(19) OFICINA ESPAÑOLA DE PATENTES Y MARCAS

ESPAÑA

(11) Número de publicación: **2 645 565**

(51) Int. CI.:

G06F 17/30 (2006.01)

(12) TRADUCCIÓN DE PATENTE EUROPEA T3

(86) Fecha de presentación y número de la solicitud internacional: **27.10.2003 PCT/US2003/033908**

(87) Fecha y número de publicación internacional: **27.05.2004 WO04044676**

(96) Fecha de presentación y número de la solicitud europea: **27.10.2003 E 03779267 (8)**

(97) Fecha y número de publicación de la concesión europea: **09.08.2017 EP 1559034**

(54) Título: **Sistema de gestión y acceso para repositorios de documentos electrónicos**

(30) Prioridad:

07.11.2002 US 289782

(45) Fecha de publicación y mención en BOPI de la traducción de la patente:
05.12.2017

(73) Titular/es:

THOMSON REUTERS GLOBAL RESOURCES UNLIMITED COMPANY (100.0%)
Neuhofstrasse 1
6340 Baar, CH

(72) Inventor/es:

BLUHM, MARK;
GETTING, BRUCE;
HAYFT, MARK y
WALZ, SHIRLEY

(74) Agente/Representante:

CURELL AGUILÁ, Mireia

DESCRIPCIÓN

Sistema de gestión y acceso para repositorios de documentos electrónicos.

Campo de la invención

La presente invención se refiere a la distribución y la gestión de datos, particularmente documentos, en repositorios de datos electrónicos de gran escala y a aplicaciones de software creadas para acceder a los repositorios de datos y utilizar los mismos.

Antecedentes de la invención

Con el uso de ordenadores y aplicaciones basadas en la web, se puede hacer que esté accesible en línea para los usuarios finales una cantidad cada vez mayor de información. En los últimos tiempos, las bases de datos en línea se especializaron en el contenido, abarcando solamente un tipo particular de registros, tales como marcas comerciales, o artículos técnicos en un campo particular. Así, las bases de datos y las herramientas de acceso estaban diseñadas particularmente teniendo en mente ese contenido. Un usuario con múltiples necesidades informativas se enfrentaba a un entorno como el que se muestra en la figura 1. Cada necesidad informativa requería trabajar con un sistema aparte y con su interfaz de usuario particular, que proporcionaba acceso a una base de datos particular (o conjunto de bases de datos relacionadas), a la que prestaba acceso un software de acceso y facturación particular para ese recurso de información. Las mejoras en la capacidad y la velocidad del almacenamiento masivo han permitido que las bases de datos aumenten de tamaño enormemente y han permitido que un único proveedor ofrezca múltiples bases de datos.

No obstante, el aumento del tamaño de las colecciones de información computarizada y las expectativas de los usuarios en cuanto a velocidad de recuperación y formas de utilización y suministro de documentos cómodas para el usuario, crean desafíos para los proveedores de información. Las interfaces de usuario y los sistemas de acceso heredados son normalmente cómodos para el usuario únicamente para aquellos muy acostumbrados a utilizarlos, y la mayoría de interfaces de usuario y sistemas de acceso personalizados para un contenido particular presentan diferencias entre ellos que hacen que resulte difícil que un usuario de un sistema se cambie fácilmente a un sistema para otro contenido. Incluso si los usuarios pueden aceptar las diferencias, los operadores han observado que no es eficiente simplemente cargar sistemas heredados independientes en procesadores más rápidos con dispositivos de almacenamiento mayores.

Además, el uso de un sistema heredado para acceder a la base de datos de otro sistema y compartir la misma es habitualmente difícil, cuando no imposible. Incluso si pueden compartirse bases de datos heredadas entre sistemas heredados, pueden surgir otras ineficiencias. Normalmente, la misma información es solicitada por usuarios diferentes con finalidades diferentes. Así, se puede hacer que la misma información esté disponible a través de múltiples recursos o canales de búsqueda (tales como un servicio de documentos jurídicos o económicos, por contraposición a un servicio de noticias) para diferentes tipos de usuarios. Para conseguir que la misma información esté disponible a través de múltiples recursos, normalmente los datos se duplican y se almacenan en bases de datos independientes. Además, se pueden usar diferentes aplicaciones de consulta de usuario, que incluyen interfaces de usuario asociadas, para acceder a cada base de datos independiente. Típicamente esta disposición es ineficiente, ya que requiere esfuerzos de desarrollo y de soporte por duplicado y un almacenamiento de la información también por duplicado. Por otra parte, dificulta todo cambio sobre sistemas existentes como respuesta a cambios de las relaciones con los clientes o de las condiciones de los mercados.

Para el proveedor de información, dos de los objetivos básicos son maximizar la información disponible para su venta a usuarios y maximizar la flexibilidad en la comercialización de la información. Estas finalidades significan que el proveedor de información puede atraer a muchos tipos diferentes de usuarios, ofrecer diferentes niveles de acceso y modos de suministro y distribuir información que está personalizada en cuanto a contenido y formato. La persecución de estos objetivos permite que el proveedor de información disponga de una mayor flexibilidad en el emparejamiento de usuarios diferentes con diferentes productos y tarificación, y en la aceptación de contenido y clientes nuevos.

Para hacer frente a estos objetivos, surge la necesidad de un sistema centralizado de gestión de información y de bases de datos de información que permita que múltiples usuarios accedan a la misma información desde aplicaciones diferentes, basadas en la web y de otro tipo. Existe una necesidad adicional de un modelo eficiente de arquitectura/infraestructura para construir dichas aplicaciones con el fin de acceder a grandes agregaciones de documentos almacenados electrónicamente.

El documento WO97/15018 da a conocer un método para proporcionar acceso a información almacenada. En un repositorio se obtienen y almacenan meta-datos sobre objetos de datos. Por medio de un servidor de pasarela, un navegador compatible con HTTP y manejado por un usuario final puede acceder al repositorio y buscar un objeto de datos específico. Después de una búsqueda exitosa, el objeto de datos se entrega al usuario final.

Breve sumario de la invención

En una de las formas de realización, la presente invención es un sistema para mantener una agregación de gran tamaño de documentos almacenados electrónicamente y para hacer que los mismos estén disponibles para usuarios que envían mensajes de consulta.

En una de las formas de realización, la presente invención es un método para entregar resultados de consulta y documentos seleccionados de entre una agregación de gran tamaño de documentos almacenados electrónicamente, a usuarios que envían mensajes de consulta. El método comprende proporcionar acceso a por lo menos una interfaz de usuario para obtener de un usuario un mensaje de consulta en formato electrónico que busca documentos almacenados en formato electrónico en una colección de datos, presentando cada documento un identificador único, y para prever la entrega del mensaje de consulta a un componente de búsqueda con el fin de procesar el mensaje de consulta para identificar documentos de la colección de datos que responden y recuperar un identificador para documentos que responden. El método incluye además proporcionar al usuario, como respuesta al mensaje de consulta, un mensaje de resultado de búsqueda que identifica uno o más documentos y, como respuesta a un mensaje de usuario que selecciona un documento del mensaje de resultado de búsqueda, entregar el documento seleccionado al usuario en un formato predeterminado basado en un perfil de usuario asociado a este último, lo cual se puede materializar en una aplicación de recursos. El método incluye también asociar al documento seleccionado un atributo de instante de tiempo, para permitir la detección de una versión actualizada del documento seleccionado.

Aunque se dan a conocer múltiples formas de realización, para aquellos versados en la materia se pondrán de manifiesto todavía otras formas de realización de la presente invención a partir de la siguiente descripción detallada. Por consiguiente, los dibujos y la descripción detallada deben considerarse de carácter ilustrativo y no limitativo.

Breve descripción de los dibujos

La figura 1 es un diagrama de bloques esquemático que muestra sistemas minoristas de información de la técnica anterior.

La figura 2 es un diagrama de bloques esquemático que muestra cómo el sistema de la presente invención utiliza aplicaciones independientes que usan recursos comunes (servicios compartidos) para proporcionar servicios de usuario diferenciados basándose en un acceso compartido a múltiples colecciones de datos.

La figura 3 es un diagrama de bloques esquemático que muestra una vista general de un sistema para mantener y distribuir una agregación de gran tamaño de documentos almacenados electrónicamente.

La figura 4 es un diagrama de bloques que ilustra los componentes de un marco de aplicaciones basadas en la web, para aplicaciones de consulta y minoristas de documentos, y su interconexión con un sistema común de software de repositorio de contenido.

La figura 5 es un diagrama esquemático que ilustra las capas de cliente, de servidor y de datos de un sistema como el de la figura 4.

La figura 6 es un diagrama esquemático de un registro de documento y un registro de metadatos relacionado, como los utilizados en la presente invención.

La figura 7 es un diagrama de bloques esquemático que muestra las herramientas de desarrollo usadas para construir una aplicación de recursos.

La figura 8 es un diagrama de flujo que muestra el proceso para la admisión de documentos.

La figura 9 es un diagrama de flujo que muestra el proceso para el procesado de enriquecimiento y metadatos de un documento.

La figura 10 es un diagrama esquemático que muestra una arquitectura de tablas de Contenidos (TOC) para la presente invención.

Las figuras 11A a D son diagramas esquemáticos que muestran estructuras de tablas de contenidos, construidas de acuerdo con la presente invención.

La figura 12 es un diagrama esquemático que muestra un ejemplo de tabla de contenido construida de acuerdo con la presente invención.

La figura 13 es un diagrama de bloques esquemático que muestra cómo funcionan los servicios de Seguridad

en la presente invención.

La figura 14 es un diagrama relacional que muestra cómo se implementa la Seguridad y Grupos y Permisos de usuario.

La figura 15 es un diagrama relacional y de definición de datos para el modelo de seguridad de la presente invención.

La figura 16 es un diagrama esquemático que muestra cómo proporciona recortes (del inglés, *clipping*) el servicio de Alerta.

La figura 17 es un diagrama esquemático que muestra cómo funciona el servicio de Historial de Seguimiento.

Las figuras 18A-18B son diagramas relacionales esquemáticos que muestran cómo funciona el servicio de suministro de documentos.

La figura 19 es un diagrama de bloques esquemático que muestra cómo funciona la renderización (del inglés, *rendering*) de documentos con una hoja de estilos de presentación mínima.

La figura 20 es un diagrama de bloques esquemático que muestra cómo funciona la renderización de documentos con una hoja de estilos personalizada y con múltiples hojas de estilos.

Descripción detallada

A. Visión general del sistema

La figura 2 es un diagrama de bloques que muestra cómo el sistema de la presente invención usa recursos comunes o servicios compartidos del sistema para proporcionar a usuarios 6 servicios de recursos de información diferenciados sobre la base de un acceso compartido a múltiples colecciones de datos 30 con metadatos 50. Más específicamente, la figura 2 muestra cómo podrían interaccionar con el sistema múltiples usuarios 6, por ejemplo, el $Usuario_a$ interesado en noticias económicas, el $Usuario_b$ interesado en documentos jurídicos y el $Usuario_z$ interesado en documentos tecnológicos y de patentes. (Estas materias de interés son ejemplos; los usuarios pueden estar interesados en áreas amplias de información jurídica, impositiva, de contabilidad, médica, de propiedad intelectual científica, de material escolar educativo o de noticias, o en sectores especializados dentro de dichos campos). Cada usuario envía un mensaje de consulta de usuario respectivo, Q_{a1}, Q_{b1} y Q_{z1}, que es recibido por el software de aplicación de recursos 15 respectivo del usuario, por ejemplo, Apl de $Recursos_A$, Apl de $Recursos_B$ y Apl de $Recursos_z$, cada uno de los cuales está diseñado para satisfacer las necesidades documentales particulares del usuario que se abona a ese recurso de información y que compra el acceso al mismo. Cada una de las aplicaciones de recursos 15 (Apl de $Recursos_A$, Apl de $Recursos_B$, y Apl de $Recursos_z$) traslada el mensaje de consulta de usuario respectivo Q_{a1}, Q_{b1} y Q_{z1} a un conjunto de servicios/herramientas de sistema compartidos 20, software y hardware que llevan a cabo una variedad de funciones necesarias para proporcionar las características de recursos de información que comprenden cada recurso de búsqueda.

Con el fin de obtener una visión general, los servicios/herramientas compartidos más importantes son el servicio de búsqueda, que procesa las consultas de usuario e intenta encontrar uno o más documentos que responden con respecto a la consulta, y los servicios de seguridad, contabilidad y comerciales que posibilitan la venta minorista de información. El servicio de búsqueda compartido analiza el contenido de las colecciones de datos apropiadas 30 (incluyendo metadatos asociados 50), cada una de las cuales incluye uno o más documentos con metadatos asociados 50. Para simplificar, cada colección de datos se muestra de manera que presenta solamente dos documentos. Así, la $Colección\ de\ Datos_1$ tiene los documentos D_{11}, D_{12}, la $Colección\ de\ Datos_2$ tiene los documentos D_{21}, D_{22} y la $Colección\ de\ Datos_n$ tiene los documentos D_{n1}, D_{n2}. Aunque cada usuario podría disponer de una suscripción que proporcionase acceso a solamente una colección de datos 30 y la colección de datos accesible de cada usuario podría ser diferente con respecto a la correspondiente de los otros, el sistema no está limitado en este sentido. En el ejemplo de la figura 2, el $Usuario_a$ ha enviado una consulta que accede a la $Colección\ de\ Datos_1$, mientras que el $Usuario_b$ ha enviado una consulta que accede tanto a la $Colección\ de\ Datos_1$ como a la $Colección\ de\ Datos_2$. Además, el $Usuario_z$ ha enviado una consulta que accede tanto a la $Colección\ de\ Datos_2$ como a la $Colección\ de\ Datos_z$.

El $Usuario_a$ recibe el documento D_{11a} de la $Colección\ de\ Datos_1$ como respuesta a su consulta Q_{a1}. D_{11a} tal como se distribuye tiene una forma y un formato particulares del documento almacenado como D_{11}, sobre la base de las características de la Apl de $Recursos_A$. El $Usuario_b$ recibe dos documentos como respuesta a su consulta Q_{b1}, el documento D_{11} de la $Colección\ de\ Datos_1$ (que es el mismo documento que respondió a la consulta Q_{a1}) y el documento D_{21} de la $Colección\ de\ Datos_2$. Tal como se muestra en la figura 2, al documento D_{11} tal como se distribuye al $Usuario_a$ se le puede asignar una forma o formato característico D_{11a} determinado por la Apl de $Recursos_A$ y que no sea igual a la forma o formato D_{11b} en el cual la Apl de $Recursos_B$ distribuye el documento

almacenado como D_{11}. El Usuario$_z$ recibe también dos documentos como respuesta a la consulta Q_{z1}, el documento D_{22} de la Colección de Datos$_2$ y el documento D_{n2} de la Colección de Datos$_n$. Nuevamente, la Apl de Recursos$_z$ distribuye cada uno de estos documentos en una forma y un formato particulares del documento almacenado como D_{22} y D_{n2}, a saber, D_{22z} y D_{n2z}, basándose en las características de la Apl de Recursos$_z$. Así, la figura 2 muestra que dos aplicaciones de recursos 15 diferentes pueden acceder, cada una de ellas, a la misma colección de datos 30, y, de hecho, pueden acceder al mismo documento de esa colección. Además, la figura 2 muestra que cada aplicación de recursos 15 hace uso de los servicios/herramientas de sistema compartidos 20, aunque puede provocar que un documento, tal como se entrega su usuario, difiera parcialmente, en cuanto a forma y formato, con respecto a la forma/formato distribuido de ese mismo documento según es distribuido por otra aplicación de recursos. Los recursos en los servicios y niveles de datos son compartidos, pero los resultados distribuidos a los usuarios pueden diferenciarse.

La figura 3 es un diagrama de bloques esquemático que muestra una visión general de un sistema para mantener y distribuir una agregación de gran tamaño de documentos almacenados electrónicamente, incluyendo la funcionalidad descrita para la figura 2. Se representan únicamente elementos funcionales de nivel superior. Debe conseguirse que estos documentos almacenados estén a disposición de una población diversa de usuarios. Los elementos del sistema incluyen una o más Interfaces de Usuario de Aplicaciones de Recursos (RA) 10a, 10b... 10n que acceden a y/o generan y distribuyen diversas pantallas estáticas e interactivas para usuarios, con el fin de sonsacar y aceptar como entrada uno o más Mensajes de Usuario 12a, 12b,... 12n; un Componente de Búsqueda 22 compartido; una o más Bases de Datos o Colecciones de Datos 30a, 30b (para simplificar se muestran únicamente dos); un Componente de Suministro de Documentos 40 compartido; uno o más Archivos de Metadatos 50a, 50b (nuevamente, para simplificar se muestran solamente dos); y un Componente de Admisión 60 para procesar documentos de una cola de Documentos de Noticias 70, con un Componente de Enriquecimiento 80, un Componente de Prioridad 90 y un Control de GUID 100.

Cada Interfaz de Usuario de RA 10a, 10b... 10n forma parte de una aplicación de recursos 15a, 15b,..., 15n, una colección de software que sirve como recurso de acceso a información en una o más áreas de materias de estudio. Una aplicación de recursos materializa una oferta comercial deseada particular (es decir, un “producto”) y/o responde a una necesidad de usuario o perfil de usuario particular. Como tal, una aplicación de recursos puede diferenciarse con respecto a las otras por: el contenido/materia de estudio accesible; el grado de enriquecimiento del documento, las características de la interfaz de usuario; los formatos o modos de suministro del documento; la tarificación; y otras características que atraen a una necesidad de recursos o un mercado de usuarios particular.

Los diversos componentes de la presente invención proporcionan un conjunto de herramientas que permiten una compartición de contenido esencialmente sin fisuras para documentos almacenados por diferentes aplicaciones de recursos, y que permiten también un acceso según una variedad de formas (por ejemplo, por medio de sitios web, intranets, extranets, inalámbricamente, y otros). La presente invención incluye también una implementación por medio de una infraestructura común de aplicaciones de recursos para proporcionar servicios y herramientas compartidos del sistema a aplicaciones de recursos (Servidor de AR 300, figura 4). Para la comercialización minorista de información, los servicios de seguridad y de facturación son servicios compartidos significativos. Cada aplicación de recursos se desarrolla para prestar servicio a ciertos perfiles de usuario y mercados facilitando acceso a por lo menos una parte de los documentos almacenados y mantenido por el software de servidor de datos y de repositorio de contenido común (Servidor de CCRDS 400, figura 4). Los beneficios de definir e implementar para aplicaciones de recursos un conjunto de servicios y herramientas compartidos, tales como los que se describen a continuación incluyen: capacidad de reutilización, tiempo reducido para desarrollar aplicaciones, y costes reducidos para el desarrollo de nuevas aplicaciones.

En uno de los sistemas aplicables a la presente invención, se almacena una agregación de gran tamaño de datos en un repositorio de contenido común gestionado por un Servidor de CCRDS 400, aunque dicha agregación se puede diseminar por una pluralidad de servidores accesibles y se puede mantener de forma redundante. Los datos cubren una amplia gama de materias de estudio y pueden acceder a ellos diversos grupos de usuarios por diferentes motivos. Por tanto, puede resultar útil ofrecer tipos diferentes de acceso de usuario a los datos a través de diferentes interfaces de usuario 10a, 10b, ... 10c. Cada interfaz de usuario se puede implementar en pantallas adaptadas para dar satisfacción a ciertas características y necesidades específicas de los usuarios. La interfaz de usuario no solamente se puede personalizar para proporcionar una facilidad de formulación de mensajes de usuario que realizan consultas o solicitan documentos, sino que también se puede personalizar para proporcionar formas y formatos apropiados del usuario para el suministro de documentos, es decir, los datos distribuidos desde el repositorio común se pueden adaptar y formatear específicamente para la interfaz de usuario particular.

Por diversos motivos, la información almacenada se almacena en forma de documentos individuales dentro de una agregación de documentos. La agregación de documentos se puede partir en una o más colecciones de documentos. Tal como se usa en la presente, un documento se define en términos generales como una unidad de datos coherente que recibe un identificador único universal (GUID), tal como un artículo periodístico, una argumentación judicial (informe jurisprudencial), una resolución reguladora, un informe, un archivo electrónico o

un registro de base de datos, u otro formato habitual (en soporte o bien de papel o bien electrónico) en el cual un autor o fuente prepara información. Un grupo de documentos relacionados se puede almacenar de manera conjunta (en términos lógicos, no necesariamente de manera física) en forma de una colección, y una o más colecciones se pueden almacenar conjuntamente (de nuevo en términos lógicos, no necesariamente de manera física) en forma de un conjunto. En general, un documento dado aparece solamente en una colección.

El uso de colecciones y conjuntos puede facilitar la amplitud de búsqueda de un usuario permitiendo que el mismo especifique (o que el sistema seleccione) una búsqueda dentro de un conjunto o colección particular, interpretada de la manera habitual, por ejemplo, un conjunto particular de informes jurisprudenciales regionales; una categoría particular de publicaciones periódicas, tales como revistas jurídicas; una colección de registros, tales como registros de defunción, de la propiedad o de marcas comerciales. Cada documento está indexado por lo menos una vez dentro de una colección y un conjunto. Esta disposición de colecciones y conjuntos permite también que el sistema reduzca la carga de búsqueda dirigiendo la búsqueda a colecciones o conjuntos particulares y no siendo necesario que cada búsqueda cubra el repositorio completo de documentos. Un repositorio de documentos puede ser extremadamente grande, conteniendo en total 20 Terabytes o más de información.

En algunos campos se generan constantemente documentos nuevos, en ocasiones incluso con una frecuencia tan elevada que los mismos aparecen o se generan cada pocos minutos o segundos y se distribuyen a la cola de documentos 70 en tiempo real por FTP u otro formato. A medida que el repositorio de datos se actualiza o sus colecciones se amplían, pueden añadirse documentos nuevos a una o más colecciones de datos 30a, 30b. Debido a que a los documentos nuevos accederán posteriormente usuarios diferentes por motivos diferentes, es deseable que las funciones de admisión de documentos llevados a cabo por el Componente de Admisión 80 proporcionen una base apropiada. No resultaría deseable tener que repetir la admisión de un documento dado o, excepto en circunstancias no habituales, tener que editar su contenido después de que se haya añadido el repositorio. Sin embargo, puede que este mismo documento necesite ser modificable como parte de su suministro, basándose en la interfaz de usuario y la aplicación de recursos que pueden utilizarse para acceder al documento (por otra parte, puede que la aplicación de recursos no exista en el momento en el que el documento entra en una colección de documentos). Así, cada documento se almacena preferentemente en XML u otro formato de documentos que permita una flexibilidad en su posterior publicación y, en el momento de la creación o entrada, se le proporcionan atributos que facilitan la flexibilidad. Además, puede crearse también un registro o entrada de archivo de metadatos asociado al documento. Este registro o entrada de archivo de metadatos permite que el contenido del documento se enriquezca de ciertas maneras que pueden resultar adecuadas para el propio documento en el momento de su admisión, y permite también una posterior mejora de la información disponible para un usuario, modificando los metadatos asociados a un documento en la admisión sin modificar el propio documento. (Tal como se usa en la presente, metadatos significa información sobre información y podría ser cualquier información sobre un documento que sea útil o bien para un usuario, o bien para el sistema, o para ambos).

La figura 6 muestra un diagrama esquemático de un registro de documento 110 tal como el que se podría almacenar en una colección de datos 30a, 30b y un archivo de metadatos 150 en el cual se almacena un registro de metadatos 152 asociado al documento 110. El registro de documento 110 incluye campos tales como título 112, autor/editor 114, fecha 116, GUID (identificador global universal) 118, y sello de PIT (instante de tiempo). El documento puede incluir campos 126 para recibir datos de enriquecimiento preparados en la admisión, y puede incluir un campo opcional “ntocview” 127 que relaciona el documento con una o más Tablas de Contenido (tal como se describe posteriormente de manera adicional). El documento puede contener uno o más enlaces insertados 122 o atributos de clasificación 124. Un documento puede contener texto, imágenes fijas o en movimiento, sonido u otras formas de contenido. La naturaleza de contenido puede ser otro atributo 111 capturado o bien en el registro de documento 110 o bien en un registro de archivo de metadatos 152.

Según se ha mencionado, un documento procesado por el Componente de Admisión 80 se puede enriquecer proporcionando atributos al archivo según se recibe de una fuente (por ejemplo, un editor de noticias, un editor de publicaciones, un mercado de valores, un tribunal o un organismo regulador). Los atributos particulares proporcionados dependerán del tipo de documento que se esté añadiendo. Los atributos se pueden especificar como parte de los archivos de metadatos. También se pueden especificar como parte de una aplicación de recursos particular.

Los atributos pueden satisfacer por lo menos dos funciones. La primera es útil dentro del sistema o para el usuario. Es decir, se puede crear contenido o modificadores de contenido específicos, así como características funcionales (por ejemplo, que muestren relaciones de navegación con respecto a otros documentos o enlaces que establecen activamente conexiones de navegación). La segunda es características de reconocimiento de marcas, ya que la percepción de una fuente es normalmente tan importante como el propio documento. Esta identidad de marca se puede establecer por el aspecto final de un documento, el cual se puede facilitar mediante ciertos atributos de identidad de marca añadidos al documento, tales como un contenido de valor añadido, derivado especial o de formato único, producido a partir del contenido según se distribuye desde una fuente.

Por ejemplo, un documento o procesado para su admisión se puede relacionar con un informe sobre valores o un análisis fundamental dado. El contenido textual o fáctico del documento es en general estático una vez que se ha creado. Dos proveedores diferentes, que presentan, cada uno de ellos, un nivel de calidad percibido, pueden ofrecer a los usuarios acceso al contenido de ese documento. De este modo, cada uno de los proveedores dará acceso a ese documento, aunque puede desear que el mismo tenga el "aspecto y percepción" de su propio sistema minorista de información. Así, los atributos y/o los archivos de metadatos del presente sistema se pueden utilizar para suplementar o enriquecer el documento según se proporciona desde su fuente, de manera que pueda percibirse como un producto más exclusivo y de marca y valor añadido cando se presenta por medio de una interfaz de usuario o aplicación de recursos específica.

Uno de los atributos que se puede asociar a un documento para su enriquecimiento en la admisión incluye enlazar el documento (por ejemplo, mediante la inserción de un hiperenlace 122) para un posible uso posterior de una manera que dependa de la aplicación de recursos. Por ejemplo, un informe jurisprudencial adecuado para una aplicación de recursos jurídicos podría incluir enlaces a otros informes jurisprudenciales referenciados internamente que se puedan encontrar en el repositorio del contenido común (o en algún otro lugar, tal como la Malla Informática Mundial). Un artículo periodístico se puede enlazar a contenido específico relevante para las personas o eventos identificados dentro del historial. La accesibilidad de dichos enlaces 122 puede ser contingente, es decir, en función de la aplicación de recursos no se puede sacar a través de una interfaz de usuario en algunos contextos. Por ejemplo, un usuario no legítimo de un servicio de noticias puede acceder a un informe jurisprudencial. Los enlaces a otros informes jurisprudenciales pueden no estar activos o no ser aptos para este usuario de la misma manera que lo estarían para un usuario profesional legítimo. Tal como se ha indicado, el registro de archivo de metadatos 152 asociado a un documento es otra de las ubicaciones para atributos añadidos a un documento en el momento en el que el mismo se añade al repositorio. El registro de metadatos 152 almacena información (o enlaces 156 a otros archivos de metadatos) que puede ser usada por una aplicación de recursos para superponer, añadir, eliminar, o modificar datos, parámetros, o formatos de presentación para un documento en relación con un mensaje de consulta de usuario, sin cambiar el propio registro de documento 110 almacenado original.

