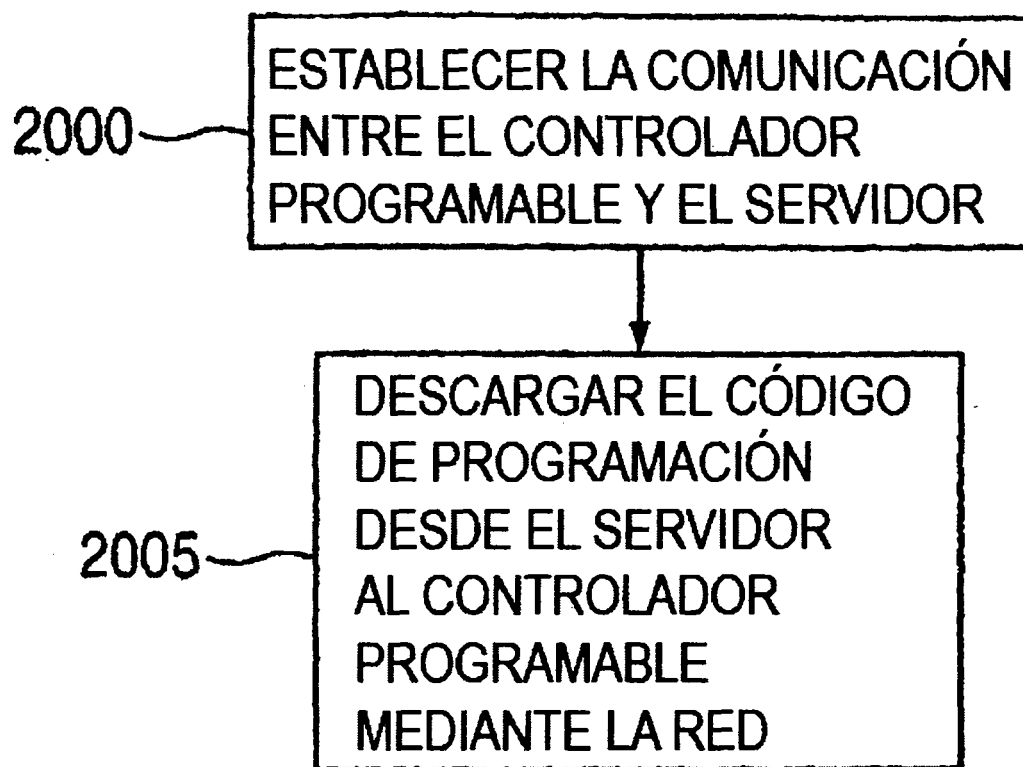


FIG. 12

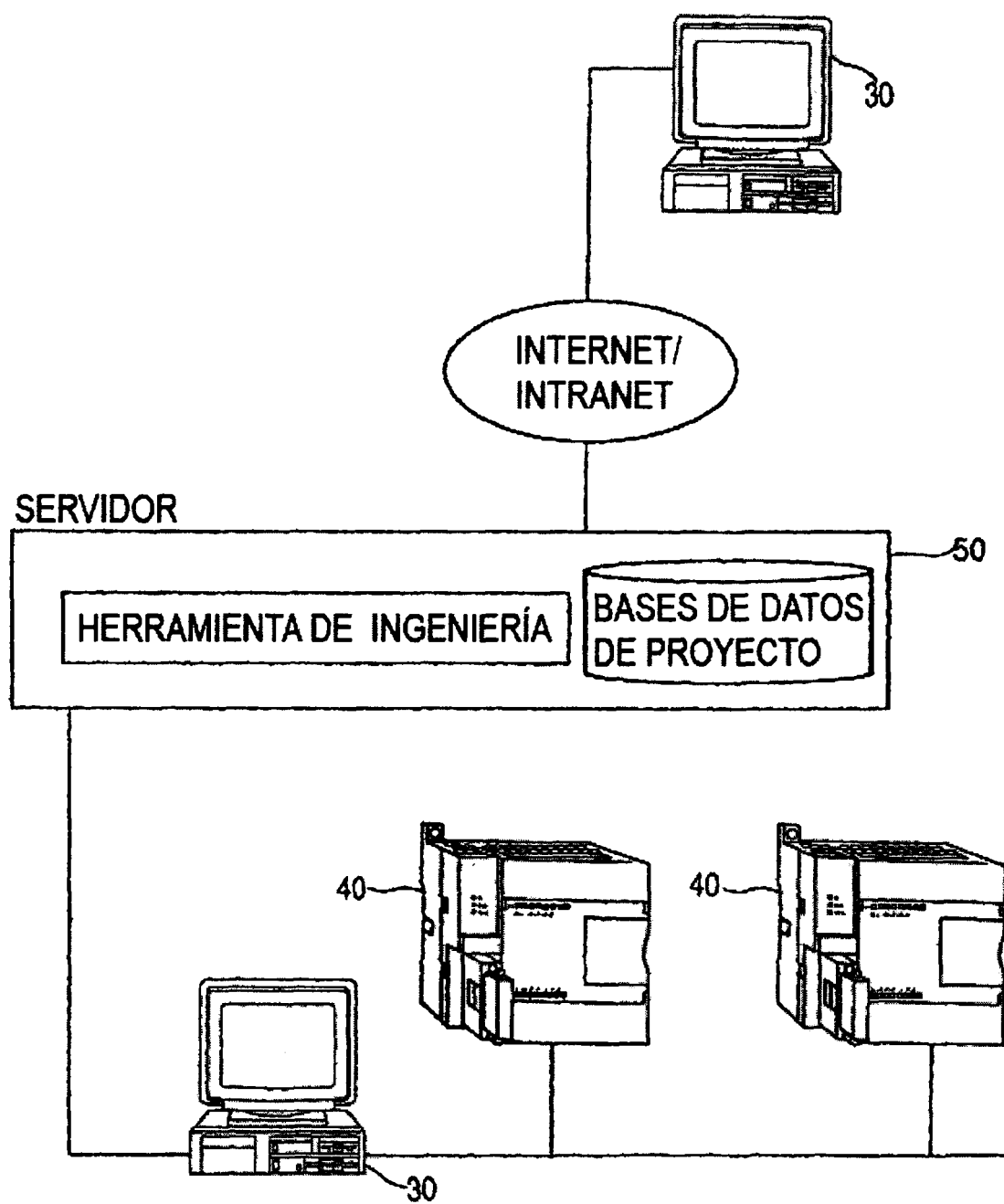


FIG. 13

