

Señores
SUPERINTENDENCIA DE INDUSTRIA Y COMERCIO
DIVISIÓN DE NUEVAS CREACIONES
E. S. D.

SUPERINDUSTRIA Y COMERCIO
Radicación : 06019553 00000002 Folios: 161
Fecha (HH): 2006-03-08 12:22:29
Trámite : 002 PATENTES D I REGISTRO/ 229 COMPLEMEN
Dependencia: 2020 DIVISION DE NUEVAS CREACIONES

Ref.: Expediente No. 06-019.553
Solicitud de Registro para Patente de
Invención denominada MAPEO DE UN
MODELO DE SISTEMA DE ARCHIVO A UN
OBJETO DE UNA BASE DE DATOS.
Solicitante: Microsoft Corporation

Darío Cárdenas Navas identificado como aparece al ple de mi firma, me dirijo a ese Despacho para adjuntar los documentos restantes dentro de la solicitud de patente en referencia:

1. Copia auténtica de los documentos donde constan las cesiones del derecho a las patente reivindicadas otorgadas por los Inventores a favor del solicitante debidamente Apostilladas y con sus correspondientes traducciones oficiales.
2. Copia auténtica del documento de prioridad correspondiente a la patente de los Estados Unidos No. 11/091.079 de fecha Marzo 28, 2005.

Tal como consta en el Expediente el recibo oficial 06-13.596 de fecha Febrero 17, 2006 correspondiente a la reivindicación de prioridad fue presentado junto con la solicitud de patente radicada ante ese Despacho el día 28 de Febrero de 2006.

Habiendo dado cumplimiento a los requerimientos de forma establecidos por la Decisión 486 de la Comunidad Andina, atentamente solicito a ese Despacho proceda con la publicación de la solicitud de patente en referencia.

Atentamente,

Darío Cárdenas Navas.
c.c. 17.066.629 de Bogotá
t.p. 1.250 C.S.J.



TRADUCCIÓN OFICIAL

APOSTILLA

(Convención de La Haya, 5 de octubre de 1961)

1. País: **Estados Unidos de América**

El presente documento público:

2. Ha sido firmado por: **L. Edelen**
3. Actuando en calidad de: **Funcionario que Certifica**
4. Lleva el sello / estampilla de: **La Oficina de Patentes y Marcas Registradas/
Patent and Trademark Office**

CERTIFICADO

5. En: **Washington, D.C.**
6. El: **Veintisiete de Diciembre de 2005**
7. Por: **El Funcionario Asistente de Autenticaciones, Departamento de Estado de los Estados Unidos.**
8. Bajo el número: **06007538-4**
9. Lleva Sello / Estampilla: **Sello del DEPARTAMENTO DE ESTADO DE LOS ESTADOS UNIDOS DE AMÉRICA / United States Department of State**
10. Firmado por: **Sonya N. Johnson**



Martha S. Sarmiento S.
Martha S. Sarmiento S.
 Traductora Oficial Juramentada
 Resolución No. 835 Minjusticia

Traducción Oficial

LOS ESTADOS UNIDOS DE AMÉRICA

A TODOS AQUELLOS QUE LES PUEDA INTERESAR

**EL DEPARTAMENTO DE COMERCIO DE LOS ESTADOS UNIDOS
LA OFICINA DE PATENTES Y MARCAS REGISTRADAS DE LOS ESTADOS UNIDOS**

21 de Diciembre de 2005

**CON ÉSTE CERTIFICAMOS QUE EL DOCUMENTO ANEXO ES COPIA
VERDADERA DE LOS REGISTROS Y ANOTACIONES REALIZADOS EN ESTA
OFICINA, DEL DOCUMENTO REGISTRADO EL 25 DE AGOSTO DE 2005.**

Autorizado por el

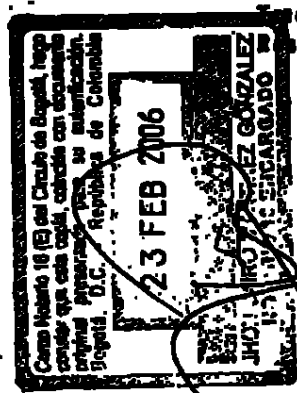
**SECRETARIO DE COMERCIO PARA PROPIEDAD INTELECTUAL Y DIRECTOR
DE LA OFICINA DE PATENTES Y MARCAS REGISTRADAS DE LOS ESTADOS
UNIDOS.**

**Firma: L. EDELEN
Funcionario Autorizado para Certificar**

**Sello Dorado: DEPARTAMENTO DE COMERCIO
OFICINA DE PATENTES Y MARCAS REGISTRADAS DE LOS ESTADOS UNIDOS**



Martha S. Sarmiento S.
Martha S. Sarmiento S.



Productora Oficial Juramentada
Bogotá No. 835 Minjuncin

TRADUCCIÓN OFICIAL

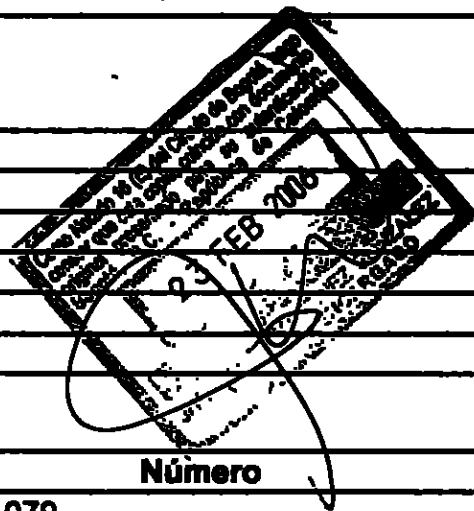
TRASPASO DE PATENTE

Versión Electrónica v1.1
Versión Impresa en Papel v1.1

TIPO DE SOLICITUD:	NUEVO TRASPASO
NATURALEZA DEL ACUERDO:	TRASPASO

DATOS DEL CEDENTE	
Nombre	Fecha de Ejecución
Nigel R. Ellis	03/24/2005
Gregory S. Friedman	03/24/2005
Bekin Demiroski	03/24/2005
Anil Kumar Nori	03/24/2005
Amit Shukla	03/24/2005
Srinivasmurthy P. Acharya	03/24/2005
Jeffrey T. Pearce	03/24/2005
Michael J. Newman	03/24/2005
Jason T. Hunter	03/24/2005

DATOS DEL CESIONARIO	
Nombre:	Microsoft Corporation
Dirección:	One Microsoft Way
Ciudad :	Redmond
Estado/ País:	WASHINGTON
Código Postal:	98052



NÚMERO Total de PROPIEDADES: 1	
Tipo de Propiedad	Número
Número de Solicitud:	11091079

500046958

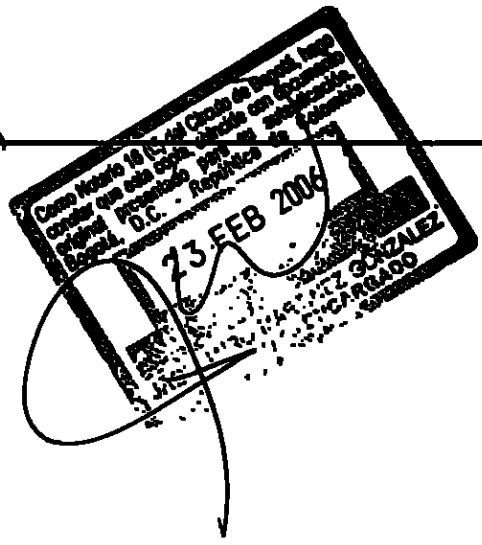
OP \$40.00 11091079

PATENTE
ROLL : 016451 CUADR : 0719

4

DATOS PARA CORRESPONDENCIA	
Número de Fax:	(216) 696-8731
<i>La correspondencia será enviada por correo de Estados Unidos cuando el envío por fax es infructuoso</i>	
Teléfono	216-696-8730
E-mail:	volszewski@thepatentattorneys.com
Nombre para correspondencia:	Himanshu S. Amin
Dirección Línea 1:	24th Floor, 1900 E. 9th Street
Dirección Línea 4:	Cleveland, OHIO 44114

