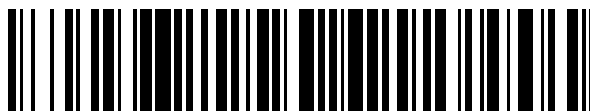


19



OFICINA ESPAÑOLA DE  
PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **2 757 916**

51 Int. Cl.:

**H04L 12/741** (2013.01)

**H04L 12/721** (2013.01)

**H04L 12/851** (2013.01)

**H04L 12/715** (2013.01)

12

TRADUCCIÓN DE PATENTE EUROPEA

T3

96 Fecha de presentación y número de la solicitud europea: **29.11.2013 E 17196071 (9)**

97 Fecha y número de publicación de la concesión europea: **04.09.2019 EP 3300320**

54 Título: **Priorización de paquetes en una red definida por software que implementa OpenFlow**

30 Prioridad:

**29.11.2012 US 201261731389 P**

**25.11.2013 US 201314089295**

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

**30.04.2020**

73 Titular/es:

**HUAWEI TECHNOLOGIES CO., LTD. (100.0%)  
Huawei Administration Building, Bantian,  
Longgang District  
Shenzhen, Guangdong 518129, CN**

72 Inventor/es:

**WANG, JIAO y  
LUO, MIN**

74 Agente/Representante:

**LEHMANN NOVO, María Isabel**

ES 2 757 916 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**

Priorización de paquetes en una red definida por software que implementa OpenFlow

**Antecedentes**

5 Las redes modernas de comunicaciones comprenden nodos, tales como encaminadores, conmutadores, puentes y otros dispositivos, que transportan datos a través de las redes. Con el paso de los años, las redes se han ido haciendo cada vez más complejas, conduciendo a mallas entrelazadas de nodos de red. Como resultado, los proveedores de nodos se han esforzado para personalizar, optimizar y mejorar el rendimiento de los nodos. El trabajo en red definido por software (SDN, software-defined networking) es una tecnología de red emergente que trata dichas personalización, optimización y mejora. SDN simplifica las redes al desacoplar la funcionalidad de envío de datos (es decir, el plano de datos) respecto de la funcionalidad de encaminamiento, recursos y otra funcionalidad de gestión (es decir, el plano de control). Como resultado, mientras que los nodos de red tradicionales pueden proporcionar tanto la funcionalidad del plano de datos como la funcionalidad del plano de control, un nodo SDN puede proporcionar la funcionalidad del plano de datos y un controlador SDN puede proporcionar la funcionalidad del plano de control. Los servicios de interfaz de programación de aplicaciones (API, application programming interface) abierta, tales como OpenFlow, pueden estandarizar las interacciones entre el plano de datos y el plano de control, y pueden permitir la implementación de nodos y controladores que no son específicos de ningún proveedor. Como resultado, SDN conjuntamente con un servicio de API abierta tal como OpenFlow pueden proporcionar beneficios, tales como personalización, optimización y rendimiento mejorados.

20 El documento WO 2012/032684 A1 da conocer un mecanismo para conseguir la extensión del número de entradas de una tabla de flujo abierto utilizando tablas en un conmutador como recursos existentes, en el que cuando el puerto de entrada es el puerto válido de OpenFlow, el conmutador transfiere el paquete desde puerto de entrada a la sección de función de OpenFlow.

**Resumen**

25 La presente invención se define en las reivindicaciones independientes. Las reivindicaciones dependientes definen realizaciones particulares de la invención.

30 En un ejemplo, la invención incluye un aparato OpenFlow de trabajo en red definido por software (SDN), que comprende un procesador, y un sistema de memoria acoplado al procesador y que comprende un conducto de flujo, en el que el conducto de flujo comprende una serie de tablas de flujo, en el que cada una de las tablas de flujo comprende por lo menos un campo de correspondencia, en el que los campos de correspondencia corresponden a una serie de servicios de red, en el que los campos de correspondencia están ordenados en base a una priorización de los servicios de red, a cuáles de los campos de correspondencia son compartidos entre los servicios de red, a una dependencia compartida de los campos de correspondencia y a la velocidad de procesamiento, y en el que la priorización está basada en qué servicios de red son los más importantes y en qué servicios de red se utilizan más frecuentemente.

35 En otro ejemplo, la invención incluye un aparato OpenFlow de trabajo en red definido por software (SDN) que comprende un procesador, y un sistema de memoria acoplado al procesador y que comprende un conducto de flujo, en el que el conducto de flujo comprende una serie de tablas de flujo numeradas de 0 a 11, en el que la tabla de flujo 0 y la tabla de flujo 1 están configuradas para implementar un servicio de lista de control de acceso (ACL, access control list), en el que la tabla de flujo 2 está configurada para implementar un servicio de localización de servicios, en el que la tabla de flujo 3 está configurada para implementar un servicio de envío de la capa dos, en el que la tabla de flujo 2, la tabla de flujo 3 y la tabla de flujo 4 están configuradas para implementar un servicio de red de área local virtual (VLAN, virtual local area network), en el que la tabla de flujo 2 y la tabla de flujo 5 están configuradas para implementar un servicio de conmutación por etiquetas multiprotocolo (MPLS, multiprotocol label switching), en el que la tabla de flujo 2, la tabla de flujo 6, la tabla de flujo 7 y la tabla de flujo 8 están configuradas para implementar un servicio de versión 4 de protocolo de internet (IPv4), en el que la tabla de flujo 2, la tabla de flujo 6, la tabla de flujo 9 y la tabla de flujo 10 están configuradas para implementar un servicio de protocolo de internet versión 6 (IPv6) y en el que la tabla de flujo 11 está configurada para implementar por lo menos un servicio adicional.

50 En otro ejemplo más, la invención incluye un procedimiento relacionado con OpenFlow de trabajo en red definido por software (SDN), comprendiendo el procedimiento seleccionar una serie de servicios de red, priorizar los servicios de red en base a qué servicios de red son los más importantes y a qué servicios de red son los más frecuentemente utilizados, determinar un campo de correspondencia correspondiente a cada uno de los servicios de red, y diseñar un conducto de flujo con una serie de tablas de flujo, en el que cada una de las tablas de flujo comprende por lo menos uno de los campos de correspondencia, y en el que los campos de correspondencia están ordenados en base a la priorización, a cuáles de los campos de correspondencia son compartidos entre los servicios de red, a la dependencia compartida de los campos de correspondencia y a la velocidad de procesamiento.

55 Estas y otras características se comprenderán más claramente a partir de la siguiente descripción detallada, tomada junto con los dibujos adjuntos y las reivindicaciones.

Breve descripción de los dibujos

Para una comprensión más completa de esta invención, se hace referencia a continuación a la siguiente breve descripción, tomada en relación con los dibujos adjuntos y la descripción detallada, donde los numerales de referencia similares representan partes similares.

5 La figura 1 es un diagrama esquemático de una red definida por software, de acuerdo con una realización de la invención.

La figura 2 es un diagrama esquemático de un dispositivo de red, según una realización de la invención.

La figura 3 es un diagrama esquemático de un conducto de flujo generalizado.

10 La figura 4 es un diagrama esquemático de una realización de un conducto de flujo para implementar calidad de servicio (QoS, quality of service).

La figura 5 es un diagrama esquemático de una realización de un conducto de flujo para implementar QoS.

La figura 6 es un diagrama esquemático de una realización de un conducto de flujo para implementar encaminamiento y envío virtuales (VRF, routing and forwarding).

15 La figura 7 es un diagrama esquemático de una realización de un conducto de flujo para implementar un cortafuegos.

La figura 8 es un diagrama esquemático de una realización de un conducto de flujo para implementar conmutación por etiquetas multiprotocolo (MPLS).

La figura 9 es un diagrama de dependencias compartidas de campos de correspondencia.

La figura 10 es un diagrama esquemático de un conducto de flujo, según una realización de la invención.

20 La figura 11 es un diagrama esquemático que muestra conmutación y envío de la capa dos en el conducto de flujo de la figura 10.

La figura 12 es un diagrama esquemático que muestra conmutación y envío de la capa tres en el conducto de flujo de la figura 10.

La figura 13 es un diagrama esquemático de un conducto de flujo, según otra realización de la invención.

25 La figura 14 es un diagrama de flujo que muestra un procedimiento, según una realización de la invención.

Descripción detallada

30 Para empezar se debe entender que, aunque a continuación se da a conocer una implementación ilustrativa de una o varias realizaciones, los sistemas y/o procedimientos dados a conocer se pueden implementar utilizando cualquier número de técnicas, ya sean actualmente conocidas o existentes. La invención no se deberá limitar en modo alguno a las implementaciones, dibujos y técnicas ilustrativas mostradas a continuación, incluyendo los diseños e implementaciones a modo de ejemplo mostrados y descritos en la presente memoria, sino que se puede modificar dentro del alcance de las reivindicaciones adjuntas junto con toda la gama de sus equivalentes.

35 Las redes pueden tener nodos con tablas de encaminamiento. Una tabla de encaminamiento puede comprender instrucciones de encaminamiento que instruyen a un nodo cómo encaminar datos a destinos de red. Los nodos de red basados en protocolo de internet (IP, Internet Protocol) tradicionales pueden tener tablas de encaminamiento relativamente grandes que comprenden muchas, posiblemente millones de instrucciones de encaminamiento. Cada una de las instrucciones de encaminamiento puede tener, por ejemplo, cinco atributos. Por lo tanto, la velocidad de entrega de los datos de dichos nodos puede ser relativamente más lenta debido a que la velocidad entrega de los datos puede depender estrechamente de la velocidad de búsqueda a través de las grandes tablas de encaminamiento. Además, dichas tablas de encaminamiento pueden estar implementadas en memoria ternaria de contenido direccionable (TCAM, ternary content-addressable memory), y la TCAM puede consumir relativamente más potencia, coste y espacio en comparación con otros tipos de memoria.

45 La especificación de conmutador OpenFlow, versión 1.0.0 (OpenFlow 1.0.0) adoptó una única tabla de encaminamiento, o de flujo, y por lo tanto heredó los problemas descritos anteriormente. Una tabla de flujo se puede definir como una tabla de encaminamiento utilizada para realizar un flujo de paquetes de datos. El flujo de paquetes puede comprender búsqueda, envío y modificación de paquetes. Un paquete se puede definir como una estructura de datos de Ethernet, IP u otra, que comprende una cabecera y una carga útil. Además, OpenFlow 1.0.0 proporciona entradas de flujo de 12 tuplas o 12 elementos. Las entradas de flujo pueden comprender datos para comparar con un paquete con estos, así como instrucciones para procesar el paquete en base al resultado de la comparación. Por  
50 lo tanto, la implementación puede requerir la duplicación de muchos atributos a través de las entradas de cinco-

tuplas, agotando de ese modo el espacio de las tablas de flujo y los recursos de hardware. Además, OpenFlow 1.0.0 puede requerir la implementación de una capa de abstracción del hardware (HAL, hardware abstraction layer) compleja. Una HAL se puede definir como una capa de abstracción, lo que significa que permite al software funcionar perfectamente con diferentes tipos de hardware. Por consiguiente, aunque los controladores de OpenFlow 1.0.0 pueden proporcionar instrucciones de encaminamiento para nodos de manera centralizada, estas instrucciones de encaminamiento pueden seguir siendo ineficientes. Finalmente, OpenFlow 1.0.0 puede no aprovechar la capacidad de reconocer tipos de aplicación o de flujo, ni aprovechar otro potencial en una red SDN-OpenFlow.