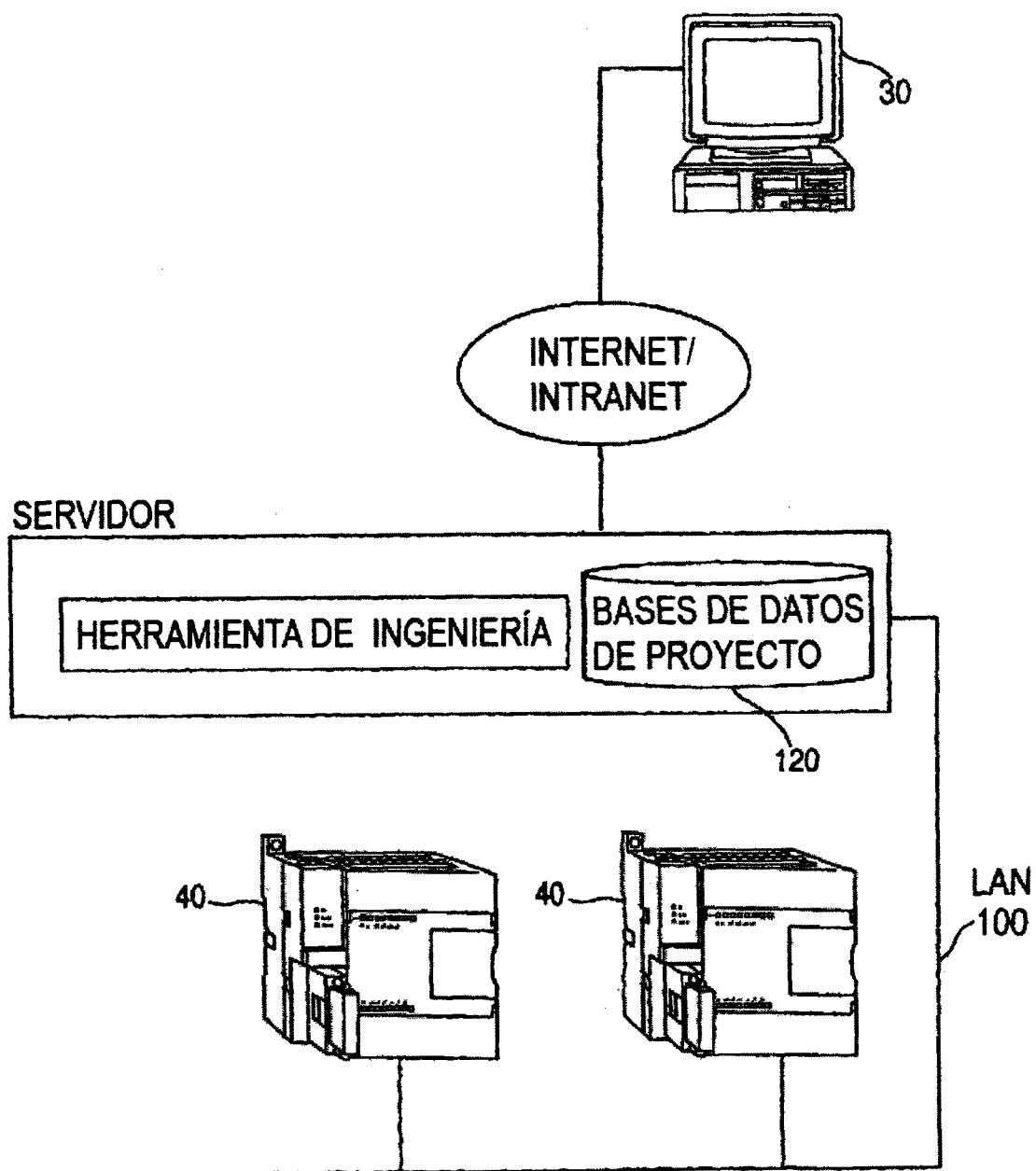


FIG. 14

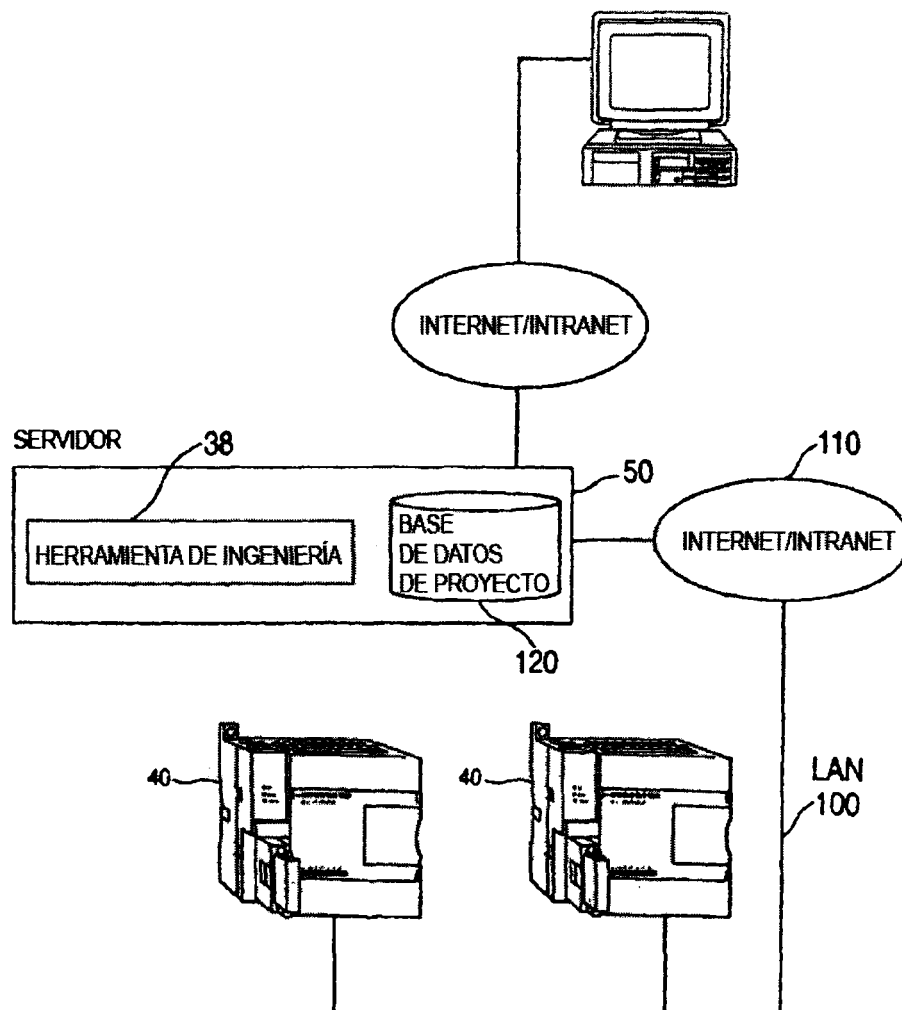