Como parte del proceso de admisión de documentos, un registro de documento 110 se puede enriquecer con material editorial. Es decir, se puede insertar contenido editorial de valor añadido en un documento, o el mismo se puede asociar o adjuntar al documento mediante la adición de un registro de metadatos. Por ejemplo, con un informe jurisprudencial, pueden crearse y añadirse notas introductorias o un resumen. Este material se puede crear de manera manual o, en algunos casos, automáticamente. Por ejemplo, se pueden comprobar, en cuanto a las citas, casos a los que se haga referencia en un informe jurisprudencial, y dichas citas se pueden actualizar automáticamente. De manera adicional, documentos nuevos se pueden etiquetar mediante una variedad de atributos de clasificación 124 que se pueden usar como indicaciones para alguna agrupación colectiva, por ejemplo, jurisdicción, tema, etcétera.

B. Visión general del proceso de consulta de usuario

La comercialización minorista de información se puede realizar con una variedad de relaciones con los clientes. No obstante, en la mayoría de los casos, existirá un algún tipo de contrato de cliente que define la suscripción o términos de acceso que ha adquirido el cliente. Este contrato puede especificar límites sobre la materia de estudio a la que se puede acceder, horas de acceso, etcétera, y puede definir la tarificación. Los contratos se pueden suscribir en papel o en línea, y con cierta antelación a cualquier uso del repositorio de información. Una vez se tenga la garantía apropiada del pago, los contratos también se podrían suscribir inmediatamente antes del uso. Una vez que se ha definido la relación contractual entre el proveedor de información y el usuario, este último tendrá acceso a por lo menos una parte del repositorio de contenido común por medio de una o más aplicaciones de recursos y su(s) interfaz(ces) de usuario.

Una de las finalidades de la invención es permitir que el proveedor de información defina esencialmente cualquier producto/servicio de información y la relación de cliente que se desee para acceder a partes del repositorio del contenido común y el suministro de documentos. Así, las relaciones pueden implicar varios parámetros que pueden variar de un cliente a otro, incluyendo: colecciones o conjuntos a los cuales se permite el acceso; horas de acceso, números de usuario u otros límites de carga sobre la disponibilidad del acceso; el aspecto o contenido de las pantallas presentadas al usuario, por medio de las cuales este último introduce consultas o solicitudes y recibe resultados; los modos y/o formatos de suministro de documentos que pueden solicitarse; y las tarifas para varias formas de uso. Así, el minorista puede desear desarrollar aplicaciones de recursos que soporten una variedad de relaciones, con el fin de permitir que el sistema proporcione servicios que se ajusten a los diversos términos comerciales acordados.

Será útil una versión general de cómo una aplicación de recursos proporciona acceso al repositorio de contenido común. Inicialmente, un usuario final busca un documento a partir del sistema en una consulta de usuario final. Tal como se muestra en la figura 3, esta solicitud está en forma de un Mensaje de Usuario 12a, 12b, ..., 12n. Se pueden introducir múltiples Mensajes de Usuario de usuarios simultáneos, a cualquier hora del día, desde cualquier lugar del mundo (a no ser que se impongan limitaciones por la relación de cliente definida). Un Mensaje

de Usuario se puede ajustar a una serie de formatos diferentes, tales como una operación de Encontrar, Buscar o Explorar. Por ejemplo, si un usuario conoce información específica, tal como el título y el autor de un documento, el usuario pediría al sistema que Encontrase el documento específico. Por otro lado, si el usuario estuviese buscando información general sobre una materia, el usuario llevaría a cabo una Búsqueda o Exploración para encontrar cualquier información pertinente o precisar una consulta. Estos son solamente ejemplos de los múltiples tipos de Mensajes de Usuario que se pueden definir en una aplicación de recursos y que pueden ser recibidos por el sistema desde el usuario.

Cada Mensaje de Usuario es introducido en el sistema por un usuario a través de la utilización de una o más Interfaces de Usuario RA 10a, 10b,... 10n. Cada Interfaz de Usuario puede tener un aspecto y percepción únicos y puede facilitar la recuperación de clases específicas de documentos por partes de los usuarios, en función del tipo de Interfaz de Usuario que se esté usando. Por ejemplo, la Interfaz de Usuario de una aplicación de recursos diseñada para recuperar documentos jurídicos estará adaptada para acceder a documentos diferentes con respecto a los de una Interfaz de Usuario diseñada para recuperar artículos de periódicos. Probablemente, estas Interfaces de Usuario de aplicaciones diferentes tendrán un aspecto y percepción diferentes, ya que están diseñadas para acceder a tipos diferentes de documentos y atraer un perfil de usuario diferente.

Después de que un usuario haya introducido una solicitud en forma de un Mensaje de Usuario 12a, 12b, ... 12n, se usa un Componente de Búsqueda 22 para encontrar documentos pertinentes. El Componente de Búsqueda usa palabras o expresiones clave de la solicitud del usuario para determinar dónde están localizados los documentos relevantes y para distribuir un mensaje de resultado de búsqueda que identifica uno o más documentos. El Componente de Búsqueda finalmente encuentra el identificador de GUID para cada documento, lo cual permitirá que el documento sea extraído fácilmente de una colección. En algunas situaciones, el componente de Búsqueda distribuye una lista de “coincidencias” en lugar de documentos concretos, y un Mensaje de Usuario adicional define la selección de un documento particular para su visión u otro suministro. El Componente de Búsqueda se describe de forma más detallada posteriormente.

Cada documento que se almacena en una o más colecciones de datos 30a, 30b se almacena con un instante de tiempo (PIT) de precisión 120, de un componente de sellos de tiempo del componente de Control de GUID. El campo de PIT puede ser un valor de tiempo real del reloj, aunque también puede ser un identificador de secuencia o versión que simplemente muestra, para un documento dado, dónde se encuentra una versión particular con respecto a otras versiones. Por ejemplo, pueden construirse identificadores de versión basándose en GUID’s: GUID.00, GUID.01, GUID.02, etcétera (esto puede resultar particularmente útil para documentos legislativos). Así, si se modifican documentos o datos relacionados a lo largo del tiempo (por ejemplo, añadiendo o cambiando los metadatos asociados), el PIT puede ayudar al sistema a detectar si se ha presentado al usuario la versión más actual de un documento y los datos asociados. Permite también presentar una versión actualizada de un documento en caso de que la función de búsqueda dé con ese documento, a pesar de que se haya presentado previamente una versión anterior del documento. Además de un tiempo de admisión PIT, una aplicación de recursos puede realizar un seguimiento del tiempo de entrega, que puede ser útil para esta última función de actualización.

Una vez que se ha solicitado un documento de una base de datos, puede utilizarse un Componente de Suministro de Documentos 40 para distribuir el documento al usuario. El Componente de Suministro 40 presenta el documento en un formato y en un modo (por ejemplo, correo electrónico, fax, por mensajero) seleccionado por el usuario de entre aquellos que ha puesto a su disposición la aplicación de recursos. Por lo tanto, el mismo documento, cuando se distribuye, puede presentar un aspecto y un modo de suministro diferente en función de qué Interfaz de Usuario y aplicación de recursos recibieron la solicitud del mismo.

Tal como se muestra en la figura 3, los documentos almacenados en las bases de datos o colecciones de datos 30a, 30b están asociados a archivos de metadatos 50a, 50b. Los archivos de metadatos pueden incluir una variedad de información adicional asociada a cada documento. Esta información no forma parte del propio contenido del documento, pero se puede acceder a ella durante una búsqueda de documentos relevantes o al mismo tiempo que se accede al propio documento para su suministro.

Cada Documento Nuevo se coloca inicialmente en una Base de Datos 30a, 30b por medio de un Componente de Admisión 60. Las Bases de Datos se actualizan con documentos de una manera constante y frecuente, y por lo menos algunos de los documentos requieren una publicación inmediata. Por ejemplo, los informes sobre precios de valores y los artículos sobre las últimas noticias deberían ponerse a disposición lo antes posible. Su relevancia normalmente dura poco tiempo y su valor está asociado a su puntualidad. El Componente de Prioridad 90 del Componente de Admisión prioriza documentos entrantes para su procesado, usando uno o más niveles de prioridad para procesar selectivamente documentos que están fuera del orden de recepción en el tiempo con requisitos de tiempo real u otros de disponibilidad especial que se pueden definir para una aplicación de recursos particular (por ejemplo, una aplicación de recursos que promete disponibilidad de informes jurisprudenciales o artículos periodísticos en el momento de la publicación). El componente de Control de GUID 100 puede comprobar la unicidad del identificador de documento asignado para cada documento, antes de que el mismo se ponga a disposición de cualquier usuario en una Colección de Datos. El Componente de Admisión

60 también puede comprobar un formato predeterminado del documento antes de hacer que el documento esté disponible para un usuario. Estas características ayudan a garantizar que documentos emitidos para las Colecciones de Datos 30a, 30b estén preparados para su acceso por parte del sistema.

Procesado de admisión

La figura 8 muestra, en forma de diagrama de flujo, el proceso de admisión 800. En 802, el sistema 5 (figura 3) recibe un archivo transmitido desde una fuente, tal como un servicio de noticias, un tribunal, un servicio de datos de mercados, y, en 804, el Componente de Admisión 60 convierte los archivos de un formato de transmisión a un formato más adecuado para el procesado de admisión. En 806, se aísla un documento individual para su procesado y, en 808, el Componente de Admisión busca un código de prioridad que puede haber sido pre-asignado por la fuente o cuya asignación pueda ser necesaria en este momento. En 810, el documento se almacena en una o más colas para un procesado adicional por prioridad. En 812, el sistema comprueba la presencia de documentos adicionales en un archivo y/o archivos adicionales recibidos y, si cualquiera de ellos está presente, vuelve al punto de ejecución apropiado para procesar el siguiente documento o archivo.

En 814, otro recurso de procesado accede a las colas de documentos seleccionando el documento con la prioridad más alta. En 816, el Componente de Admisión busca un GUID que pueda haber sido pre-asignado por la fuente (en coordinación con el sistema, que debe garantizar la unicidad de los GUID's) o cuya asignación pueda ser necesaria en este momento sobre la base del historial y algoritmos que garantizan la unicidad. En 818, se comprueba el formato de almacenamiento del documento para garantizar su correcta disposición para el procesado con el Componente de Enriquecimiento 80 en 820.

El Componente de Enriquecimiento se puede usar para mejorar cada documento a medida que el mismo se coloca en una Colección de Documentos 30a, 30b. El Componente de Enriquecimiento añade varias características a cada documento, que incrementan el valor del documento para uno o más grupos de usuarios. El Componente de Enriquecimiento asocia cada documento a uno o más de los siguientes: material editorial adicional preparado por un agente humano; material editorial adicional preparado por un agente automatizado; un enlace que proporciona un puntero a otro documento en la base de datos; o una entrada asociada al documento que aparece en un archivo de metadatos. Estas características de enriquecimiento permiten que el usuario final reciba un producto de valor añadido en forma de un documento individual combinado con contenido adicional de cierto tipo. Puede haber disponibles diferentes formas de enriquecimiento en función de la aplicación de recursos 15 usada para prestar servicio a un usuario particular y distribuir un documento particular.

Tras el procesado de enriquecimiento en 820, el documento se puede someter al procesado de un componente de extracción de metadatos en 822. Los metadatos desarrollados en este procesado implican generalmente la extracción de datos que son importantes para conectar este documento en una o más colecciones. Así, el contenido del documento se puede analizar para desarrollar una clasificación lingüísticamente inteligente de este documento con respecto a otros documentos en la misma o diferentes colecciones. Se pueden desarrollar varias formas de metadatos que ayuden en el almacenamiento o la recuperación, y en la modificación o personalización de un documento con el fin de proporcionar una base para características de una o más aplicaciones de recursos.

Todavía en referencia a la figura 8, después de que se extraigan metadatos, en 824 se almacena un documento con un PIT correspondiente a su tiempo de emisión para el acceso. En 826, el sistema determina si hay más documentos en las colas de prioridad a procesar. En caso negativo, el procesador de documentos pasa a un estado de espera en 828, si hay más documentos, entonces el control pasa al punto de ejecución en el cual se selecciona el siguiente documento de entre las colas de prioridad.

Los documentos en por lo menos una colección de datos en la que se coloca un documento nuevo se pueden partir en por lo menos un subconjunto de colección, y el componente de admisión para recibir documentos nuevos puede garantizar que cada documento adicional tenga un identificador único y se asigne a por lo menos un subconjunto de colección. Otra colección de datos puede tener por lo menos un conjunto de documentos que sea una agregación de los documentos de uno o más subconjuntos de colección.

Procesado de enriquecimiento y de metadatos

La figura 9 muestra un diagrama de flujo correspondiente al proceso 900 de enriquecimiento de documentos y extracción de metadatos según se ha hecho referencia en la figura 8. En 902, los componentes usados para el procesado de enriquecimiento de documentos reciben un control y reciben un documento para su enriquecimiento. En 904, se aplica un agente de enriquecimiento automatizado, y las características de enriquecimiento producidas por el agente se usan para potenciar el documento. Por ejemplo, el agente podría buscar nombres de individuos o empresas en un artículo periodístico o en un caso, y, a continuación, podría construir un archivo para su presentación en la barra lateral, que podría ser consultado por una persona que navegase por el documento. Tras la aplicación del agente de enriquecimiento automatizado, puede tomarse un camino de derivación 905 a la etapa 910 cuando no sea necesaria una asignación a un editor de enriquecimiento

humano. Si no se toma la derivación 905, en 906, el documento se asigna a un editor de enriquecimiento humano para su revisión y edición. En 908, el editor de enriquecimiento humano devuelve un archivo con un documento más potenciado. En 910, el documento potenciado se distribuye para un procesado de metadatos y, en 912, el componente de extracción de metadatos recibe el documento distribuido. En 914, se aplica un motor de metadatos automatizado al documento, para extraer metadatos, y, en 916, los archivos de metadatos se recopilan y se asocian al documento. Por ejemplo, los metadatos extraídos se pueden desarrollar en una capa de metadatos en forma de sentencias del Marco de Descripción de Recursos (RDF) construidas en una capa de metadatos o datos XML. En 918, los archivos de metadatos para este documento se enlazan con archivos de metadatos para otros documentos. Por ejemplo, si el procesado de metadatos ha dado como resultado cierta clasificación lingüística del documento, una tabla, un índice, una tabla de contenido u otro archivo de metadatos a nivel de colección se puede actualizar con información de este documento y/o una referencia al mismo. En 920, a un archivo de metadatos se le pueden añadir etiquetas condicionadas a la aplicación de recursos. Estas son usadas por aplicaciones de recursos particulares para etiquetas metadatos para su inclusión o exclusión en los servicios de recursos de documentos ofrecidos por las aplicaciones de recursos particulares. En algunos casos, se excluirán metadatos de la búsqueda o presentación, sobre la base de la presencia o ausencia de etiquetas que son accesibles por aplicaciones de recursos.

En 922 se almacenan archivos de metadatos. Estos se pueden almacenar acoplados a un documento o no acoplados; es decir, puede haber una asociación de almacenamiento física o simplemente una asociación lógica. En 924, el sistema marca el archivo (o partes) como candidato o no candidato para la adición de metadatos futuros que se puedan obtener mediante el uso de un análisis, por reglas estadísticas o heurísticas, de patrones de uso de documentos a lo largo del tiempo. Con el almacenamiento de los metadatos, el documento que se está procesando resulta preparado para el acceso del usuario, aunque metadatos relacionados pueden cambiar posteriormente. Si el sistema dispone de agentes para realizar un seguimiento de patrones de uso y analizar los mismos, este marcaje puede garantizar que se realiza un seguimiento del uso de este documento cuando resulte apropiado, y que los resultados (por ejemplo, metadatos de uso 154 en la figura 6) se registran. Por otra parte, a medida que se desarrolla información sobre los patrones de uso, los metadatos que se han almacenado en el momento de la admisión del documento se pueden actualizar. Por ejemplo, si este documento es parte frecuentemente de un patrón de búsqueda observado, el archivo de metadatos puede llegar a reflejar otros documentos próximos en una cadena de búsqueda que haya tenido lugar, para ayudar al guiado de usuarios posteriores a lo largo de esa misma cadena. En 926, los componentes de enriquecimiento y de extracción de metadatos devuelven el control al sistema.

Los metadatos son la información de valor añadido que se crea para disponer, describir, realizar un seguimiento de y mejorar de otras maneras el acceso a objetos de información. Tal como se explica de forma más detallada posteriormente, en la presente invención se desarrollan metadatos en forma de tablas de contenidos, derivaciones de tablas de contenidos obtenidas por filtración u otras manipulaciones, datos de patrones de uso obtenidos a partir de información de rastreo del usuario, firmas de documentos desarrolladas para la detección de duplicados, e indexación por *tokens* entre otros métodos. En agregaciones de gran tamaño de documentos, los metadatos pueden ser jerárquicos, por cuanto pueden desarrollarse metadatos de nivel superior para ayudar a dotar de sentido a metadatos de nivel inferior. En otras circunstancias, los metadatos serán no jerárquicos pero, sin embargo, estarán relacionados con otros metadatos mediante enlaces u otros medios no jerárquicos de apuntamiento. Las tablas de contenidos, que se describen a continuación, presentan una excelente oportunidad para desarrollar metadatos de valor añadido.

Construcción de Tablas de Contenido (TOC)

Una de las formas de procesado de metadatos es la construcción de Tablas de Contenido (TOC). Según se implementa en la presente invención, una TOC requiere que se definan dos tipos de colección diferentes. Las colecciones de TOC contienen la relación jerárquica de las TOC. Las colecciones de documentos (DOC) contienen documentos. Una TOC puede hacer referencia a documentos en una, dos o muchas colecciones de DOC. Los GUID persistentes son un requisito para lograr los beneficios del presente diseño de TOC. Cuando un sistema ofrece múltiples tipos de información a los usuarios, el mismo dispondrá típicamente de por lo menos una TOC para cada tipo de información.

La jerarquía de TOC reside en una colección en el repositorio de contenido común y contiene referencias a documentos. Los documentos a los que se hace referencia existen en una o más Colecciones de DOC. Se usan Conjuntos de Colecciones para vincular una única Colección de TOC con la(s) Colección(es) de DOC que contienen los documentos a los que se hace referencia. La figura 10 es una vista general esquemática de la arquitectura de TOC. Los siguientes son detalles adicionales sobre la implementación de la TOC.

a. Carga de datos. Se cargan datos de TOC en la colección de TOC. Se cargan datos de DOC en la(s) colección(es) de DOC. Las colecciones tanto de TOC como de DOC podrían estar cargando datos simultáneamente. Para mantener en sincronización los datos de TOC y de los documentos, hay disponible una promoción síncrona con el fin de permitir que un cliente promueva múltiples colecciones de forma sincronizada.

b. Restricción de una búsqueda basada en un Nodo de TOC. Puede añadirse un elemento “n-tocview” a cargas de datos de documentos para soportar restricciones de vista-consulta-búsqueda de TOC. El elemento “n-tocview” 127 (figura 6) contiene los GUID de TOC que un cliente desea asociar a un documento. El siguiente es un ejemplo del XML usado para actualizar la estructura de TOC de muestra, simplificada, en la figura 11A, en la cual el nodo sombreado representa un nodo de TOC que apunta al documento “d2”.

```
<n-document guid=“d2” control=“ADD”>
    <n-tocview>n1 n2 n5</n-tocview>
    <n-docbody>document data</n-docbody>
</n-document>
<n-node guid=“n1” control=“ADD”>
    <n-label>label information</n-label>
</n-node>
<n-node guid=“n2” control=“ADD”>
<n-parent-guid>n1</n-parent-guid>
    <n-label>label information</n-label>
    <n-rank>1</n-rank>
</n-node>
<n-node guid=“n5” control=“ADD”>
<n-parent-guid>n2</n-parent-guid>
<n-doc-guid>d2</n-doc-guid>
<n-label>label information</n-label>
    <n-rank>2</n-rank>
</n-node>
```

Nota: no se verifica la existencia de los GUIDs especificados en el n-tocview, por el repositorio de contenido común, ni si los mismos son información relacionada dentro de un conjunto de colecciones.

c. API de Envoltura. Se usan Conjuntos de Colecciones para vincular la Colección de TOC con la(s) Colección(es) de DOC que contienen los documentos a los que se hace referencia. La API de Envoltura contiene API de TOC para usar con Colecciones o Conjuntos de Colecciones. Un Conjunto de Colecciones proporciona un punto único con el que se puede usar la API de Envoltura.

d. XML de TOC. Un elemento n-nodo (*n-node*) creará, actualizará y eliminará nodos de TOC. Cada elemento n-nodo contiene información que describe un nodo de TOC. Los datos de TOC no están indexados por *tokens* (como con los documentos), por lo que no se pueden buscar por el repositorio del contenido común. La información de n-tocview se puede situar dentro del documento y, por lo tanto, se puede indexar para la búsqueda.

Un n-nodo tiene dos atributos:

guid – el GUID del nodo de TOC
control – indica la acción que va a producirse.

Valor	Descripción de la acción
“ADD”	Añade un nodo de TOC a esta fase
“DEL”	Elimina un nodo de TOC de esta fase
“DELBRANCH”	Elimina el nodo de TOC y todos los hijos del nodo de TOC para esta fase.

Un n-nodo tiene los siguientes elementos:

n-parent-guid – GUID padre del nodo. Un nodo raíz no contendrá este elemento.

n-doc-guid – GUID de un documento si este nodo de TOC hace referencia a un documento. Un nodo de TOC puede tener ningún o un documento asociado al mismo. Cualquier contenido en este elemento indica que el nodo de TOC hace referencia a un documento.

n-anchor-guid – GUID de un ancla si este TOC hace referencia a un ancla.

n-label – Campo de texto con un límite de tamaño de 598 bytes.

n-rank – Número real usado para ordenar nodos de TOC para que la aplicación los presente en orden de rango.

n-name – Contenido específico del nodo de TOC devuelto a la aplicación. Estos datos no tienen ningún significado para la definición de la TOC dentro del repositorio de contenido común. El valor máximo para n-name es 20 bytes

n-value – Contenido específico del nodo TOC que se devuelve a la aplicación con un valor(es) de n-name. Estos datos no tienen ningún significado para la definición de la TOC dentro del repositorio de contenido común. El valor máximo para n-value es 200 bytes.

n-meta-data – Contiene información de metadatos sobre la TOC.

e. DTD de TOC

```
<!ELEMENT n-node (n-parent-guid?, n-doc-guid?, n-anchor-guid?, n-label?, n-rank?, n-name?, n-value?, n-meta-data?)>

<!ATTLIST    n-node guid ID    #REQUIRED
                    control   (ADD I DEL I DELBRANCH) “ADD”>
                    <!ELEMENT n-label         %n-labelcontent;>
                    <!ELEMENT n-parent-guid    #PCDATA>
                    <!ELEMENT n-doc-guid       #PCDATA>
                    <!ELEMENT n-anchor-guid        #PCDATA>
                    <!ELEMENT n-rank          #PCDATA>
                    <!ELEMENT n-name          #PCDATA>
                    <!ELEMENT n-value         #PCDATA>
                    <!ELEMENT n-meta-data      #PCDATA>
```

f. Ejemplo de XML de TOC y de Documentos En referencia a continuación a la Fig. 11B, los nodos sombreados representan nodos de TOC que hacen referencia a documentos. Dos nodos de TOC hacen referencia al documento “d1”. El nodo de TOC “n7” es un ancla de “n5”.

XML de TOC	XML de documento
<n-node guid=“n1” control=“ADD”> <n-label>*label information*</n-label </n-node> <n-node guid=“n2” control=“ADD”> <n-parent-guid>n1</n-parent-guid> <n-label>*label information*</n-label> <n-rank>1</n-rank> </n-node> <n-node guid=“n4” control=“ADD”> <n-parent-guid>n2</n-parent-guid> <n-doc-guid>d1</n-doc-guid> <n-label>*label information*</n-label> <n-rank>1</n-rank> </n-node> <n-node guid=“n5” control=“ADD”> <n-parent-guid>n2</n-parent-guid>	<n-document guid=’d1’ control=’add’> <n-tocview>n1 n2 n4</n-tocview> <n-tocview>n1 n3 n6</n-tocview> <n-docbody>*document text*</n-docbody> </n-document> <n-document guid=’d2’ control=’add’> <n-tocview>n1 n2 n5</n-tocview> <n-docbody>*document text* <anchor>d2.1</anchor>*anchor info* </n-docbody> </n-document> <n-document guid=’d3’ control=’add’> <n-tocview>n1 n3</n-tocview> <n-docbody>*document text*</n-docbody> </n-document>

XML de TOC	XML de documento
<n-doc-guid>d2</n-doc-guid> <n-label>*label information*</n-label> <n-rank>2</n-rank> </n-node> <n-node guid="n3" control="ADD"> <n-parent-guid>n1</n-parent-guid> <n-doc-guid>d3</n-doc-guid> <n-label>*label information*</n-label> <n-rank>2</n-rank> </n-node> <n-node guid="n6" control="ADD"> <n-parent-guid>n3</n-parent-guid> <n-doc-guid>d1</n-doc-guid> <n-label>*label information*</n-label> </n-node> <n-node guid="n7" control="ADD"> <n-parent-guid>n5</n-parent-guid> <n-doc-guid>d2</n-doc-guid> <n-anchor-guid>d2.1</n-anchor-guid> <n-label>*label information*</n-label> </n-node>	

Obsérvese que:

El documento "d1" podría haber tenido una n-tocview combinada de: <n-tocview>n1 n2 n4 n3 n6</n-tocview> Las anclas contenidas dentro del documento se especifican con etiquetas específicas del cliente. El sistema no tiene ninguna etiqueta de ancla requerida dentro de un documento.

g. Actualización de reglas para n-nodos

1. Sin GUIDS duplicados. Un GUID no se puede añadir dos veces, o eliminar y añadir en la misma carga. Si no se cumple esta condición, la carga falla con errores con de datos.

2. Un n-nodo definido es un nodo sustituido. Si se define un n-nodo en el XML, es necesario redefinir toda la información del n-nodo. Los datos se cargarán con éxito. El nodo reflejará solamente la última definición.

3. Un n-nodo sin hijos se puede eliminar con la función de eliminación. Si no se cumple esta condición, la carga falla con errores de datos.