En lugar de nodos que tienen una única tabla de flujo, la especificación de conmutador OpenFlow, versión 1.1.0 (OpenFlow 1.1.0), que se incorpora como referencia como si se reprodujera en su totalidad, introduce nodos que tienen conductos de flujo con múltiples tablas de flujo. Cada tabla de flujo comprende entradas de flujo, que comprenden campos de correspondencia. Un conducto de flujo se puede definir como un conjunto de tablas de flujo conectadas. Un campo de correspondencia se puede definir como un campo con el que se pone en correspondencia un paquete. Sin embargo, la introducción de conductos de flujo puede hacer más complejos el diseño y la implementación de los nodos. Por ejemplo, puede ser difícil separar servicios de red tales como QoS y encaminamiento. Asimismo, existe un desafío en diseñar conductos de flujo que utilicen de manera efectiva los recursos, tales como la memoria de acceso aleatorio (RAM, random access memory) y la TCAM, y que sean flexibles, fáciles de implementar y capaces de buscar rápidamente correspondencias. Para facilitar la adopción de OpenFlow para grandes redes, existe asimismo la necesidad de extender la abstracción de las tablas del núcleo y establecer conductos de flujo flexibles, pero optimizados, con múltiples tablas de flujo que proporcionen de manera efectiva lo siguiente: (1) encaminamiento de internet del núcleo con políticas QoS; (2) encaminamiento conducido por tipos de aplicación o características de flujo; (3) encaminamiento utilizando el hardware actual pero tablas TCAM relativamente menores; y (4) capacidades de conmutación de la capa dos y la capa tres. La capa dos se puede referir a la capa de enlace de datos en el modelo de interconexión de sistemas abiertos (OSI, Open Systems Interconnection). La capa tres se puede referir a la capa de red en el modelo OSI. A partir de OpenFlow 1.1.0, se han implementado las subsiguientes especificaciones de OpenFlow. La especificación de conmutador OpenFlow, versión 1.3.1 (OpenFlow 1.3.1), que se incorpora como referencia como si se reprodujera en su totalidad, es la especificación de OpenFlow más reciente. Se debe entender que aunque OpenFlow 1.3.1 es la especificación de OpenFlow más reciente, la técnica dada conocer no se limita a OpenFlow 1.3.1.

En la presente memoria se dan a conocer sistemas y procedimientos para un diseño mejorado de conducto de flujo en conmutadores basados en OpenFlow SDN y dispositivos similares. En general, la técnica dada a conocer se puede centrar en servicios de red clave, priorizar dichos servicios de red clave y seleccionar para las tablas de flujo una secuencia de campos de correspondencia que puede ser necesaria para proporcionar dichos servicios de red clave. Más específicamente, la técnica dada a conocer se puede centrar en optimizar el tiempo de procesamiento (búsqueda y puesta en correspondencia para campos de correspondencia) para servicios importantes y utilizados frecuentemente, tales como lista de control de acceso (ACL), envío de la capa dos, red de área local virtual (VLAN), MPLS y red de envío de la capa tres. Además, la técnica dada a conocer puede proporcionar campos de correspondencia adicionales para localizar diferentes tipos de servicio con el fin de acelerar el envío. Se puede utilizar una tabla de flujo final para incluir campos de correspondencia relacionados con otros servicios más. Por lo tanto, la técnica dada a conocer puede proporcionar encaminamiento basado en OpenFlow SDN con políticas QoS; encaminamiento conducido por tipos de aplicación o características de flujo; encaminamiento utilizando el hardware actual, pero con tablas TCAM relativamente más pequeñas; y capacidades de conmutación de la capa dos y la capa tres.

La figura 1 es un diagrama esquemático de una red definida por software 100, según una realización de la invención. La red 100 puede comprender un controlador 110, un canal 130, un nodo 140 y una serie de puntos extremos 160.

El controlador 110 puede ser cualquier dispositivo o proceso de comunicación capaz de comunicar con el nodo 140. Por ejemplo, el controlador 110 puede ser un servidor informático que pueda recibir instrucciones de un usuario por medio de una interfaz de usuario 120 y transmitir dichas instrucciones al nodo 140. El controlador 110 puede gestionar el nodo 140 por medio de OpenFlow.

El canal 130 puede ser cualquier canal que pueda proporcionar un trayecto de comunicación entre el controlador 110 y el nodo 140. Por ejemplo, el canal 130 puede ser un medio físico o una conexión lógica. El canal 130 puede ser un canal seguro y puede utilizar OpenFlow para facilitar la comunicación entre el controlador 110 y el nodo 140.

El nodo 140 puede ser cualquier dispositivo de comunicación capaz de comunicar con el controlador 110 y entre los puntos extremos 160. Por ejemplo, el nodo 140 puede ser un encaminador, un conmutador, un puente u otro dispositivo similar. El nodo 140 puede recibir paquetes desde un punto extremo 160 y encaminar dichos paquetes a otro punto extremo 160 en base a instrucciones recibidas desde el controlador 110 e implementadas en un módulo de conducto de flujo (FPM, flow pipeline module) 150, que se describirá en mayor detalle a continuación.

Los puntos extremos 160 pueden ser cualesquiera dispositivos de comunicación capaces de comunicar con el nodo 140 y entre sí. Por ejemplo, los puntos extremos 160 pueden ser un teléfono móvil, un ordenador de tableta, un ordenador de sobremesa, un ordenador portátil u otro dispositivo similar.

La figura 2 es un diagrama esquemático de un dispositivo de red 200, según una realización de la invención. El dispositivo de red 200 puede comprender una serie de puertos de entrada 210 y/o unidades de receptor (Rx) 220 para datos, un procesador o unidad lógica 230 para procesar señales, una serie de puertos de salida 240 y/o unidades de transmisor (Tx) 250 para transmitir datos a otros componentes, y una memoria 260. El dispositivo de red 200 puede ser adecuado para implementar las características, procedimientos y dispositivos descritos en la presente memoria, incluyendo el nodo 140. Por ejemplo, si el dispositivo de red 200 está implementando el nodo 140, entonces los puertos de entrada 210 y los puertos de salida 240 pueden estar acoplados al canal 130 y a los puntos extremos 160. Además, el procesador 230 y/o la memoria 260 pueden comprender el FPM 150.

El procesador 230, que se puede denominar una unidad central de proceso (CPU, central processing unit), puede estar en comunicación con los puertos de entrada 210, las unidades de receptor 220, los puertos de salida 240, las unidades de transmisor 250 y la memoria 260. El procesador 230 puede estar implementado como uno o varios chips de CPU, núcleos (por ejemplo, un procesador multi-núcleo), matrices de puertas programables in situ (FPGA, field-programmable gate arrays), circuitos integrados de aplicación específica (ASIC, application specific integrated circuits) y/o procesadores de señal digital (DSP, digital signal processors) y/o puede formar parte de uno o varios ASIC.

La memoria 260 se puede componer de uno o varios discos, unidades de cinta o unidades de estado sólido; puede ser utilizada para almacenamiento no volátil de datos y como un dispositivo de almacenamiento de datos en desbordamiento; puede ser utilizada para almacenar programas cuando dichos programas son seleccionados para ejecución; y puede ser utilizada para almacenar instrucciones y quizás datos que se lean durante la ejecución del programa. La memoria 260 puede ser volátil y/o no volátil y puede ser memoria de sólo lectura (ROM, read only memory), RAM, TCAM, memoria estática de acceso aleatorio (SRAM, static random-access memory) o cualquier combinación de las mismas. La memoria 260 puede estar situada en un chip con los otros componentes del dispositivo de red 200 o situada en un chip independiente.

La figura 3 es un diagrama esquemático de un conducto de flujo generalizado 300. El conducto de flujo 300 es de OpenFlow 1.3.1. El conducto de flujo 300 puede ser implementado en el nodo 140 y, en particular, en el FPM 150. El conducto de flujo 300 puede comprender tablas de flujo numeradas de 0 a n, donde n es un entero igual o mayor que uno. Cada tabla puede comprender cualquier número de entradas de flujo. El controlador 110 puede añadir, actualizar y eliminar dichas entradas de flujo en cualquier momento. Las entradas de flujo pueden comprender seis atributos: campos de correspondencia, una prioridad, contadores, instrucciones, límites de tiempo y una cookie. Los campos de correspondencia se han definido anteriormente. La prioridad puede definir una precedencia de correspondencia de una entrada de flujo. Los contadores se pueden actualizar cuando se ponen en correspondencia paquetes. Las instrucciones informan al conducto de flujo 300 de cómo procesar un paquete si se produce una correspondencia. Los límites de tiempo se pueden referir a la cantidad máxima de tiempo que un paquete se puede encaminar a través del conducto de flujo 300 antes de expirar. Una cookie puede no utilizarse para paquetes de proceso, pero puede utilizarse para filtrar estadísticas de flujo y para modificación y eliminación. La técnica dada a conocer se puede centrar en campos de correspondencia de entradas de flujo, contadores e instrucciones.

Un paquete puede entrar en el conducto de flujo 300 por medio de un puerto de entrada. El puerto de entrada y cualesquiera puertos subsiguientes pueden ser puertos físicos o lógicos. El flujo de paquetes puede ser unidireccional, de tal modo que un paquete progresa secuencialmente de la tabla 0 a la tabla n. El paquete puede comenzar siempre en la tabla 0. En la tabla 0, el FPM 150 puede extraer del paquete una cabecera del paquete y comparar un valor de la cabecera con cada campo de correspondencia en cada entrada de flujo en la tabla 0. La comparación puede tener uno de tres resultados: el valor se corresponde con un campo de correspondencia, el valor se corresponde con más de un campo de correspondencia o el valor no se corresponde con ningún campo de correspondencia. Si el valor se corresponde con un campo de correspondencia, entonces el FPM 150 puede actualizar un contador de dicho campo de correspondencia, ejecutar instrucciones asociadas con dicho campo de correspondencia y enviar el paquete según dichas instrucciones. Por ejemplo, el FPM 150 puede enviar el paquete a la tabla 1. Si el valor se corresponde con más de un campo de correspondencia, el FPM 150 puede determinar qué campo de correspondencia tiene la máxima prioridad para la tabla 0, actualizar un contador del campo de correspondencia de máxima prioridad, ejecutar instrucciones asociadas con dicho campo de correspondencia y enviar el paquete según dichas instrucciones. Por ejemplo, el FPM 150 puede enviar el paquete a la tabla 1. Si el valor no se corresponde con ningún campo de correspondencia, el FPM 150 puede entonces enviar el paquete al controlador 110, desechar el paquete o enviar el paquete según una instrucción de inexistente en la tabla ("table-miss").