NOMBRE DEL SOLICITANTE:	Himanshu S. Amin
Total Anexos: 2	
Fuente = MS311235.01 Traspaso #página1.tif	
Fuente = MS311235.01 Traspaso #página2.tif	



PATENTE
ROLLO: 016451 CUADRO: 0720

5

Traducción Oficial

Expediente MS No. 311235.01
(Expediente OC No. MSFTP652U6)

TRASPASO DE PATENTE

Nosotros Nigel R. Ellis, Gregory S. Friedman, Bekin Demiroski, Anil Kumar Nori, Amit Shukla, Srinivasmurthy P. Acharya, Jeffrey T. Pearce, Michael J. Newman y Jason T. Hunter ("Los CEDENTES") quienes hemos creado un invento ("El INVENTO") revelado y/o reivindicado en la solicitud de Registro de Patente denominada "MAPPING OF A FILE SYSTEM MODEL TO A DATABASE OBJECT/ MAPEADO DE UN MODELO DE SISTEMA DE ARCHIVOS A UN OBJETO DE BASE DE DATOS" ("LA SOLICITUD ") que:

☒ Será radicada sin ejecutar este TRASPASO DE PATENTE. El CEDENTE autoriza y solicita la representación legal del CESIONARIO denominado Law Offices/ las Oficinas legales de Amin and Torucy, LLC, ubicadas en el 1900 E, 9th Street, Cleveland, Ohio 44114, empresa que se encuentra registrada bajo el número de cliente 27195, para radicar aquí, en paréntesis, el (No. Serial U.S. 11/091,079, registrado el 3/28/05), se incluyen, cuando se conocen, este número serial para la Solicitud de patente en los Estados Unidos y la fecha de Radicación.

- ☐ Fue radicada el _____ y se le asignó el número serial _____;
- ☐ Se radica simultáneamente con este traspaso;

Microsoft Corporation, ("EL CESIONARIO"), una Corporación con sede en el Estado de Washington, en su nombre y en el de sus sucesores y cesionarios tiene derecho a y esta interesada en adquirir en su totalidad los derechos exclusivos, el título y los intereses del INVENTO y de la SOLICITUD DE PATENTE (y de todas las solicitudes y patentes derivadas de esta, como las solicitudes de continuidad y renovación, solicitadas en y para los Estados Unidos, sus territorios y para todos los países extranjeros (DERIVADOS DEL REGISTRO DE PATENTE));

— 6 —

Por consiguiente, como resultado de esta negociación, el CEDENTE da reconocimiento al recibo fruto de esta transacción. En este traspaso vende, asigna y transfiere al CESIONARIO la totalidad de los derechos exclusivos, título e intereses relacionados con este INVENTO, y con la solicitud de registro de patente (y los DERIVADOS DEL REGISTRO DE PATENTE);

El CEDENTE se compromete a ejecutar todos los instrumentos y documentos requeridos para diligenciar y procesar la SOLICITUD DE REGISTRO DE PATENTE, (y los DERIVADOS DEL REGISTRO DE PATENTE), así como para litigación relacionada con dicho REGISTRO DE PATENTE, o con el propósito de proteger el nombre de la SOLICITUD DE REGISTRO DE PATENTE (y los DERIVADOS DEL REGISTRO DE PATENTE).

03/24/2005
Fecha

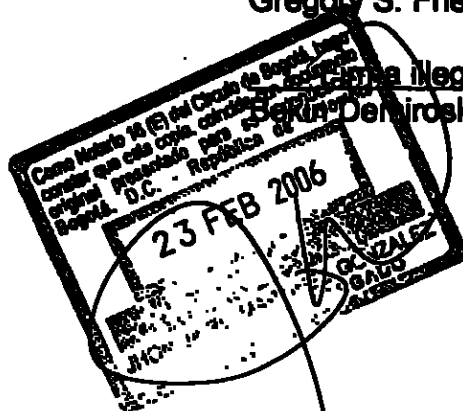
Firma ilegible
Nigel R. Ellis

03/24/2005
Fecha

Firma ilegible
Gregory S. Friedman

03/24/2005
Fecha

Firma ilegible
Berin Demiroski



PATENTE
ROLLO: 016461 CUADRO: 0721

7

Expediente MS No. 311238.01
(Expediente OC No. MSFTP882US)

03/24/2005
Fecha

Firma ilegible
Anil Kumar Nori

03/24/2005
Fecha

Firma ilegible
Amit Shukla

03/24/2005
Fecha

Firma ilegible
Srinivasmurthy P. Acharya

03/24/2005
Fecha

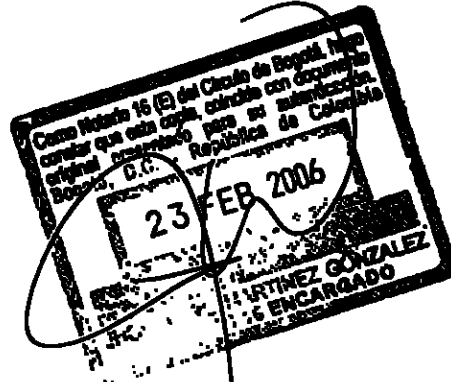
Firma ilegible
Jeffrey T. Pearce

03/24/2005
Fecha

Firma ilegible
Michael J. Newman

03/24/2005
Fecha

Firma ilegible
Jason T. Hunter



REGISTRADO: 08/25/2005

PATENTE
ROLL : 016451 CUADR : 0722

CONSTANCIA DE TRADUCCIÓN OFICIAL

Yo, Martha Sofía Sarmiento Solano, identificada con Cédula de Ciudadanía No. 41.686.291, de Bogotá, **HAGO CONSTAR** que el presente documento es versión fiel y completa en idioma Español de los siguientes documentos:

- Apostilla No. 06007538- 4, del 27 de Diciembre de 2005 y firmada por Sonya N. Johnson, Funcionario Asistente de Autenticaciones del Departamento de Estado de Los Estados Unidos.
- Certificado del Departamento de Comercio de los Estados Unidos firmado por: L. EDELEN, Funcionario Autorizado para Certificar, con fecha 21 de Diciembre de 2005.
- Traspaso de Patente identificado con el Número de Solicitud 11091079, y los números de ROLLO/CUADRO: 016451/0719 y 016451/0720.
- Traspaso de Patente firmado por Nigel R. Ellis, Gregory S. Friedman, Bekin Demiroski, Anil Kumar Nori, Amit Shukla, Srinivasmurthy P. Acharya, Jeffrey T. Pearce, Michael J. Newman y Jason T. Hunter. Registrado el 25 de Agosto de 2005.

Martha S. Sarmiento

Martha S. Sarmiento S.

Traductora Oficial

Resolución Minjusticia 835



Martha S. Sarmiento S.

Traductora Oficial Juramentada

Resolución No. 835 Minjusticia

APOSTILLE*(Convention de La Haye du 5 octobre 1961)*

1. Country: United States of America

This public document

2. has been signed by L. Edelen

3. acting in the capacity of Certifying Officer

4. bears the seal/stamp of Patent and Trademark Office

Certified

5. at Washington, D.C.

6. the twenty-seventh of December, 2005

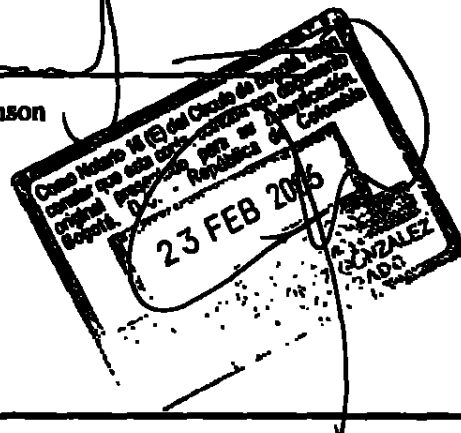
7. by Assistant Authentication Officer, United States Department of State

8. No. 06007538-4

9. Seal/Stamp:

10. Signature:

Sonya N. Johnson



to

A 2406782



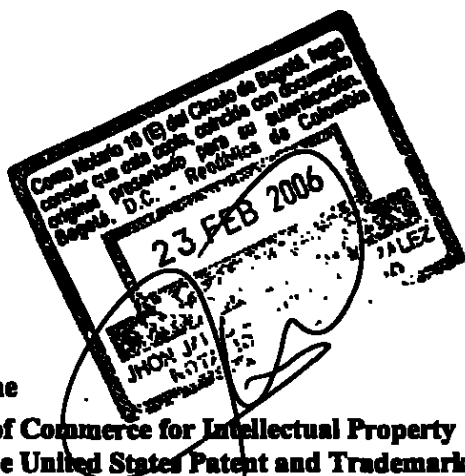
THE UNITED STATES OF AMERICA

TO ALL TO WHOM THESE PRESENTS SHALL COME:

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office

December 21, 2005

THIS IS TO CERTIFY THAT ANNEXED IS A TRUE COPY FROM THE
RECORDS OF THIS OFFICE OF A DOCUMENT RECORDED ON
August 25, 2005.