4. La eliminación de una rama implica que se eliminan el n-nodo y todos sus nodos hijos.

5. Los n-nodos de una rama eliminada no se pueden modificar o añadir en la misma carga que la rama eliminada (véase la regla uno). Si no se cumple esta condición, la carga falla con errores de datos.

6. Un guid padre de un n-nodo debe ser un nodo existente. Si no se cumple esta condición, la carga falla con errores de datos. (Un nodo existente es un nodo que ya existe en la TOC o un nodo que existe en la carga actual. No es necesario cargar los nodos en orden de rango. La verificación de nodos perdidos se producirá al final del proceso de carga. No obstante, puede producirse velocidades de carga mejores si se cargan nodos de una manera jerárquica (por rango)).

h. Casos prácticos de carga de TOC –

Los siguientes son casos prácticos que muestran implicaciones reales de la TOC. Una gran parte de estos ejemplos se basa en la estructura de TOC de la figura 11C. Esta misma estructura se usó en el ejemplo de XML anteriores.

1. ¿Cómo pueden cargarse grandes cantidades de datos con la estructura de TOC?

Supongamos que vamos a cargar 50 giga-bytes de datos sin procesar y que podemos realizar la carga a una velocidad de 500 mega-bytes/hora para una colección dada. Si usamos una TOC y tres Colecciones de Documentos, podemos cargar estos mismos datos en poco más de un día. Si un cliente está dispuesto

a descomponer datos de los documentos por múltiples colecciones, los datos se pueden cargar de manera rápida.

2. ¿Cómo puede eliminarse una rama de la TOC?

Usando nuestro ejemplo, vamos a eliminar una rama de TOC que comienza en el nodo “n2”, y también eliminaremos todo nodo por debajo del nodo “n2”. Ya no se hará referencia al documento “d2” en la TOC. La documentación no se eliminará a no ser que se pase un eliminar para “d2” al repositorio de contenido común.

Aquí se presenta el XML para la eliminación de la rama comenzando con “n2”.

```
<n-node guid=“n2” control=”DELBRANCH”/>  //elimina nodos de TOC “n2”, “n4”, “n5”
```

Aquí se presenta el XML para eliminar el documento “d2”.

```
<n-document guid=“d2” control=”DEL”/>  //elimina documento “d2”
```

3. ¿Cómo puedo cambiar el texto del documento “d1”?

Recargar documento “d1” con los datos de documento nuevos e información de n-tocview que permite restricciones de búsqueda de TOC. Aquí se presenta el XML.

```
<n-document guid=“d1” control=“ADD”>
        <n-tocview>n1 n2 n4</n-tocview>
        <n-tocview>n1 n3 n6</n-tocview>
        <n-docbody>new document data</n-docbody>
</n-document>
```

4. ¿Cómo puede cambiarse la etiqueta del nodo de TOC “n4”?

Recargar el nodo de TOC “n4” con la información de etiqueta nueva. Este mismo ejemplo funciona si se están cambiando los campos de meta-datos, valor, nombre o rango de TOC. Aquí se presenta el XML.

```
<n-node guid=“n4” control=“ADD”>
    <n-parent-guid>n2</n-parent-guid>
    <n-doc-guid>d1</n-doc-guid>
    <n-label>new label information</n-label>
        <n-rank>1</n-rank>
</n-node>
```

5. ¿Cómo puede eliminarse el documento “d3” y dejar el nodo “n3” en la TOC?

Enviar un eliminar para el documento “d3” y redefinir el nodo de TOC “n3” para que no incluya una referencia al documento “d3”. Aquí se presenta el XML.

```
<n-document guid=“d3” control=“DEL”/>      //deletes document “d3”

<n-node guid=“n3” control=“ADD”>                  //redefines TOC
nodes “n3”
        <n-parent-guid>n1</n-parent-guid>
    <n-label>label information</n-label>
        <n-rank>2</n-rank>
</n-node>
```

6. ¿Cómo puede eliminarse el documento “d1” y quitar todos los nodos de TOC que hacen referencia al mismo?

Enviar un eliminar para el documento “d1” y un eliminar para los nodos “n4” y “n6”. Aquí se presenta el XML.

```
<n-document guid=“d1” control=“DEL”>    //deletes document “d1”
<n-node guid=“n4” control=“DEL”/>       //deletes node “n4”
<n-node guid=“n6” control=“DEL”/>       //deletes node “n6”
```

7. ¿Cómo puedo insertar un nodo de TOC nuevo “n7” entre los nodos de TOC “n1” y “n3”?

   La estructura de TOC nueva sería como la de la figura 11D.

   Recargar el documento “d1” y “d3” con una n-tocview nueva para soportar el GUID “n7” como una restricción de vista de consulta. Crear también un nodo de TOC nuevo para “n7” y redefinir el nodo de TOC “n3” para que apunte a “n7” en lugar de “n1”. Aquí se presenta el XML.

```
<n-document guid=“d1” control=“ADD”>   //adds document “d1”
<n-tocview>n1 n2 n4</n-tocview>
    <n-tocview>n1 n7 n3 n6</n-tocview>
    <n-docbody>document data</n-docbody>
</n-document>

<n-document guid=“d3” control=“ADD”>   //adds document “d3”
    <n-tocview>n1 n7 n3</n-tocview>
    <n-docbody>document data</n-docbody>
</n-document>

<n-node guid=“n7” control=“ADD”>                    //inserts toc node “n7”
        <n-parent-guid>n1</n-parent-guid>
    <n-label>label information</n-label>
        <n-rank>2</n-rank>
</n-node>

<n-node guid=“n3” control=“ADD”>                    //inserts toc node “n3”
        <n-parent-guid>n7</n-parent-guid>
    <n-doc-guid>d3</n-doc-guid>
    <n-label>label information</n-label>
        <n-rank>3</n-rank>
</n-node>
```

En resumen, las TOC’s proporcionan una infraestructura para almacenar metadatos jerárquicos sobre documentos. Las TOC’s están constituidas por nodos. Los GUID’s identifican nodos, nodos padre, documentos a los que se hace referencia y anclas. Toda entrada para la construcción de una TOC se encuentra en XML. Una TOC puede ser una estructura recursiva. Esto ocurre cuando el n-doc-guid de un nodo contiene el GUID de un nodo de TOC en lugar del GUID de un documento. Entonces, un nodo de TOC hace referencia a un nodo de TOC. El vocabulario de las etiquetas de nodo en una TOC se puede usar como vocabulario para sentencias de RDF en metadatos.

Un documento puede existir solamente en una colección de DOC en cualquier instante de tiempo. No obstante, con una TOC pueden representarse documentos en múltiples lugares. Los documentos de una o más colecciones de DOC se pueden representar en una TOC.

Los datos jerárquicos de TOC se almacenan en una colección de TOC. Los datos de documentos se almacenan en una o más colecciones de DOC. Una TOC particular reside en una colección. Esa colección de TOC y una o más colecciones de DOC a la que hace referencia la colección de TOC están vinculadas entre sí por un conjunto de colecciones. La figura 12 ilustra una muestra de una TOC simplificada que hace referencia a dos documentos (etiquetados como DG1 y DG2).

El presente diseño de TOC aporta varias características útiles:

Navegación por la TOC: se proporcionan API’s para navegar por los nodos de una TOC. Las siguientes operaciones de muestra se pueden ejecutar a través de una API del tipo mencionado: recuperar nodo raíz de la TOC; dado un nodo, recuperar sus hijos; dado un resultado de búsqueda de TOC y un nodo de TOC, recuperar el siguiente nodo o un nodo previo en el orden de la TOC; y, dado un nodo, recuperar su padre.

TOC con coincidencias: cuando una búsqueda produce coincidencias de documentos, estas se fusionarán para devolver el número de coincidencias en cada nodo de TOC.

TOCs filtradas: si una aplicación de recursos envía una referencia a una búsqueda y un nodo de TOC, se eliminarán las partes de la TOC que no se corresponden con la búsqueda. Si una aplicación de recursos envía una referencia a un manejador de suscripción (una limitación sobre una búsqueda, basándose en una suscripción), se eliminará cualquier TOC que no cumpla los criterios de suscripción.

Encontrar nodos: si una aplicación de recursos envía una referencia a un nombre y/o valor, una TOC devolverá nodos relacionados.

Anclas de TOC: pueden usarse anclas para reflejar una jerarquía dentro de un documento.

Indexación

Como preparación para el procesado de metadatos, normalmente un documento se indexa mediante la creación de un archivo de índices. Dicho archivo de índices se obtiene según la manera convencional mediante tokenización, eliminación de palabras vacías, radicación, eliminación de mayúsculas e inversión. Véase, por ejemplo, la patente de Estados Unidos 6.389.412. La relación del proceso de indexación con respecto a la extracción de metadatos puede resultar interesante. Debido a que la indexación da como resultado cierta pérdida de información semántica, puede que la indexación no sea deseable para algunas colecciones de documentos. En otras colecciones, la indexación es aceptable, pero es mejor llevar a cabo una extracción de metadatos en un documento que no se presenta en formato indexado, en caso de que la información a extraer recaiga total o parcialmente en características perdidas en la indexación. Los metadatos pueden estar o no indexados. En una forma de realización, los datos de TOC no se indexan y, por lo tanto, los mismos no se pueden buscar por medio de motores de búsqueda que se basan en la indexación. No obstante, está disponible para el usuario la exploración según se explica de forma más detallada posteriormente.

Los documentos de una colección de datos se pueden partir en por lo menos un subconjunto de colección. El sistema puede tener un servicio de índices que mantiene un índice de palabras clave que aparecen por lo menos una vez en el subconjunto de colección, con una asociación entre las palabras clave del índice y la posición de su aparición en el subconjunto de colección.

Vista de implementación de los componentes – visión general

La figura 4 es un diagrama de bloques que muestra los componentes principales que implementan los elementos funcionales de nivel superior mostrados o descritos anteriormente. Los componentes principales incluyen el Servidor Web 200, el Componente de Servidor de Recursos de Aplicaciones (AR) 300, y una o más aplicaciones de recursos. En un sistema basado en la web, el Servidor de CCRDS 400 y el Servidor de AR 300 actúan como los servidores, con las aplicaciones de recursos como cliente. Específicamente, el Servidor de CCRDS 400 es un servidor de base de datos (colecciones), y el Servidor Web 200 y el Servidor de AR 300 son, respectivamente, un servidor web y de aplicaciones. Otros componentes del sistema son los Servicios Comerciales en Línea del Sistema 500 y los Sistemas Comerciales 600 y una API de Publicación 700. Dentro de los Sistemas Comerciales 600 se encuentran los componentes de SAP y de Facturación en Línea.

Los componentes del Servidor de AR 300 proporcionan un marco de aplicación usado para desplegar aplicaciones de recursos basadas en la web que acceden a documentos residentes en el repositorio del Servidor de CCRDS 400. Este marco está destinado a proporcionar un cambio radical rápido de aplicaciones de recursos nuevas a las cuales pueden abonarse los usuarios, tales como servicios de noticias, servicios jurídicos, etcétera. En una de las implementaciones, los componentes del Servidor de AR proporcionan objetos serializables que se pueden usar sobre contenedores J2EE.

Aunque se anticipa que los mayores beneficios y la capacidad del sistema de proporcionar información de cliente compartida a través de las múltiples aplicaciones de recursos, se producen con el uso completo de los componentes compartidos del Servidor de AR, una aplicación no tiene que utilizar necesariamente todas las funciones proporcionadas por las arquitecturas del servidor. Algunas de las otras ventajas de la arquitectura del servidor de la figura 4 incluyen capacidad de reutilización, brevedad del plazo de comercialización de aplicaciones nuevas o actualizadas, y costes reducidos para el desarrollo de productos nuevos.

En general, los componentes del Servidor de CCRDS 400 proporcionan acceso a una agregación de gran tamaño de documentos almacenados, indexados y ordenados electrónicamente. Estos documentos se añaden en el repositorio de contenido común y se enriquecen para permitir una recuperación más sencilla y un contenido de valor añadido. Cuando un usuario busca o solicita un documento a través de una aplicación de recursos, el Servidor de CCRDS 400 interacciona con el Servidor de AR 300 para proporcionar los resultados de la búsqueda o el documento de una manera eficiente. El Servidor de CCRDS 400 utiliza una serie de herramientas para enriquecer los documentos. Estas herramientas se describen de forma más detallada posteriormente.

En general, los componentes del Servidor de AR 300 proporcionan un marco de aplicación usado para desplegar aplicaciones de recursos basadas en la web. Los componentes implementan un marco común de servicios y herramientas que reduce el tiempo de desarrollo y los costes para cada aplicación de recursos que recupera documentos usando el Servidor de CCRDS 400. Así, se pueden crear fácil y rápidamente aplicaciones de recursos nuevas por medio de unidades comerciales que permiten una interfaz personalizada, al mismo tiempo que proporcionando acceso a un núcleo centralizado de datos y servicios. Adicionalmente, los componentes del

Servidor de AR 300 promueven una compartición de información sobre un cliente a través de varias aplicaciones. En una de las formas de realización, el marco establece un modelo fijo para el desarrollo de aplicaciones, tal como la Edición Empresarial de Java 2 (J2EE) y otras directrices recomendadas.

El marco ofrece Interfaces de Programación de Aplicaciones (API) para producir un lenguaje de marcado genérico, tal como HTML o XML, aunque el desarrollador de aplicaciones correspondiente a la aplicación de recursos es responsable de proporcionar la interfaz, ya sea una hoja de estilos XML o un objeto Java, que convierte el HTML o XML genérico en el formato requerido por la aplicación de recursos. Los componentes del Servidor de AR 300 proporcionan servicios y herramientas comunes, eliminando así la necesidad de que cada aplicación de recursos desarrolle estos servicios individualmente. Estos servicios y herramientas se explican de forma más detallada posteriormente.

Cada aplicación de recursos es una aplicación única diseñada para proporcionar a usuarios de un mercado particular un producto personalizado para localizar y recuperar documentos. Tal como se ha explicado anteriormente, una aplicación de recursos utiliza servicios y herramientas especiales proporcionados por el Servidor de AR 300 para acceder a un gran repositorio de contenido común de documentos gestionados por el Servidor de CCRDS 400. Con el Servidor de AR 300 y el Servidor de CCRDS 400 pueden interaccionar simultáneamente más de una aplicación de recursos, con el fin de acceder y solicitar el mismo documento; no obstante, al documento se le pueden proporcionar un aspecto y una percepción únicos basándose en la aplicación de recursos usada para distribuir el documento.

Cada aplicación de recursos se desarrolla con sus propios componentes de interfaz, tales como HTML, imágenes JPEG, Páginas de Servidor Java (JSP), Servlets, hojas de estilos personalizadas, etcétera. No obstante, en lugar de que cada aplicación de recursos 15 (figura 3) utilice herramientas y servicios personalizados para comunicarse con un usuario, procesar Mensajes de Usuario, acceder a los documentos almacenados en el repositorio de contenido común, y aplicar la totalidad del resto de reglas comerciales para las transacciones minoristas de información, el Servidor de AR 300 y el Servidor de CCRDS 400 tienen componentes normalizados que permiten que cada aplicación de recursos utilice herramientas y servicios pre-programados. Por ejemplo, el componente de Seguridad del Servidor de AR 300 permite que cada aplicación de recursos utilice las mismas características de seguridad, aunque cada una de ellas puede presentar las características de seguridad en un formato diferente, en función de los componentes seleccionados para desarrollar la aplicación. Las diversas herramientas y servicios proporcionados por el Servidor de AR 300 y el Servidor de CCRDS 400, y la forma en la que estos interaccionan con las diversas aplicaciones de recursos, se describen posteriormente.

El Servidor de AR 300 proporciona un modelo de arquitectura/infraestructura común para construir aplicaciones web sobre empresas. El Servidor de CCRDS 400 proporciona una etapa de trastienda reutilizable para la búsqueda, el suministro de documentos, y la Tabla de Contenido. El Servidor de AR 300 proporciona la misma capacidad de reutilización para las aplicaciones web.

Servidor de CCRDS

El Servidor de CCRDS 400 es un sistema de gestión y de repositorio de contenido común que facilita la introducción de documentos nuevos y la recuperación de documentos existentes. El Servidor de CCRDS 400 incluye las siguientes utilidades para introducir, enriquecer, encontrar y recuperar documentos: Motor de Búsqueda, Tabla de Contenidos (TOC), Doc, Utilidad, CCI, Gestión de Cargas, Gestión de Datos, y Registro.

Motor de búsqueda

El Motor de Búsqueda proporciona una serie de herramientas para localizar documentos de diferentes maneras. Por ejemplo, se puede proporcionar una operación de Buscar, una de Encontrar y una de Explorar.

La operación de Buscar permite que un usuario reciba, como respuesta a un Mensaje de Usuario con una consulta apropiada, una o múltiples “coincidencias” que den satisfacción a la consulta a partir del repositorio de contenido común. En general, la interfaz de usuario solicitará al usuario que especifique términos de consulta y que seleccione colecciones de contenido y/o tipos de contenido deseados como parte de la operación de Buscar. Los usuarios de la operación Buscar pueden tener capacidades e interpretaciones diferentes de la funcionalidad de búsqueda en línea. Algunos usuarios tendrán experiencias previas con productos de motores de búsqueda privativos, y otros tendrán experiencia con motores de búsqueda de Internet. Las interfaces de usuario se pueden diseñar para presentar un aspecto familiar para aquellos con la experiencia del tipo mencionado.

Uno de los usos de Buscar incluye una búsqueda de información por Términos de Consulta con Operadores booleanos. El usuario introducirá un(os) término(s) de consulta en una casilla de consulta y sus expectativas serán que la lista de Resultados comprenda documentos que contienen este término(s). En este caso, el usuario desea construir una consulta con términos y operadores booleanos. Las expectativas del usuario será que se soporten todos los operadores booleanos, es decir, que sean reconocidos por el motor de búsqueda, y que se

recuperen solamente documentos que cumplan las condiciones de la cadena de consulta.

Las búsquedas con lenguaje booleano se pueden ampliar mediante el uso de “información por campos”. Esta técnica permite que el usuario busque metadatos y atributos de contenido específicos, de los datos, para filtrar adicionalmente la búsqueda. Los campos típicos incluyen elementos tales como fechas de documentos de varios tipos, autores, títulos, publicación, clasificaciones temáticas, etcétera.

La búsqueda por tema (donde los temas asociados a un documento particular se han asignado de antemano mediante un proceso de editorial) se logra usando extensiones de búsqueda del tipo Por Campos en la búsqueda booleana, aunque puede que esto no se revele para el usuario de la misma manera que una búsqueda más convencional. Los campos también se pueden ampliar de manera que presenten una mayor especificidad para las colecciones de contenidos, por ejemplo: partes involucradas en un caso, un juez, un número de expediente, etcétera.

Otra forma de búsqueda es el uso de Buscar Información usando lenguaje natural. En este caso, el usuario desea introducir los términos de consulta en una sintaxis de lenguaje natural. Por ejemplo: “obtener casos sobre fraudes a seguros”. Se espera que los resultados de búsqueda devueltos a partir de búsquedas con Lenguaje Natural tengan relevancia en relación con los términos de la búsqueda, omitiendo la sintaxis de construcción de las frases. Por ejemplo, en la búsqueda anterior, los resultados devueltos deberían incluir la expresión “fraudes a seguros”.

Puede ofrecerse una función de Búsqueda compuesta, conocida como Alerta, en donde el usuario desea que se le ponga al día cuando algo ha cambiado o hay información nueva que es relevante para su campo(s) de actividad. Los usuarios configuran una cartera de Alertas de Búsqueda que se ejecutan automáticamente de forma periódica. Cada Alerta se fija para ejecutar una búsqueda particular con respecto a una colección de contenido particular con cierto intervalo definido. Cada Alerta permitirá la definición de ciertos atributos. Para búsquedas, estos atributos podrían incluir, por ejemplo, términos de consulta, colección de contenidos, áreas temáticas, etcétera. Los usuarios deberían poder definir múltiples Alertas así como detener, iniciar, eliminar o cambiar la frecuencia para cada Alerta de su cartera. Los documentos encontrados por el servicio de Alertas se pueden distribuir según la manera de un servicio o convencional de recortes.

La operación Encontrar permite que un usuario recupere un documento individual de la colección de contenidos. En general, no debería ser necesario que el usuario especificase campos, categorías o áreas particulares, tales como, área jurisdiccional o campo de actividad, como parte de la operación Encontrar. Los usuarios de la operación Encontrar tienen un conocimiento previo de que existe un documento y desean acceder a ese documento específico. Dichos usuarios dispondrán de información identificadora específica de ese documento, tal como, por ejemplo, una cita, un título, las partes involucradas, o un nombre común para el documento.

Ciertas referencias pueden no ser suficientes para describir un documento particular. Se producen ejemplos de este tipo de problema con múltiples textos del mismo documento, el mismo documento en idiomas diferentes, o fuentes distintas correspondientes a una abreviatura de una cita particular. Por ejemplo, la abreviatura “ALR” es insuficiente para distinguir entre las publicaciones de Informes Jurídicos Americanos o Australianos (*American or Australian Legal Report*). En tales casos, la operación Encontrar recuperará todas las versiones de un documento particular que encajen con la referencia y permitirá que el usuario seleccione el documento particular que le interese.

La operación Encontrar es diferente de Buscar o Explorar. Encontrar permite que el usuario acceda a un único documento específico. Buscar permite que usuarios exploren una colección en búsqueda de documentos que coincidan con un conjunto de criterios definido por ellos. Los usuarios de Explorar escudriñan la taxonomía en busca de documentos que puedan satisfacer sus necesidades.

Se incluye también con la orden Encontrar una operación Encontrar por Atributos que permite que el usuario recupere un documento especificando uno o más atributos del documento o sus metadatos. Los ejemplos de Encontrar por Atributo incluyen: Encontrar por Título, Encontrar por Partes (buscar partes involucradas), y Encontrar por Nombre Común.

En función de la aplicación, habrá ocasiones en las que resulte apropiado establecer pre-filtros antes de ejecutar una operación Encontrar. Dichos filtros permitirían que el usuario limitase los resultados, por ejemplo, por Código de país, Idioma, Dominios de aplicación, Conjuntos de contenido definidos por la aplicación, Tipos de contenido (jurídico, regulador, impositivo, noticias), Campos de actividad, Jurisdicciones, Particiones de clasificaciones, etcétera. Los usuarios deberían poder invalidar dichos atributos de filtración por defecto en caso de que deseasen encontrar documentos dentro de bases de contenido más amplias.

Las operaciones de Encontrar se pueden canalizar (*pipelined*) con otras operaciones para crear nuevas operaciones o productos únicos. Por ejemplo, la salida de la operación Encontrar se puede enviar directamente a un servidor de impresión o de correo electrónico *push* para crear un suministro de documento simple. Para una

operación Encontrar dirigida contra múltiples colecciones, puede usarse un perfil de usuarios que contenga el conjunto de pre-filtros por defecto y automáticos, con el fin de limitar el número de resultados no únicos.

Los requisitos de los datos para implementar la operación Encontrar se deberían determinar para la aplicación del contenido durante el tiempo de diseño. La aplicación puede proporcionar nombres normalizados y canónicos, referencias y otra información para cada documento, suficiente para proporcionar la funcionalidad Encontrar.

En el nivel de búsqueda, Encontrar es similar a la operación Buscar. En general, Encontrar es un problema de la implementación y/o de la interfaz de usuario. Desde el punto de vista del usuario, Encontrar tiene que ver con la extracción de documentos conocidos a partir del corpus de contenido, mientras que la operación Buscar explora el corpus de contenido en busca de uno o más documentos potenciales que respondan a las condiciones de consulta y presenta, por lo tanto, un modelo de tarea diferente para el usuario final.

Funciones de las Tablas de Contenido (TOC)

La función de TOC según la proporciona el Servidor de CCRDS 400 es la versión electrónica de una tabla de contenido de un libro físico, aunque mejorada con tecnología apropiada para permitir la expansión/contracción de los niveles de encabezados y los enlaces a documentos. Las TOCs están compuestas por nodos raíz en la parte superior de la jerarquía, ramas intermedias opcionales y nodos hoja en los extremos terminales. Los nodos hoja están enlazados de manera individual con documentos o secciones de dentro de un documento.

Una operación de Explorar Tabla de Contenidos (TOC) permite que el usuario lea detenidamente una vista jerárquica del contenido de una colección. Puesto que una colección puede estar compuesta por uno o más documentos, las TOCs correspondientes pueden representar una TOC para múltiples documentos, un único documento, o subsecciones de un documento particular. A la inversa, una colección individual de documentos puede tener múltiples TOCs. Una TOC puede estar adaptada para un tipo de usuario particular así como para las colecciones de DOC particulares a las que hace referencia.

Mientras se explora una TOC, el usuario puede tener un conocimiento previo de un documento particular que está intentando encontrar; puede estar buscando orientación sobre un ámbito legislativo y/o un campo de actividad con los que no está familiarizado; o puede estar usando la TOC para ayudar a enfocar una cuestión o problema. Cuando la colección de contenido direccionada por una TOC es un documento, entonces la TOC asociada puede reflejar la estructura del documento. El usuario requerirá este tipo de TOC para navegar por grandes documentos, por ejemplo, Legislaciones.

De manera similar, cuando la colección de contenido contiene múltiples documentos, por ejemplo, Artículos Científicos, Estatutos, o formularios, puede crearse una TOC que muestre la presencia de cada documento. Esta es una característica importante para el usuario que requiere una lista de todos los documentos con el fin de poder explorar y seleccionar aquel que se ha apropiado.

Las funciones de exploración de TOC incluyen el acceso por navegación de material enlazado. Para la navegación, las estructuras de TOC pueden ser estrechas, amplias, profundas, o someras, en función de la naturaleza y el tamaño de la colección. La TOC puede tener niveles de jerarquía que se expanden (mostrando niveles inferiores) o se contraen (mostrando niveles superiores) para ayudar a navegar sobre la pantalla.

Los usuarios descienden por la TOC siguiendo enlaces desde nodos de nivel superior a nodos intermedios y terminales, incrementándose la especificidad de cada nivel. Dichos enlaces se pueden mostrar o bien explícitamente de forma esquemática, o bien como carpetas que se pueden abrir o cerrar, o usando otros métodos jerárquicos de interfaz de usuario. A medida que los usuarios descienden por la TOC, se crea un historial de seguimiento de tipo “migas de pan” que proporciona enlaces de vuelta a cada nivel visitado. Los usuarios navegan por la TOC realizando selecciones desde los nodos de nivel superior y moviéndose de vuelta hacia abajo según otro camino o buscando en la TOC.

La TOC debería ser accesible cuando se visiona cualquier documento dentro de esa colección. La posición relativa de ese documento en relación con otros documentos en la colección será mostrada por la TOC. El usuario puede navegar por la TOC al mismo tiempo que visiona cualquiera de los documentos de la colección de contenido; es decir, el documento siga abierto mientras el usuario navega por la TOC buscando contenido adicional.