Las instrucciones pueden ordenar cambios en el paquete, un conjunto de acciones asociadas con el paquete, metadatos asociados con el paquete o cualquier combinación de los mismos. Los cambios en el paquete lo pueden ser en los datos incluidos en el propio paquete. El conjunto de acciones se puede definir como el conjunto de acciones de procesamiento aplicadas a un paquete cuando éste sale del conducto de flujo 300. Los metadatos se pueden definir como valores de registro que se pueden enmascarar, utilizados para llevar información de una tabla a la siguiente tabla. Para conseguir una correspondencia, el valor extraído de la cabecera tiene que ser exactamente igual que el campo de correspondencia. Alternativamente, una entrada de flujo puede comprender un campo de correspondencia comodín, que significa que el valor extraído de la cabecera se puede corresponder con el campo de correspondencia si el valor de la cabecera es cualquier valor. Además de buscar correspondencias en cabeceras

de paquete, el FPM 150 puede buscar correspondencias en puertos de entrada, puertos de salida y campos de metadatos. El proceso de búsqueda de correspondencia se puede repetir en cada tabla hasta que el paquete sale de la tabla n y se ejecuta el conjunto de acción finalizada o hasta que el paquete se desecha o se envía al controlador 110 debido a una falta de correspondencia.

5 Las figuras 4 y 5 son diagramas esquemáticos de realizaciones de conductos de flujo para implementar QoS. Las figuras 6 a 8 son diagramas esquemáticos de realizaciones de conductos de flujo para implementar VRF, un cortafuegos y MPLS, respectivamente. Los conductos de flujo de las figuras 4 a 8 se han diseñado con múltiples tablas de flujo, pero dichos conductos de flujo están diseñados para servicios de red únicos.

10 Tal como se ha explicado anteriormente, un conducto de flujo se puede diseñar para proporcionar múltiples servicios de red seleccionando primero los servicios de red clave que se utilizan más frecuentemente. Dichos servicios de red pueden incluir ACL, envío de la capa dos, VLAN, MPLS y envío de la capa tres. En segundo lugar, se seleccionan los campos de correspondencia clave correspondientes a dichos servicios. Dichos campos de correspondencia pueden ser tal como se muestra en la tabla 1.

**Tabla 1: campos de correspondencia clave**

ACL	Envío de la capa dos
ETH TYPE	ETH SRC
IP SRC	ETH DST
IP DST	Metadatos
IP PROTO	
DST PORT	

15

VLAN	MPLS	Envío de la capa tres
ETH DST	MPLSLABEL	IP SRC
VLAN VID	MPLSTC	IP DST
VLAN PCP	Metadatos	ETH TYPE
Metadatos		IP TOS
		IP PROTO
		Metadatos

20

Además, los campos de correspondencia ETH TYPE e IN PORT se pueden utilizar para localizar diferentes tipos de servicio con el fin de acelerar el envío de paquetes. En tercer lugar, algunos campos de correspondencia pueden requerir que se utilice asimismo otro campo de correspondencia. Dichos requisitos se pueden denominar dependencias compartidas y se muestran en la figura 9, que es un diagrama de dependencias compartidas de campos de correspondencia. A modo de ejemplo, tal como se muestra en la tabla 1, ACL tiene un campo de correspondencia IP PROTO. Tal como se muestra en la figura 9, para poner en correspondencia IP PROTO es necesario asimismo poner en correspondencia ETH TYPE. Volviendo a la tabla 1, ETH TYPE es otro campo de correspondencia para ACL, de tal modo que se satisface la dependencia de los campos de correspondencia en la tabla 1. En cuarto lugar, en base a la priorización de los servicios de red, a una determinación de qué campos de correspondencia tienen en común los servicios de red, y a las dependencias compartidas de los campos de correspondencia, se puede diseñar un conducto de flujo optimizado.

25

30

La priorización puede estar basada en qué servicios de red se utilizan más frecuentemente. Un controlador tal como el controlador 110, un nodo de red tal como el nodo 140, un administrador tal como administrador asociado con el nodo 110 u otra entidad adecuada pueden llevar a cabo la priorización. El controlador, el nodo de red u otra entidad adecuada pueden analizar tráfico de datos, determinar qué servicios de red se utilizan más frecuentemente y priorizar en consecuencia los servicios de red. Por ejemplo, si hay 10 servicios de red disponibles, el controlador, el nodo de red u otra entidad adecuada pueden analizar el tráfico de datos, determinar los cinco primeros servicios de red disponibles en términos de frecuencia de utilización, y seleccionar dichos cinco primeros servicios de red disponibles. El controlador, el nodo de red u otra entidad adecuada pueden a continuación priorizar los cinco servicios de red disponibles por frecuencia de utilización.

35

Alternativamente, el administrador puede seleccionar y priorizar servicios de red arbitrariamente, en base a su conocimiento, o en base a por lo menos un criterio que le es conocido.

En la tabla 1, el campo de correspondencia ETH TYPE puede describir un tipo de trama Ethernet. El campo de correspondencia IP SRC puede corresponder al campo de correspondencia IPV4 SRC, que puede describir una dirección de origen IPv4, o bien al campo de correspondencia IPV6 SRC, que puede describir una dirección de origen IPv6. El campo de correspondencia IP DST puede corresponder al campo de correspondencia IPV4 DST, que puede describir una dirección de destino IPv4, o bien al campo de correspondencia IPV6 DST, que puede describir una dirección de destino IPv6. El campo de correspondencia IP PROTO puede describir un número de protocolo IPv4 o IPv6. El campo de correspondencia DST PORT puede corresponder al campo de correspondencia TCP DST, que puede describir el puerto de destino del protocolo de control de transmisión (TCP, transmission control protocol), o bien al campo de correspondencia UDP DST, que puede describir el puerto de destino del protocolo de datagramas de usuario (UDP, user datagram protocol). El campo de correspondencia ETH SRC puede describir una dirección de origen de Ethernet. El campo de correspondencia ETH DST puede describir una dirección de destino de Ethernet. El campo de correspondencia de metadatos puede describir metadatos, que se han definido anteriormente. El campo de correspondencia VLAN VID puede describir la identificación (ID) VLAN. El campo de correspondencia VLAN PCP puede describir una prioridad VLAN. El campo de correspondencia MPLS LABEL puede describir una etiqueta MPLS. El campo de correspondencia MPLS TC puede describir una clase de tráfico (TC, traffic class) MPLS. El campo de correspondencia IP TOS puede corresponder al campo de correspondencia IP DSCP, que puede describir un tipo de servicio (TOS, type of service) IP en seis bits, o bien al campo de correspondencia IP ECN, que puede describir un IP TOS en dos bits. Finalmente, el campo de correspondencia IN PORT puede describir un puerto de entrada de conmutación.

Es importante señalar que, aunque no se muestran, los campos de correspondencia dados a conocer pueden comprender prefijos y otras notaciones, tales como las mostradas en la página 49 de OpenFlow 1.3.1. Por ejemplo, el campo de correspondencia ETH TYPE se puede describir en OpenFlow 1.3.1 como OFPXMT\_OFB\_ETH\_TYPE. Aunque las denominaciones de campo de correspondencia pueden cambiar a través de diferentes versiones de OpenFlow, los propósitos de los campos de correspondencia se mantienen idénticos en general.

La figura 10 es un diagrama esquemático de un conducto de flujo 1000, según una realización de la invención. El conducto de flujo 1000 puede estar diseñado en base a la técnica anterior, y puede estar implementado en el nodo 640, y en particular, el FPM 650. Las tablas 0/1 pueden implementar ACL; la tabla 2 puede implementar localización de servicios; la tabla 3 puede implementar envío de la capa dos; las tablas 2 a 4 pueden implementar VLAN; las tablas 2 y 5 pueden implementar MPLS; las tablas 2, 6 y 7/8 pueden implementar envío IPv4 ((capa tres); las tablas 2, 6 y 9/10 pueden implementar envío IPv6 (capa tres); y la tabla 11 puede incluir todos los campos de correspondencia restantes para implementar cualesquiera servicios restantes. La denotación de las tablas 0/1 puede indicar que la tabla 0 y la tabla 1 comprenden los mismos campos de correspondencia; sin embargo, la tabla 0 puede ser una tabla de correspondencia exacta mientras que la tabla 1 puede ser una tabla de correspondencia comodín. Las denotaciones de las tablas 7/8 y las tablas 9/10 pueden indicar la misma relación de correspondencia exacta y correspondencia comodín. Cada tabla de correspondencia exacta puede estar almacenada y ser consultada en SRAM mientras que cada tabla de correspondencia comodín puede estar almacenada y ser consultada en TCAM. Esta relación puede reducir la utilización de TCAM, que de lo contrario sería necesaria para implementar el conducto de flujo 1000. Además de tener un bit 0 y un bit 1 para poner o no en correspondencia un campo, TCAM puede tener un tercer bit prescindible ("do-not-care") que puede ser adecuado para correspondencia comodín. La correspondencia exacta y la correspondencia comodín se pueden producir secuencial o simultáneamente. Por ejemplo, la tabla 0 puede en primer lugar realizar una búsqueda de correspondencia y puede estar seguida por la tabla 1, o las tablas 0 y 1 pueden llevar a cabo búsqueda de correspondencia al mismo tiempo. Si existe una correspondencia exacta y una correspondencia comodín, la correspondencia exacta puede tener prioridad. El conducto de flujo 1000 puede soportar servicios de red clave, así como conmutación y envío rápidos. Por ejemplo, algunos servicios pueden no requerir poner en correspondencia cada tabla, sino que pueden saltarse tablas en el conducto de flujo 1000. Las siguientes figuras proporcionan ejemplos de este tipo. Además, puede haber tablas de grupos asociadas. Las tablas de grupos pueden comprender conjuntos de acciones para desbordamiento, pueden permitir un envío más complejo y pueden permitir enviar múltiples entradas de flujo a un único identificador. Finalmente, el conducto de flujo 1000 se puede extender dinámicamente a otros campos de correspondencia utilizados con menor frecuencia y proporcionar soporte para futuras entradas de flujo utilizando la tabla 11.

La figura 11 es un diagrama esquemático que muestra conmutación y envío de la capa dos en el conducto de flujo 1000 de la figura 10. Un paquete puede comenzar en las tablas 0/1 para correspondencia de ACL. Si el paquete se corresponde, se puede entonces ordenar que el paquete salte a la tabla 2. Si el paquete no se corresponde, el paquete puede entonces ser enviado al controlador 610, desechado o enviado según una instrucción de inexistente en la tabla. En la tabla 2, si el paquete se corresponde, puede entonces ordenarse que el paquete salte a la tabla 3. Si el paquete no se corresponde, el paquete puede entonces enviarse a la tabla 3, según una instrucción de inexistente en la tabla. En la tabla 3, si el paquete se corresponde, se puede entonces ordenar que el paquete salte al puerto correspondiente del nodo 640 y puede comenzar el proceso de envío de datos. Si el paquete no se corresponde, el paquete puede entonces enviarse al controlador 610 o desecharse.