By Authority of the
Under Secretary of Commerce for Intellectual Property
and Director of the United States Patent and Trademark Office

L. Edelen

L. EDELEN
Certifying Officer

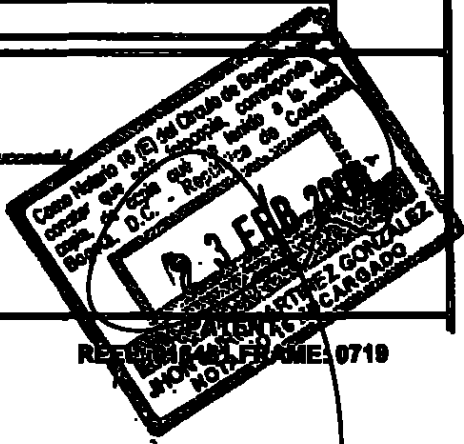


PATENT ASSIGNMENT

Electronic Version v1.1
Stylesheet Version v1.1

SUBMISSION TYPE:	NEW ASSIGNMENT																				
NATURE OF CONVEYANCE:	ASSIGNMENT																				
CONVEYING PARTY DATA																					
<table><thead><tr><th>Name</th><th>Execution Date</th></tr></thead><tbody><tr><td>Nigal R. Ellis</td><td>03/24/2006</td></tr><tr><td>Gregory S. Friedman</td><td>03/24/2006</td></tr><tr><td>Dustin Demiroski</td><td>03/24/2006</td></tr><tr><td>Anil Kumar Neri</td><td>03/24/2006</td></tr><tr><td>Anil Shukla</td><td>03/24/2006</td></tr><tr><td>Brinivemurthy P. Acharya</td><td>03/24/2006</td></tr><tr><td>Jeffrey T. Pearce</td><td>03/24/2006</td></tr><tr><td>Michael J. Newman</td><td>03/24/2006</td></tr><tr><td>Jason T. Hunter</td><td>03/24/2006</td></tr></tbody></table>		Name	Execution Date	Nigal R. Ellis	03/24/2006	Gregory S. Friedman	03/24/2006	Dustin Demiroski	03/24/2006	Anil Kumar Neri	03/24/2006	Anil Shukla	03/24/2006	Brinivemurthy P. Acharya	03/24/2006	Jeffrey T. Pearce	03/24/2006	Michael J. Newman	03/24/2006	Jason T. Hunter	03/24/2006
Name	Execution Date																				
Nigal R. Ellis	03/24/2006																				
Gregory S. Friedman	03/24/2006																				
Dustin Demiroski	03/24/2006																				
Anil Kumar Neri	03/24/2006																				
Anil Shukla	03/24/2006																				
Brinivemurthy P. Acharya	03/24/2006																				
Jeffrey T. Pearce	03/24/2006																				
Michael J. Newman	03/24/2006																				
Jason T. Hunter	03/24/2006																				
RECEIVING PARTY DATA																					
<table><tr><td>Name:</td><td>Microsoft Corporation</td></tr><tr><td>Street Address:</td><td>One Microsoft Way</td></tr><tr><td>City:</td><td>Redmond</td></tr><tr><td>State/Country:</td><td>WASHINGTON</td></tr><tr><td>Postal Code:</td><td>98052</td></tr></table>		Name:	Microsoft Corporation	Street Address:	One Microsoft Way	City:	Redmond	State/Country:	WASHINGTON	Postal Code:	98052										
Name:	Microsoft Corporation																				
Street Address:	One Microsoft Way																				
City:	Redmond																				
State/Country:	WASHINGTON																				
Postal Code:	98052																				
PROPERTY NUMBERS Total: 1																					
<table><thead><tr><th>Property Type</th><th>Number</th></tr></thead><tbody><tr><td>Application Number:</td><td>11061079</td></tr></tbody></table>		Property Type	Number	Application Number:	11061079																
Property Type	Number																				
Application Number:	11061079																				
CORRESPONDENCE DATA																					
Fax Number: (216)886-8731																					
Correspondence will be sent via US Mail when the fax attempt is unsuccessful																					
Phone: 216-886-8730																					
Email: volzseweld@thepatentattorneys.com																					
Correspondent Name: Himanshu S. Amin																					
Address Line 1: 24th Floor, 1900 E. 8th Street																					
Address Line 4: Cleveland, OHIO 44114																					

500048958



+25

NAME OF SUBMITTER:	Himanshu S. Amin
Total Attachments: 2 source=MS311236.01.AssignmentPage1.tif source=MS311236.01.AssignmentPage2.tif	



15

PATENT ASSIGNMENT

We, Nigel R. Ellis, Gregory S. Friedman, Bekim Demiroski, Anil Kumar Nori, Amit Shukla, Srinivasamurthy P. Acharya, Jeffrey T. Pearce, Michael J. Newman, and Jason T. Hunter (ASSIGNORS), have invented subject matter ("INVENTION") disclosed and/or claimed in a patent application entitled "MAPPING OF A FILE SYSTEM MODEL TO A DATABASE OBJECT" ("APPLICATION"), which:

- ☒ will be filed ~~without~~ this executed PATENT ASSIGNMENT. ASSIGNORS hereby authorize and request, ASSIGNEE'S legal representatives, of Amin and Turocy, LLC, 1900 E. 9th Street, Cleveland, Ohio 44114, who are associated with customer number 27195, to insert here in parenthesis (U.S. Serial No. 11/081,078, filed 3/28/05) this APPLICATION's U.S. Serial Number and filing date, when known;
- ☐ was filed on and was given U.S. Serial No.;
- ☐ is filed concurrently herewith;

Microsoft Corporation, a Washington Corporation, on behalf of itself and its successors and assigns (ASSIGNEE), is entitled to, and is desirous of acquiring, the entire and exclusive rights, title and interest in the INVENTION and the APPLICATION (and all other applications and patents derived therefrom, such as continuing applications, in and for the United States, its territories, and all foreign countries ("APPLICATION DERIVATIVES"));

For good and valuable consideration, the receipt of which is hereby acknowledged by the ASSIGNORS, the ASSIGNORS hereby sell, assign and transfer to the ASSIGNEE, the entire and exclusive rights, title and interest in the INVENTION and the APPLICATION (and APPLICATION DERIVATIVES);

ASSIGNORS agree to execute all instruments and documents required for the making and prosecution of the APPLICATION (and APPLICATION DERIVATIVES), for litigation regarding letters patent derived therefrom, and for the purpose of protecting and perfecting title to the APPLICATION (and APPLICATION DERIVATIVES).