Existen varias formas según las cuales se pueden construir TOCs para una colección de contenido particular, por ejemplo, en términos editoriales, programáticamente, mediante filtración para crear una TOC virtual, por composición para crear una TOC virtual. Evidentemente, pueden crearse TOCs manualmente usando un planteamiento convencional. Las TOCs se pueden crear programáticamente aprovechando marcas contenidas en el contenido. En tales casos, la TOC se crea dinámicamente y se puede organizar según una variedad de maneras. Una vez que se ha creado una TOC, la misma proporciona un cuerpo flexible de metadatos que se puede usar de maneras diferentes por parte de aplicaciones de recursos diferentes. Las TOCs se pueden filtrar

dinámicamente por medio de una aplicación de recursos para producir una o más vistas de subconjunto de la TOC completa. Dichas vistas se pueden usar para producir encabezados y pies de página que muestran una subsección particular del documento dentro del contexto mayor de la TOC de un documento o una colección. Se pueden crear vistas filtradas extrayendo propiedades de una TOC que limitan la vista a subconjuntos temáticos, jurisdiccionales, administrativos o temporales. Se pueden extraer dinámicamente múltiples vistas de subconjunto extraídas de una o más TOCs, y las mismas se pueden combinar para producir una TOC virtual correspondiente a un documento virtual que no existe como documento individual en un espacio físico.

Los filtros de extracción de subconjuntos para producir vistas de TOC virtuales se pueden aplicar con respecto a la TOC completa en todos los niveles o con respecto a múltiples TOCs de conjunto de contenido. Tal como anteriormente, los resultados de estas extracciones de subconjuntos recortan las partes seleccionadas deseadas. A continuación, las secciones recortadas se pueden secuenciar para producir una nueva TOC virtual compuesta. La TOC virtual comunica al usuario el aspecto de un único documento virtual que apunta a múltiples referencias en las mismas colecciones de contenido o en colecciones diferentes.

También puede proporcionarse un índice. El índice establece correspondencias de etiquetas y códigos de XML específicos con el texto dentro de un documento, y también establece correspondencias del texto total dentro de un documento, una colección o un conjunto con una herramienta con capacidad de búsqueda completa.

Gestión de cargas

La presente arquitectura facilita el escalado del hardware y de otros recursos que son sensibles habitualmente a la carga. Con recursos duplicados es necesario equilibrar las cargas, de manera que las tareas no se sitúen de manera excesiva en la cola en ciertos recursos, cuando hay otros intercambiables para dichas tareas. Por consiguiente, la presente invención adopta una gestión de cargas de tipo licitación. Esto requiere que los recursos inactivos o con baja carga informen de su disponibilidad para el procesado adicional de tareas que se encuentran en la cola. El modelo de licitación se puede implementar en parte mediante el uso de LDAP por parte del componente de Monitorización.

Registro

El Registro realiza un seguimiento de eventos solicitados por los servicios/herramientas compartidos, y permite un diagnóstico basándose en lo que se introdujo realmente. Es decir, se realiza un seguimiento tanto de la carga de documentos como de la búsqueda del usuario en la etapa frontal, para proporcionar una comprobación histórica de errores y una monitorización en tiempo real.

Gestión de datos

El componente de Gestión de Datos proporciona un mantenimiento y una optimización del sistema básico.

CCI (Información de Control Central)

El componente de CCI gestiona los lugares en los que se almacenan todos los metadatos y los monitoriza en cuanto a forma en cada Colección de Datos. Durante la admisión, al CCI se le asignan Conjuntos de Carga de una colección. Los conjuntos de carga son tablas que contienen reglas para definir cómo las herramientas/servicios compartidos van a procesar los datos XML. Hay Conjuntos de Carga que contienen reglas de indexación detalladas, reglas de procesado para DOC, TOC, y MM, y reglas en relación con qué elementos son procesados por qué constructores. Un Conjunto de Carga puede ser compartido por más de una Colección de Datos.

DOC

DOC es el servicio que toma solicitudes, devuelve documentos, y modifica, pone marcas en y configura documentos para su suministro al componente de Renderización. Esto incluye mecanismos proporcionados por el motor de recuperación de DOC para el filtrado de documentos. DOC proporciona también y opciones de filtrado diseñadas para identificar y recuperar partes bien formadas de documentos XML completos.

Utilidad

El Servicio de Utilidad es un servicio general diseñado para reunir servicios misceláneos que no justifican ser un servicio en sí mismo (lo cual significa tener su propia cola MQ, etcétera). Los siguientes servicios se alojarán dentro del Servicio de Utilidad:

1. Localizador de Documentos

Este servicio se usa para localizar qué colección contiene un documento, dado un GUID. Se usa en general

cuando se validan y/o se siguen hiperenlaces (que contendrán solamente el GUID objetivo).

2. Navegación de Resultados

Este servicio proporciona funciones para navegación básica dentro de un objeto resultado de búsqueda. El Servicio de Búsqueda crea objetos resultado de búsqueda. El Servicio de DOC se usa para recuperar el texto de documentos. Este Servicio de Navegación de Resultados vinculará entre sí los dos mencionados permitiendo que un cliente solicite la información de documentos (GUID) para rangos particulares. Esta información se extraerá a partir del objeto resultado de búsqueda y la misma será devuelta. A continuación, el cliente dispondrá de la información necesaria con la cual invocar al Servicio de DOC para recuperar el texto del documento.

3. Obtener PIT

Los clientes usarán un valor de PIT (instante de tiempo) como mecanismo para “congelar” su vista del mundo. Siempre que se use el mismo PIT para llamadas sucesivas del servicio de repositorio del contenido común, la vista permanecerá constante (no verán ningún dato nuevo que haya sido cargado). Cuando un cliente solicita un PIT nuevo, el mismo reinicializará su vista al instante de tiempo que era actual en el momento de la solicitud.

4. Destructores de objetos persistentes

Tal como se define en la especificación del Servicio de Persistencia, la destrucción de objetos persistentes será responsabilidad del cliente. El Servicio de Destructores de Objetos Persistentes proporcionará las APIs por medio de las cuales los clientes pueden provocar que se produzca esta destrucción. Se creará una API independiente y única para destruir cada tipo de objeto persistente.

Persistencia

Aunque el servicio de Persistencia aparece en la figura 4 como parte del Servidor de AR 300, el mismo está íntimamente asociado al repositorio del contenido común y al Servidor de CCRDS 400. La función del componente de Persistencia es almacenar resultados de búsquedas para un acceso posterior sin que sea necesaria la nueva ejecución de la búsqueda. Por ejemplo, una búsqueda dada podría conducir a la recuperación de identificadores para cien documentos relevantes. Podrían visualizarse los documentos del uno al diez, mientras que los documentos del once al cien se guardan. Así, si el usuario selecciona el documento cincuenta, este último puede determinarse posteriormente a partir del componente de Persistencia accediendo a identificadores almacenados, sin tener que volver a ejecutar la búsqueda. Con múltiples usuarios accediendo al repositorio de contenido común con el componente de búsqueda, el componente de Persistencia alivia la carga sobre el componente de búsqueda.

Servidor web, servidor de AR

Los componentes del Servidor Web 200 y el Servidor de AR 300 proporcionan un marco de aplicación usado para crear y desplegar aplicaciones basadas en la web, que se basan en datos residentes en el repositorio de documentos del Servidor de CCRDS 400. Como parte de este marco, el Servidor de AR 300 tiene un objetivo de alto nivel para promover información compartida en un cliente pasando por unidades comerciales participantes. Un repositorio común almacena información de usuario para las unidades comerciales participantes. Estos componentes soportan también un único inicio de sesión para múltiples aplicaciones por parte de un usuario.

El Servidor de AR 300 es una plataforma hospedante individual que permite acceso desde una pluralidad de diferentes componentes de interfaz de usuario asociados a diferentes aplicaciones de recursos. Esta plataforma tiene un conjunto de componentes que proporcionan cierta funcionalidad común, así como servicios de Renderización, Localización y Alerta. La plataforma incluye también un diseño normalizado común para implementar persistencia con el fin de soportar conmutación a sistemas redundantes (*failover*) y componentes que soportan una alta disponibilidad. Además, se ponen a disposición componentes de datos persistentes genéricos, reutilizables. Se proporciona un modelo de seguridad para la autenticación y el control de acceso con el fin de garantizar una única vista de un cliente. La plataforma incluye además procedimientos comunes para monitorización, gestión y despliegue.

Los componentes del Servidor de AR 300 proporcionan un kit de herramientas para que los desarrolladores de aplicaciones de recursos personalicen aplicaciones para interfaces de usuario específicas que extraen partes comunes del repositorio de datos del Servidor de CCRDS 400. A continuación sigue una descripción de los componentes significativos del Servidor de AR 300.

Detección de duplicados

El Servicio de Detección de Duplicados actúa como filtro para evitar que se muestre nuevamente el mismo documento, a no ser que este haya sido modificado desde que se vio por última vez. El problema del documento duplicado surge cuando, por ejemplo, un usuario presenta una consulta a una colección de DOC y recibe una lista de documentos que contienen duplicados, con diferencias únicamente marginales en cuanto a título, fuente o versión. Esto puede suceder, por ejemplo, con noticias, de las cuales se puede haber informado de manera similar en varios periódicos que proporcionan artículos al repositorio del contenido común. Se ha observado que hasta un 30% de los documentos que se devuelven tras una búsqueda de noticias pueden ser miembros de un conjunto de documentos duplicados. Del conjunto de documentos duplicados, más de la mitad de todos los documentos que se pueden considerar duplicados se sitúan en la categoría de duplicados exactos. No obstante, también puede resultar interesante incluir nociones de duplicación más vagas para detectar documentos muy similares pero no idénticos.

La detección de duplicación de documentos puede interpretarse, con un nivel suficiente de abstracción, como la comparación de dos cadenas de texto; no obstante, esta vez, en lugar de una cita y un documento fuente candidato, se dispone de un documento y un documento fuente candidato. Se ha determinado que los n (donde n es un entero relativamente pequeño) primeros términos (incluyendo tokens y sus posiciones mutuamente relativas) de idf de un documento son suficientes para aportar una “huella digital” del documento con fines comparativos. Aquí, idf significa “frecuencia inversa de documento” que, para un término dado, es el inverso de la “frecuencia de documento” para ese término, es decir, 1 dividido por el número de documentos en la colección bajo consideración que contienen el término.

Esta huella digital se debe preparar como un campo de metadatos para cada documento que se va a usar en un sistema de detección de duplicados. Aporta una tarea computacional que debe llevarse a cabo en el momento de la admisión del documento, aunque también puede realizarse más tarde, para una colección de documentos ya cargada en el repositorio de contenido común. Para ayudar a repartir la carga computacional de ejecutar concretamente las comparaciones de huellas digitales cuando se producen resultados de búsqueda que pueden contener duplicados, la tarea de comparación se divide entre el lado de cliente (desde el cual se origina la solicitud de búsqueda) y el lado de servidor. Así, la detección de duplicados implica esencialmente tres etapas:

A. Generación de metadatos – durante un proceso de carga por lotes

Durante una sesión de admisión de documentos, para cada documento, se almacenará una firma de documento completa en forma de metadatos (magnitud escalar de longitud + vector de huella digital).

La “magnitud escalar de longitud” (en tokens, excluyendo la fuente, el título, el autor y otra información de encabezados) del documento se almacenará como parte de la firma.

El “vector de huella digital” consistirá en los n (donde n es de cuatro a treinta, preferentemente de cuatro a seis, y con la mayor preferencia, seis) primeros términos de idf únicos para el documento (excluyendo información de encabezados), junto con sus posiciones mutuamente relativas, por ejemplo, {dolo[76], rehén[0], conspicuo[25], intransigencia[121], brutalidad[163], teatro[13]} (clasificados por valores de idf).

Obsérvese que los términos bajo consideración excluirían el título del documento y otros encabezados (ya que los mismos pueden variar claramente en documentos debido a diferentes títulos, editores, ediciones, etcétera).

Obsérvese también que los términos con una idf inhabitualmente alta, es decir, idf > 0,8, no se considerarían como los seis primeros candidatos ya que estos tienden a ser anomalías (es decir, erratas y faltas de ortografía).

A continuación, se aplicará una función *hash* en el vector de huella digital para obtener una clave de longitud manejable, por ejemplo [!x9v^4#w+z2%7t$d] (16 bytes). Los términos de idf más alta del documento se permite que aparezcan solamente una vez en los n primeros términos de idf del vector, incluso si aparecen en el documento más de una vez.

B. Operación de comparación de documentos – en el lado del servidor, dada una lista de resultados de búsqueda

En el servidor, comenzando con el documento de clasificación más alta en los resultados de búsqueda y el siguiente documento que falta por comparar con el mismo,

se compararán longitudes de los documentos: si el documento de comparación está dentro de ±M caracteres (por ejemplo, donde M es de 0 a 256, preferentemente, 40 caracteres) del documento de base, se continúa; en cualquier otro caso, se finaliza la comparación (±M sirve para compensar potenciales diferencias en el texto cerca del material de encabezado);

a continuación se compararán las huellas digitales de los documentos: si el documento de comparación tiene una

huella digital idéntica a la del documento de base, entonces el documento duplicado se señaliza como duplicado; si no, se finaliza la comparación;

los documentos señalizados por un estado de duplicado, se moverán efectivamente a la carpeta de Duplicados en el Cliente.

A continuación, el siguiente documento de clasificación más alta y que no se ha señalizado ya por duplicación, se comparará con la totalidad del resto de documentos de rango inferior en la lista de resultados y que no se han señalizado previamente.

El proceso continuará hasta que se haya comparado el último par de documentos sin señalizar.

C. Renderización de documentos – en el lado del cliente

(1) Los documentos sin duplicados aparecen en la lista convencional de resultados de búsqueda;

(2) Los documentos clasificados más arriba que tienen duplicados aparecen en la lista de resultados de búsqueda convencional, pero se marcan para indicar que sus documentos duplicados correspondientes aparecen en la carpeta “Duplicados” (por ejemplo, que aparece en la esquina inferior izquierda de la pantalla);

(3) Los restantes documentos duplicados aparecen en la carpeta “Duplicados”.

La implementación de este sistema de detección de duplicados implica algunas consideraciones adicionales:

los documentos no señalizados, con la clasificación más alta, se mantendrán en la lista de resultados convencional.

Las idfs sin duda cambiarán con el tiempo; una huella digital generada hoy podría no corresponderse con una huella digital que se genera el año siguiente si la colección en base a la cual se obtienen las idfs cambia. Para evitar la necesidad de reproducir periódicamente una huella digital de un documento, es importante entonces realizar un mantenimiento de la colección estable, de gran tamaño, en la cual se basan las puntuaciones de las idfs. Alternativamente, una vez que se ha determinado la colección estable, de gran tamaño, los términos y sus idfs correspondientes simplemente se podrían almacenar rentablemente en una tabla de consulta.

En un documento tal como un artículo periodístico, es posible que la totalidad de los n primeros términos del valor IDF provengan del mismo párrafo. Así, esto parecería representar una cobertura interdocumental deficiente. No obstante, un artículo periodístico no es largo, con una longitud de aproximadamente una página por término medio. Aun cuando existe una remota posibilidad de que la totalidad de los n términos de idf más alta se produzca en una localización relativamente pequeña, esto no significa que su cobertura del documento se vea disminuida en modo alguno. La cobertura de las huellas digitales permanece intacta puesto que la ausencia de los términos de idf más alta en otras secciones del documento es también útil para la detección.

Se podría añadir un nivel de “vaguedad” al proceso de detección de duplicados mediante la elección de no aplicar la función *hash* al vector de huella dactilar y, en su lugar, permitir una relación de ±1, ±2, o ±N entre términos en los vectores que se estén comparando. Así, para ajustar un sistema a un nivel deseado de detección de duplicados, las diferencias de la huella digital y/o del parámetro escalar de longitud entre dos documentos se pueden medir con respecto a un umbral de similitud predeterminado, aunque ajustable, y multi-factorial.

La aplicación de la función *hash* puede añadir un nivel extra de rigurosidad a las firmas de los documentos a comparar, puesto que unas variaciones modestas en los valores de la idf pueden hacer que cambie la ordenación de los n primeros términos de idf, pero no los propios términos. Por lo tanto, el valor *hash* del términoA[0], términoB[25]... diferirá con respecto a términoB[25], términoA[0]... Así, a no ser que los cálculos de idf se estabilicen usando una colección maestra normalizada, podría fallar un número mayor de comparaciones debido al fenómeno anterior.

Componente de renderización de documentos

El componente de Renderización de Documentos mapea documentos con una hoja de estilos específica de la aplicación. En cada documento se incorpora una etiqueta de referencia de hoja de estilos de acuerdo con las normas del Servidor de AR. El componente de Renderización requiere entradas externas. Estas entradas incluyen las hojas de estilo personalizadas del desarrollador de la aplicación y un mapeo de hojas de estilos con GUIDs de hoja de estilos, asociados. Las entradas son recuperadas por los componentes de renderización usando el sistema de archivos y el sistema de repositorio de contenido común.

El servicio de renderización de documentos mapea y almacena en memoria caché hojas de estilos XSL. La

aplicación de recursos usa el toHTML() para diseñar el XML. La figura 19 muestra esquemáticamente cómo se desarrolla la renderización de documentos con una hoja de estilos de presentación mínima. La figura 20 muestra esquemáticamente cómo se desarrolla la renderización de documentos con una hoja de estilos personalizada y con múltiples mapas de hojas de estilos.

Almacenamiento de información específica de cada aplicación en línea en favoritos

Este servicio permite que un usuario seleccione, almacene y acceda a documentos, colecciones, cadenas de búsqueda, o conjuntos usados de forma rutinaria. Este componente permite también que un usuario añada comentarios a un documento dado, que, a continuación, se almacenan con ese documento para ese usuario. Otros ejemplos de información que se pueden almacenar incluyen, Búsquedas Guardadas, Enlaces Directos Guardados a Documentos, Enlaces Directos Guardados a Definiciones de Alertas. La información se almacena como una jerarquía dinámica que puede ser manipulada por el usuario. La característica es similar a la característica de Favoritos de navegadores web convencionales.

Conversión de imágenes

El componente de Conversión de Imágenes de los servicios/herramientas compartidos convierte imágenes TIFF o JPEG, o lleva a cabo conversiones de otros formatos de imagen. El componente soporta también una característica para redimensionar imágenes y soporta manipulaciones de imágenes, incluyendo escalado, rotación, recortado y filtrado. Pueden usarse componentes convencionales de conversión de imágenes.

Localización

Este componente permite que los usuarios modifiquen individualmente su interfaz de usuario local y que obtengan una personalización. Por ejemplo, una interfaz de usuario se puede traducir a español o se puede desarrollar en español para un mercado particular. Además, en la medida en la que se permitan consultas en lenguaje natural, la localización puede requerir que la totalidad o partes de un motor de búsqueda en idioma inglés se sustituyan con componentes de búsqueda específicos de un lenguaje local.

Puede especificarse un lugar para un usuario, por idioma, país, y variantes de los mismos. Pueden localizarse tanto texto como imágenes. Se configura un archivo de propiedades y un directorio por cada lugar.

API y servicio de alerta

El servicio de Alerta permite que clientes seleccionen consultas de búsqueda, las cuales se ejecutan a intervalos especificados. Se distribuyen resultados de búsqueda al usuario final cada vez que la consulta de búsqueda devuelve un resultado nuevo. El servicio de Alerta usa el mecanismo de suministro de documentos de servicios/herramientas compartidos para correo electrónico y facsímiles. Los siguientes componentes forman parte del servicio de Alerta: una base de datos para contener las entradas de Alerta; una API para manipular entradas de Alerta de la interfaz de usuario; un servicio para ejecutar las entradas de Alerta y distribuir los resultados al cliente. Estos componentes permiten que cada aplicación de recursos use los mismos servicios de Alerta para su aplicación específica.

Tal como puede verse mejor en la figura 16, la Alerta para cualquier usuario se configura con entradas en un directorio usando la API de Alerta 1602. Las entradas de Alerta se pueden crear, editar, eliminar o ejecutar. La frecuencia para la ejecución de los datos de selección de documentos definidos en una entrada de Alerta se puede fijar para cada día, en días laborables, semanalmente, bisemanalmente, mensualmente o se puede guardar (conservarla pero sin ejecutarla nunca). Un Servicio de Alerta 1604 interacciona con el repositorio de contenido común 1606 y con una base de datos de Alerta 1608, a continuación distribuye los documentos recortados por medio del servicio de suministro de documentos 1610. El usuario puede realizar recortes por múltiples colecciones de DOC.

API y servicio de suministro de documentos

El Servicio de Suministro permite que el usuario cree copias electrónicas locales o físicas de documentos en línea. Puede accederse a la función de suministro en cualquier punto del proceso de indagación. Para distribuir un documento, los usuarios generalmente especifican la siguiente información: qué distribuir, inclusiones y omisiones, destinos del suministro, y formato.

Para determinar qué distribuir, se supone que la aplicación de recursos presenta por lo menos una forma de acceder a la función de Suministro para un documento o artefacto específico. Ya se realice mediante indicación directa (por ejemplo, un botón en la página) o indirectamente (por ejemplo, enlace de impresión), la funcionalidad es idéntica. En general, se supone que se distribuirá un documento o artefacto completo. Para ciertos documentos grandes, debería permitirse al usuario distribuir solamente partes específicas del documento. Pueden usarse mecanismos tales como una Tabla de Contenidos para seleccionar las partes de un documento a

distribuir.

Cuando se determina lo que se incluye u omite, el modo por defecto para la operación de suministro es incluir el texto completo, y el conjunto total de imágenes y tablas asociadas a un documento particular. Del trabajo de suministro pueden incluirse u omitirse elementos adicionales, en función del tipo de documento u otras propiedades. Los usuarios deberían poder deseleccionar cada elemento según resulte apropiado para el tipo de documento.

El destino se determinará basándose en la preferencia del usuario y la disponibilidad del dispositivo de destino. Por ejemplo, algunos destinos pueden incluir una impresora conectada (la impresora conectada al ordenador o red de área local del usuario), una dirección de correo electrónico (en lugar de imprimir el trabajo, se envía una copia formateada del archivo a la dirección de correo electrónico del usuario), una máquina de fax (se envía una copia formateada del archivo a la dirección de fax del usuario), o una descarga (se guarda una copia formateada del archivo en el destino especificado por el usuario en su unidad de disco duro del ordenador). Los usuarios no deberían tener que especificar una dirección de destino hasta que seleccionan ese destino como valor por defecto.

Las preferencias del usuario para el suministro (y una variedad de otras opciones) se especifican en un archivo que contienen valores por defecto para aplicaciones de recursos y los servicios/herramientas compartidos.

Los formatos soportados para el suministro de documentos incluyen HTML, RTF, PDF, PostScript y archivos de texto. Las figuras 18A y 18B muestran las relaciones entre diversos componentes involucrados en el suministro de documentos. Los documentos a distribuir se conservan en unos medios de almacenamiento temporal desde los cuales se accede a los primeros por medio del servicio de suministro, que los suministra al componente de renderización. El Servicio de Suministro tiene un componente de renderización y acepta documentos XML y XSLt. Los distribuye al exterior en formatos XSL FO, HTML y de Texto. Un procesador de XSL FO produce documentos HTML y PDF/Postscript que se pueden enviar por correo SMTP. El procesador de RTF produce documentos RTF para la Web. También pueden distribuirse documentos mixtos de Texto/HTML.

API y servicio de historial de seguimiento

El Servicio de Historial de Seguimiento mantiene un historial de las transacciones de eventos de aplicación reproducibles. Esto permite que un usuario encuentre rápidamente un documento sin ser necesario que el usuario reformule los eventos de indagación para recrear los resultados de documentos. Es decir, el sistema conserva información sobre la indagación según se ha llevado a cabo mediante consulta al repositorio de contenido común y los resultados generados, pero no el texto del(de los) propio(s) documento(s) resultante(s). Por ejemplo, cada documento se identifica con un GUID (Identificador Global Universal), y la función de Historial de Seguimiento se puede almacenar como referencia un “lo-mejor-de-los-GUIDs-buscados”. Cada vez que un usuario inicia una nueva sesión en una de las aplicaciones de recursos, se crea un historial de seguimiento de las solicitudes y recuperaciones del usuario. Cuando un usuario necesita volver a una solicitud previa, el componente de Historial de Seguimiento proporciona un acceso rápido al documento usando el historial de seguimiento establecido durante el proceso de búsqueda. El componente de Historial de Seguimiento también puede permitir que un usuario aproveche la indagación de una sesión previa permitiendo que el mismo guarde historiales de seguimiento o que acceda a historiales de seguimiento previos.

Las funciones del Historial de Seguimiento permiten que el usuario acceda a una indagación previa de una manera sencilla y rápida, reuniendo la secuencia de operaciones que fueron ejecutadas por el usuario en una estructura de datos contenida en la Base de Datos de Historiales de Seguimiento 1702 (véase la figura 17). Para especificar las acciones de creación, eliminación, modificación y recuperación se usa un Directorio de Historiales de Seguimiento. Las funciones del Historial de Seguimiento conceden acceso a los usuarios y les permite manipular la estructura de datos de historiales de seguimiento.

El registro de Historial de Seguimiento se crea durante la sesión de indagación y se mantiene en forma de una estructura de datos en la aplicación de recursos. La estructura de datos del Historial de Seguimiento es específica de una contraseña particular, y, cuando se registre mediante métodos de autenticación, del ID de cliente también. Los eventos registrados en un historial de seguimiento pueden corresponderse con eventos de indagación tarificables básicos, por ejemplo, principalmente extracciones de documentos, búsquedas, y solicitudes de citadores. Cada evento registrado en el historial de seguimiento se puede recrear; haciendo clic en el evento, el usuario puede volver a un documento, volver a ejecutar la búsqueda o citador, etcétera.

La aplicación de recursos expone una interfaz que permite que el usuario (por medio del consumidor de eventos de historial de seguimiento 1706) acceda al mecanismo de historial de seguimiento en cualquier momento durante una sesión de indagación. De manera similar, el mecanismo de historial de seguimiento permite que el usuario vuelva a entrar sin interrupciones en la sesión de indagación en el punto en el que el usuario la dejó.

El flujo de aplicación para un Historial de Seguimiento funciona de la siguiente manera. La aplicación crea un

historial de seguimiento nuevo, fijando información específica (por ejemplo, nombre del historial de seguimiento, producto, id de usuario, id de cliente, etcétera). El historial de seguimiento también contiene la fecha de creación, la fecha del último acceso, y la fecha de expiración. Pueden definirse parámetros adicionales y los mismos se pueden usar específicamente por un producto. Estas “propiedades” se almacenan en una cadena XML, la cual no es una base de datos en la que se puedan realizar búsquedas. Alternativamente, la aplicación obtiene un historial de seguimiento preexistente específico mediante una clave de historial de seguimiento única.