- La figura 12 es un diagrama esquemático que muestra la conmutación y envío de la capa 3, en el conducto de flujo 1000 de la figura 10. Un paquete puede comenzar en las tablas 0/1 para correspondencia ACL. Si el paquete se corresponde, se puede entonces ordenar que el paquete salte a la tabla 2. Si el paquete no se corresponde, el paquete puede entonces ser enviado al controlador 610, desechado o enviado según una instrucción de inexistente en la tabla. En la tabla 2, si el paquete se corresponde, puede entonces ordenarse que el paquete salte a la tabla 6. Por ejemplo, si el paquete tiene ETH TYPE = 0x0806 e IN PORT = 10, entonces se puede ordenar que el paquete salte a la tabla 6. Si el paquete no se corresponde, el paquete puede entonces ser enviado al controlador 610, desechado o enviado según una instrucción de inexistente en la tabla. En la tabla 6, se determina qué tipo de envío de la capa tres puede ser apropiado. Si el paquete se corresponde con IPv4, se puede ordenar entonces que el paquete salte a las tablas 7/8. Si el paquete se corresponde con IPv6, se puede ordenar entonces que el paquete salte a las tablas 9/10. Si el paquete no se corresponde, el paquete puede entonces enviarse al controlador 610 o desecharse. En la tabla 9/10, si el paquete se corresponde, se puede entonces ordenar que el paquete salte al puerto correspondiente del nodo 640 y puede comenzar el proceso de envío de datos. Si el paquete no se corresponde, el paquete puede entonces enviarse al controlador 610 o desecharse.
- En base a la definición de las múltiples tablas de flujo, las tablas 0 y 1 se pueden extender de acuerdo con algunos requisitos específicos. Una extensión de este tipo que se centra en envío de la capa tres puede comprender los campos de correspondencia mostrados en la tabla 2 e implementados en la figura 13.

**Tabla 2: diseño de tablas de flujo extendidas**

Tabla 0:	Tabla 1:	Tabla 2:
ETHTYPE	IPV4 SRC	IPV4 SRC
IP PROTO	IPV4 DST	IPV4 DST
(Correspondencia exacta)	TCP SRC	TCP SRC
	TCP DST	TCP DST
	(Correspondencia exacta)	(Correspondencia comodín)
Tabla 3:	Tabla 4:	Tabla 5:
IPV4 SRC	IPV4 SRC	IPV6 SRC
IPV4 DST	IPV4 DST	IPV6 DST
UDP SRC	UDP SRC	TCP SRC
UDP DST	UDP DST	TCP DST
(Correspondencia exacta)	(Correspondencia comodín)	(Correspondencia exacta)
Tabla 6:	Tabla 7:	Tabla 8:
IPV6 SRC	IPV6 SRC	IPV6 SRC
IPV6 DST	IPV6 DST	IPV6 DST
TCP SRC	UDP SRC	UDP SRC
TCP DST	UDP DST	UDP DST
(Correspondencia comodín)	(Correspondencia exacta)	(Correspondencia comodín)

- Igual que para las figuras 10 a 12, la denotación en la figura 13 de las tablas 1/2, 3/4, 5/6 y 7/8 puede indicar que las tablas 1, 3, 5 y 7 pueden ser tablas de correspondencia exacta mientras que las tablas 2, 4, 6 y 8 pueden ser tablas de correspondencia comodín.

- En la tabla 2 pueden seguir aplicando las descripciones de campos de correspondencia para la tabla 1. Además, el campo de correspondencia TCP SRC puede describir el puerto de origen TCP. El campo de correspondencia UDP SRC puede describir el puerto de origen UDP.



5 La figura 14 es un diagrama de flujo que muestra un procedimiento 1400, según una realización de la invención. El procedimiento 1400 se puede implementar en el nodo 140, por ejemplo, en el FPM 150. En la etapa 1410, se pueden seleccionar una serie de servicios de red. En la etapa 1420, los servicios de red se pueden priorizar en base a qué servicios de red son los más importantes y a qué servicios de red son los utilizados más frecuentemente. En la etapa 1430, se puede determinar un campo de correspondencia correspondiente a cada uno de los servicios de red. En la etapa 1440, se puede diseñar un conducto de flujo con una serie de tablas de flujo. Cada una de las tablas de flujo puede comprender por lo menos uno de los campos de correspondencia, y los campos de correspondencia pueden estar ordenados en base a la priorización, a cuáles de los campos de correspondencia son compartidos entre los servicios de red, a una dependencia compartida de los campos de correspondencia y a la velocidad de procesamiento.

10 Aunque en la presente invención se han dado conocer varias realizaciones, se debe entender que los sistemas y procedimientos se pueden realizar de muchas otras formas específicas sin apartarse del o alcance de la presente invención. Los presentes ejemplos se deben considerar como ilustrativos y no limitativos, y no se pretende limitarse a los detalles proporcionados en la presente memoria. Por ejemplo, los diversos elementos o componentes se pueden combinar o integrar en otro sistema, o determinadas características pueden ser omitidas, o no implementarse.

15 Además, las técnicas, sistemas, subsistemas y procedimientos descritos y mostrados en las diversas realizaciones como separados o independientes, se pueden combinar o integrar con otros sistemas, módulos, técnicas u procedimientos sin apartarse del alcance de la presente invención. Otros elementos mostrados o descritos como acoplados, o acoplados directamente o comunicando entre sí, pueden estar acoplados indirectamente o comunicando a través de alguna interfaz, dispositivo o componente intermedio ya sea eléctrica, mecánicamente o de otro modo. Los expertos en la materia pueden ser capaces de determinar otros ejemplos de cambios, sustituciones y modificaciones, y estos se pueden realizar sin apartarse del y alcance dado a conocer en la presente memoria.

## REIVINDICACIONES

1. Un aparato (200), que comprende:

un receptor configurado para recibir instrucciones;

un sistema de memoria (260) acoplado al receptor y configurado para almacenar las instrucciones; y

5 un procesador (230) acoplado al sistema de memoria y configurado para ejecutar las instrucciones con el fin de provocar que el aparato:

implemente un conducto de flujo (300), en el que el conducto de flujo comprende una serie de tablas de flujo, en el que las tablas de flujo comprenden campos de correspondencia, en el que los campos de correspondencia se ordenan en base a una priorización de servicios de red y a dependencias compartidas de campos de correspondencia, y

encamine un paquete utilizando el conducto de flujo.

2. El aparato según la reivindicación 1, en el que los servicios de red comprenden un servicio de lista de control de acceso, ACL, un servicio de envío de la capa dos, un servicio de red de área local virtual, VLAN, un servicio de conmutación por etiquetas multiprotocolo, MPLS, un servicio de envío de la capa tres y un servicio de localización de servicios.

3. El aparato según la reivindicación 2, en el que el servicio de envío de la capa tres comprende un servicio de protocolo de internet versión 4, IPv4, y un servicio de protocolo de internet versión 6, IPv6.

4. El aparato según la reivindicación 3, en el que las tablas de flujo están numeradas de 0 a 11, en el que la tabla de flujo 0 y la tabla de flujo 1 hacen que el procesador implemente el servicio ACL, en el que la tabla de flujo 2 hace que el procesador implemente el servicio de localización de servicios, en el que la tabla de flujo 3 hace que el procesador implemente el servicio de envío de la capa dos, en el que la tabla de flujo 2, la tabla de flujo 3 y la tabla de flujo 4 hacen que el procesador implemente el servicio VLAN, en el que la tabla de flujo 2 y la tabla de flujo 5 hacen que el procesador implemente el servicio MPLS, en el que la tabla de flujo 2, la tabla de flujo 6, la tabla de flujo 7 y la tabla de flujo 8 hacen que el procesador implemente el servicio IPv4, en el que la tabla de flujo 2, la tabla de flujo 6, la tabla de flujo 9 y la tabla de flujo 10 hacen que el procesador implemente el servicio IPv6 y en el que la tabla de flujo 11 hace que el procesador implemente por lo menos un servicio adicional.

5. El aparato según la reivindicación 4, en el que la tabla de flujo 0 y la tabla de flujo 1 comprenden cada una campos de correspondencia ETH TYPE, IP SRC, IP DST, IP PROTO y DST PORT, en el que la tabla de flujo 2 comprende campos de correspondencia ETH TYPE e IN PORT, en el que la tabla de flujo 3 comprende campos de correspondencia ETH SRC, ETH DST y de metadatos, en el que la tabla de flujo 4 comprende campos de correspondencia VLAN VID, VLAN PCP y de metadatos, en el que la tabla de flujo 5 comprende campos de correspondencia MPLS LABEL, MPLS TC y de metadatos, en el que la tabla de flujo 6 comprende campos de correspondencia IP TOS, IP PROTO y de metadatos, en el que la tabla de flujo 7 y la tabla de flujo 8 comprenden cada una campos de correspondencia IPV4 SRC, IPV4 DST y de metadatos, en el que la tabla de flujo 9 y la tabla de flujo 10 comprenden cada una campos de correspondencia IPV6 SRC, IPV6 DST y de metadatos, en el que la tabla de flujo 11 comprende campos de correspondencia adicionales y en el que el campo de correspondencia ETH TYPE describe un tipo de trama de Ethernet, el campo de correspondencia IP SRC describe una dirección de origen de protocolo de internet, IP, el campo de correspondencia IP DST describe una dirección de destino IP, el campo de correspondencia IP PROTO describe un número de protocolo IPv4 o IPv6, el campo de correspondencia DST PORT describe un puerto de destino, el campo de correspondencia IN PORT describe un puerto de entrada de conmutación, el campo de correspondencia ETH SRC describe una dirección de origen de Ethernet, el campo de correspondencia ETH DST describe una dirección de destino de Ethernet, el campo de correspondencia de metadatos describe metadatos, el campo de correspondencia VLAN VID describe una identificación, ID, de VLAN, el campo de correspondencia VLAN PCP describe una prioridad VLAN, el campo de correspondencia MPLS LABEL describe una etiqueta MPLS, el campo de correspondencia MPLS TC describe una clase de tráfico, TC, MPLS, el campo de correspondencia IP TOS describe un tipo de servicio, TOS, IP, el campo de correspondencia IPV4 SRC describe una dirección de origen IPv4, el campo de correspondencia IPV4 DST describe una dirección de destino IPv4, el campo de correspondencia IPV6 SRC describe una dirección de origen IPv6 y el campo de correspondencia IPV6 DST describe una dirección de destino IPv6.

6. El aparato según la reivindicación 1, en el que el sistema de memoria comprende memoria estática de acceso aleatorio, SRAM, para implementar una correspondencia exacta de los campos de correspondencia, y memoria ternaria de contenido direccionable, TCAM, para implementar correspondencia comodín de los campos de correspondencia.