3/24/2005
 DATE

3/24/2005
 DATE

3/24/2005
 DATE

Nigel R. Ellis
 Nigel R. Ellis

Gregory S. Friedman
 Gregory S. Friedman

Bekim Demiroski
 Bekim Demiroski



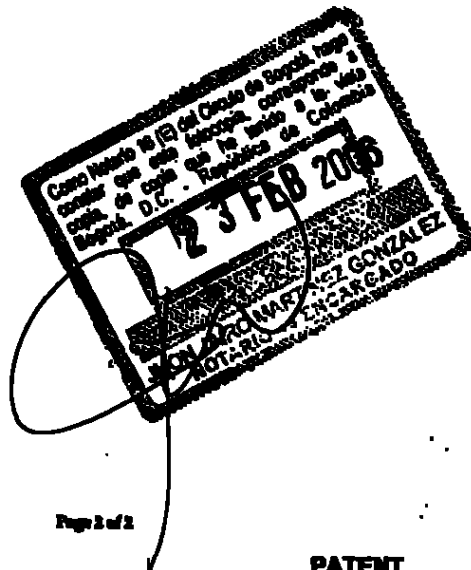
PATENT
 REEL: 016461 FRAME: 0721

T

MS Docket No. MSB11225-01
(DC Docket No. MSB1763215)

3/24/2005
DATE
3/24/2005
DATE
3/24/2005
DATE
3/24/2005
DATE
3/24/05
DATE
3/24/05
DATE

Anil Kumar Nori
Anil Kumar Nori
Amit Shukla
Amit Shukla
Srinivasamurthy P. Acharya
Srinivasamurthy P. Acharya
Jeffrey T. Pearce
Jeffrey T. Pearce
Michael J. Neumann
Michael J. Neumann
Jason T. Hunter
Jason T. Hunter



Page 3 of 2

RECORDED: 08/26/2006

PATENT
REEL: 016461 FRAME: 0722

15/2

Traducción Oficial**LOS ESTADOS UNIDOS DE AMÉRICA
A TODOS AQUELLOS QUE LES PUEDA INTERESAR****DEPARTAMENTO DE COMERCIO DE LOS ESTADOS UNIDOS
OFICINA DE PATENTES Y MARCAS REGISTRADAS DE LOS ESTADOS UNIDOS****22 de Diciembre de 2005**

CON ESTE CERTIFICAMOS QUE EL DOCUMENTO ANEXO ES COPIA VERDADERA TOMADA DE LOS REGISTROS DE LA OFICINA DE PATENTES Y MARCAS REGISTRADAS DE LOS ESTADOS UNIDOS DE LOS DOCUMENTOS DE LA SOLICITUD DE PATENTE ABAJO IDENTIFICADOS, QUE CUMPLIERON LOS REQUISITOS PARA QUE SE LES OTORGARA UNA FECHA DE RADICACIÓN BAJO EL 35 DEL USC 111.

NÚMERO DE SOLICITUD: 11/091,079**FECHA DE RADICACIÓN: 28 de Marzo de 2005**

EL CÓDIGO DEL PAÍS Y EL NÚMERO DE SOLICITUD DE PRIORIDAD QUE SE DEBE UTILIZAR EN SOLICITUDES EN OTROS PAÍSES BAJO LA CONVENCIÓN DE PARIS, ES **US 11/091,079**.

Autorizado por el

**SECRETARIO DE COMERCIO PARA PROPIEDAD INTELECTUAL, DIRECTOR
DE LA OFICINA DE PATENTES Y MARCAS REGISTRADAS DE LOS ESTADOS
UNIDOS.**

**Firma: E. BORNETT****Funcionario Autorizado para Certificar****Sello Dorado: DEPARTAMENTO DE COMERCIO****OFICINA DE PATENTES Y MARCAS REGISTRADAS DE LOS ESTADOS UNIDOS**

Handwritten signature or mark.

Traducción Oficial

PT050374-05

Acetabular foramen: 1975-1976, 1978-1979, 1981-1982, 1984-1985, 1987-1988, 1990-1991, 1993-1994, 1996-1997, 1999-2000, 2002-2003, 2005-2006, 2008-2009, 2011-2012, 2014-2015, 2017-2018, 2020-2021, 2023-2024, 2026-2027, 2029-2030, 2032-2033, 2035-2036, 2038-2039, 2041-2042, 2044-2045, 2047-2048, 2050-2051, 2053-2054, 2056-2057, 2059-2060, 2062-2063, 2065-2066, 2068-2069, 2071-2072, 2074-2075, 2077-2078, 2080-2081, 2083-2084, 2086-2087, 2089-2090, 2092-2093, 2095-2096, 2098-2099, 2101-2102, 2104-2105, 2107-2108, 2110-2111, 2113-2114, 2116-2117, 2119-2120, 2122-2123, 2125-2126, 2128-2129, 2131-2132, 2134-2135, 2137-2138, 2140-2141, 2143-2144, 2146-2147, 2149-2150, 2152-2153, 2155-2156, 2158-2159, 2161-2162, 2164-2165, 2167-2168, 2170-2171, 2173-2174, 2176-2177, 2179-2180, 2182-2183, 2185-2186, 2188-2189, 2191-2192, 2194-2195, 2197-2198, 2199-2200, 2202-2203, 2205-2206, 2208-2209, 2211-2212, 2214-2215, 2217-2218, 2220-2221, 2223-2224, 2226-2227, 2229-2230, 2232-2233, 2235-2236, 2238-2239, 2241-2242, 2244-2245, 2247-2248, 2250-2251, 2253-2254, 2256-2257, 2259-2260, 2262-2263, 2265-2266, 2268-2269, 2271-2272, 2274-2275, 2277-2278, 2280-2281, 2283-2284, 2286-2287, 2289-2290, 2292-2293, 2295-2296, 2298-2299, 2301-2302, 2304-2305, 2307-2308, 2310-2311, 2313-2314, 2316-2317, 2319-2320, 2322-2323, 2325-2326, 2328-2329, 2331-2332, 2334-2335, 2337-2338, 2340-2341, 2343-2344, 2346-2347, 2349-2350, 2352-2353, 2355-2356, 2358-2359, 2361-2362, 2364-2365, 2367-2368, 2370-2371, 2373-2374, 2376-2377, 2379-2380, 2382-2383, 2385-2386, 2388-2389, 2391-2392, 2394-2395, 2397-2398, 2399-2400, 2402-2403, 2405-2406, 2408-2409, 2411-2412, 2414-2415, 2417-2418, 2420-2421, 2423-2424, 2426-2427, 2429-2430, 2432-2433, 2435-2436, 2438-2439, 2441-2442, 2444-2445, 2447-2448, 2450-2451, 2453-2454, 2456-2457, 2459-2460, 2462-2463, 2465-2466, 2468-2469, 2471-2472, 2474-2475, 2477-2478, 2480-2481, 2483-2484, 2486-2487, 2489-2490, 2492-2493, 2495-2496, 2498-2499, 2501-2502, 2504-2505, 2507-2508, 2510-2511, 2513-2514, 2516-2517, 2519-2520, 2522-2523, 2525-2526, 2528-2529, 2531-2532, 2534-2535, 2537-2538, 2540-2541, 2543-2544, 2546-2547, 2549-2550, 2552-2553, 2555-2556, 2558-2559, 2561-2562, 2564-2565, 2567-2568, 2570-2571, 2573-2574, 2576-2577, 2579-2580, 2582-2583, 2585-2586, 2588-2589, 2591-2592, 2594-2595, 2597-2598, 2599-2600, 2602-2603, 2605-2606, 2608-2609, 2611-2612, 2614-2615, 2617-2618, 2620-2621, 2623-2624, 2626-2627, 2629-2630, 2632-2633, 2635-2636, 2638-2639, 2641-2642, 2644-2645, 2647-2648, 2650-2651, 2653-2654, 2656-2657, 2659-2660, 2662-2663, 2665-2666, 2668-2669, 2671-2672, 2674-2675, 2677-2678, 2680-2681, 2683-2684, 2686-2687, 2689-2690, 2692-2693, 2695-2696, 2698-2699, 2701-2702, 2704-2705, 2707-2708, 2710-2711, 2713-2714, 2716-2717, 2719-2720, 2722-2723, 2725-2726, 2728-2729, 2731-2732, 2734-2735, 2737-2738, 2740-2741, 2743-2744, 2746-2747, 2749-2750, 2752-2753, 2755-2756, 2758-2759, 2761-2762, 2764-2765, 2767-2768, 2770-2771, 2773-2774, 2776-2777, 2779-2780, 2782-2783, 2785-2786, 2788-2789, 2791-2792, 2794-2795, 2797-2798, 2799-2800, 2802-2803, 2805-2806, 2808-2809, 2811-2812, 2814-2815, 2817-2818, 2820-2821, 2823-2824, 2826-2827, 2829-2830, 2832-2833, 2835-2836, 2838-2839, 2841-2842, 2844-2845, 2847-2848, 2850-2851, 2853-2854, 2856-2857, 2859-2860, 2862-2863, 2865-2866, 2868-2869, 2871-2872, 2874-2875, 2877-2878, 2880-2881, 2883-2884, 2886-2887, 2889-2890, 2892-2893, 2895-2896, 2898-2899, 2901-2902, 2904-2905, 2907-2908, 2910-2911, 2913-2914, 2916-2917, 2919-2920, 2922-2923, 2925-2926, 2928-2929, 2931-2932, 2934-2935, 2937-2938, 2940-2941, 2943-2944, 2946-2947, 2949-2950, 2952-2953, 2955-2956, 2958-2959, 2961-2962, 2964-2965, 2967-2968, 2970-2971, 2973-2974, 2976-2977, 2979-2980, 2982-2983, 2985-2986, 2988-2989, 2991-2992, 2994-2995, 2997-2998, 2999-3000, 3002-3003, 3005-3006, 3008-3009, 3011-3012, 3014-3015, 3017-3018, 3020-3021, 3023-3024, 3026-3027, 3029-3030, 3032-3033, 3035-3036, 3038-3039, 3041-3042, 3044-3045, 3047-3048, 3050-3051, 3053-3054, 3056-3057, 3059-3060, 3062-3063, 3065-3066, 3068-3069, 3071-3072, 3074-3075, 3077-3078, 3080-3081, 3083-