FIG. 15

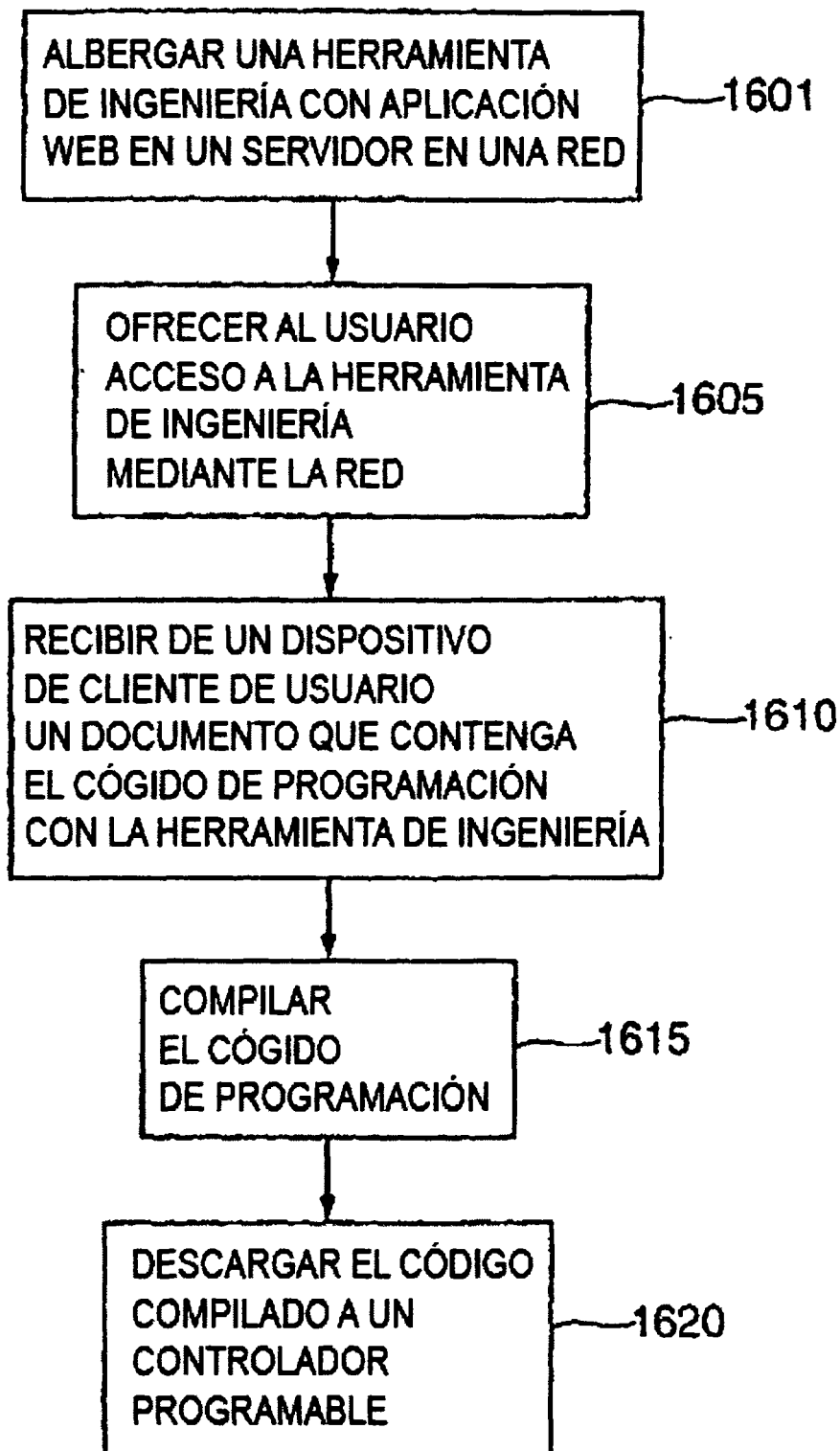


FIG. 16

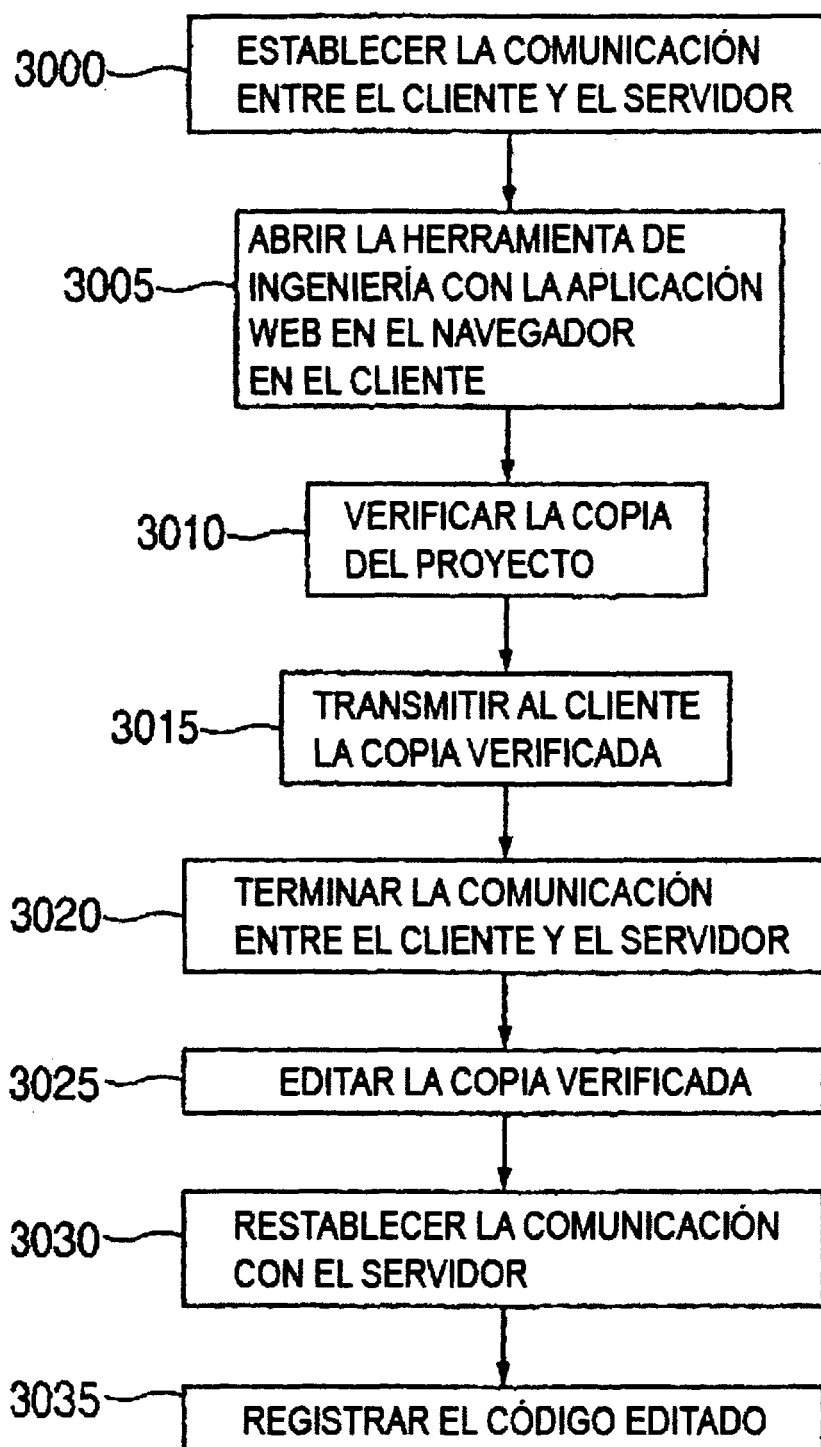