7. El aparato según la reivindicación 1, en el que las tablas de flujo están numeradas de 0 a 8, en el que la tabla de flujo 0 comprende campos de correspondencia ETH TYPE e IP PROTO, en el que la tabla de flujo 1 y la tabla de flujo 2 comprenden cada una campos de correspondencia IPV4 SRC, IPV4 DST, TCP SRC y TCP DST, en el que la

- 5 tabla de flujo 3 y la tabla de flujo 4 comprenden cada una campos de correspondencia IPV4 SRC, IPV4 DST, UDP SRC y UDP DST, en el que la tabla de flujo 5 y la tabla de flujo 6 comprenden cada una campos de correspondencia IPV6 SRC, IPV6 DST, TCP SRC y TCP DST, en el que la tabla de flujo 7 y la tabla de flujo 8 comprenden cada una campos de correspondencia IPV6 SRC, IPV6 DST, UDP SRC y UDP DST y en el que el campo de correspondencia
- 10 ETH TYPE describe un tipo de trama de Ethernet, el campo de correspondencia IP PROTO describe un número de protocolo IPv4 o IPv6, el campo de correspondencia IPV4 SRC describe una dirección de origen IPv4, el campo de correspondencia IPV4 DST describe una dirección de destino IPv4, el campo de correspondencia TCP SRC describe un puerto de origen del protocolo de control de transmisión, TCP, el campo de correspondencia DST TCP PORT describe un puerto de destino TCP, el UDP SRC describe un puerto de origen del protocolo de datagramas de usuario, UDP, el UDP DST describe un puerto de destino UDP, el campo de correspondencia IPV6 SRC describe una dirección de origen IPv6, y el campo de correspondencia IPV6 DST describe una dirección de destino IPv6.
8. El aparato según la reivindicación 7, en el que la tabla de flujo 0, la tabla de flujo 1, la tabla de flujo 3, la tabla de flujo 5 y la tabla de flujo 7 son tablas de flujo de correspondencia exacta, y en el que la tabla de flujo 2, la tabla de flujo 4, la tabla de flujo 6 y la tabla de flujo 8 son tablas de flujo de correspondencia comodín.
- 15 9. Un procedimiento relacionado con OpenFlow de trabajo en red definido por software, SDN, comprendiendo el procedimiento:
- seleccionar (1410) servicios de red;
- priorizar (1420) los servicios de red;
- determinar (1430) campos de correspondencia correspondientes a los servicios de red; y
- 20 diseñar (1440) un conducto de flujo con una serie de tablas de flujo, en el que cada una de las tablas de flujo comprende por lo menos uno de los campos de correspondencia, y en el que los campos de correspondencia se ordenan en base a la priorización y a las dependencias compartidas de campos de correspondencia.
10. El procedimiento según la reivindicación 9, que comprende además enviar instrucciones para implementar el conducto de flujo y para encaminar paquetes utilizando el conducto de flujo.
- 25 11. El procedimiento según la reivindicación 9, en el que la priorización se basa en cuáles son los servicios de red utilizados con mayor frecuencia.
12. El procedimiento según la reivindicación 9, en el que las tablas de flujo están numeradas de 0 a 11, en el que la tabla de flujo 0 y la tabla de flujo 1 implementan un servicio de lista de control de acceso, ACL, en el que la tabla de flujo 2 implementa un servicio de localización de servicios, en el que la tabla de flujo 3 implementa un servicio de envío de la capa dos y en el que la tabla de flujo 2, la tabla de flujo 3 y la tabla de flujo 4 implementan un servicio de red de área local virtual, VLAN, en el que la tabla de flujo 2 y la tabla de flujo 5 implementan un servicio de conmutación por etiquetas multiprotocolo, MPLS, en el que la tabla de flujo 2, la tabla de flujo 6, la tabla de flujo 7 y la tabla de flujo 8 implementan un servicio de protocolo de internet versión 4, IPv4, en el que la tabla de flujo 2, la tabla de flujo 6, la tabla de flujo 9 y la tabla de flujo 10 implementan un servicio de protocolo de internet versión 6, IPv6, y en el que la tabla de flujo 11 implementa por lo menos un servicio adicional.
- 30 13. El procedimiento según la reivindicación 12, en el que la tabla de flujo 0 y la tabla de flujo 1 comprenden cada una campos de correspondencia ETH TYPE, IP SRC, IP DST, IP PROTO y DST PORT, en el que la tabla de flujo 2 comprende campos de correspondencia ETH TYPE e IN PORT, en el que la tabla de flujo 3 comprende campos de correspondencia ETH SRC, ETH DST y de metadatos, en el que la tabla de flujo 4 comprende campos de correspondencia VLAN VID, VLAN PCP y de metadatos, en el que la tabla de flujo 5 comprende campos de correspondencia MPLS LABEL, MPLS TC y de metadatos, en el que la tabla de flujo 6 comprende campos de correspondencia IP TOS, IP PROTO y de metadatos, en el que la tabla de flujo 7 y la tabla de flujo 8 comprenden cada una campos de correspondencia IPV4 SRC, IPV4 DST y de metadatos, en el que la tabla de flujo 9 y la tabla de flujo 10 comprenden cada una campos de correspondencia IPV6 SRC, IPV6 DST y de metadatos, en el que la
- 45 tabla de flujo 11 comprende campos de correspondencia adicionales y en el que el campo de correspondencia ETH TYPE describe un tipo de trama de Ethernet, el campo de correspondencia IP SRC describe una dirección de origen de protocolo de internet, IP, el campo de correspondencia IP DST describe una dirección de destino IP, el campo de correspondencia IP PROTO describe un número de protocolo IPv4 o IPv6, el campo de correspondencia DST PORT describe un puerto de destino, el campo de correspondencia IN PORT describe un puerto de entrada de conmutación, el campo de correspondencia ETH SRC describe una dirección de origen de Ethernet, el campo de correspondencia ETH DST describe una dirección de destino de Ethernet, el campo de correspondencia de metadatos describe metadatos, el campo de correspondencia VLAN VID describe una identificación, ID, de VLAN, el campo de correspondencia VLAN PCP describe una prioridad VLAN, el campo de correspondencia MPLS LABEL describe una etiqueta MPLS, el campo de correspondencia MPLS TC describe una clase de tráfico, TC, MPLS, el
- 50 campo de correspondencia IP TOS describe un tipo de servicio, TOS, IP, el campo de correspondencia IPV4 SRC describe una dirección de origen IPv4, el campo de correspondencia IPV4 DST describe una dirección de destino IPv4, el campo de correspondencia IPV6 SRC describe una dirección de origen IPv6 y el campo de correspondencia IPV6 DST describe una dirección de destino IPv6.
- 55

14. Un aparato (110, 140, 200) que comprende: un primer procesador configurado para crear instrucciones para implementar un conducto de flujo (300), estando además configurado para llevar a cabo las siguientes etapas:

seleccionar una serie de servicios de red;

priorizar los servicios de red en base a qué servicios de red se utilizan más frecuentemente;

5 determinar un campo de correspondencia correspondiente a cada uno de los servicios de red; y

diseñar un conducto de flujo con una serie de tablas de flujo;

en el que cada una de las tablas de flujo puede comprender al menos uno de los campos de correspondencia, y los campos de correspondencia están basados en la priorización y en una dependencia compartida de los campos de correspondencia.