Oficina de Patentes y Marcas Registradas de los Estados Unidos. DEPARTAMENTO DE COMERCIO DE LOS ESTADOS UNIDOS

Para la Administración de Recaudación de Tráfico de 1995, la mayor proporción está asociada con la cantidad de vehículos, pero el dato sugiere un régimen de control sobre EMB.

TRANSMISIÓN DE SOLICITUD DE PATENTE DE UTILIDAD <i>(Exclusivamente para solicitudes nuevas no provisionales hechas el 31 del CFR 1.83 (a))</i>		Expediente de Abogado No.		M8311235.01/M8FTP05	
		Primer Inventor		Nigel R. Ellis, et al.	
		Nombre	MAPPING OF A FILE SYSTEM MODEL TO A DATABASE OBJECT MAPPEADO DE UN MODELO DE SISTEMA DE ARCHIVOS A UN OBJETO DE BASE DE DATOS		
		Etiqueta de Correo Externo No.		EV628534781U8	

ELEMENTOS DE LA SOLICITUD				DIRIGIR A:	
Ver MPEP Capítulo 600 en relación con las instrucciones de solicitudes de patentes de utilidad				P.O. Box 1400 Alexandria, VA 22313-1400	
1	☒	Forma del formato de información (ver ejemplo PTO/GB-117) (suministrado original y copia para liquidación de la Tasa)		7.	☐ CD-ROM o CD-R en duplicado, tableta gráfica o programa de computador (Anexo)
2	☐	El solicitante reiniciará estados de empresa pequeña Ver el 37 del CFR 1.27		8.	Presentación de secuencia de nucleótidos y/o aminoácidos (si aplica, todo a tresceros)
3	☒	Especificaciones (El orden preferido se señala a continuación) - Título descriptivo del invento - Referencias cruzadas para las solicitudes relacionadas - Descripción en relación con prior art Federal R&D - Referencia de la lista de secuencias, tableta o un anexo con lista en computadora. - Antecedentes del invento. - Breve Resumen del invento - Breve Descripción de los Ejemplos (si se adjuntó). - Descripción detallada. - Revisiones/claims. - Resumen de declaración de Información Confidencial. Total Páginas 35		a.	☐ Formato Legible en Computador (FLC)
				b.	Lista de secuencia de las especificaciones etc.
				i.	☐ CD-ROM o CD-R (2 copias) o
				ii.	☐ papel.
				c.	☐ Declaración verificando la veracidad de los datos
CONTENIDO QUE ACOMPAÑA LA SOLICITUD					
4	☒	Obligato (35 del U.S.C. 113) (Total Páginas 12)		9.	☐ Documentos de trabajo (hoja de presentación y documentación)
5		Declaración o juramento. (Total Páginas 3)		10.	☐ Declaración de acuerdo con el 37 del CFR 3.73(a) ☒ Poder de Abogado (suministrado en un copias)
	a.	☒ Realizada Reservación (original o copia)		11.	☐ Documento de traducción si aplica (si aplica)
	b.	☐ Copia de una solicitud anterior según el 37 del CFR 1.63 (a) (para solicitud de continuación odividida con el mismo o diferente invento)		12.	☐ Declaración de Información Confidencial. ☐ Copias de Clam IDB (PTO-1448) Modificación preliminar
	c.	☐ DECLARACIÓN DE INVENTORES Declaración firmada o firmada inventores) designado(s) en la solicitud anterior. Ver el 37 del CFR 1.63 (a)(2) y 1.63 (a).		13.	☒ Tarjeta para constancia de recibo de correo. (MPEP 603) (Debe ser sellada específicamente)
6	☒	Hoja de datos de la solicitud. Ver el 37 del CFR 1.76		14.	☐ Copia del Certificado del documento de prioridad (si se retiene prioridad de extranjero)
15	Si es una SOLICITUD DE CONTINUACIÓN marque la casilla apropiada y suministre la información solicitada a continuación y en la primera página de la solicitud, en la parte superior de la hoja de datos para solicitud bajo el 37 del CFR 1.76. ☐ Continuación ☐ División ☐ Continuación parcial (CIP)				
(Información de la solicitud priora: Examinador Control Grupo de División) Unicamente para SOLICITUDES DE CONTINUACIÓN O DIVISIÓN: La divulgación completa de la primera solicitud, de la cual se deriva la presente, debe ser incluida en la solicitud de continuación o división de la cual se deriva la presente. Si se adjuntan como parte de los documentos divulgados que corresponden a la solicitud de continuación o división de la cual se deriva la presente, se adjuntará como parte de la presente una copia de los documentos de la primera solicitud que corresponden a la presente.					
16. DIRECCIÓN PARA CORRESPONDENCIA					
☒ Número de cliente 27185 ☐ La					
Nombre	Amin & Turetzky, LLP				
Dirección	1800 E. 9 th Street National City Center, 24th Floor				
Ciudad	Cleveland	Estado	Ohio	Código Postal	44114
País	Estados Unidos	Teléfono	(216) 998-6730	Fax	
Nombre	Himanshu G. Amin		Registro (Abogado/Agencia) No.		30 50
Firma	Firma Real		Fecha		20 de Marzo de 2008

[illegible]

El reciente aumento del desempleo exige tomar medidas. Llame al 1-800-FTD-8188 y solicite la opción 3.

20427 U.S. PTO
CÓDIGO DE BARRAS
732015

113211 U.S. FTO
11/08/07
CODIGO DE BARRAS
032805

Copia Suministrada por la USPTO de la Base de datos de Imagen IFW el 12/19/2005

77

Traducción Oficial

PTO/BB/17(12-04)

Aprobado para uso hasta el 07/31/2008 OMB 0951-0032

Oficina de Patentes y Marcas Registradas, DEPARTAMENTO DE COMERCIO DE LOS ESTADOS UNIDOS

Esta es la Sumatoria de Pagos de Tarifas de 1995. Ninguna cuota será requerida a presentar a una solicitud de información solo si esta contiene un número de control válido (CIP)

Vigencia desde 1/30/2004

Las tasas se basan según el acto legislativo
Consolidated Appropriations Act, del 2006 (P.L.A. 481a)

Declaración de Gastos
para el Año Fiscal 2005

☐ El Solicitante declara status de Pequeña Empresa ver el 37 del CFR 1.37

Valor Total de pago (\$) 1,000.00

Complete si conoce la información

Número de Solicitud	
Fecha de presentación	Aquí Incluida
Nombre del primer inventor	Nigel R. Ellis, et al.
Nombre del Examinador	
Unidad de Diseño	
Expediente de Abogado No.	MS311235.01/MSFTP852US

Método de Pago (Marque todas las casillas que aplican)

☐ Cheques ☐ Tarjeta de crédito ☐ Giro ☐ Ninguno ☐ Otro (por favor especifique):

EDCuenta de Ahorros: Número de la cuenta de Ahorros 92-1082 Nombre del Titular de la Cuenta de Ahorros Amín & Turner, LLP

Teniendo en cuenta la cuenta de ahorros arriba identificada, se autoriza al director para: (Marque todas las casillas que aplican)

- ☐ Cobrar la(s) tasa(s) indicada(s) a continuación ☐ Cobrar la(s) tasa(s) indicadas con confirmación, excepto la Tasa de Radicación y Registro.
☐ Cobrar cualquier tasa(s) adicional(es) o cualquier saldo pendiente. ☐ Acreditar saldo a favor.
Confirme al 37 del CFR 1.16 Y 1.17.