FIG. 17

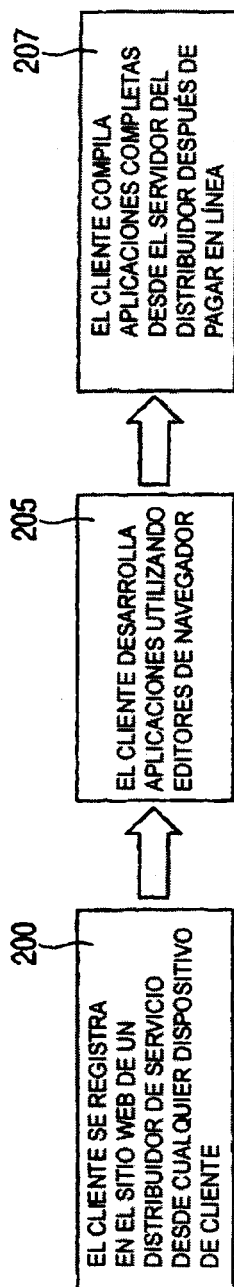


FIG. 18

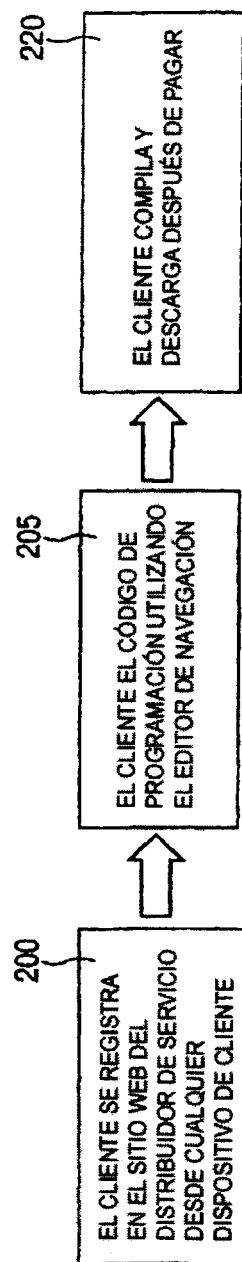