Para un “evento reproducible” (por ejemplo, resultado de búsqueda, documento, etcétera) de un producto, la aplicación creará un nuevo elemento de historial de seguimiento y lo añadirá al historial de seguimiento. El elemento de historial de seguimiento almacena información específica, tal como el tipo de elemento (por ejemplo, búsqueda, documento, etcétera), y la fecha de creación. Pueden almacenarse parámetros adicionales, definidos y usados específicamente por un producto. Estas “propiedades” se almacenan en una cadena XML y se pueden usar para recrear el evento para ese producto.

El sistema coloca solicitudes de historiales de seguimiento en una cola y un servicio de antecedentes las procesa. Para procesar las solicitudes se usa un formato FIFO (Primero en Entrar-Primero en Salir). El servicio está monitorizando constantemente la cola en busca de solicitudes nuevas. Si se produce un fallo de la base de datos, la cola realizará una copia de seguridad y postergará la información del historial de seguimiento para el usuario. Esto no ralentizará el rendimiento de la aplicación.

La aplicación de recursos visualiza la información más actualizada disponible de los historiales de seguimiento. La aplicación usa la API de Historial de Seguimiento 1708 para crear una lista de elementos de historial de seguimiento (por ejemplo, una página web con enlaces a los eventos reproducibles) para el cliente (obsérvese que la recreación de los eventos será controlada por la aplicación para garantizar que se crean o evitan eventos de facturación correctos en función de las reglas comerciales correspondientes a ese producto particular y, por lo tanto, el evento no se recreará en el historial de seguimiento).

La aplicación cerrará el historial de seguimiento cuando la sesión del usuario llegue a su fin, o cuando la aplicación cierre explícitamente el historial de seguimiento.

Otro de los usos de la información de los Historiales de Seguimiento es la mejora del sistema para adaptarse a las necesidades y las expectativas de los usuarios. Así, si el sistema tiene uno o más archivos de historial de seguimiento para contener información sobre procesos de búsqueda de usuarios en una aplicación de recursos particular e identificadores para documentos encontrados como respuesta a mensajes de consulta, esta información se puede proporcionar a un componente de análisis de historiales de seguimiento para procesar archivos de historial de seguimiento con el fin de determinar patrones de uso comunes para una aplicación de recursos particular. Este análisis puede conducir al ajuste de parámetros de la aplicación de recursos para presentar opciones de búsqueda y resultados de búsqueda de una manera más concordante con dichos patrones de uso.

Si el análisis produce patrones de uso comunes que implican la secuencia en la cual los usuarios visualizan documentos identificados en resultados de búsqueda con respecto al orden de prioridad en el cual se presentan los documentos como resultados de búsqueda, los parámetros ajustados pueden afectar al orden de prioridad en el cual se presentan los documentos como resultados de búsqueda. Si el análisis produce patrones de uso comunes que implican la revisión, de los usuarios, de documentos identificados en resultados de búsqueda como duplicados, los parámetros ajustados pueden afectar al umbral de similitud para un servicio de detección de duplicados. El análisis de Historiales de Seguimiento también puede conducir a patrones de uso comunes que se capturan en archivos de metadatos para materializar dichos patrones de uso y hacer que los mismos estén disponibles para aplicaciones de recursos relevantes. Por ejemplo, si el análisis de Historiales de Seguimiento de los usuarios más experimentados de una aplicación de recursos revela un patrón de uso de las TOC como mejor práctica y que muestra un uso reducido de ciertas ramas de TOC o un cierto patrón de exploraciones desde un nodo de TOC, esto puede conducir a la poda, expansión o reorganización de las TOC para crear una TOC que se pueda ofrecer de forma que materialice las mejores prácticas observadas y que pueda tener valor para usuarios menos experimentados.

<u>API de registro/facturación de eventos</u>

Los servicios/herramientas compartidos proporcionan una API que permite que los desarrolladores de aplicaciones de recursos produzcan registros de facturación de la aplicación web. Los registros de facturación son genéricos, permitiendo que los desarrolladores de aplicaciones de recursos capturen datos necesarios para sus necesidades de facturación. La API proporciona la información necesaria en XML, aunque el desarrollador de las aplicaciones es responsable de aportar la aplicación de recursos que convierte el XML genérico en el formato correcto para su sistema de facturación.

La API crea eventos con pares de nombre/valor específicos de la aplicación de recursos. La función de facturación/registro distribuye información de eventos facturables a un consumidor del sistema comercial. Para

los eventos facturables existen ciertas propiedades por defecto: valor del hilo, nombre de la máquina, sello de tiempo y EventGUID (ID globalmente único para el evento).

Servicios de seguridad; API de control de seguridad y de acceso

Una de las partes del control de seguridad y de acceso implica una autenticación, la cual incluye un Inicio de Sesión y un Fin de Sesión. La operación de inicio de sesión identifica al usuario e inicia la sesión de indagación de aplicaciones de recursos. El Inicio de Sesión requiere un identificador de usuario (id de usuario) y un autenticador de usuario (contraseña). Estos pueden ser generados por la aplicación de recursos o por el usuario, incluyendo alias fáciles de recordar (siempre que se proporcione una cadena identificadora única). La operación de Fin de Sesión cierra la sesión.

Las operaciones de Control de Acceso son necesarias cuando la aplicación de recursos necesita monitorizar el uso y/o restringir el acceso; el usuario requiere una personalización de la interfaz de forma individual o por grupos; debe monitorizarse el acceso a nivel de contenido y/o de funciones; debe compartirse el acceso del usuario sobre tipos de aplicación que hacen referencia al mismo contenido (por ejemplo, acceso web y por intranet a las mismas colecciones de contenido); o el acceso del usuario debe compartirse sobre múltiples aplicaciones de recursos.

Con la presente invención, múltiples aplicaciones de recursos se enlazan sobre un sistema común. Por ello, el proceso de autenticación proporciona acceso a un usuario para la totalidad de las diversas aplicaciones de recursos para cuyo uso tiene derecho el usuario. No obstante, puesto que estas diversas aplicaciones de recursos presentan sus propios requisitos de uso y parámetros de facturación, la conexión y el registro del ID/contraseña de usuario con respecto a las aplicaciones de facturación y seguimiento de uso son responsabilidad de cada aplicación de recursos. En otras palabras, se crea y se utiliza un perfil de usuario común. La figura 13 proporciona una vista general esquemática de cómo los componentes de seguridad compartidos en línea se comunican con aplicaciones de recursos 1302 e interpretan información de seguridad almacenada en una base de datos de seguridad 1304, para proporcionar validación de los usuarios y sus mensajes de consulta basándose en información de suscripción de los usuarios. La figura 13 muestra además cómo los Sistemas Comerciales 1306 que mantienen definiciones de clientes, productos, planos de precios, suscripciones, etcétera, envían, sin solicitud previa, definiciones de seguridad de aplicación a la base de datos de seguridad a través de servicios administrativos 1308. Los servicios administrativos revelan una API que permite el mantenimiento de la base de datos. La API se basa en mensajes de respuesta a solicitudes XML. Otras aplicaciones pueden usar información de seguridad específica de una aplicación de recursos.

La seguridad según se implementa en los servicios de aplicación compartidos proporciona una visión homogénea de un cliente. Mediante la creación de una Entidad de usuario en el Modelo de Seguridad de servicios de aplicación compartidos, los clientes pueden moverse fácilmente entre sitios participantes (o aplicaciones de recursos) sin necesidad de recordar un conjunto específico de credenciales para cada sitio. Esto significa que los usuarios necesitan solamente un ID y una contraseña de inicio de sesión para todos los sitios participantes, y que las credenciales de los usuarios se almacenan en un lugar seguro (Seguridad).

Así, los sitios comunes que comparten Seguridad para autenticar usuarios ahorran tiempo y dinero al no tener que construir, comprar, albergar y mantener su propio sistema de autenticación. Los desarrolladores pueden concentrarse en las características y funcionalidad de su propio sitio.

La seguridad soporta un perfil básico del usuario. La información tal como preferencia de idioma y nombre y apellidos se puede rellenar en el momento de creación de la cuenta. Al usuario dentro del modelo de Seguridad se le asigna un GUID (Identificador Global Universal) de Usuario cuando se crea la cuenta. Es esta identificación la que se usa para identificar este usuario.

El servicio de Seguridad lleva a cabo una variedad de tareas de seguridad. Las tareas incluyen: autenticar un Usuario de Seguridad Existente; permitir que la aplicación de recursos aplique una autorización; actualizar un Usuario de Seguridad Existente; añadir un Usuario de Seguridad Nuevo a la Base de Datos de Seguridad; asociar o desvincular un Usuario de Seguridad con respecto a un Grupo para conceder acceso a ciertas características. Otras características de seguridad incluyen, realizar un seguimiento en relación con límites de volumen o tarificación, regular límites sobre permisos, control de exportaciones.

La figura 14 muestra esquemáticamente el paradigma de seguridad usado. La seguridad incluye un usuario 1402, un grupo de usuarios 1404, y permisos 1406 para el usuario. Puede usarse una sola definición de usuario para todas las aplicaciones de recursos. Así, puede utilizarse un ID y una contraseña de usuario para permitir que este último acceda a los documentos desde cualquier aplicación de recursos. A cada usuario se le conceden permisos específicos que llegan a estar disponibles una vez que se han introducido el ID y la contraseña correctos. El ID y la contraseña de usuario se establecen y cambian usando la API de Seguridad. Cada permiso asocia una característica 1408 a un recurso 1410, tal como un conjunto de colecciones.

Un usuario puede pertenecer a un grupo de usuarios. Un grupo de usuarios representa una clase de usuarios. A todos los usuarios del grupo se les concede el mismo permiso. Una vez más, un grupo de usuarios puede definirse una vez, y, a continuación, puede ser usado por todas las aplicaciones de recursos.

La figura 15 muestra los componentes del modelo de seguridad para una forma de realización y sus relaciones. Para establecer este modelo, en primer lugar es necesario establecer un dominio 1502 que define un calificador de nombre para entidades específicas de la aplicación. Esto permite nombres duplicados sobre aplicaciones, por ejemplo, Fiji: búsqueda. A continuación, el Propietario 1504 define un ID y una contraseña de usuario para el administrador. Seguidamente se define un usuario 1506. Tal como se ha indicado, puede aplicarse una sola definición de usuario para todas las aplicaciones (los usuarios son independientes de los dominios). Se asignan un ID, una contraseña de usuario y otra información del perfil de usuario. El GUID de Usuario se encuentra en la base de la definición.

Si el Usuario Definido forma parte de un grupo, se usa la entidad UsuarioGrupo 1508 para añadir el usuario a un Grupo 1510, el cual representa una clase de usuarios. La definición de un grupo simplifica la administración dejando que se definan permisos una vez y que los mismos se asignen a muchos usuarios. Pueden definirse grupos en términos de jerarquías. Un grupo padre puede tener uno o más grupos hijos. Los grupos hijos heredan permisos de su grupo padre.

Una Característica 1512 se define como una función de una aplicación de recursos que puede requerir un control o tarificación aparte (por ejemplo, extracción de doc, recortes). A través del modelo de seguridad, a un Usuario se le puede conceder o denegar el uso de una característica para acceder a un contenido específico o una función. Un Recurso de Contenido 1514 representa un subconjunto definido de contenido que requiere un control o tarificación aparte (por ejemplo, noticias de Fiji, casos en Fiji). Los tipos de recursos comunes definidos para el modelo de seguridad son una colección de DOC o un conjunto de colecciones.

En el modelo de seguridad también puede definirse un objeto de control de acceso 1516. Un control de acceso concede o deniega permiso para acceder a un recurso de contenido (por ejemplo, noticias mundiales) a través de una característica (por ejemplo, Búsqueda). Dicho control de acceso se puede asignar a un grupo o a un individuo.

Cuando se define una suscripción, el usuario, el UsuarioGrupo, el Grupo, el control de acceso, la característica y elementos de recursos del modelo de seguridad se vinculan entre sí.

Entorno de múltiples capas

Las aplicaciones de recursos utilizan los componentes del Servidor de AR 300 y del Servidor de CCRDS 400 para desarrollar productos únicos con el fin de recuperar documentos.

La figura 5 ilustra la estructura de la capa de cliente, la capa de servidor y la capa de servidor de datos correspondiente a componentes que forman una aplicación de recursos y su infraestructura. Se utiliza un usuario para representar el lado del cliente, aunque al sistema accederán muchas interfaces de usuario de cliente diferentes y diversas. La interfaz de usuario del cliente accederá a un servidor web proporcionado por el entorno de suministro en línea, el cual, a su vez, proporciona el servidor de aplicación y el protocolo apropiados en función de la interfaz de usuario específica. A medida que se ejecutan búsquedas, el servidor de servicios compartidos, el servidor de directorios y el servidor de aplicaciones interaccionan con la capa de datos para obtener acceso a las bases de datos con el fin de recuperar documentos.

Entorno de desarrollo

La figura 7 muestra un entorno de desarrollo para aplicaciones de recursos bajo la presente invención. El proceso comienza con el cuestionario que define las características deseadas del producto, el dimensionamiento del entorno, las expectativas del nivel de servicio y la formación de agrupamientos (del inglés, *clustering*). El desarrollo incluye también varios tipos de registro para rendimiento, depuración e informes comerciales. La escalabilidad del diseño se considera con la planificación de la capacidad. Herramientas y procedimientos para la construcción y el despliegue son necesarias para el proceso de desarrollo, así como procedimientos para la gestión de cambios, la gestión de problemas, la transferencia a instancias superiores y foros de discusión. A medida que se construyen componentes, surge una necesidad de pruebas unitarias y de regresión y pruebas de rendimiento y de estrés. En el desarrollo se usan también varias herramientas para la administración de funciones de Alerta, Suministro e Historial de Seguimiento. Finalmente, el proceso de desarrollo de aplicaciones de recursos debe hacer frente a las actualizaciones de los sistemas operativos, los servidores web, los servidores de aplicaciones y las bases de datos.

Los anteriores servicios y herramientas de desarrollos se utilizan juntos, y se usa una monitorización para los diversos servidores web, servidores de aplicación, servicios/herramientas compartidos y servidores de bases de datos. Además, el proceso de desarrollo recurre necesariamente a los servicios comerciales en línea, los

sistemas comerciales, los servicios de contenido común y los componentes de publicación que forman parte del repositorio de contenido común.

REIVINDICACIONES

1. Sistema para mantener una agregación de gran tamaño de documentos almacenados electrónicamente y para hacer que los mismos estén disponibles para usuarios que envían mensajes de consulta, que comprende:

por lo menos una colección de datos (30, 30a, 30b) para almacenar documentos en forma electrónica, presentando cada documento un identificador exclusivo (118);

un componente de admisión (60) para recibir unos documentos nuevos (70) que deben ser añadidos a dicha por lo menos una colección de datos (30, 30a, 30b);

un componente de enriquecimiento (80) asociado al componente de admisión (60) para procesar un documento recibido para enriquecer el documento, siendo los documentos colocados en dicha por lo menos una colección de datos (30, 30a, 30b) mejorados por asociación con una o más características de enriquecimiento;

una pluralidad de aplicaciones de recursos (15, 15a, 15b, 15n), presentando cada aplicación de recursos (15, 15a, 15b, 15n) un componente de interfaz de usuario (10a, 10b, 10n) para recibir por lo menos un mensaje de consulta de usuario (12a, 12b, 12n) que busca información de entre la colección de datos (30, 30a, 30b);

un componente de búsqueda (22) para procesar dicho por lo menos un mensaje de consulta de usuario (12a, 12b, 12n) para identificar documentos en la colección de datos (30, 30a, 30b) que responden a dicho por lo menos un mensaje de consulta de usuario (12a, 12b, 12n) y recuperar un identificador (118) para esos documentos; y

un componente de suministro (40) que responde a una solicitud de documento de usuario para distribuir un documento solicitado al componente de interfaz de usuario (10a, 10b, 10n), siendo el documento solicitado de entre los documentos que responden para el suministro seleccionado por un mensaje de usuario adicional,

caracterizado por que

el sistema está adaptado para proporcionar formas diferentes de las características de enriquecimiento en función de la aplicación de recursos (15, 15a, 15b, 15n) usada por un usuario particular para el suministro del documento seleccionado.

2. Sistema según la reivindicación 1, que además comprende un servicio de persistencia para almacenar unos identificadores recuperados para un acceso posterior sin que el componente de búsqueda vuelva a procesar dicho por lo menos un mensaje de consulta de usuario (12a, 12b, 12h).

3. Sistema según la reivindicación 1, en el que los documentos en dicha por lo menos una colección de datos (30, 30a, 30b) se parten en por lo menos un subconjunto de colección, y en el que el componente de admisión (60) garantiza que cada documento adicional tenga un identificador exclusivo (118) y está asignado a por lo menos un subconjunto de colección.

4. Sistema según la reivindicación 1, en el que la agregación de documentos comprende por lo menos 20 terabytes de información.

5. Sistema según la reivindicación 1, en el que:

el componente de interfaz de usuario (10a, 10b, 10n) está adaptado a una aplicación de recursos (15, 15a, 15b, 15n) que se ejecuta en el sistema;

dicha por lo menos una colección de datos (30, 30a, 30b) está adaptada para almacenar documentos para su suministro a un usuario como respuesta a dicho por lo menos un mensaje de consulta (12a, 12b, 12n);

el sistema además comprende uno o más archivos de información de metadatos (150) para contener unos metadatos (50, 50a, 50b) para facilitar las búsquedas de los documentos almacenados en dicha por lo menos una colección de datos (30, 30a, 30b); y

el componente de admisión (60) tiene unos módulos de extracción de metadatos para desarrollar unos metadatos (50, 50a, 50b) a partir de un documento nuevo (70) y para almacenar por lo menos una parte de los metadatos (50, 50a, 50b) en uno o más archivos de información de metadatos (150), almacenados con el documento nuevo (70) preparado para el acceso de usuario en dicha por lo menos una colección de datos (30, 30a, 30b).

6. Sistema según la reivindicación 1 o 5, que comprende un servicio de tablas de contenidos para mantener por

lo menos una tabla de contenido con una pluralidad de nodos, identificando uno o más de dichos nodos uno o más documentos asociados a dicho uno o más nodos.

7. Sistema según la reivindicación 6, en el que el servicio de tablas de contenidos soporta dos o más tablas de contenidos, estando cada tabla de contenido adaptada a un tipo de usuario particular.

8. Sistema según la reivindicación 6, en el que dicha por lo menos una tabla de contenido hace referencia a documentos que están en dos o más colecciones de datos (30, 30a, 30b).

9. Sistema según la reivindicación 7, en el que una de las dos o más tablas de contenidos hace referencia a un nodo en otra de las dos o más tablas de contenidos para definir una estructura recursiva.

10. Sistema según la reivindicación 1 o 5, en el que los documentos en por lo menos una colección de datos (30, 30a, 30b) se parten en por lo menos un subconjunto de colección, y el sistema tiene un servicio de índices que mantiene un índice de palabras clave que aparecen por lo menos una vez en el subconjunto de colección, con una asociación entre las palabras clave del índice y la ubicación de su aparición en el subconjunto de colección.

11. Sistema según la reivindicación 5, en el que los documentos de por lo menos una colección de datos (30, 30a, 30b) se parten en por lo menos un subconjunto de colección, y en el que los medios para recibir documentos nuevos (70) garantizan que cada documento adicional presenta un identificador exclusivo y está asignado a por lo menos un subconjunto de la colección.

12. Sistema según la reivindicación 1 o 5, en el que dicha por lo menos una colección de datos (30, 30a, 30b) tiene por lo menos un conjunto de documentos que es una agregación de los documentos en uno o más subconjuntos de colección.

13. Sistema según la reivindicación 1 o 5, que además comprende un componente de servicio de seguridad que recibe información de identificación de usuario y unos mensajes de consulta, y valida la información de identificación y los mensajes de consulta con respecto a la información de suscripción de los usuarios.

14. Sistema según la reivindicación 1 o 5, en el que los módulos de extracción de metadatos procesan un documento para provocar la asociación del documento con por lo menos uno de los siguientes:

un material editorial adicional preparado por un agente humano;
un material editorial adicional preparado por un agente automatizado;
un enlace que proporciona un puntero a otro documento en la colección de datos (30, 30a, 30b);
unos metadatos basados en citas (50) para documentos jurídicos o bibliográficos; o
una entrada asociada al documento que aparece en un archivo de metadatos.

15. Sistema según la reivindicación 1 o 5, en el que el componente de admisión (60) prioriza documentos y procesa fuera de un orden normal basado en el tiempo de recepción que son portadores de un indicador sensible al tiempo.

16. Sistema según la reivindicación 1 o 5, en el que el componente de admisión (60) comprueba la unicidad de un identificador de documento asignado antes de que se haga que un documento nuevo (70) con dicho identificador de documento esté disponible en cualquier colección de datos (30, 30a, 30b).

17. Sistema según la reivindicación 1 o 5, en el que el componente de admisión (60) comprueba un formato de admisión predeterminado de un documento nuevo (70), antes de que se haga que dicho documento nuevo (70) esté disponible en cualquier colección de datos (30, 30a, 30b).

18. Sistema según la reivindicación 5, en el que una o más colecciones de datos (30, 30a, 30b) comprenden por lo menos 20 terabytes de información.

19. Sistema según la reivindicación 5, que además comprende unos servicios de seguridad y facturación compartidos por dos o más aplicaciones de recursos para controlar el acceso a una o más o colecciones de datos (30, 30a, 30b) como respuesta a mensajes de consulta y desarrollar información para usuarios de la facturación para el acceso a una o más colecciones de datos (30, 30a, 30b).

20. Sistema según la reivindicación 19, en el que una o más colecciones de datos (30, 30a, 30b) contienen uno o más de los siguientes tipos de información:

jurídica, impositiva, de contabilidad, médica, científica, de propiedad intelectual, material escolar educativo o de noticias.

21. Sistema según la reivindicación 20, que además comprende un servicio de tablas de contenidos para

mantener para cada uno de los tipos de información por lo menos una tabla de contenidos con una pluralidad de nodos, identificando uno o más de dichos nodos uno o más documentos asociados a dicho uno o más nodos y por lo menos un documento está identificado en más de una tabla de contenido.

22. Sistema según la reivindicación 5, en el que los metadatos (50, 50a, 50b) están en forma de sentencias del Marco de Descripción de Recursos (RDF).

23. Sistema según la reivindicación 22, que además comprende un servicio de tablas de contenidos para mantener por lo menos una tabla de contenido con una pluralidad de nodos para cada uno de los tipos de información, y por lo menos uno de los módulos de extracción de metadatos usa como vocabulario para las sentencias de RDF, las etiquetas de nodos de dicha por lo menos una tabla de contenido.

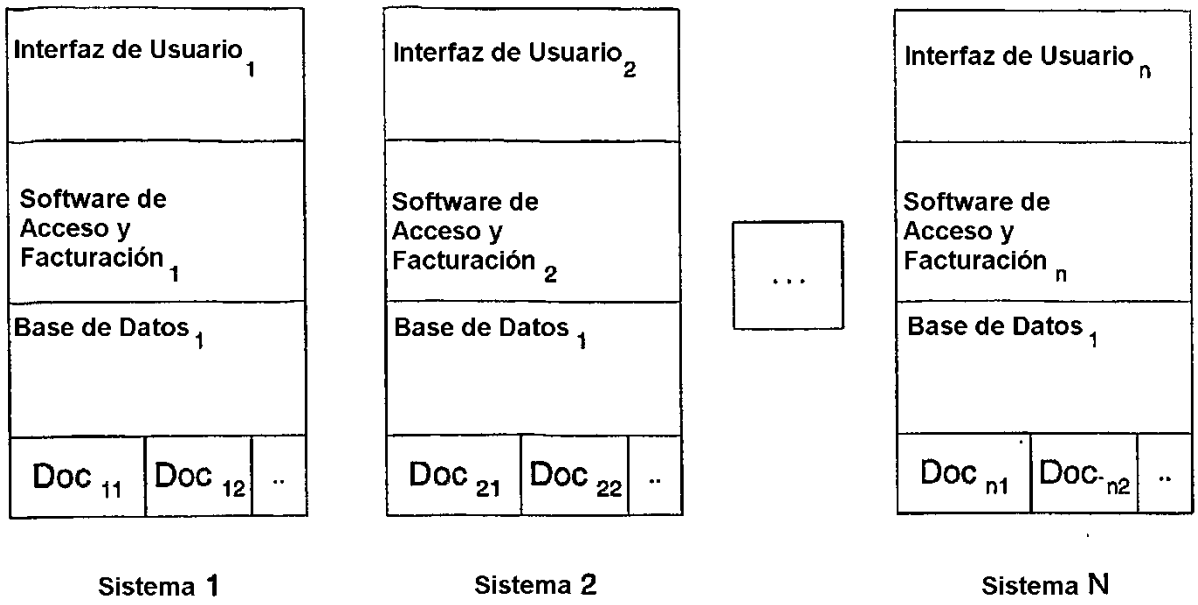


FIG. 1 (Técnica Anterior)