ADVERTENCIA: La información sobre este formato podrá ser pública. No se debe incluir en este formato datos sobre la tarjeta de crédito. Dicha información y autorización de debe suministrar en el formato PTO-2038.

CÁLCULO DE LA TASA

1. TASA BÁSICA DE PRESENTACIÓN, BÚSQUEDA Y ESTUDIO O EXAMEN DE LA SOLICITUD

Tipo de Solicitud	TASA DE RADICACIÓN		TASA DE BÚSQUEDA		TASA DE EXAMEN		Tasa acumulada \$
	Pequeña Empresa	Empresa	Pequeña Empresa	Empresa	Pequeña Empresa	Empresa	
Utilidad	300	150	500	250	300	100	1,000
Diseños	200	100	100	50	130	65	0
Plantas	200	100	300	150	160	80	0
Re-emisión	300	150	500	250	600	300	0
Provisional	200	100	0	0	0	0	0

2. TASAS ADICIONALES POR RECLAMACIÓN

Descripción de la tasa

Cada reivindicación superior a 20, o por Reemisión, o cada reivindicación superior a 20 y superiores a las de la Patente Original.

Cada reivindicación independiente superior a 3 o, por Reemisión, o cada reivindicación independiente superior a la de la Patente Original.

Múltiples reivindicaciones dependientes

Total reivindicaciones	Reivindicaciones adicionales	Tasa \$	Tasa acumulada \$
20	20 o HP=	0	0

HP = El número más alto de reivindicaciones pagadas si son superiores a 20

Reivindicaciones independientes	Reivindicaciones adicionales	Tasa \$	Tasa acumulada \$
3	3 o HP=	0	0

HP = El número más alto de reivindicaciones independientes pagadas si son superiores a 3

3. TASA SEGÚN EL TAMAÑO DE LA SOLICITUD

Si la especificación y los dibujos exceden 100 hojas de papel, la suma a pagar de acuerdo con el tamaño de la solicitud para cada 50 hojas o fracción adicional de esta. Ver el 35 del U.S.C. 41 (a)(1)(X) y el 37 del CFR 1.16(a)

Total Hojas	Hojas adicionales	Número adicional de cada 50 hojas o fracción de esta	Tasa acumulada \$
47	-100 =	0	0

4. Otra(s) tasa(s)

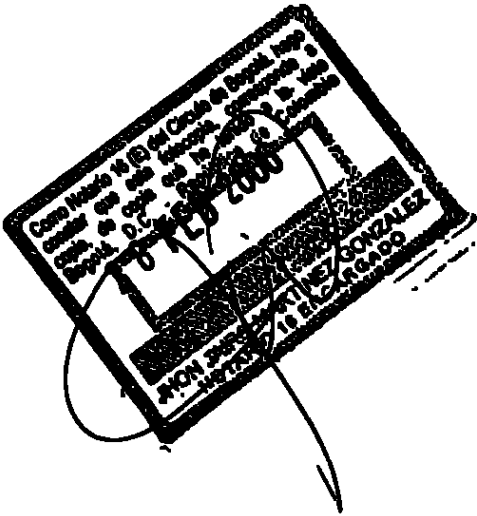
Especificaciones en idioma diferente al inglés. Valor de la tasa \$130 (no hay descuento para Pequeñas Empresas)

Otra:

Presentado por			
Nombre	Hiramah S. Arán	Número de Registro 40,884	Teléfono (216) 688-8730
		(Abogado/Agente)	
Firma	Firma legible		Fecha 26 de Marzo de 2003

La orden de recolección de esta información se encuentra estipulada en el 37 del CFR 136. Esta información es necesaria para obtener o mantener un beneficio público que consiste en recibir y registrar una solicitud para que pueda ser procesada por la oficina USPTO. Su confidencialidad se encuentra garantizada bajo el 35 del USC 122 y por el 37 del CFR 1.14. Se estima que este formato se completa en 30 minutos, incluyendo la recuperación, preparación y presentación de la solicitud totalmente diligenciada ante la oficina de la USPTO. El tiempo variará dependiendo de las necesidades de cada caso. Cualquier comentario sobre la cantidad de tiempo que usted utilice para completar este formato y/o sugerencias para reducirlo se debe enviar al Oficial en Jefe de Información de la Oficina de Patentes y Marcas Registradas de los Estados Unidos/ chief Information Officer, U.S. Patent and Trademark Office, U.S. Department of Commerce, P.O. Box 1480, Alexandria, VA 22313-1480. NO ENVIAR PAGOS POR CONCEPTO DE TASAS O FORMULARIOS DILIGENCIADOS A ESTA DIRECCIÓN. ENVELOPS A: Commissioner for Patents P.O. Box 1480, Alexandria, VA 22313-1480

Si necesita asesoría para diligenciar este formato, llame al 1-800 -- PTO -- 6188 y seleccione la opción 2



TRADUCCIÓN OFICIAL

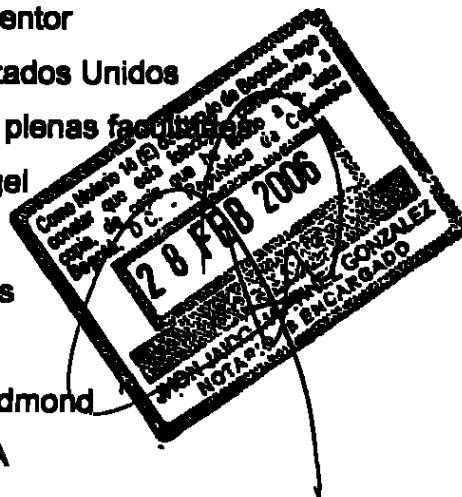
H ja d S licitud y Datos

Información de la Solicitud de Patente

Fecha de Radicación:	03/28/05
Tipo de Solicitud:	Regular
Asunto - Tema	Utilidad
Clasificación sugerida:	
Unidad de Arte sugerida:	
¿Formato Legible por computador (CRF)?:	No
Nombre:	MAPPING OF A FILE SYSTEM MODEL TO A DATABASE OBJECT/ MAPEADO DE UN MODELO DE SISTEMA DE ARCHIVOS A UN OBJETO DE BASE DE DATOS
Número de Expediente de Abogado:	MS311235.01/MSFTP852US
¿Solicitud de pronta publicación?	No
¿Solicitud de no Publicación?	Si
Diseño de Figura Sugerido:	1
Número Total de hojas con Diseños:	12
¿Empresa Pequeña?:	No
¿Se incluye petición?:	No
¿Solicitud de Patente Principal con Orden de Reserva?:	No

Información de los Solicitantes

Tipo de Especialidad del Solicitante:	Inventor
País de Origen:	Estados Unidos
Estatus:	En plenas facultades
Primer Nombre:	Nigel
Segundo Nombre:	R.
Apellido:	Ellis
Sufijo del Nombre:	
Ciudad de Residencia:	Redmond
Estado o Provincia de residencia:	WA



Fecha de Registro 03/28/05

20-

21

• Apellido:	Demiroski
Sufijo del Nombre	
Ciudad de Residencia:	Redmond
Estado o Provincia de residencia:	WA
País de Residencia:	Estados Unidos de América
Dirección de Correo:	6005 142 nd Court NE
Ciudad para envío de correo:	Redmond
Estado o provincia para correo:	WA
País para envío de correo:	Estados Unidos
Código Postal:	98052
Tipo de Especialidad del Solicitante:	Inventor
País de Origen:	Estados Unidos de América
Estatus:	En plenas facultades
Primer Nombre:	Anil
Segundo Nombre:	Kumar
Apellido:	Nori
Sufijo del Nombre	
Ciudad de Residencia:	Redmond
Estado o Provincia de residencia:	WA
País de Residencia:	Estados Unidos de América
Dirección de Correo:	7421 204 th Drive NE
Ciudad para envío de correo:	Redmond
Estado o provincia para correo:	WA
País para envío de correo:	Estados Unidos de América
Código Postal:	98053
Tipo de Especialidad del Solicitante:	Inventor
País de Origen:	India