FIG. 19

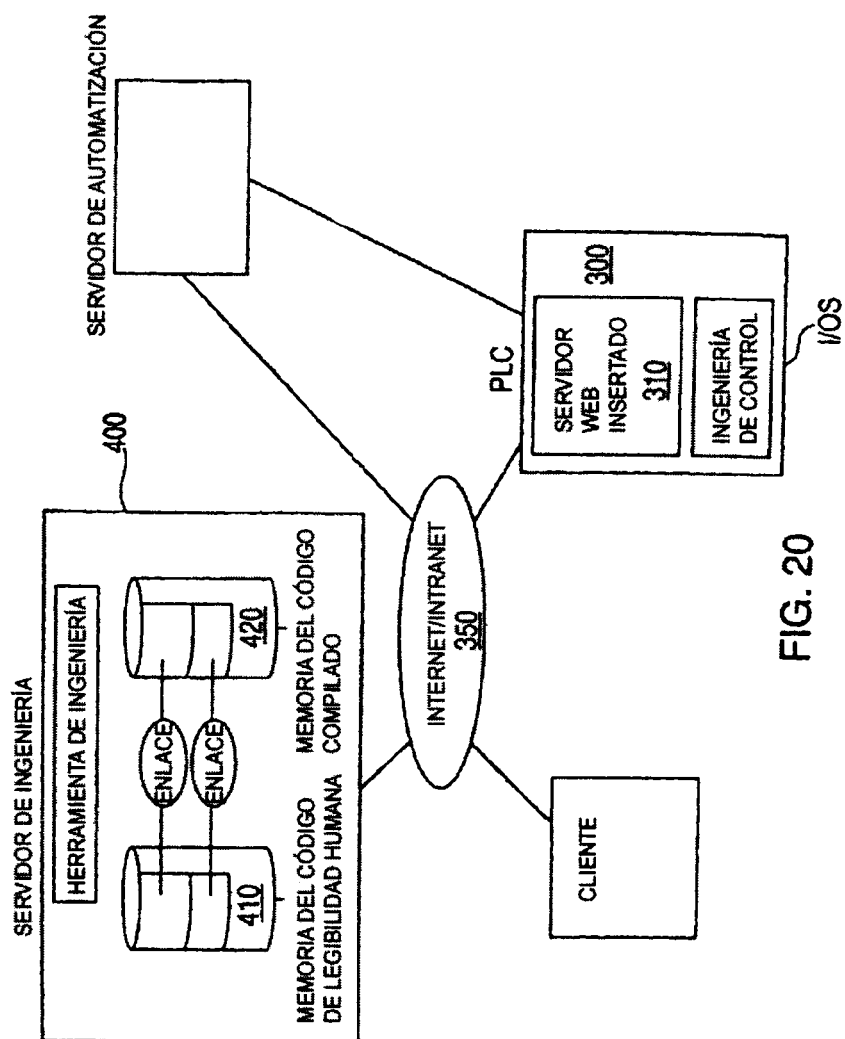
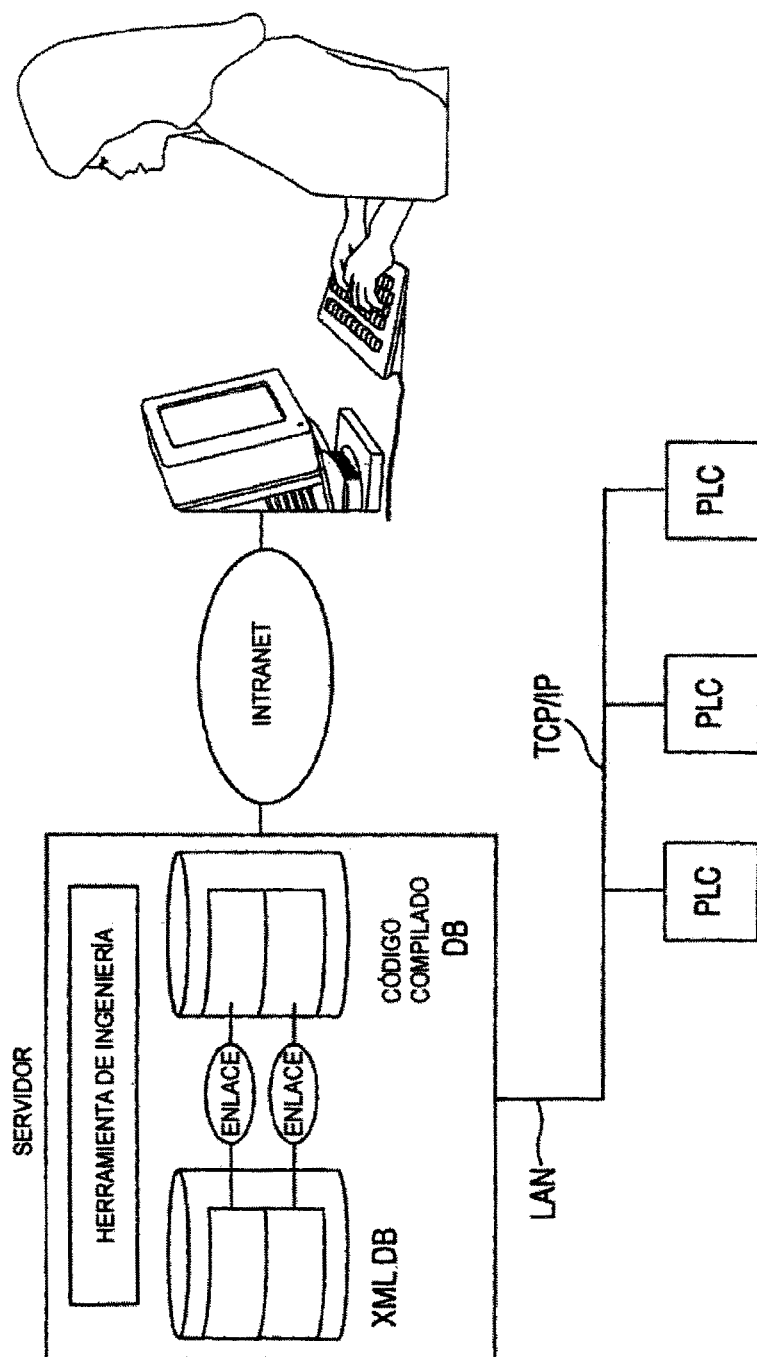