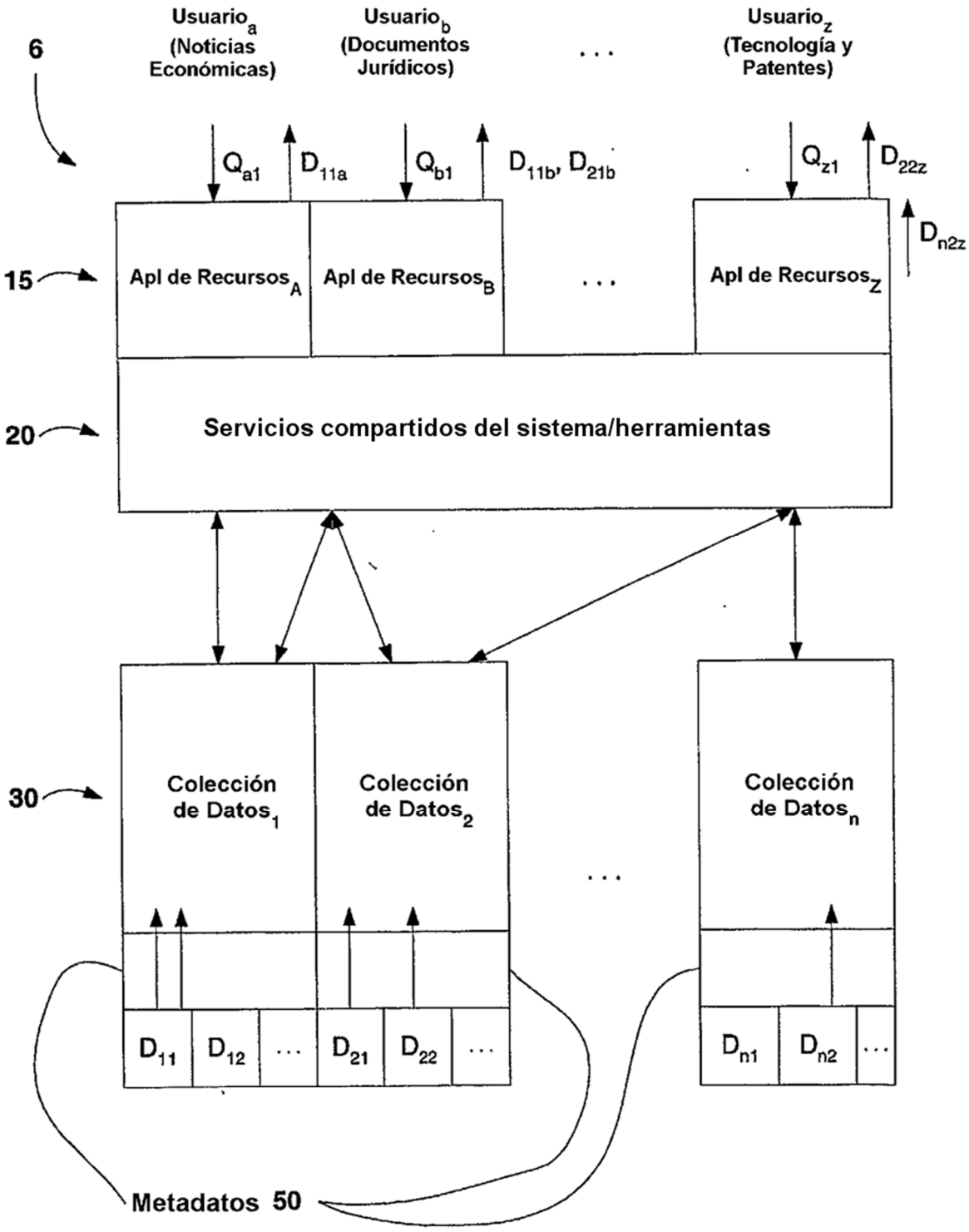

6
Usuario$_a$ (Noticias Económicas)
Usuario$_b$ (Documentos Jurídicos)
Usuario$_z$ (Tecnología y Patentes)
Q_{a1}
D_{11a}
Q_{b1}
D_{11b}, D_{21b}
Q_{z1}
D_{22z}
D_{n2z}
15
Apl de Recursos$_A$
Apl de Recursos$_B$
Apl de Recursos$_Z$
20
Servicios compartidos del sistema/herramientas
30
Colección de Datos$_1$
Colección de Datos$_2$
Colección de Datos$_n$
D_{11} D_{12} ... D_{21} D_{22} ...
D_{n1} D_{n2} ...
Metadatos 50

FIG. 2

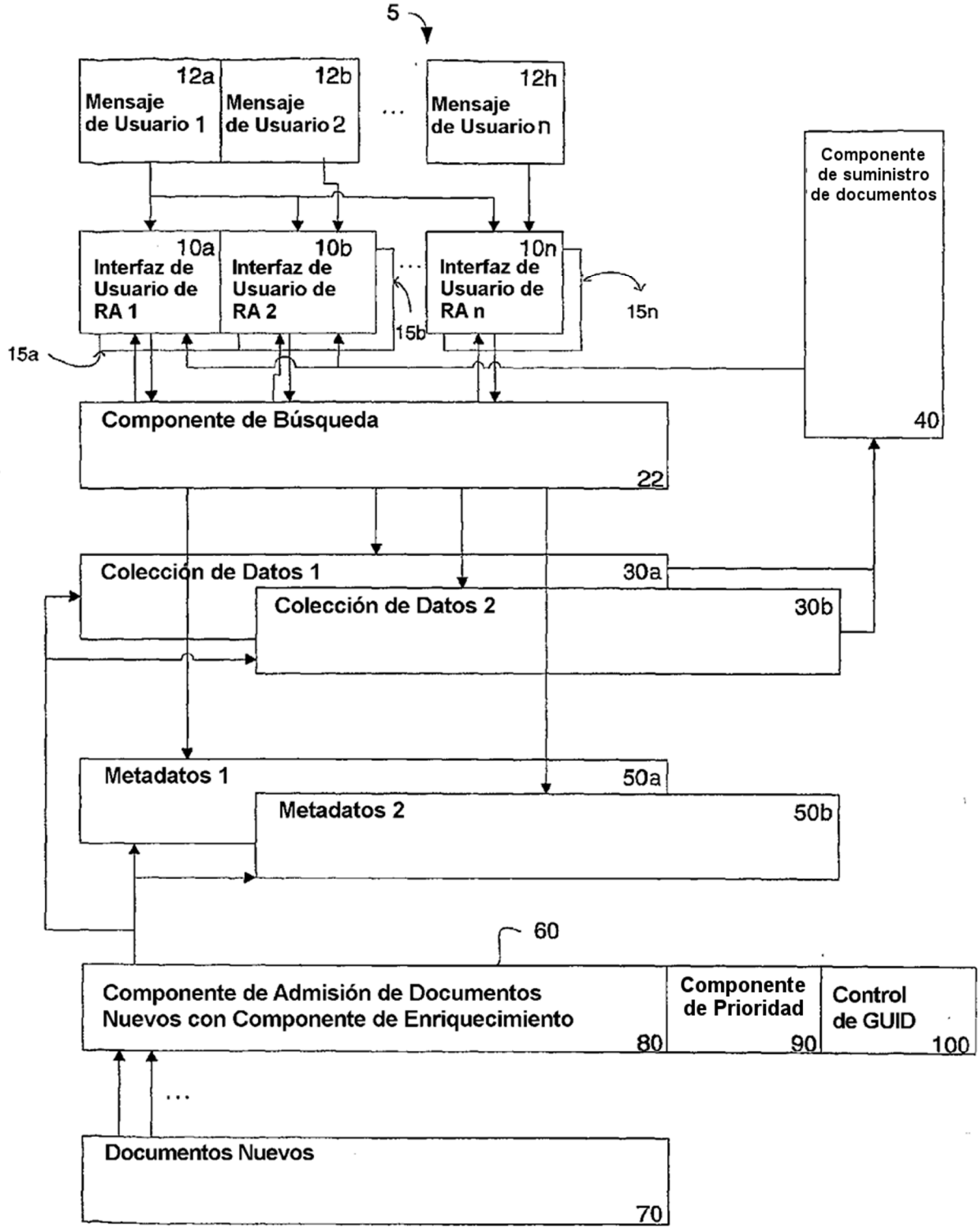

5
12a
Mensaje de Usuario 1
12b
Mensaje de Usuario 2
...
12h
Mensaje de Usuario n
Componente de suministro de documentos
10a
Interfaz de Usuario de RA 1
10b
Interfaz de Usuario de RA 2
10n
Interfaz de Usuario de RA n
15a
15b
15n
Componente de Búsqueda
22
40
Colección de Datos 1
30a
Colección de Datos 2
30b
Metadatos 1
50a
Metadatos 2
50b
60
Componente de Admisión de Documentos Nuevos con Componente de Enriquecimiento
80
Componente de Prioridad
90
Control de GUID
100
Documentos Nuevos
70

FIG. 3

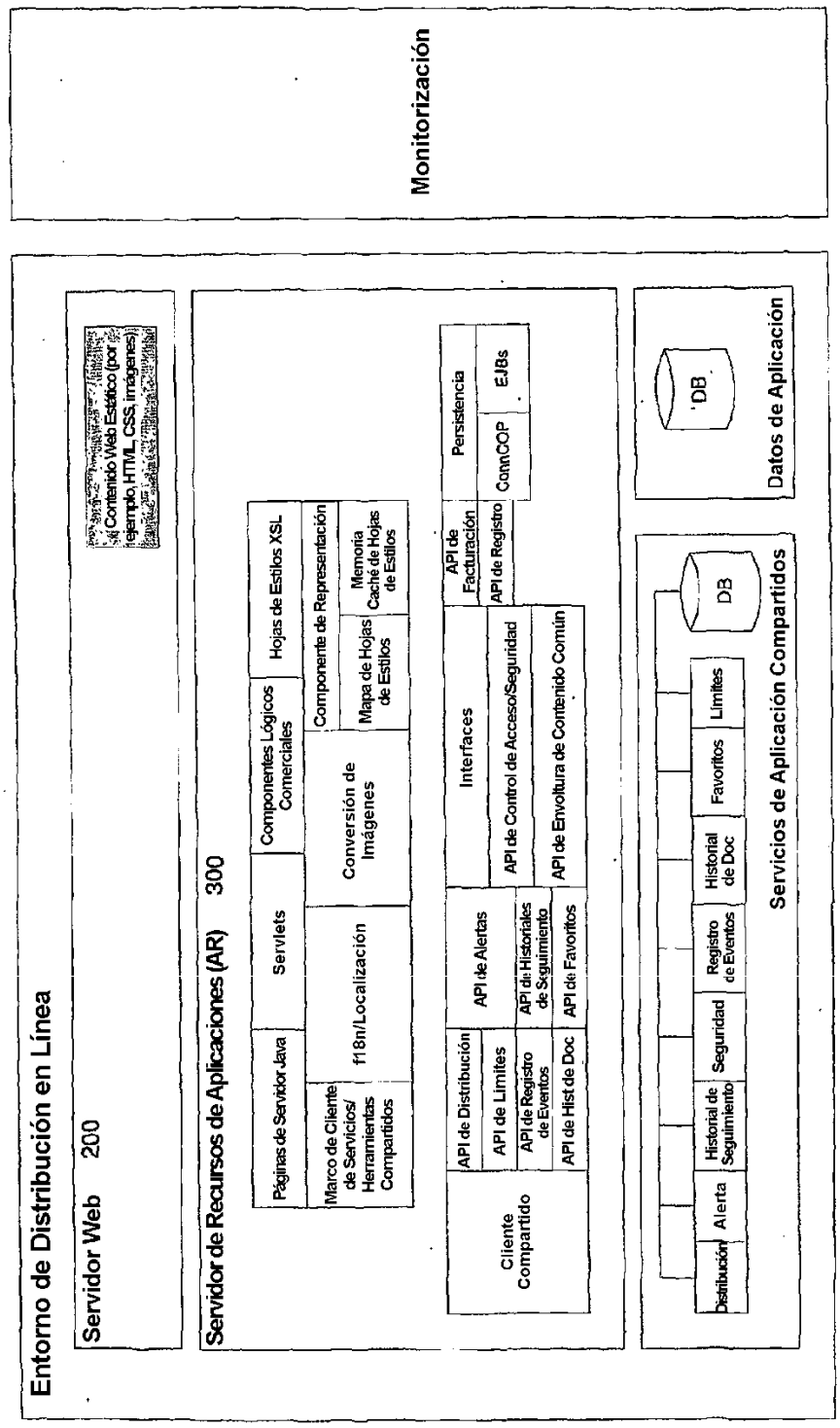

Entorno de Distribución en Línea
Servidor Web 200
Contenido Web Estático (por ejemplo, HTML, CSS, imágenes)
Servidor de Recursos de Aplicaciones (AR) 300
Páginas de Servidor Java
Servlets
Componentes Lógicos Comerciales
Hojas de Estilos XSL
Componente de Representación
Marco de Cliente de Servicios/ Herramientas Compartidos
f18n/Localización
Conversión de Imágenes
Mapa de Hojas de Estilos
Memoria Caché de Hojas de Estilos
Cliente Compartido
API de Distribución
API de Límites
API de Registro de Eventos
API de Hist de Doc
API de Alertas
API de Historiales de Seguimiento
API de Favoritos
Interfaces
API de Control de Acceso/Seguridad
API de Envoltura de Contenido Común
API de Facturación
API de Registro
Persistencia
ConnCOP
EJBs
Monitorización
Distribución
Alerta
Historial de Seguimiento
Seguridad
Registro de Eventos
Historial de Doc
Favoritos
Límites
DB
Servicios de Aplicación Compartidos
DB
Datos de Aplicación
Cargos de Contabilidad
500
Servicios Comerciales en Línea del Sistema
Búsqueda
TOC
Doc
Utilidad
CCI
Gestión de Carga
Gestión de Datos
LOG
Servicios de Contenido Común
Servidor de CCRDS
400
700
API de Publicación de Contenido Común
Publicación
Facturación en Línea
SAP
Sistemas Comerciales
600

FIG. 4

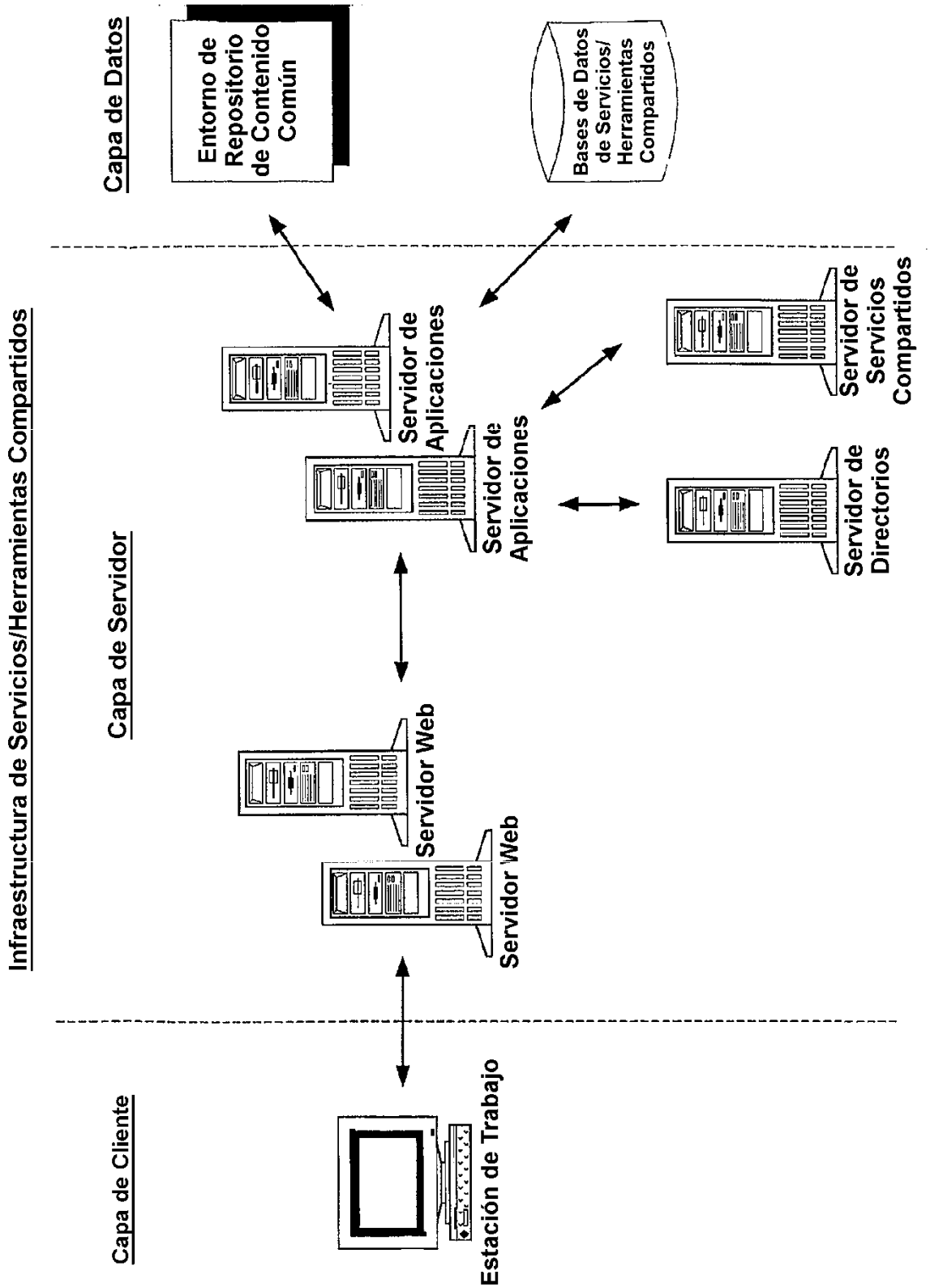

Infraestructura de Servicios/Herramientas Compartidos
Capa de Datos
Entorno de Repositorio de Contenido Común
Bases de Datos de Servicios/ Herramientas Compartidos
Capa de Servidor
Servidor de Aplicaciones
Servidor de Aplicaciones
Servidor de Servicios Compartidos
Servidor de Directorios
Servidor Web
Servidor Web
Capa de Cliente
Estación de Trabajo

FIG. 5

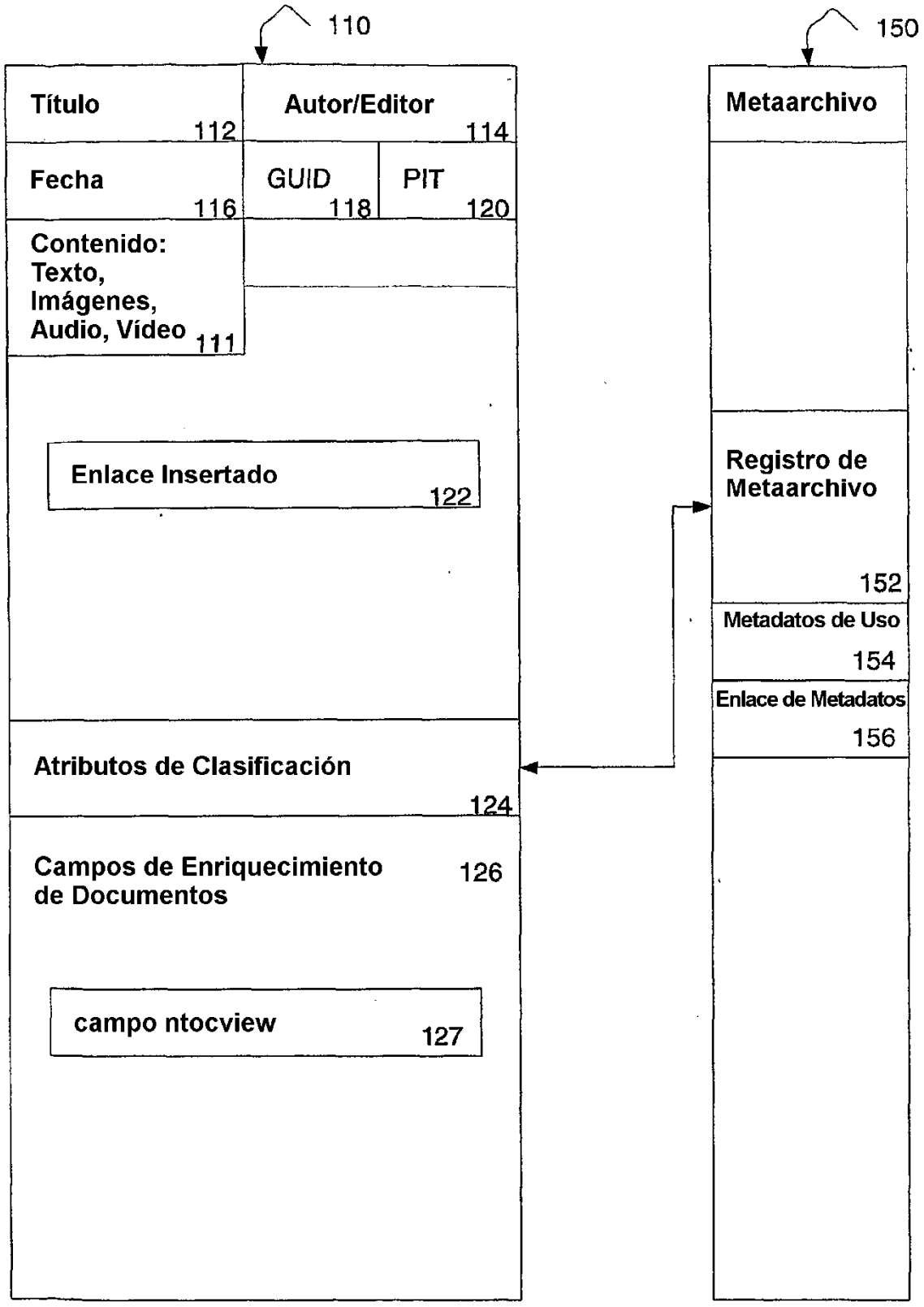

110
Título
112
Autor/Editor
114
Fecha
116
GUID
118
PIT
120
Contenido:
Texto,
Imágenes,
Audio, Vídeo
111
Enlace Insertado
122
Atributos de Clasificación
124
Campos de Enriquecimiento
de Documentos
126
campo ntocview
127
150
Metaarchivo
Registro de
Metaarchivo
152
Metadatos de Uso
154
Enlace de Metadatos
156

FIG. 6

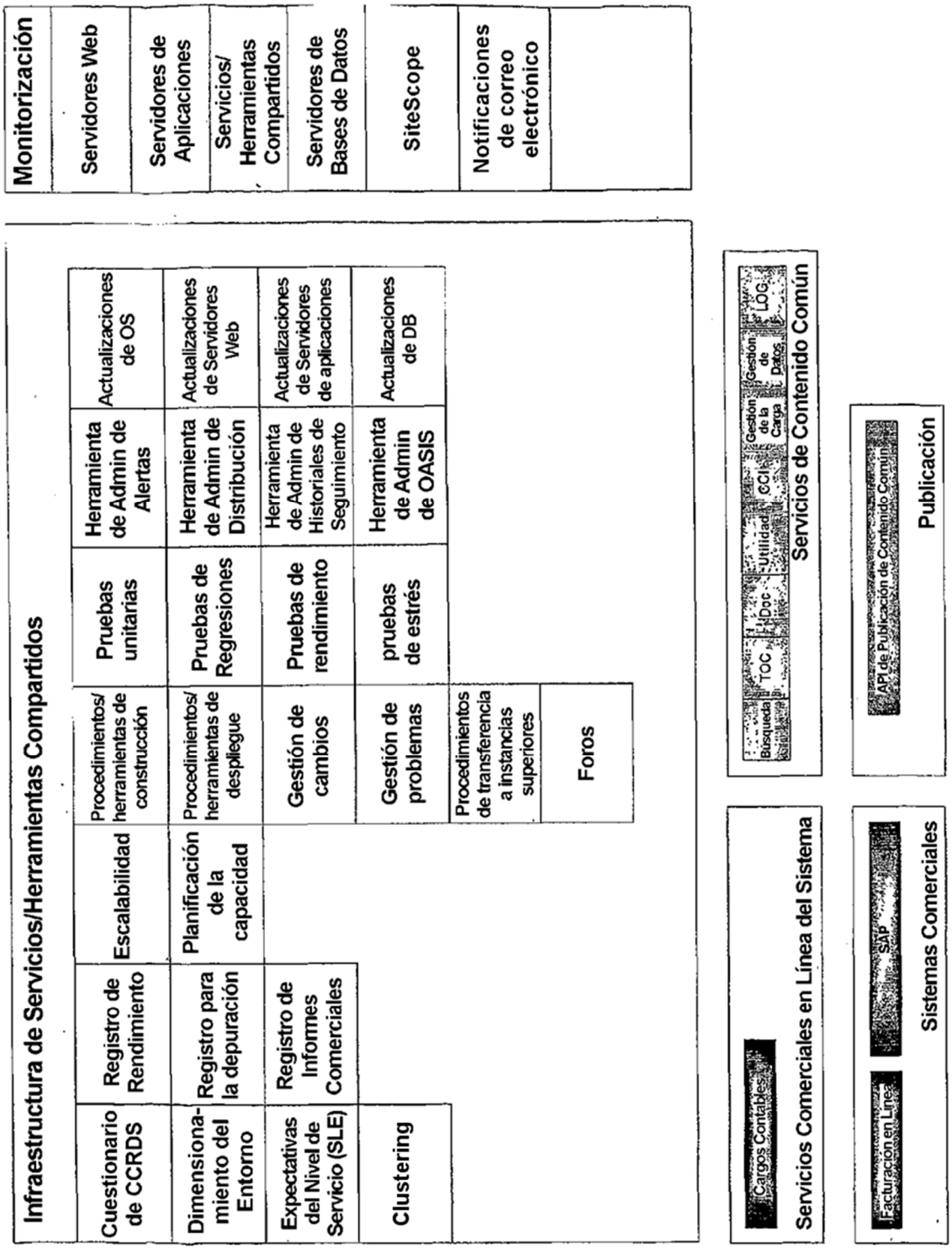

Infraestructura de Servicios/Herramientas Compartidos
Cuestionario de CCRDS
Registro de Rendimiento
Escalabilidad
Procedimientos/ herramientas de construcción
Pruebas unitarias
Herramienta de Admin de Alertas
Actualizaciones de OS
Dimensiona-miento del Entorno
Registro para la depuración
Planificación de la capacidad
Procedimientos/ herramientas de despliegue
Pruebas de Regresiones
Herramienta de Admin de Distribución
Actualizaciones de Servidores Web
Expectativas del Nivel de Servicio (SLE)
Registro de Informes Comerciales
Gestión de cambios
Pruebas de rendimiento
Herramienta de Admin de Historiales de Seguimiento
Actualizaciones de Servidores de aplicaciones
Clustering
Gestión de problemas
pruebas de estrés
Herramienta de Admin de OASIS
Actualizaciones de DB
Procedimientos de transferencia a instancias superiores
Foros
Monitorización
Servidores Web
Servidores de Aplicaciones
Servicios/ Herramientas Compartidos
Servidores de Bases de Datos
SiteScope
Notificaciones de correo electrónico
Cargos Contables
Servicios Comerciales en Línea del Sistema
Búsqueda
TOC
Utilidad
Gestión de la Carga
Gestión de Datos
LOG
Servicios de Contenido Común
Facturación en Línea
SAP
Sistemas Comerciales
API de Publicación de Contenido Común
Publicación