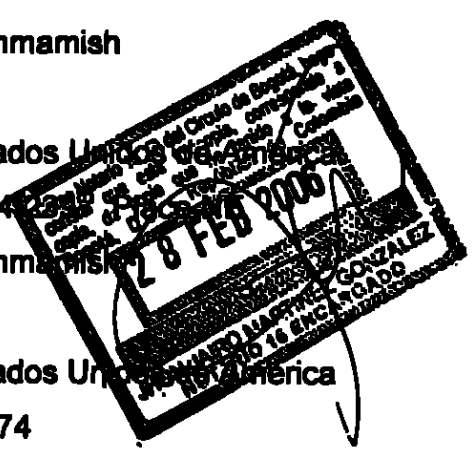
A rectangular stamp from the USPTO. The date "28 FEB 2005" is stamped in the center. Above the date, there is some small, mostly illegible text. Below the date, there is a signature that appears to read "JONAS J. GONZALEZ". To the right of the signature, there is a handwritten number "175".

Fecha de Registro 03/28/05

2

Estatús: En plenas facultades
Primer Nombre: Amit
Segundo Nombre:
Apellido: Shukla
Sufijo del Nombre
Ciudad de Residencia: Redmond
Estado o Provincia de residencia: WA
País de Residencia: Estados Unidos de América
Dirección de Correo: 10835 170th Court NE
Ciudad para envío de correo: Redmond
Estado o provincia para correo: WA
País para envío de correo: Estados Unidos de América
Código Postal: 98052

Tipo de Especialidad del Solicitante: Inventor
País de Origen: Estados Unidos de América
Estatus: En plenas facultades
Primer Nombre: Srinivasmurthy
Segundo Nombre: P.
Apellido: Acharya
Sufijo del Nombre
Ciudad de Residencia: Sammamish
Estado o Provincia de residencia: WA
País de Residencia: Estados Unidos de América
Dirección de Correo: 2304 22nd Avenue NE
Ciudad para envío de correo: Sammamish
Estado o provincia para correo: WA
País para envío de correo: Estados Unidos de América
Código Postal: 98074

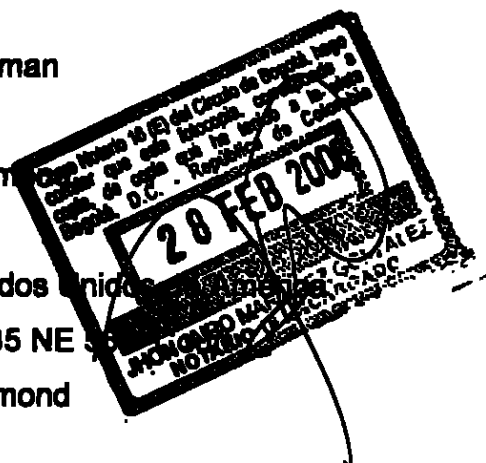


Fecha de Registro 03/28/05

Handwritten mark or signature.

• Tipo de Especialidad del Solicitante: Inventor
 País de Origen: Estados Unidos de América
 Estatus: En plenas facultades
 Primer Nombre: Jeffrey
 Segundo Nombre: T.
 Apellido: Pearce
 Sufijo del Nombre
 Ciudad de Residencia: Sammamish
 Estado o Provincia de residencia: WA
 País de Residencia: Estados Unidos de América
 Dirección de Correo: 437 237th Avenue SE
 Ciudad para envío de correo: Sammamish
 Estado o provincia para correo: WA
 País para envío de correo: Estados Unidos de América
 Código Postal: 98074

Tipo de Especialidad del Solicitante: Inventor
 País de Origen: Estados Unidos de América
 Estatus: En plenas facultades
 Primer Nombre: Michael
 Segundo Nombre: J.
 Apellido: Newman
 Sufijo del Nombre
 Ciudad de Residencia: Redmond
 Estado o Provincia de residencia: WA
 País de Residencia: Estados Unidos de América
 Dirección de Correo: 18685 NE
 Ciudad para envío de correo: Redmond
 Estado o provincia para correo: WA



Fecha de Registro 03/28/05

País para envío de correo: Estados Unidos de América
Código Postal: 98052

Tipo de Especialidad del Solicitante: Inventor
País de Origen: Estados Unidos de América
Estatus: En plenas facultades
Primer Nombre: Jason
Segundo Nombre: T.
Apellido: Hunter
Sufijo del Nombre
Ciudad de Residencia: Redmond
Estado o Provincia de residencia: WA
País de Residencia: Estados Unidos de América
Dirección de Correo: 4632 167th Court NE
Ciudad para envío de correo: Redmond
Estado o provincia para correo: WA
País para envío de correo: Estados Unidos de América
Código Postal: 98052

Información para Correspondencia
Número del cliente para correspondencia: 27195

Información del Representante Legal
Número del Representante del cliente: 27195

Información del Beneficiario de la Patente
Nombre del Cesionario: Microsoft Corporation
Dirección: One Microsoft Way
Ciudad para envío de correo: Redmond
Estado o provincia para envío de correo: WA
País para envío de correo: Estados Unidos de América
Código Postal: 98052



Fecha de Registro 03/28/05

45

Traducción Oficial

SOLICITUD DE PATENTE

DECLARACIÓN Y PODER DE ABOGADO

REGISTRO DE ABOGADO No. MSFTP682US

Expediente MS No. 311238.01

En mi calidad de inventor, por medio del presente documento declaro que:
Mi dirección de residencia/ dirección postal y nacionalidad aparecen consignados más abajo en este documento, enseguida de mi nombre;
Afirmo ser el inventor original, primero y único (si hay un solo nombre en la lista), o uno de los inventores originales y colectivos, si se encuentran varios nombres en la lista, del asunto que se reivindica y para el cual se solicita la patente del invento titulado: "MAPPING OF A FILE SYSTEM MODEL TO A DATABASE OBJECT/ MAPEADO DE UN MODELO DE SISTEMA DE ARCHIVOS A UN OBJETO DE BASE DE DATOS", cuyas especificaciones se presentan con este documento, a menos que esté indicado o marcado en el paréntesis que se encuentra a continuación:

() Se radicó el _____ con el número de serie para Solicitud de Patente en Estados Unidos o Solicitud Internacional PCT Número _____ y fue modificada el _____ (si aplica).

Declaro que he revisado y tengo pleno conocimiento y entendimiento de las especificaciones arriba mencionadas, incluyendo las ~~requisiciones~~ y modificaciones a las que se hace referencia arriba. Reconozco y consiento en revelar y publicar toda la información que se considera material para la patente según lo que se define en el 37 del Código CFR 1.56.