FIG. 20



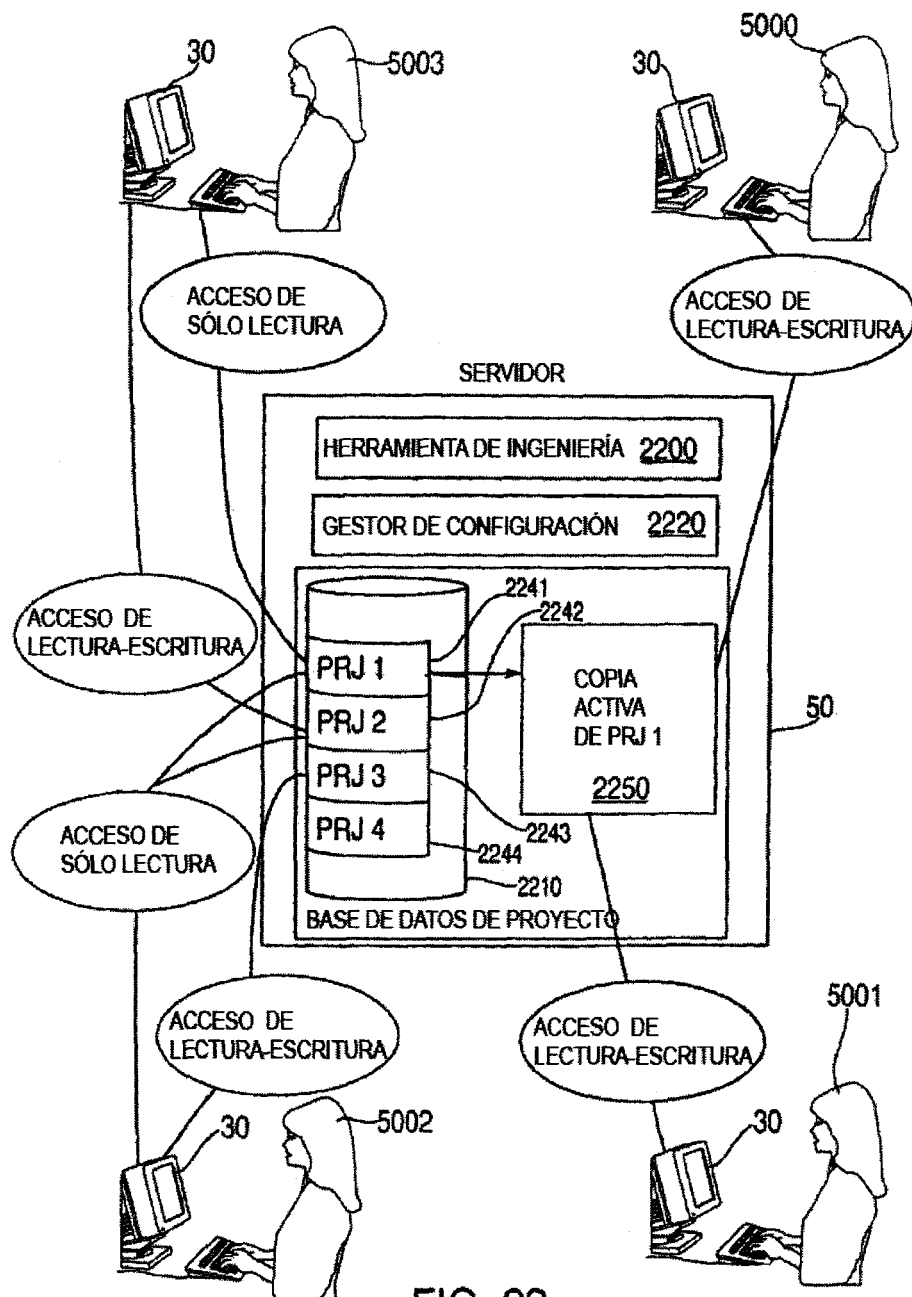


FIG. 22

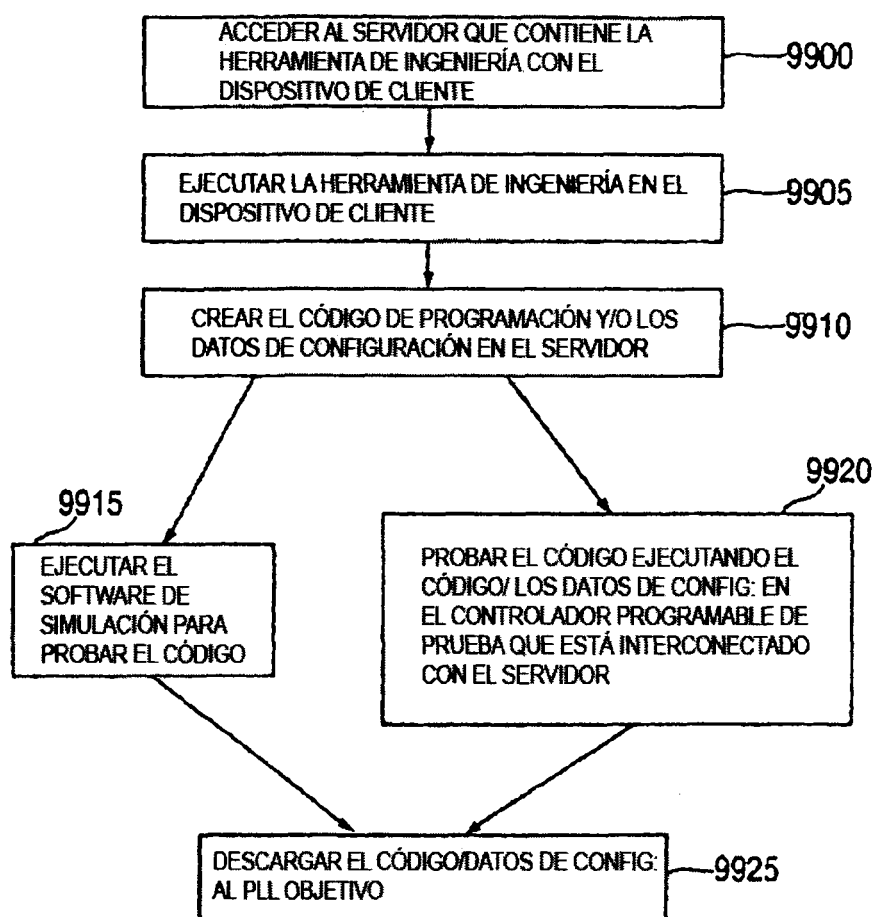