10

100

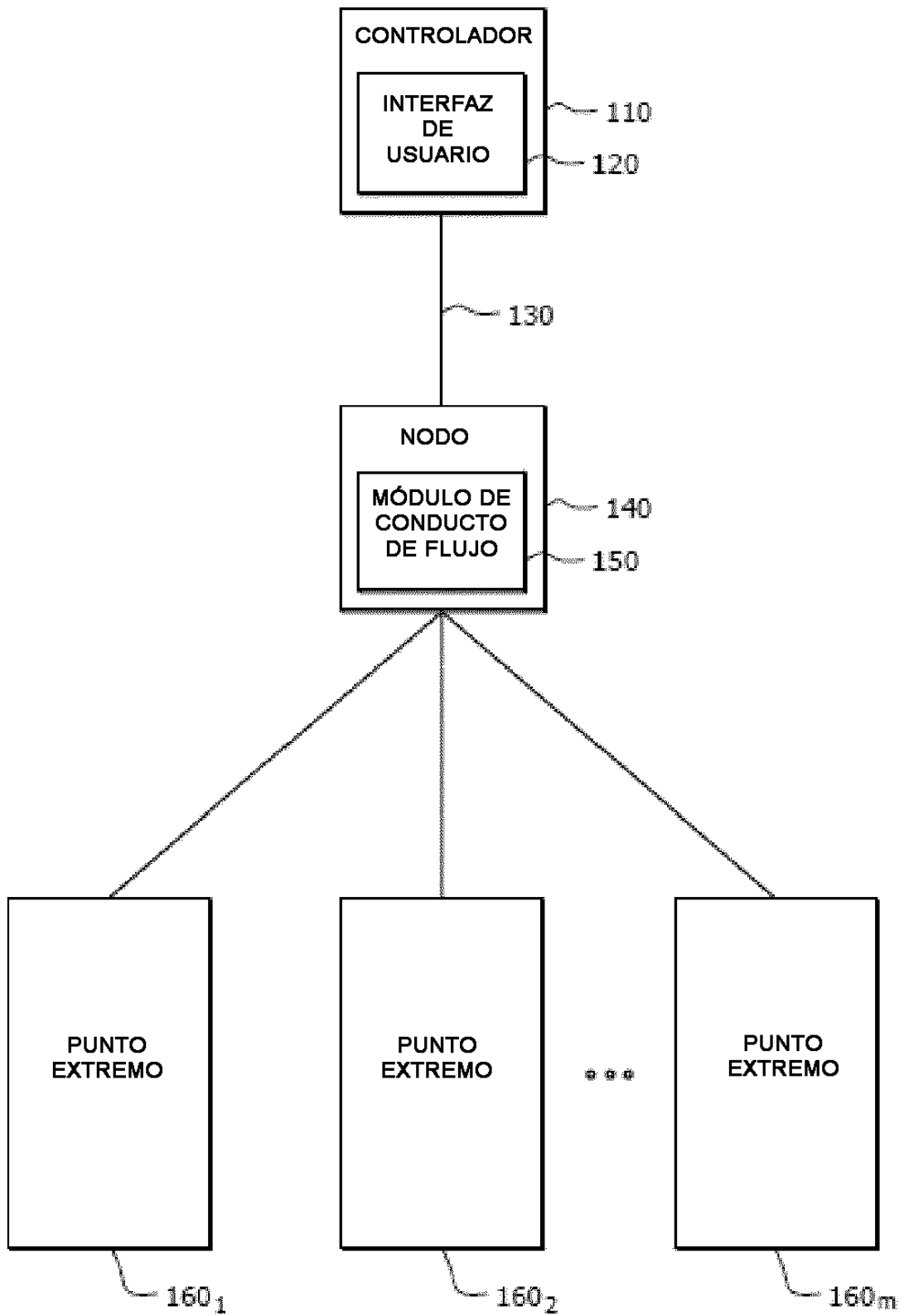


FIG. 1

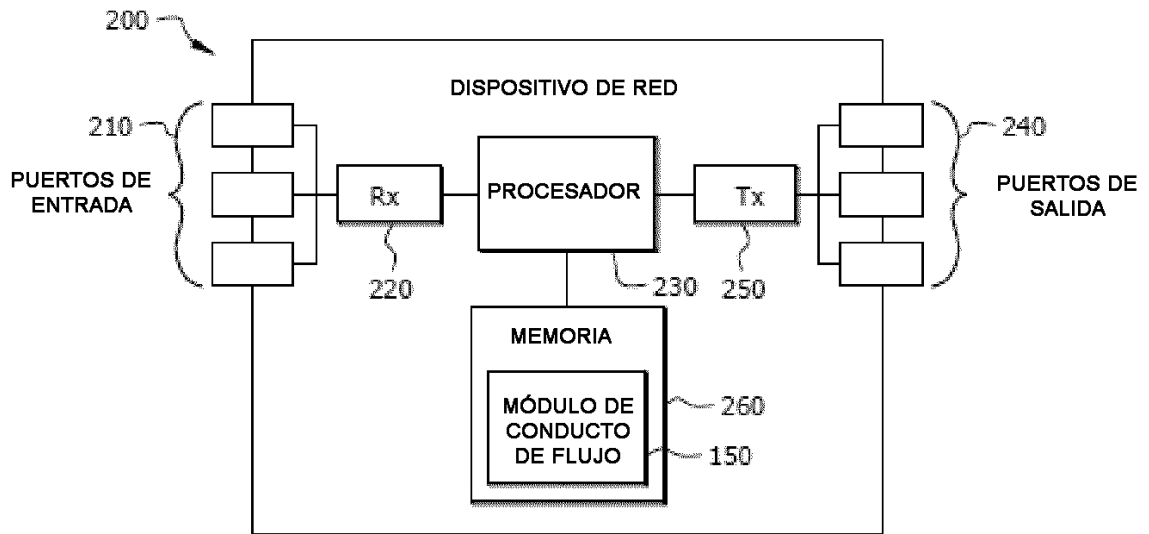


FIG. 2

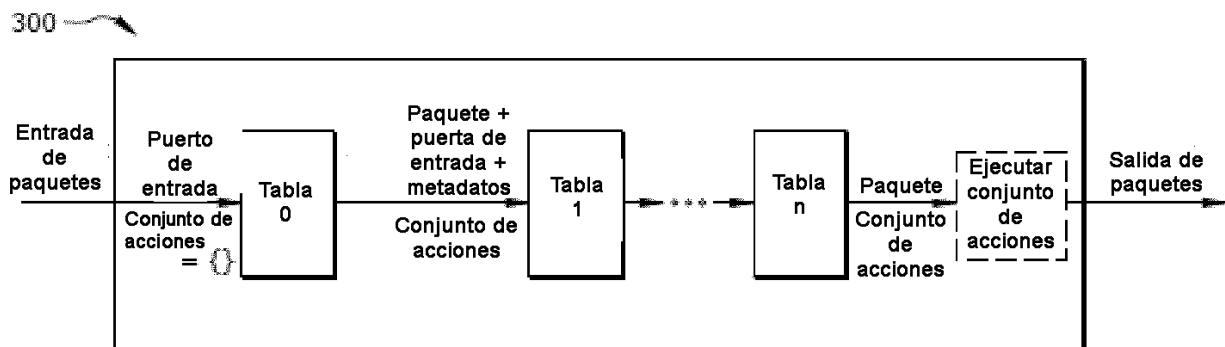


FIG. 3

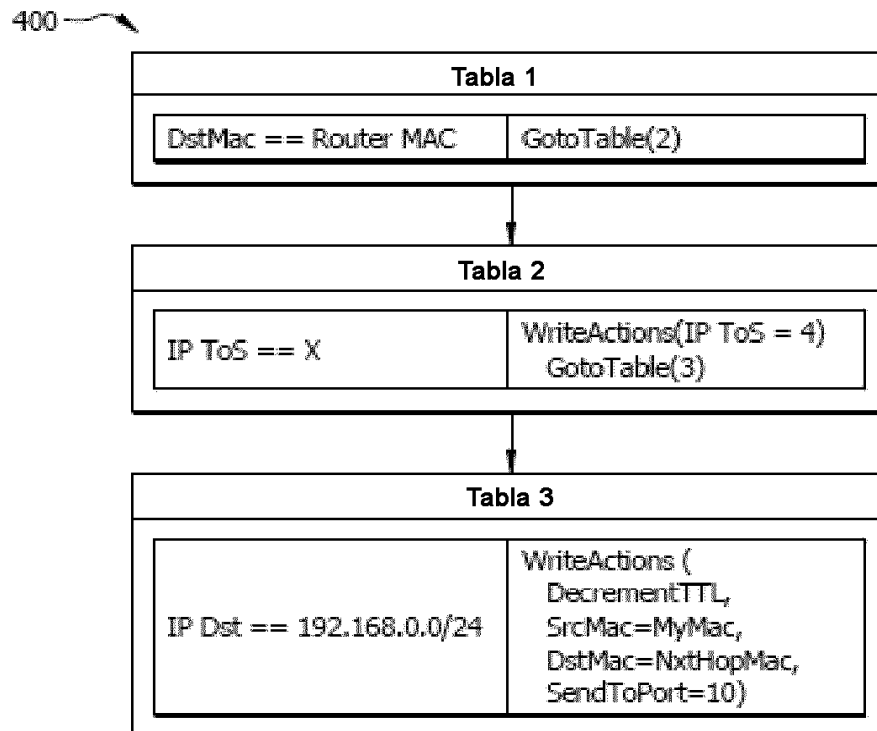


FIG. 4

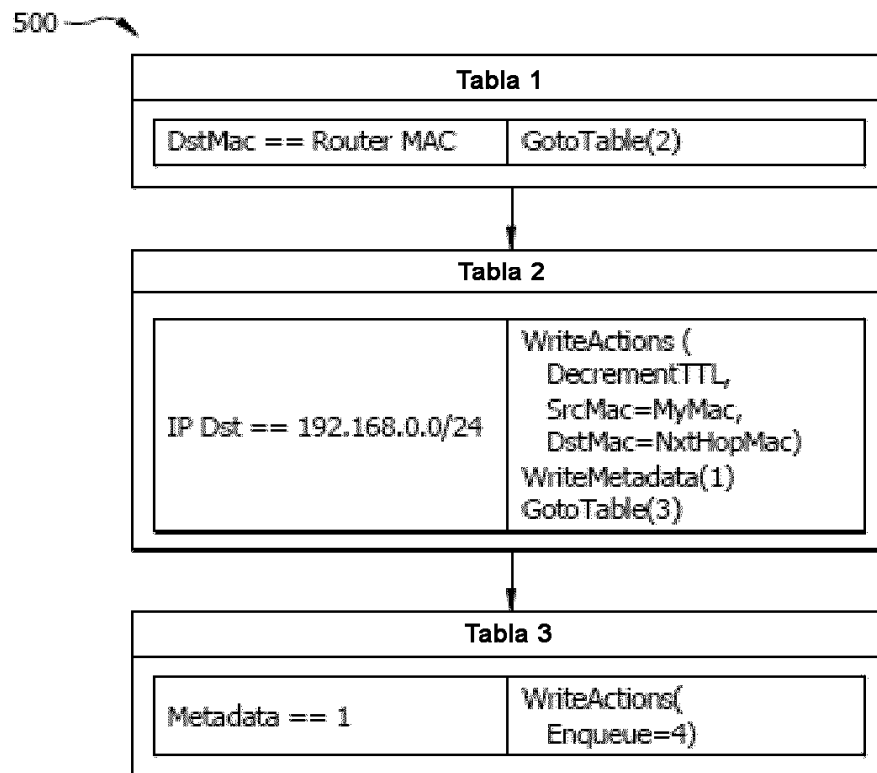