FIG. 7

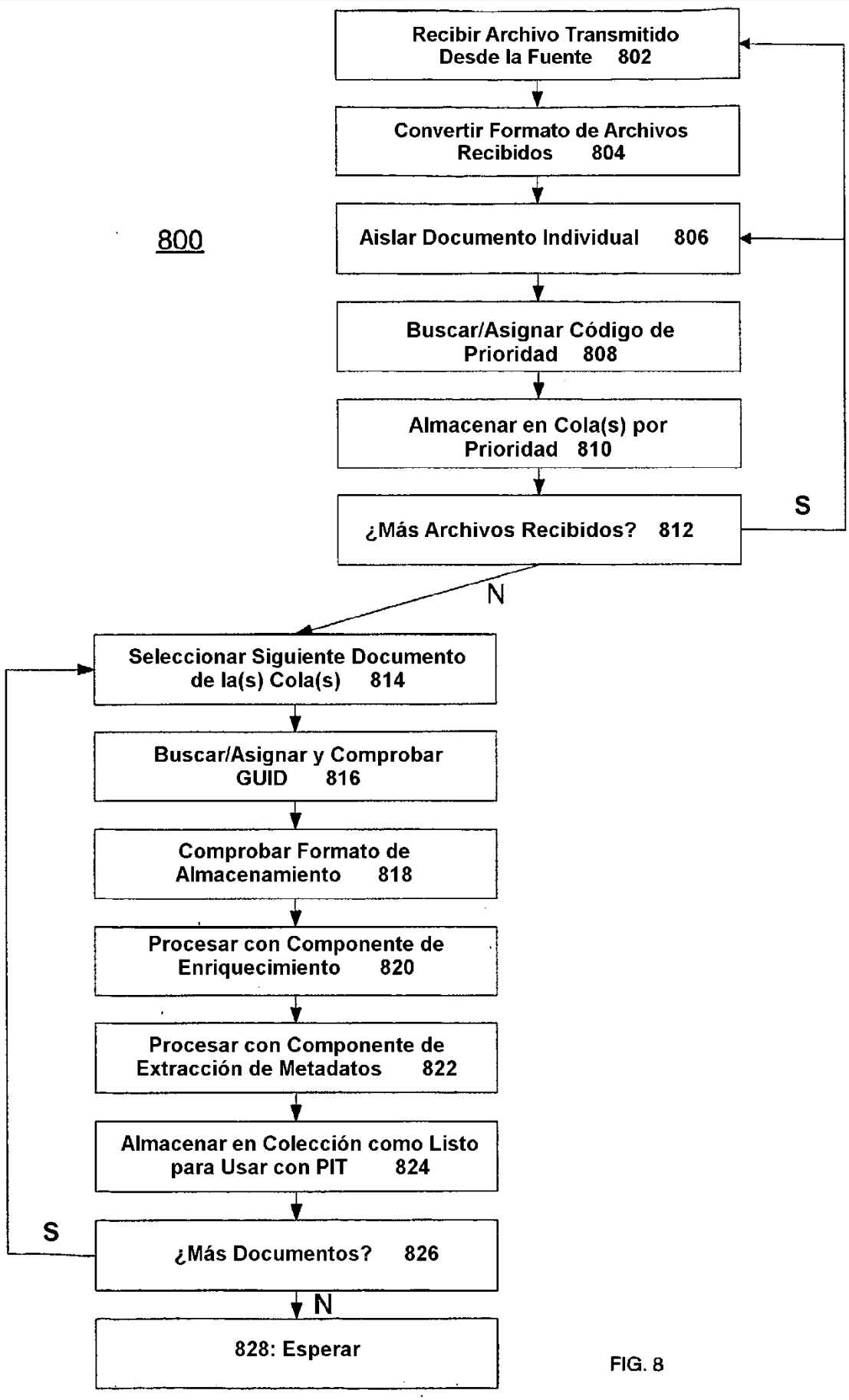

800
Recibir Archivo Transmitido Desde la Fuente 802
Convertir Formato de Archivos Recibidos 804
Aislar Documento Individual 806
Buscar/Asignar Código de Prioridad 808
Almacenar en Cola(s) por Prioridad 810
¿Más Archivos Recibidos? 812
S
N
Seleccionar Siguiente Documento de la(s) Cola(s) 814
Buscar/Asignar y Comprobar GUID 816
Comprobar Formato de Almacenamiento 818
Procesar con Componente de Enriquecimiento 820
Procesar con Componente de Extracción de Metadatos 822
Almacenar en Colección como Listo para Usar con PIT 824
S
¿Más Documentos? 826
N
828: Esperar

FIG. 8

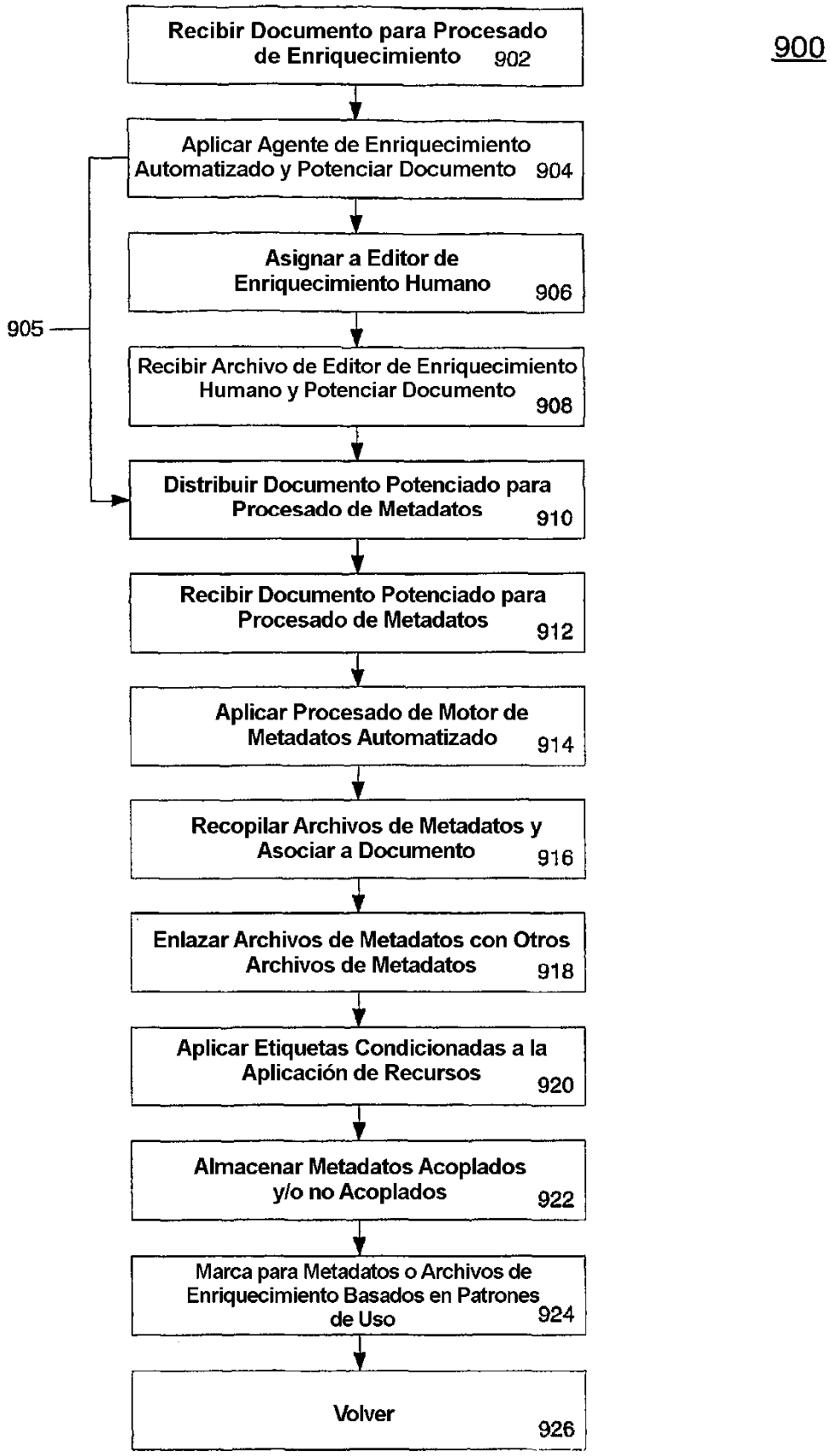

900
Recibir Documento para Procesado de Enriquecimiento 902
Aplicar Agente de Enriquecimiento Automatizado y Potenciar Documento 904
Asignar a Editor de Enriquecimiento Humano 906
905
Recibir Archivo de Editor de Enriquecimiento Humano y Potenciar Documento 908
Distribuir Documento Potenciado para Procesado de Metadatos 910
Recibir Documento Potenciado para Procesado de Metadatos 912
Aplicar Procesado de Motor de Metadatos Automatizado 914
Recopilar Archivos de Metadatos y Asociar a Documento 916
Enlazar Archivos de Metadatos con Otros Archivos de Metadatos 918
Aplicar Etiquetas Condicionadas a la Aplicación de Recursos 920
Almacenar Metadatos Acoplados y/o no Acoplados 922
Marca para Metadatos o Archivos de Enriquecimiento Basados en Patrones de Uso 924
Volver 926

FIG. 9

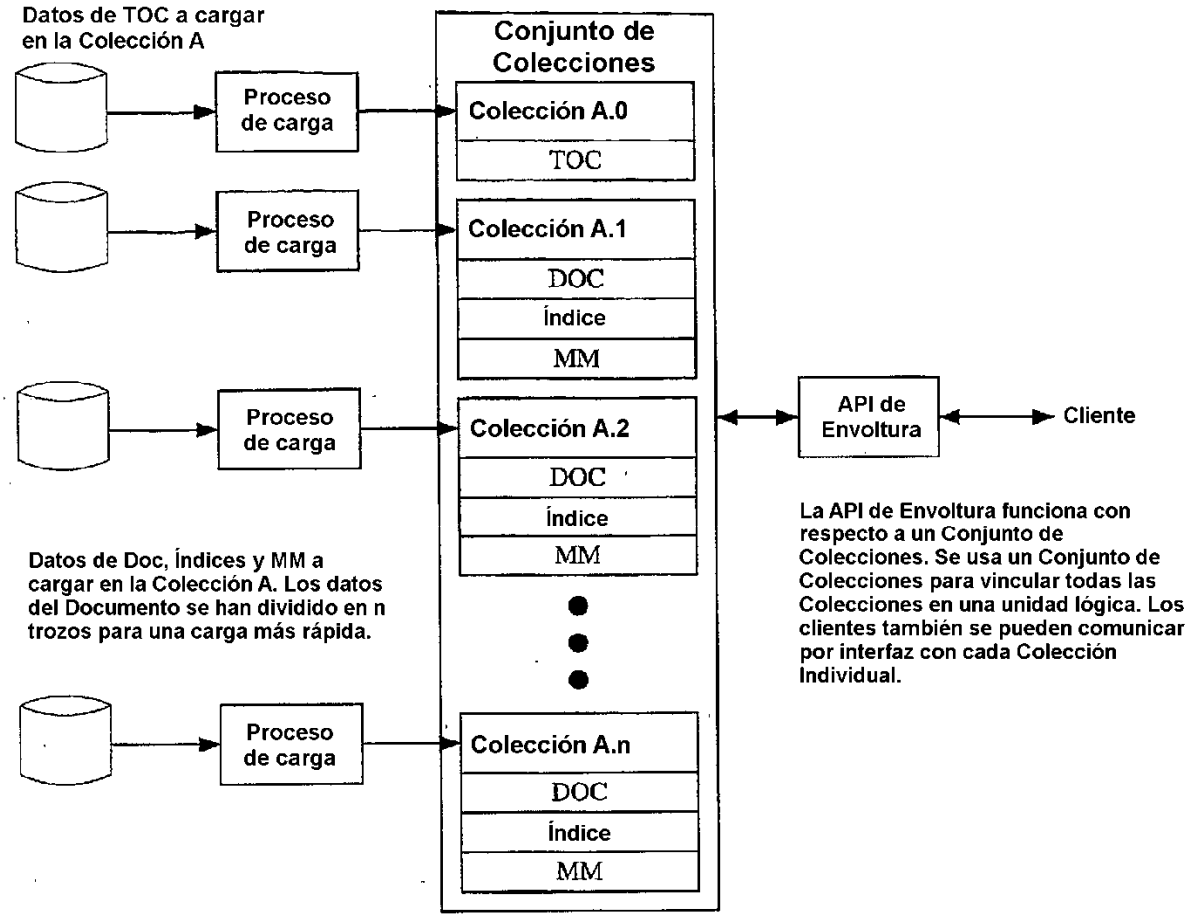

Datos de TOC a cargar en la Colección A
Proceso de carga
Conjunto de Colecciones
Colección A.0
TOC
Proceso de carga
Colección A.1
DOC
Índice
MM
Proceso de carga
Colección A.2
DOC
Índice
MM
API de Envoltura
Cliente
La API de Envoltura funciona con respecto a un Conjunto de Colecciones. Se usa un Conjunto de Colecciones para vincular todas las Colecciones en una unidad lógica. Los clientes también se pueden comunicar por interfaz con cada Colección Individual.
Datos de Doc, Índices y MM a cargar en la Colección A. Los datos del Documento se han dividido en n trozos para una carga más rápida.
Proceso de carga
Colección A.n
DOC
Índice
MM

FIG. 10

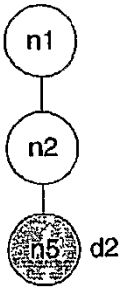


FIG. 11A

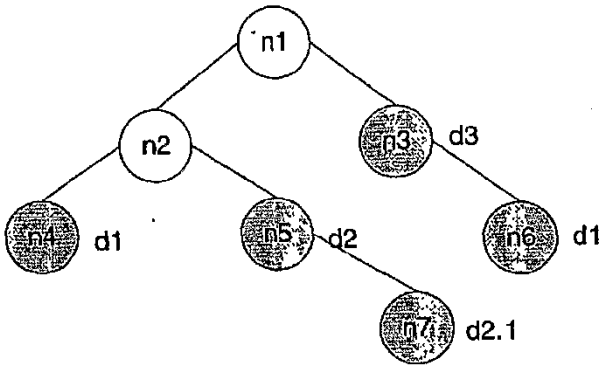


FIG. 11B

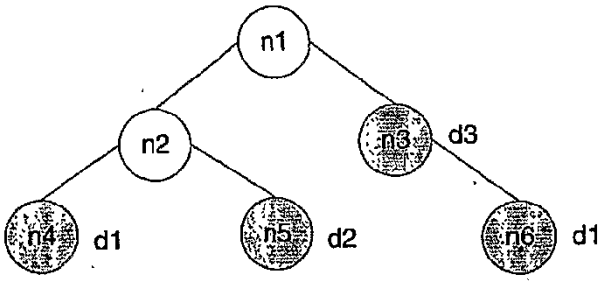


FIG. 11C

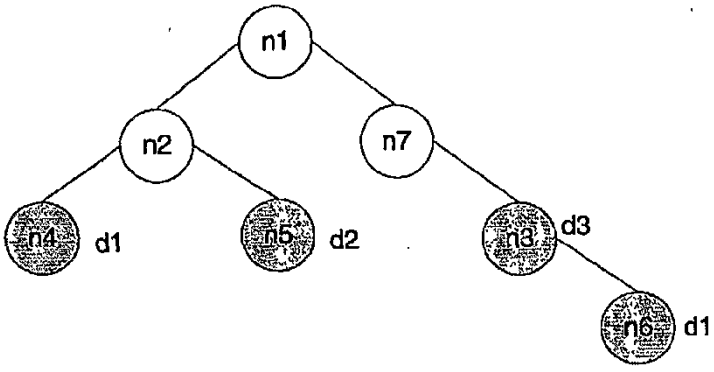
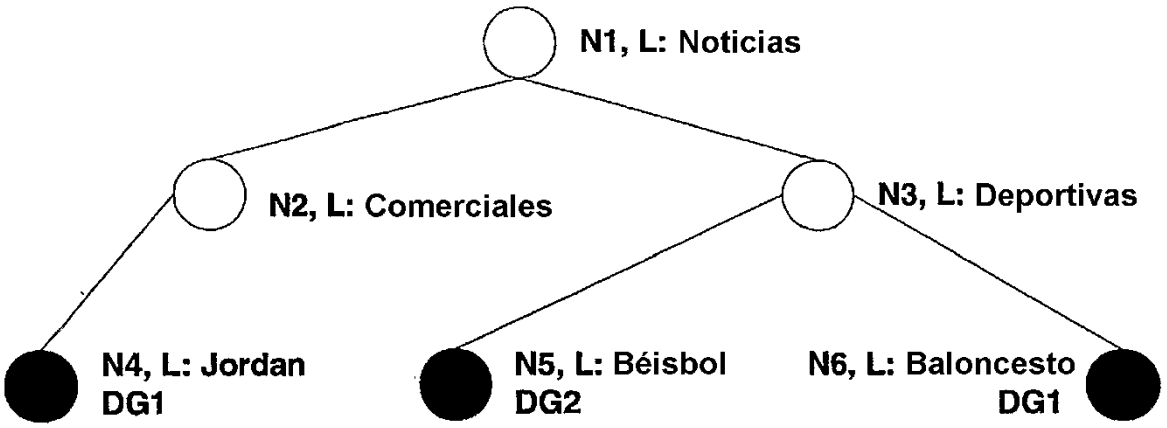


FIG. 11D



GUID de Nodo: Etiqueta	GUID de Doc: Contenido
N1: Noticias	
N2: Comerciales	
N3: Deportivas	
N4: Jordan	DG1: Promociones de Jordan
N5: Béisbol	DG2: Partido 5 de la Serie Mundial
N6: Baloncesto	DG1: Promociones de Jordan

FIG. 12

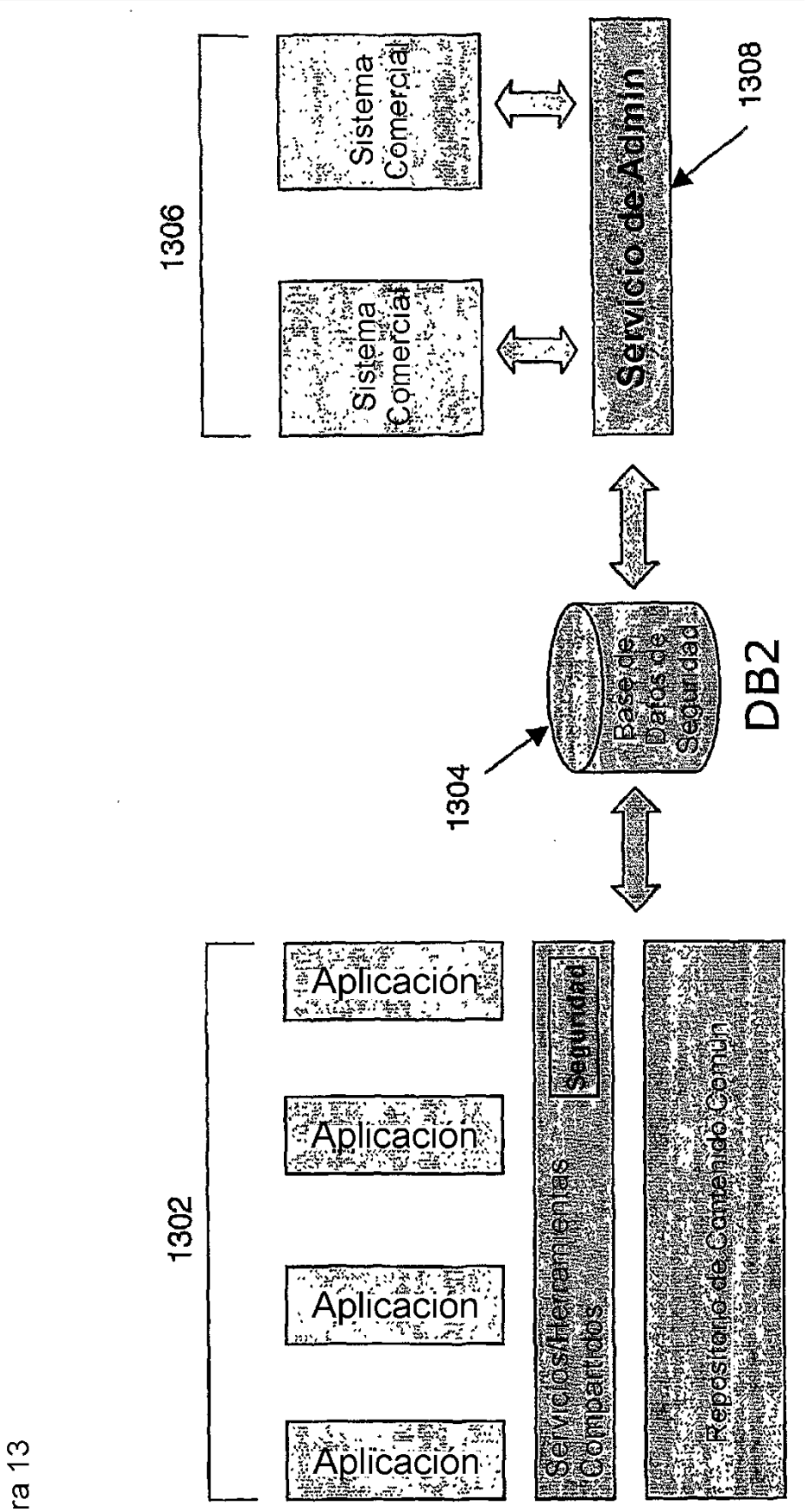
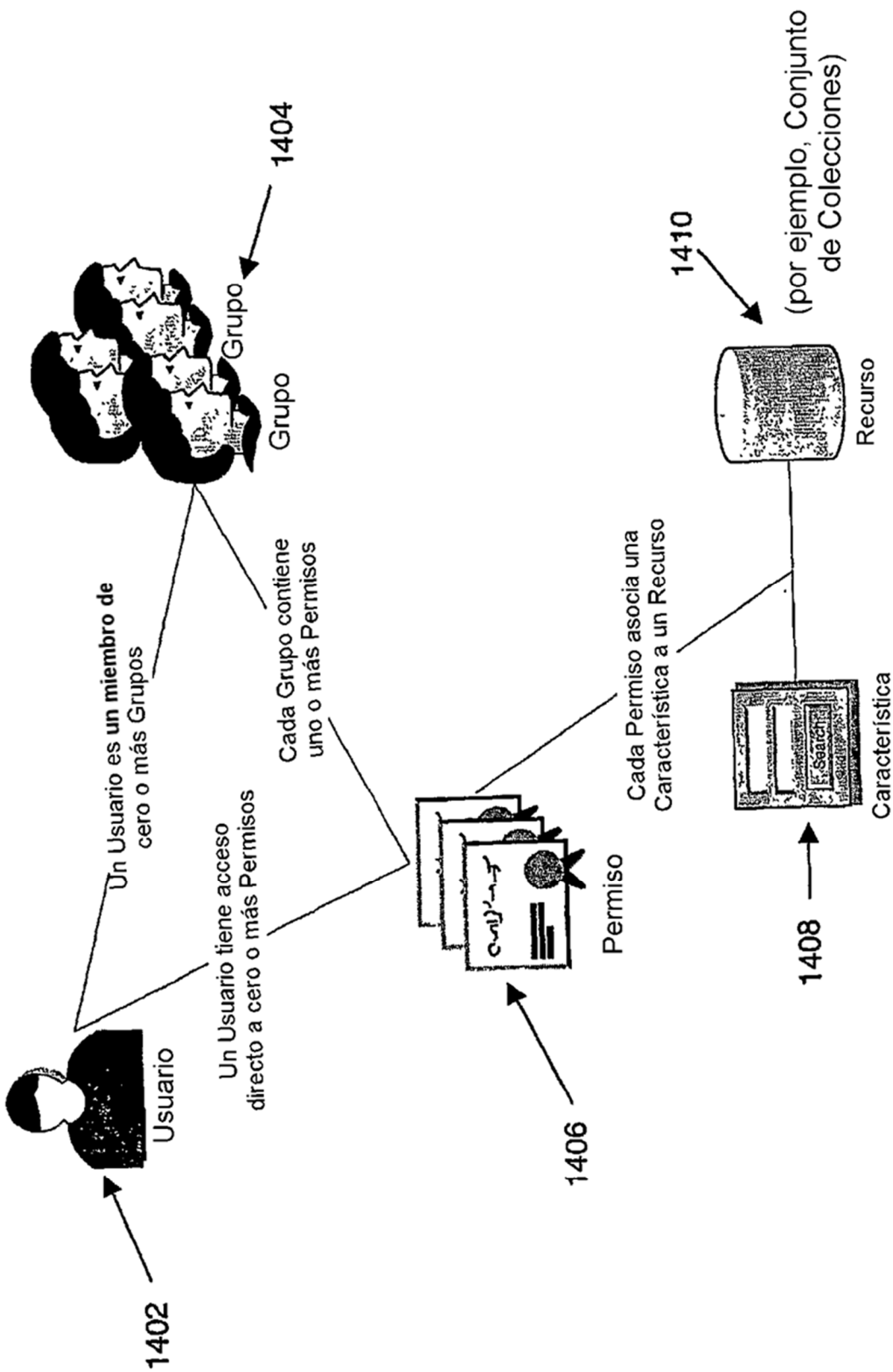

1306
Sistema Comercial
Sistema Comercial
Servicio de Admin
1308
1304
Base de Datos de Seguridad
DB2
1302
Aplicación
Aplicación
Aplicación
Aplicación
Seguridad
Servicios/Herramientas Compartidos
Repositorio de Contenido Común

Figura 13

Figura 14


1402
Usuario
Un Usuario es un miembro de cero o más Grupos
Un Usuario tiene acceso directo a cero o más Permisos
1404
Grupo
Grupo
Cada Grupo contiene uno o más Permisos
1406
Permiso
Cada Permiso asocia una Característica a un Recurso
1408
Característica
1410
(por ejemplo, Conjunto de Colecciones)
Recurso

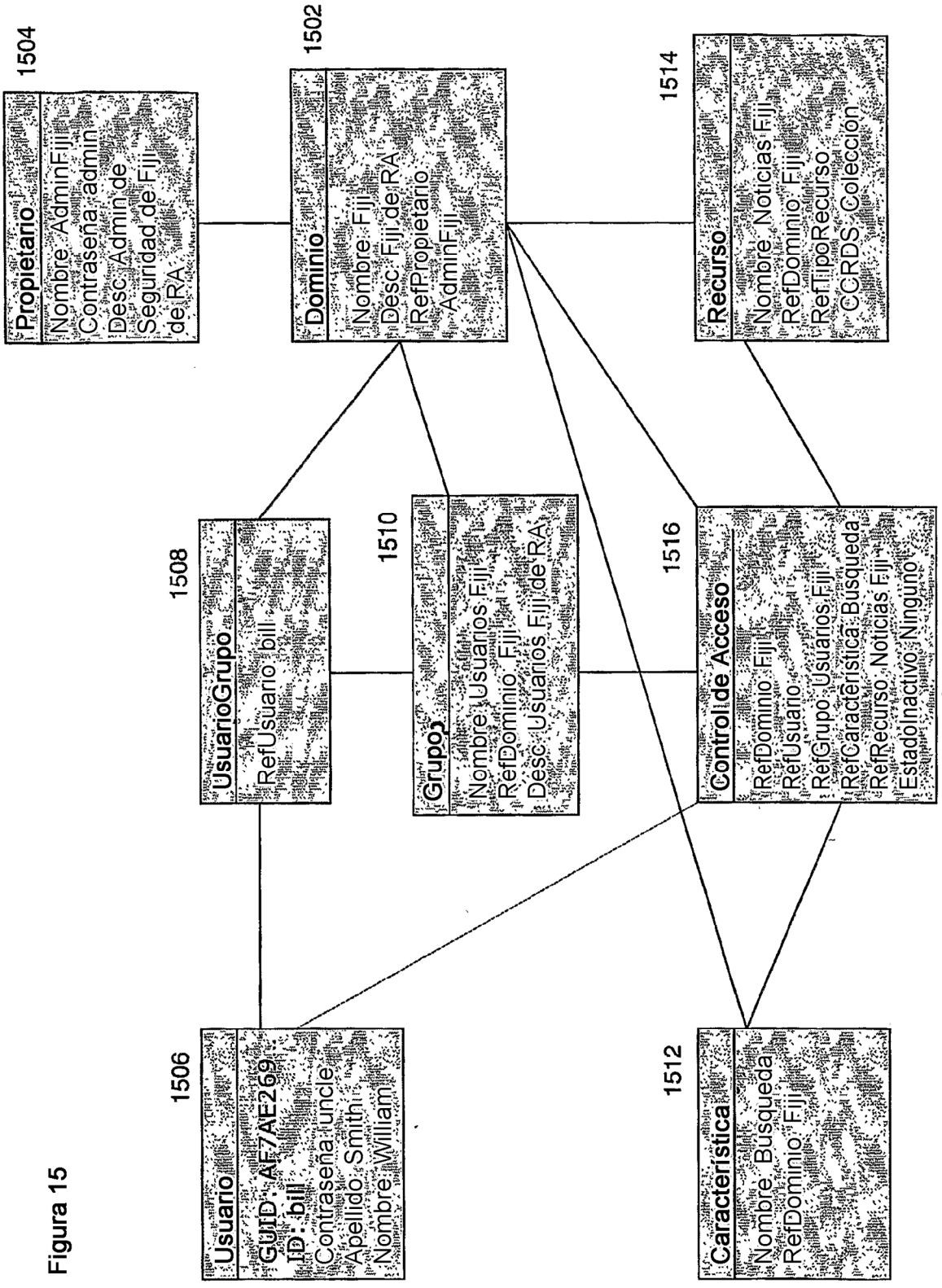

1504
Propietario
Nombre AdminFiji
Contraseña admin
Desc Admin de Seguridad de Fiji de RA
1502
Dominio
Nombre Fiji
Desc Fiji de RA
RefPropietario AdminFiji
1514
Recurso
Nombre Noticias Fiji
RefDominio Fiji
RefTipoRecurso CCRDS Coleccion
1508
UsuarioGrupo
RefUsuario bill
1510
Grupo
Nombre Usuarios Fiji
RefDominio Fiji
Desc Usuarios Fiji de RA
1516
Control de Acceso
RefDominio Fiji
RefUsuario
RefGrupo Usuarios Fiji
RefCaracteristica Busqueda
RefRecurso Noticias Fiji
EstadoInactivo Ninguno
1506
Usuario
GUID AE7AE269
ID bill
Contraseña uncle
Apellido Smith
Nombre William
1512
Caracteristica
Nombre Busqueda
RefDominio Fiji