FIG. 23

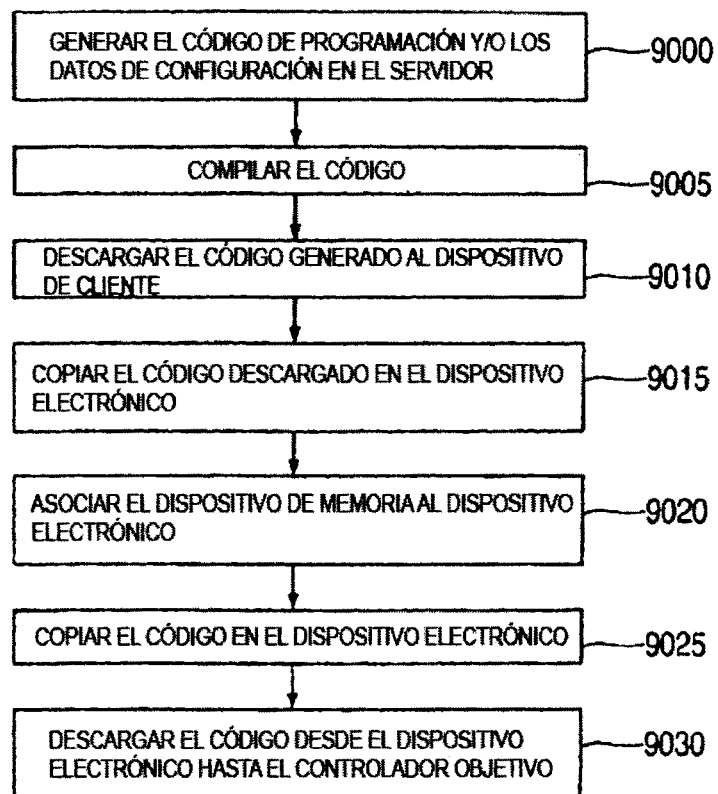


FIG. 24

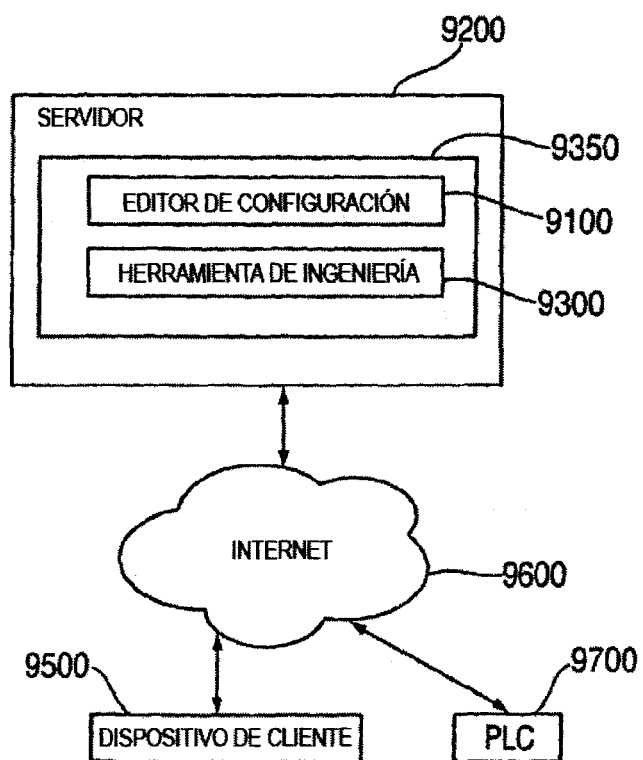