FIG. 5

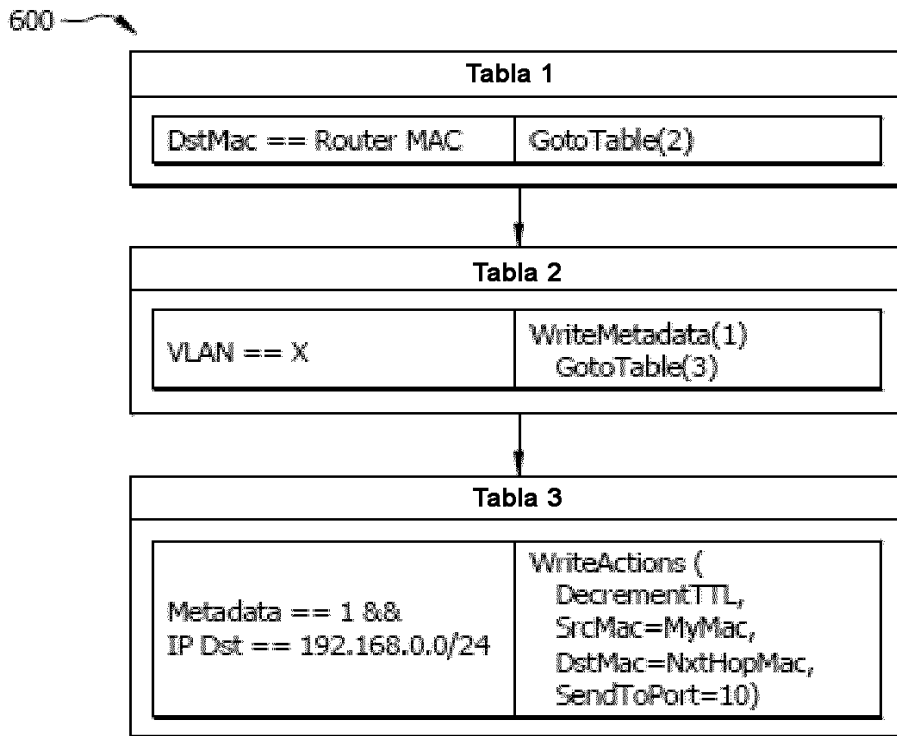


FIG. 6

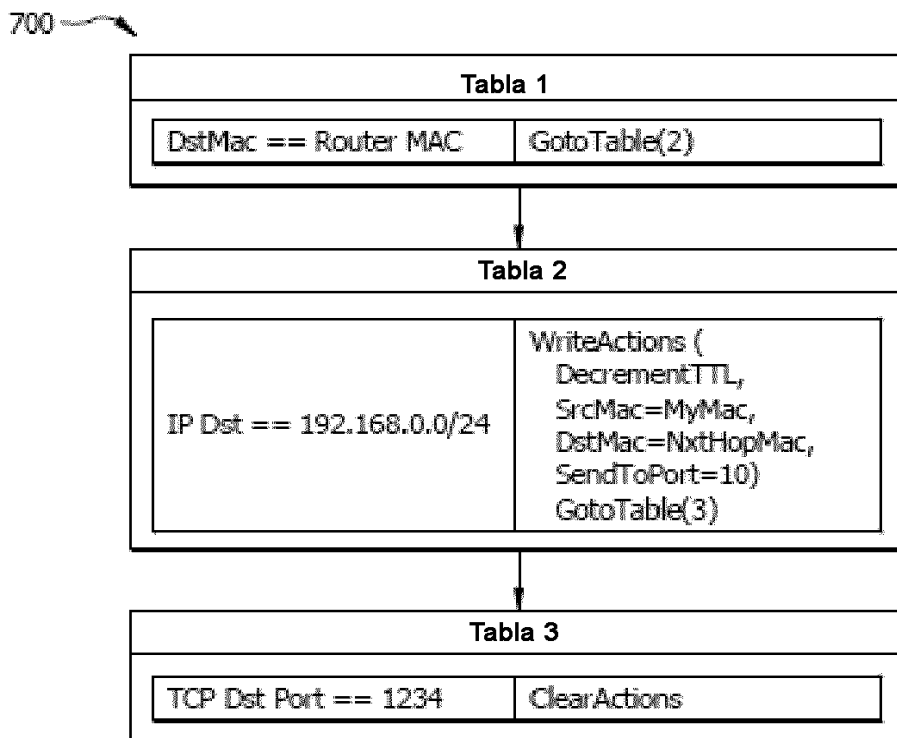


FIG. 7



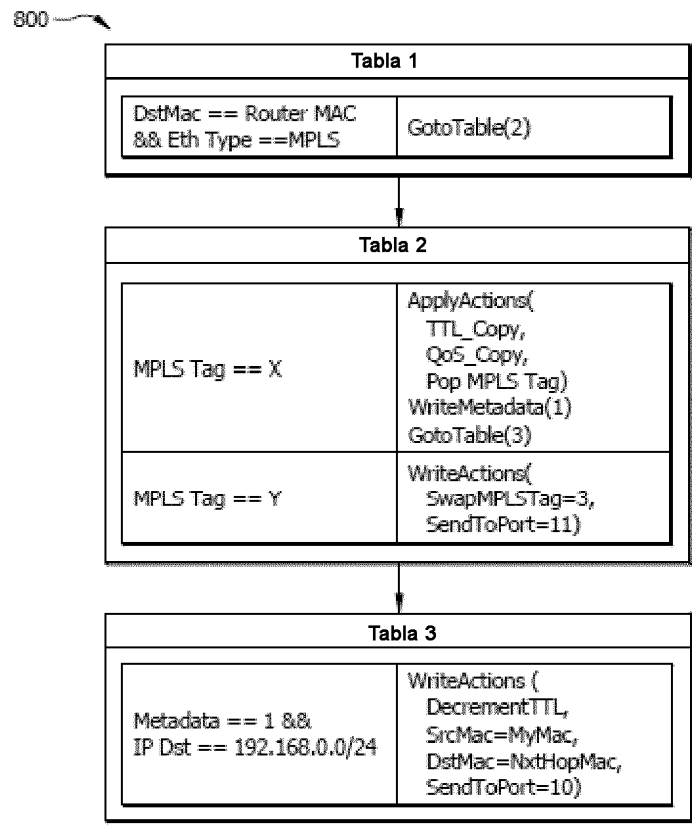


FIG. 8

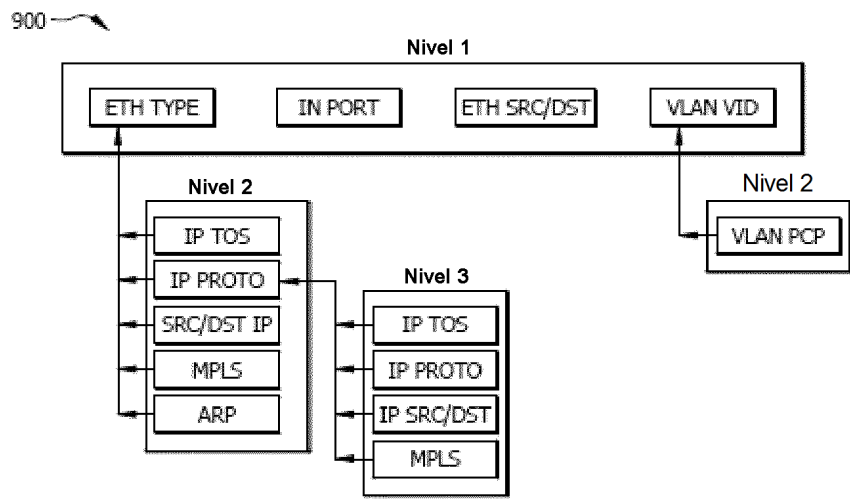


FIG. 9

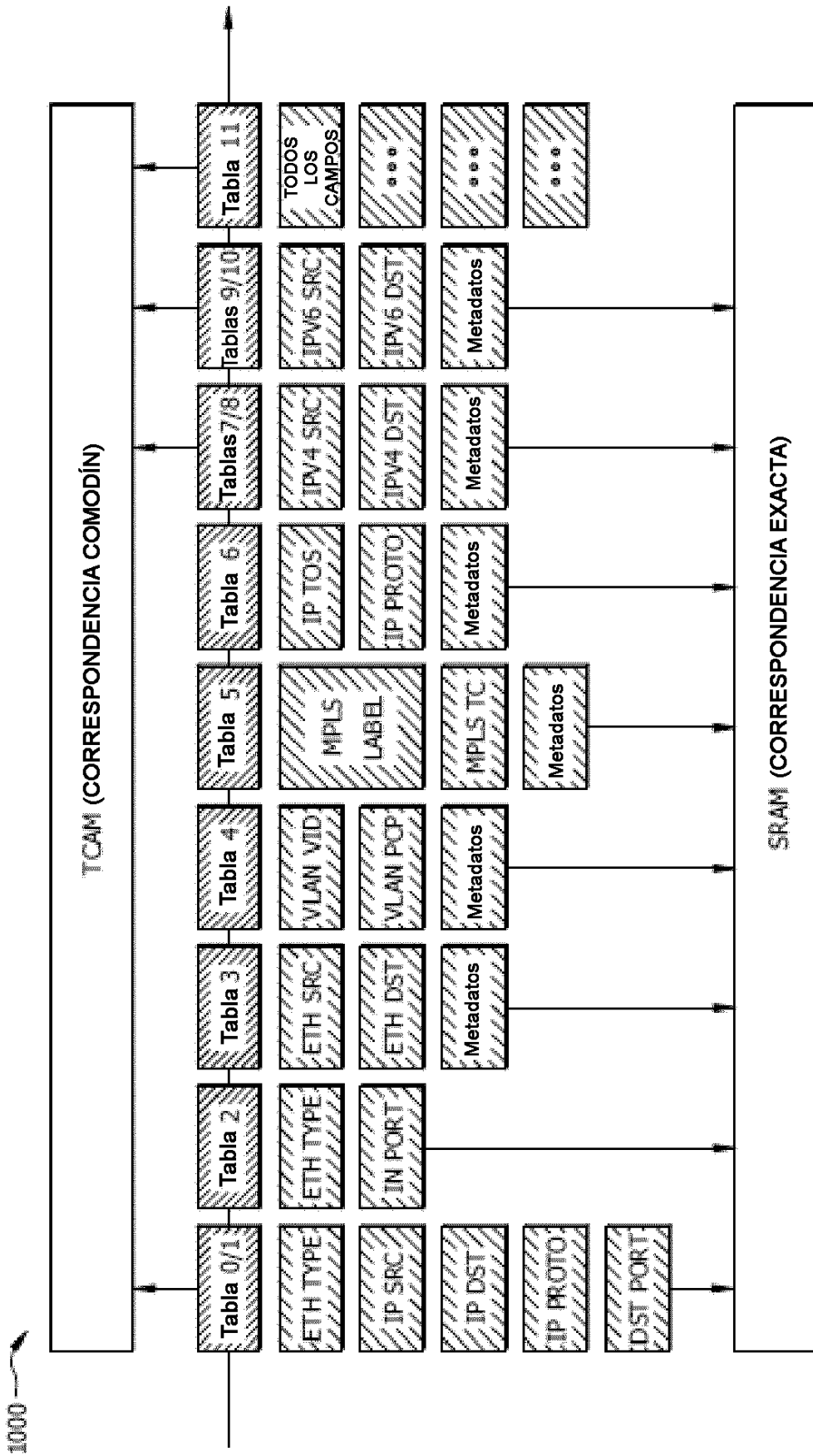