Figura 15

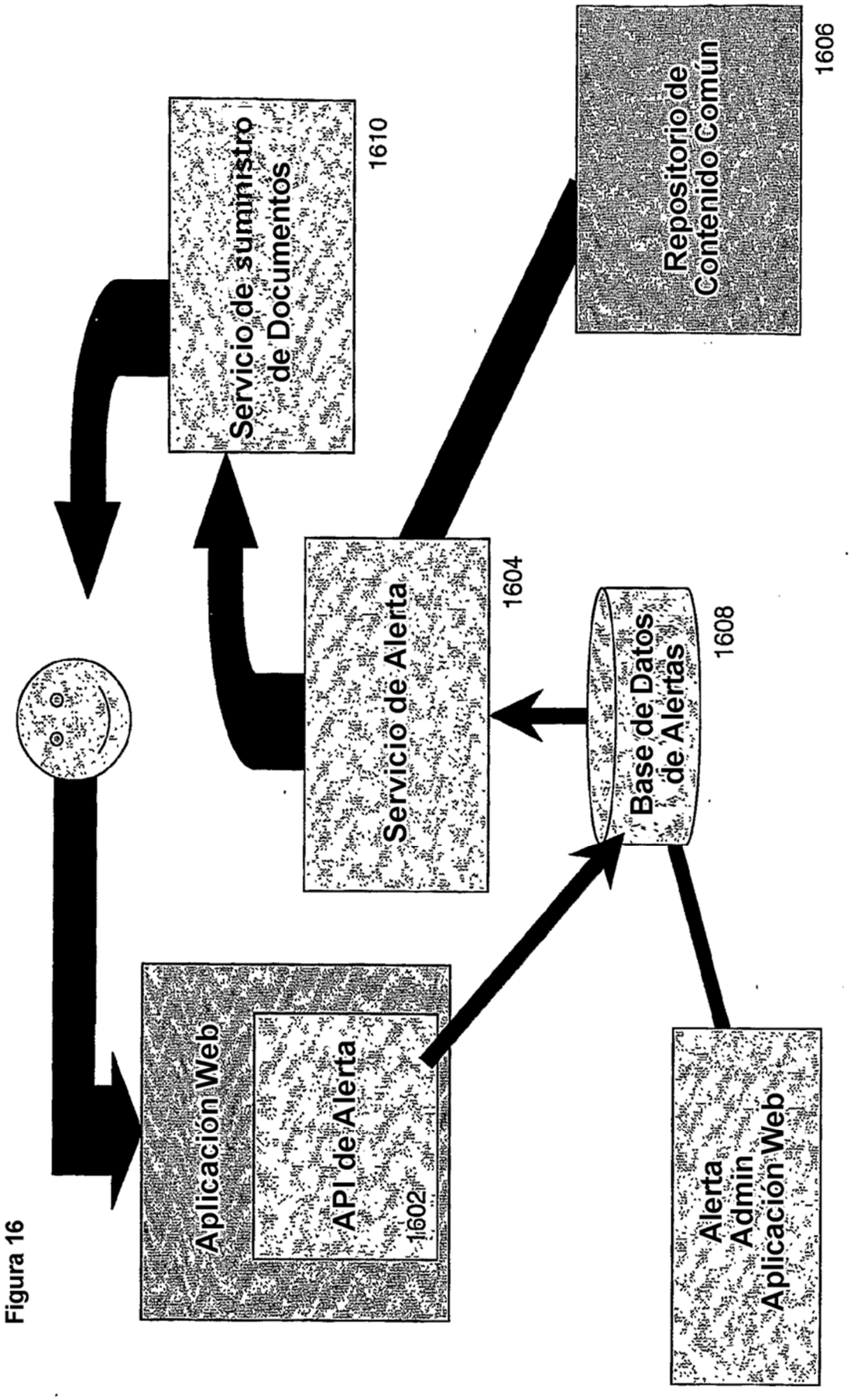

Repositorio de Contenido Común
1606
Servicio de suministro de Documentos
1610
Servicio de Alerta
1604
Base de Datos de Alertas
1608
Aplicación Web
API de Alerta
1602
Alerta Admin Aplicación Web

Figura 16

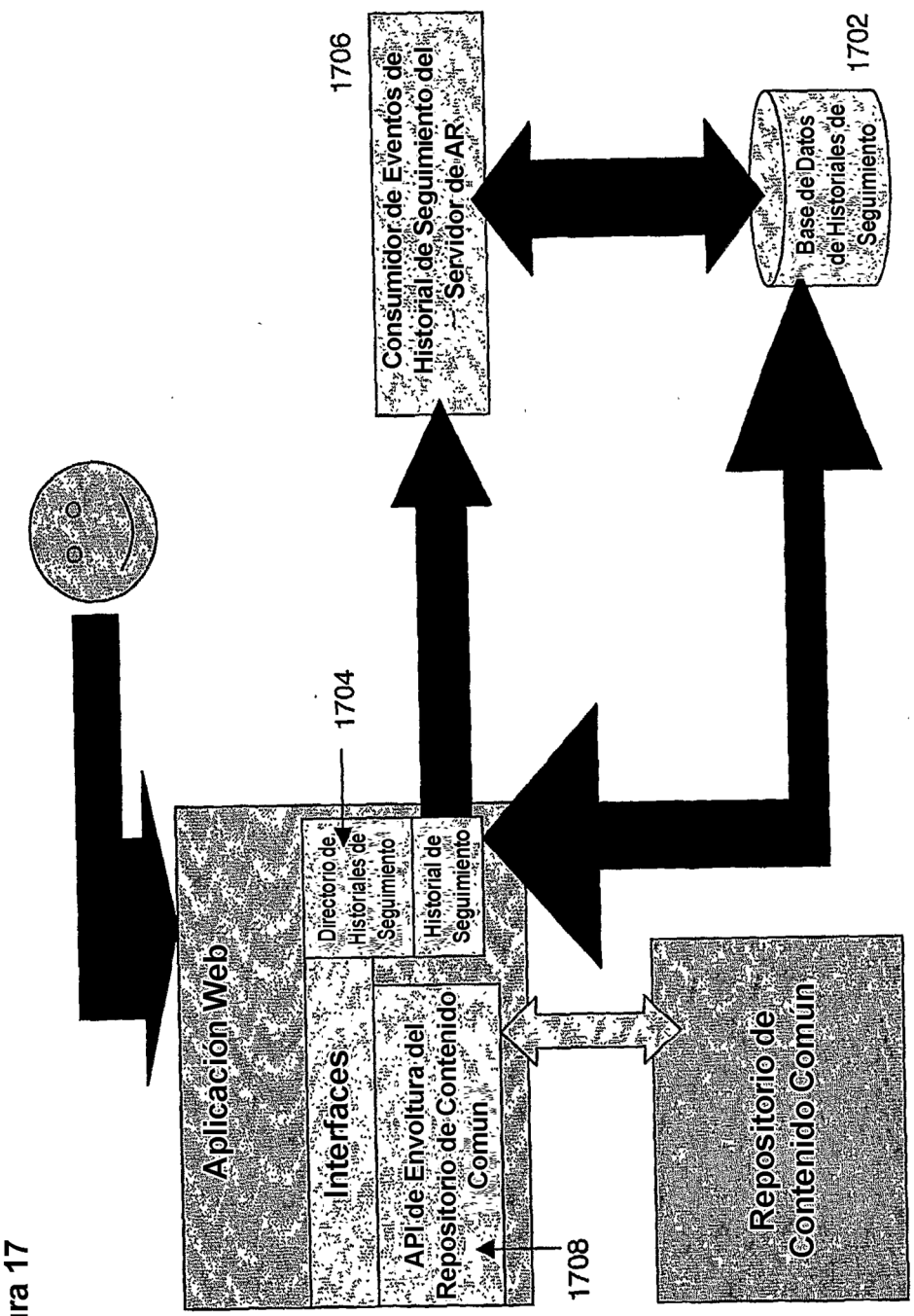

1706
Consumidor de Eventos de Historial de Seguimiento del Servidor de AR
1702
Base de Datos de Historiales de Seguimiento
1704
Aplicación Web
Interfaces
Directorio de Historiales de Seguimiento
Historial de Seguimiento
API de Envoltura del Repositorio de Contenido Común
1708
Repositorio de Contenido Común

Figura 17

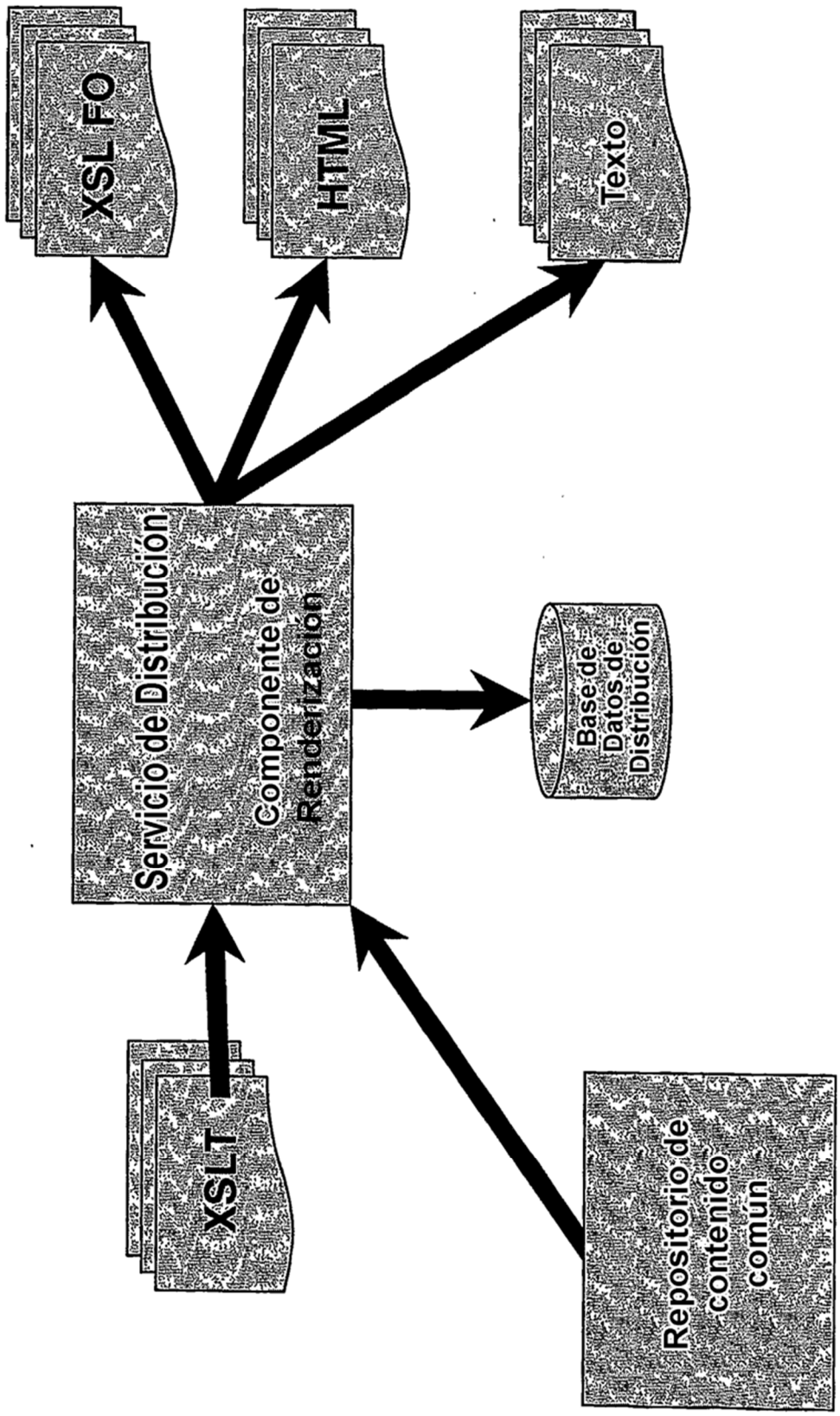

XSL FO
HTML
Texto
Servicio de Distribución
Componente de Renderización
Base de Datos de Distribución
XSLT
Repositorio de contenido común

Figura 18A

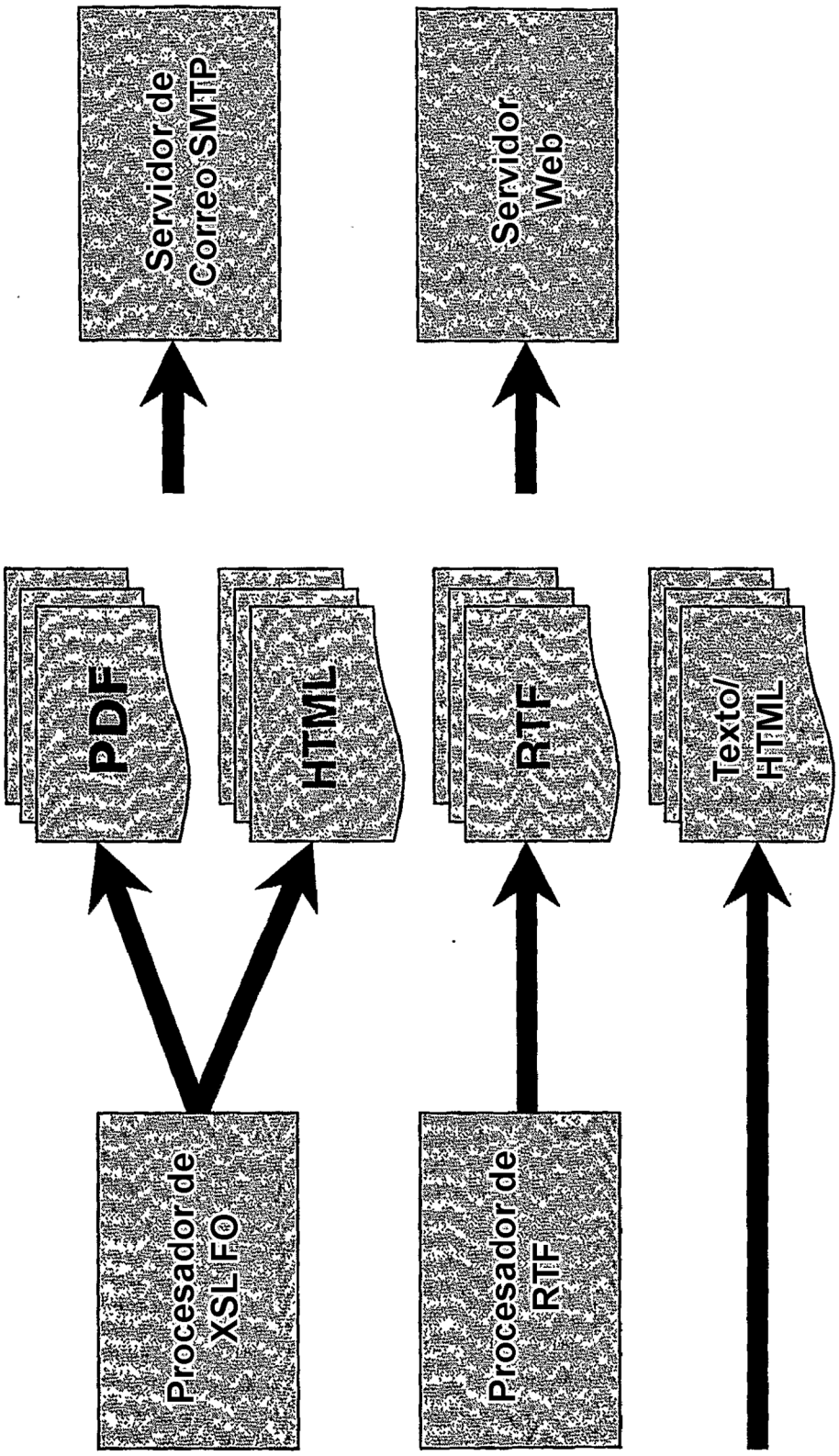

Servidor de Correo SMTP
Servidor Web
PDF
HTML
RTF
Texto/ HTML
Procesador de XSL FO
Procesador de RTF

Figura 18B

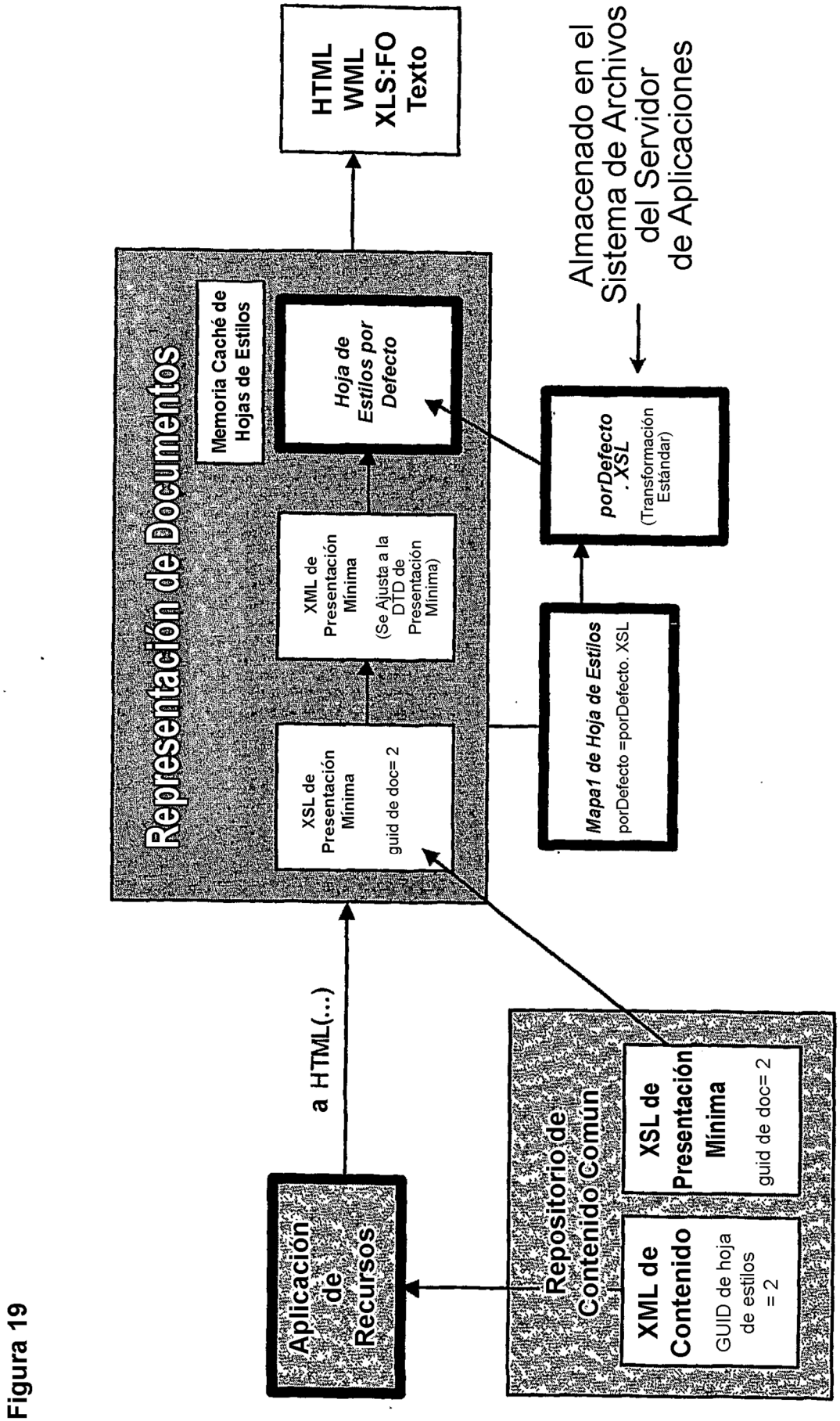

HTML
WML
XLS:FO
Texto
Almacenado en el Sistema de Archivos del Servidor de Aplicaciones
Representación de Documentos
Memoria Caché de Hojas de Estilos
Hoja de Estilos por Defecto
XML de Presentación Mínima
(Se Ajusta a la DTD de Presentación Mínima)
XSL de Presentación Mínima
guid de doc= 2
porDefecto .XSL
(Transformación Estándar)
Mapa1 de Hoja de Estilos
porDefecto =porDefecto. XSL
a HTML(...)
Aplicación de Recursos
Repositorio de Contenido Común
XML de Contenido
GUID de hoja de estilos = 2
XSL de Presentación Mínima
guid de doc= 2

Figura 19

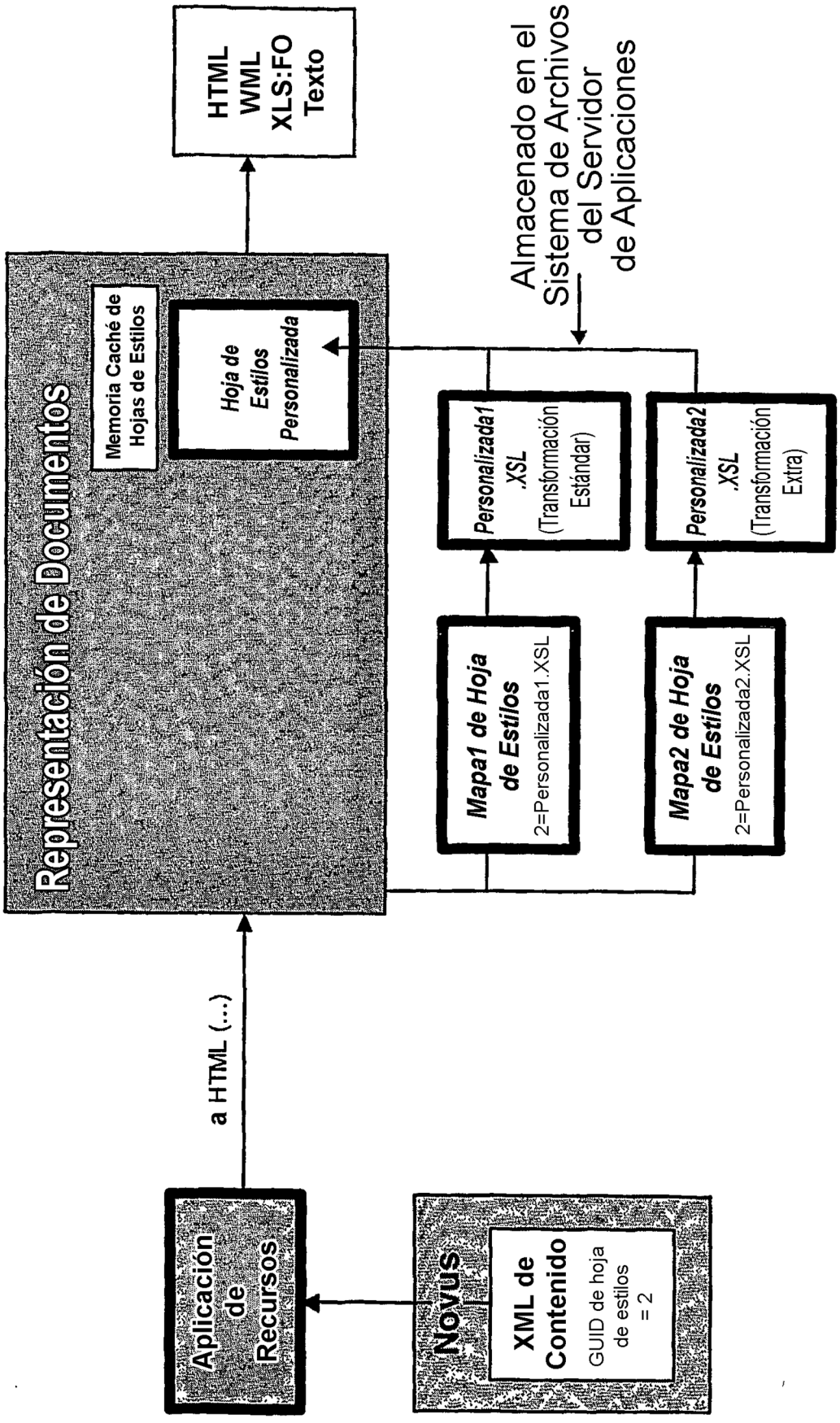

HTML
WML
XLS:FO
Texto
Almacenado en el Sistema de Archivos del Servidor de Aplicaciones
Representación de Documentos
Memoria Caché de Hojas de Estilos
Hoja de Estilos Personalizada
Personalizada1 .XSL (Transformación Estándar)
Personalizada2 .XSL (Transformación Extra)
Mapa1 de Hoja de Estilos 2=Personalizada1.XSL
Mapa2 de Hoja de Estilos 2=Personalizada2.XSL
a HTML (...)
Aplicación de Recursos
NOVUS
XML de Contenido
GUID de hoja de estilos = 2

Figura 20