FIG. 25

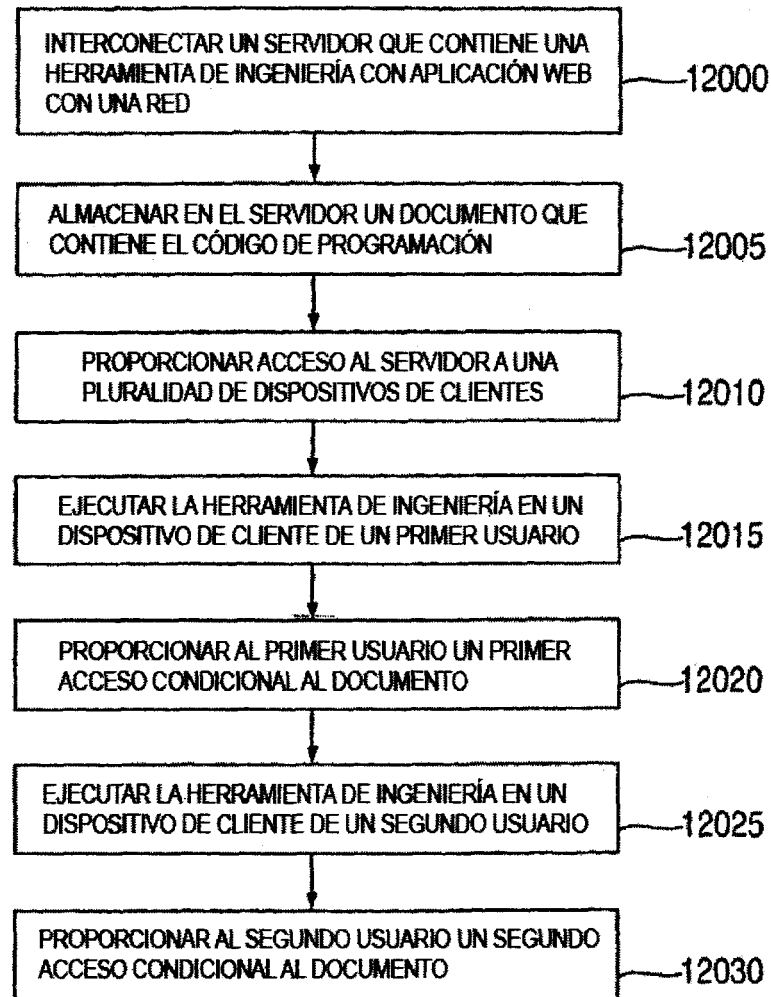


FIG. 26

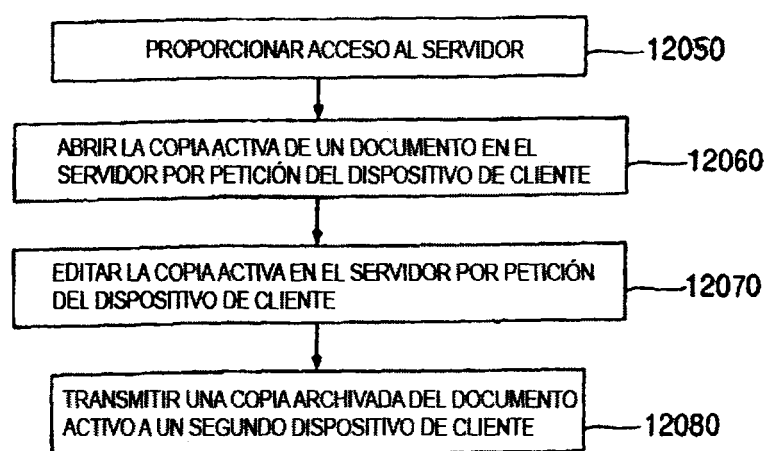


FIG. 27