FIG. 10

1000

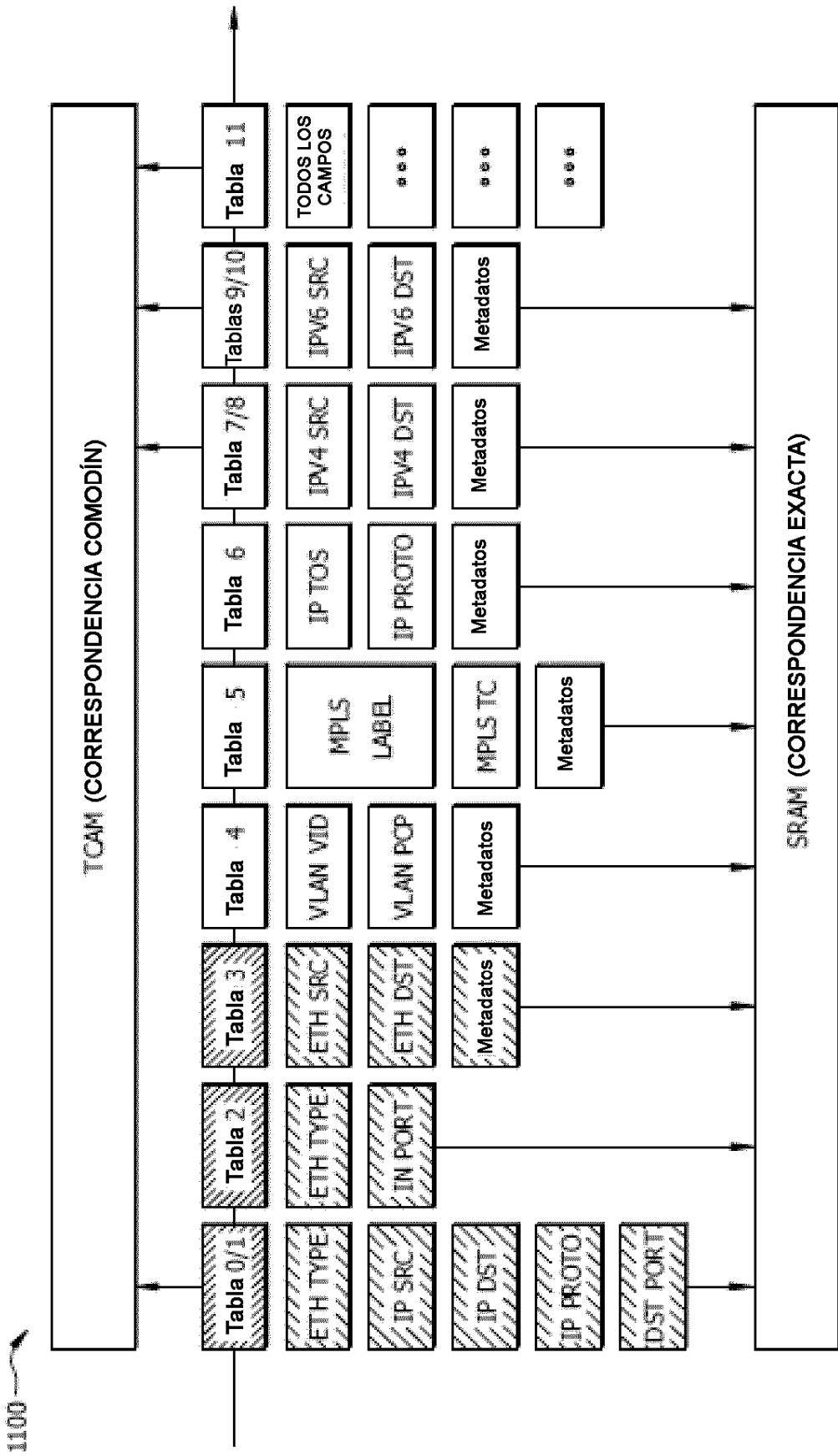


FIG. 11

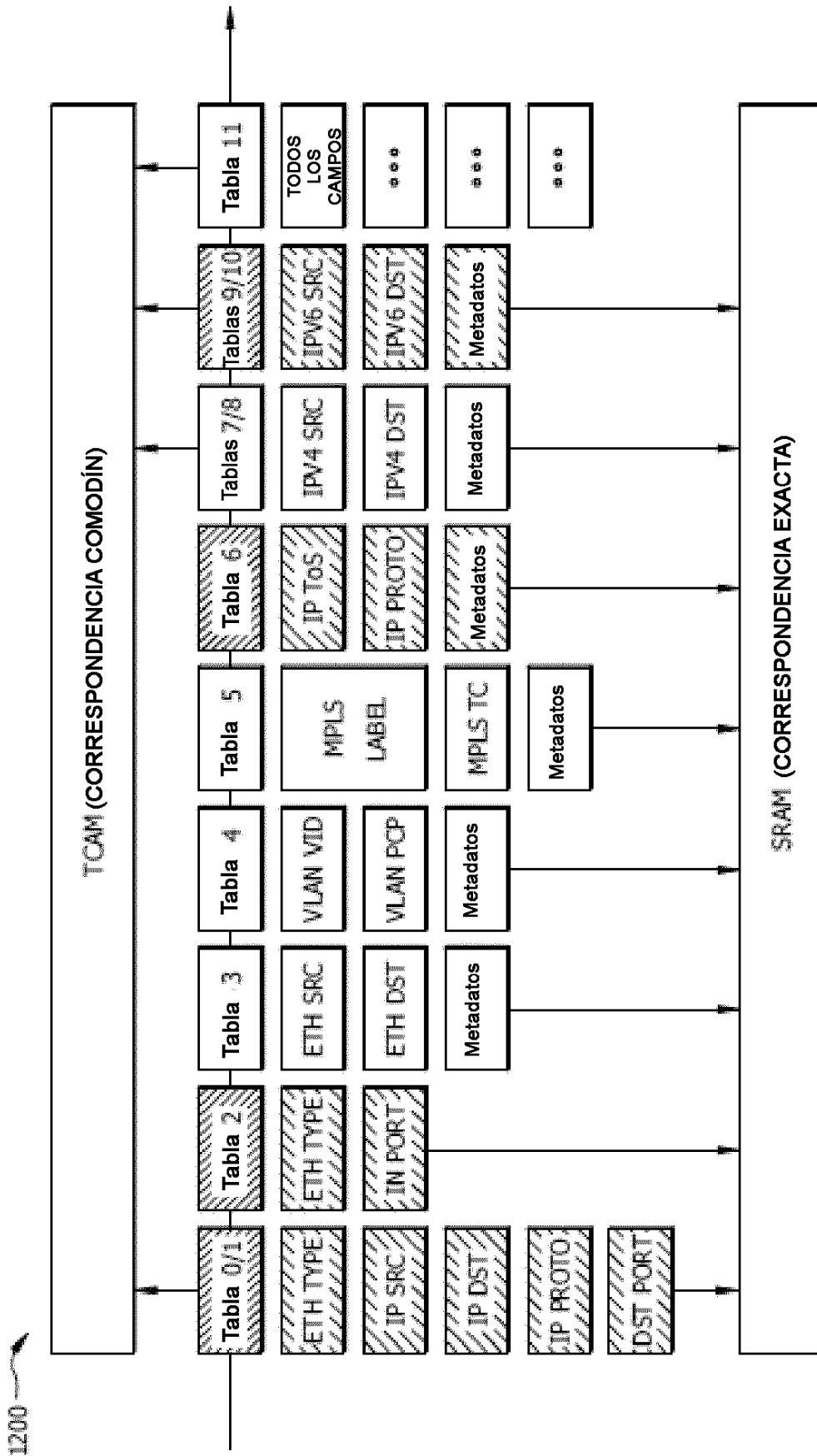


FIG. 12

1200

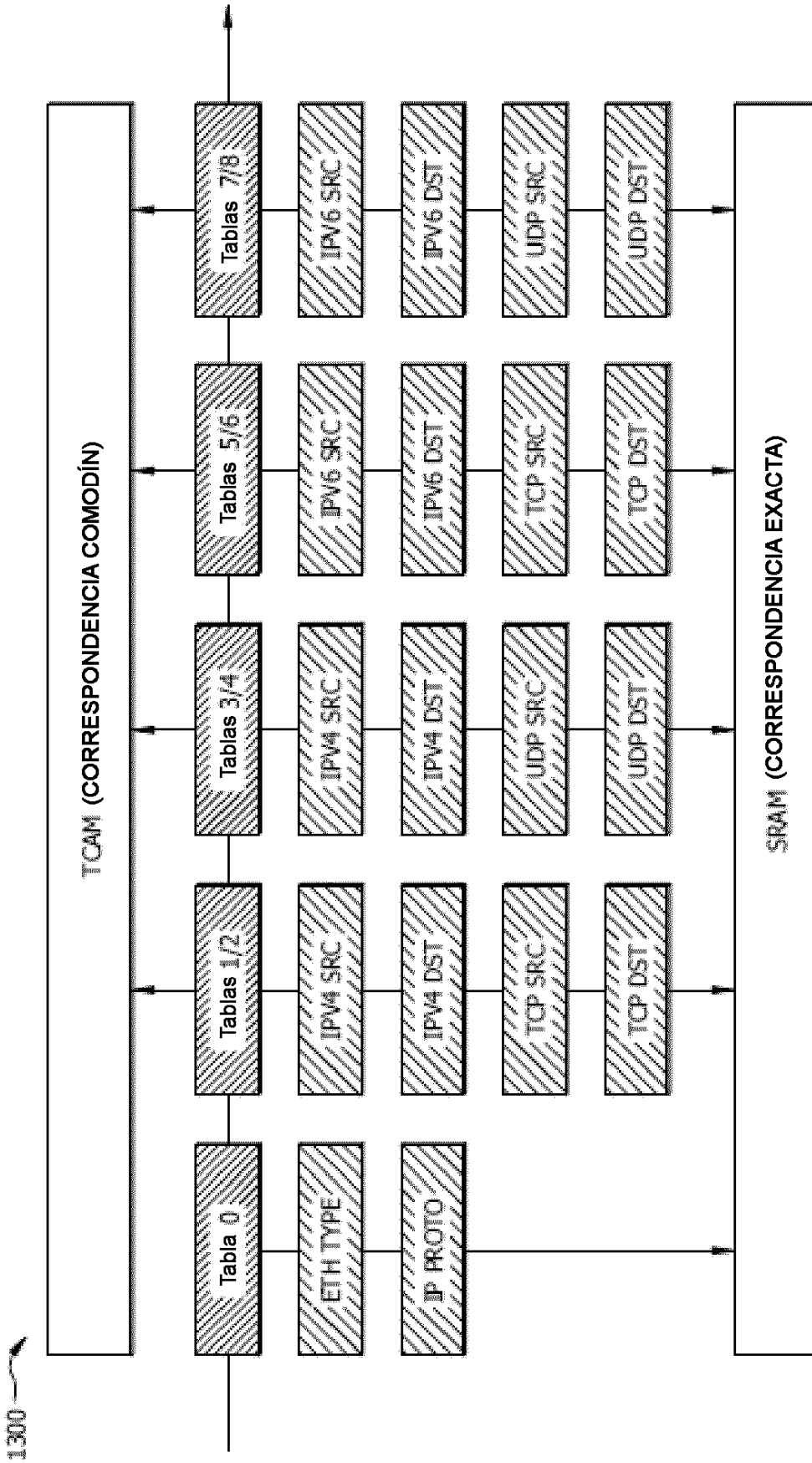
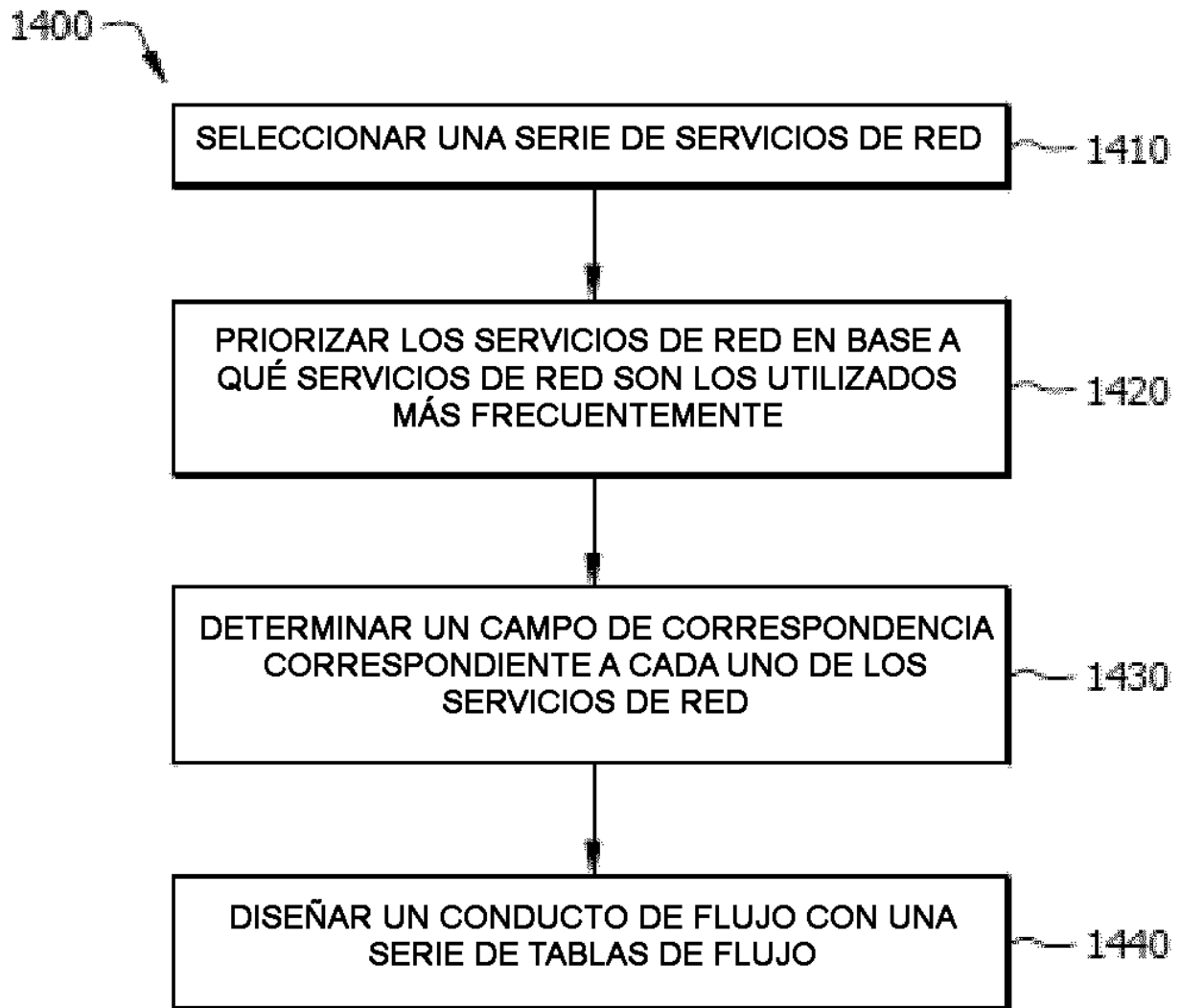


FIG. 13



*FIG. 14*