LA MEJOR COPIA DISPONIBLE

26-

Solicitudes Internacionales y/o Petición de Prioridad de Extranjería

Con el presente documento solicito los beneficios de Prioridad de Extranjería establecidos en el Título 35 del Código de los Estados Unidos, en su Artículo 119 (*United States Code, Section 119*) y que se confieren a la(s) solicitud(es) de Patente(s) a nivel Internacional o a los certificados de inventor(es) que se relacionan e identifican en la lista a continuación, como también a las solicitudes de inscripción de Patentes Internacionales o de Certificados de Inventor(es) que se hayan radicado con anterioridad a la fecha en que se solicita la prioridad :

PAÍS	SOLICITUD NÚMERO	FECHA DE RADICACIÓN	PRIORIDAD SOLICITADA BAJO EL TÍTULO 35 DEL ARTÍCULO 119 DEL U.S.C.
			SI: ☐ NO: ☐
			SI: ☐ NO: ☐

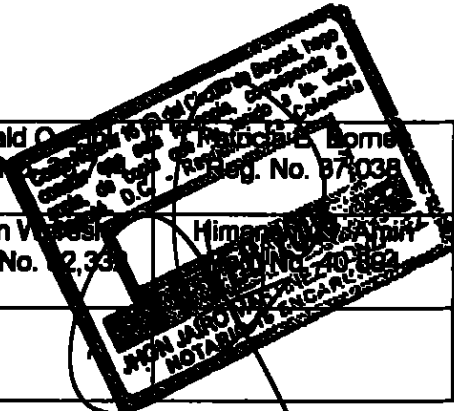
PODER DE REPRESENTACIÓN DE ABOGADO:

En mi calidad de inventor, en el presente documento designo al (los) siguiente(s) abogado(s) y/o representante(s) asociado(s) con

El Número de Cliente 27195

Para que realicen todas las gestiones relacionadas con esta solicitud ante la Oficina de Patentes y Marcas Registradas

David Bartley Eppenauer Reg. No. 35,499	Martin L. Shively Reg. No. 33,553	Ronald O. ... Reg. No. ...
Jeffrey L. Ranck Reg. No. 38,590	Michael W. Bocianowski Reg. No. 28,692	John V. ... Reg. No. ...
Gregory Turocy Reg. No. 36,952		Himant ... Reg. No. ...



LA MEJOR COPIA DISPONIBLE

27

Enviar Correspondencia a:		Llamar directamente a:
Nombre del contacto:	Himanshu S. Amin	Nombre del contacto:
Nombre de la firma:	Amin & Turocy, LLP	Himanshu S. Amin
Dirección de la Firma:	1900 E. 9th Street, National City Center – 24th floor	Teléfono del Contacto:
Ciudad, Estado, Código Postal:	Cleveland, OH 44114	(216)-696-8730



Página 3 de 6

LA MEJOR COPIA DISPONIBLE

 Copia Suministrada por la USPTO de la Base de datos de Imagen IFW el 12/19/2005

DECLARACIÓN Y PODER DE ABOGADO

REGISTRO DE ABOGADO No. MSFTP852U8

Expediente MS No. 311235.01

Con el presente documento declaro que todas estas afirmaciones las hago bajo mi pleno conocimiento y son verdaderas, además que toda la información que suministro es cierta; como también que todas estas aseveraciones las hago sabiendo que las afirmaciones falsas que se hacen de manera intencional son consideradas de carácter penal y se castigan con multas o prisión o de ambas formas, según lo establecido bajo el Título 18 del Código de los Estados Unidos en su Artículo 1001 y que dicho tipo de afirmaciones falsas ponen en peligro la validez de la solicitud o cualquier patente otorgada a partir de ellas.

Nombre Completo del Inventor:

Nigel R. Ellis

Ciudadano de :

Estados Unidos de AméricaDirección Residencia: Redmond, WashingtonDirección Postal: 12125 194th Avenue NERedmond, Washington 98053Ilegible3/24/2005

Firma del Inventor

Fecha

Nombre Completo del Inventor:

Gregory S. Friedman

Ciudadano de :

Estados Unidos de AméricaDirección Residencia: Redmond, WashingtonDirección Postal: 9812 173rd Court NERedmond, Washington 98052Ilegible3/24/2005

Firma del Inventor

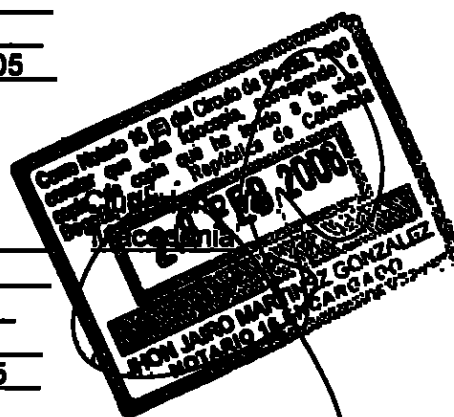
Fecha

Nombre Completo del Inventor:

Bekim DemiroskiDirección Residencia: Redmond, WashingtonDirección Postal: 6005 142nd Court NERedmond, Washington 98052Ilegible3/24/05

Firma del Inventor

Fecha



LA MEJOR COPIA DISPONIBLE

Página 4 de 6

Copia Suministrada por la USPTO de la Base de datos de Imagen IFW el 12/19/2005

A handwritten signature in the bottom right corner of the page.

DECLARACIÓN Y PODER DE ABOGADO

REGISTRO DE ABOGADO No. MSFTP652US

Expediente MS No. 311235.01

Con el presente documento declaro que todas estas afirmaciones las hago bajo mi pleno conocimiento y son verdaderas, además que toda la información que suministro es cierta; como también que todas estas aseveraciones las hago sabiendo que las afirmaciones falsas que se hacen de manera intencional son consideradas de carácter penal y se castigan con multas o prisión o de ambas formas, según lo establecido bajo el Título 18 del Código de los Estados Unidos en su Artículo 1001 y que dicho tipo de afirmaciones falsas ponen en peligro la validez de la solicitud o cualquier patente otorgada a partir de ellas.

Nombre Completo del Inventor:

Anil Kumar Nori

Ciudadano de :

Estados Unidos de AméricaDirección Residencia: Redmond, WashingtonDirección Postal: 7421 204th Drive NE
Redmond, Washington 98053Ilegible 3/24/2005
Firma del Inventor Fecha

Nombre Completo del Inventor:

Amit ShuklaCiudadano de : IndiaDirección Residencia: Redmond, WashingtonDirección Postal: 10835 170th Court NE
Redmond, Washington 98052Ilegible 3/24/05
Firma del Inventor Fecha

Nombre Completo del Inventor:

Srinivasmurthy P. AcharyaDirección Residencia: Sammamish, WashingtonDirección Postal: 2304 239th Place NE
Sammamish, Washington 98074Ilegible 3/24/2005
Firma del Inventor Fecha

LA MEJOR COPIA DISPONIBLE

30

DECLARACIÓN Y PODER DE ABOGADO

REGISTRO DE ABOGADO No. MSFTP852U8

Expediente MS No. 311235.01

Con el presente documento declaro que todas estas afirmaciones las hago bajo mi pleno conocimiento y son verdaderas, además que toda la información que suministro es cierta; como también que todas estas aseveraciones las hago sabiendo que las afirmaciones falsas que se hacen de manera intencional son consideradas de carácter penal y se castigan con multas o prisión o de ambas formas, según lo establecido bajo el Título 18 del Código de los Estados Unidos en su Artículo 1001 y que dicho tipo de afirmaciones falsas ponen en peligro la validez de la solicitud o cualquier patente otorgada a partir de ellas.

Nombre Completo del Inventor:

Jeffrey T. Pearce

Ciudadano de :

Estados Unidos de América

Dirección Residencia: Sammamish, Washington

Dirección Postal: 437 237th Avenue SE

Sammamish, Washington 98074

Ilegible

3/24/2005

Firma del Inventor

Fecha

Nombre Completo del Inventor:

Michael J. Newman

Ciudadano de :

Estados Unidos de América

Dirección Residencia: Redmond, Washington

Dirección Postal: 18685 NE 56th Court

Redmond, Washington 98052

Ilegible

3/24/2005

Firma del Inventor

Fecha

Nombre Completo del Inventor:

Jason T. Hunter

Dirección Residencia: Redmond, Washington

Dirección Postal: 4632 167th Court NE

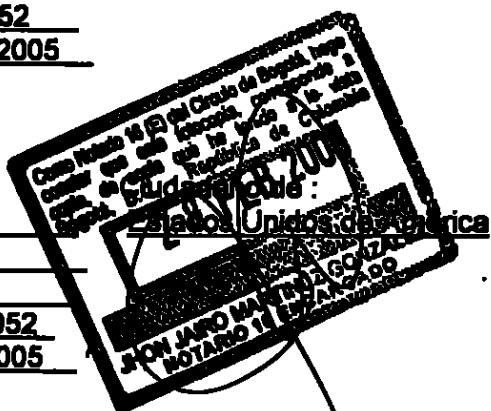
Redmond, Washington 98052

Ilegible

3/24/2005

Firma del Inventor

Fecha



LA MEJOR COPIA DISPONIBLE

Title: MAPEADO DE UN MODELO DE SISTEMA DE ARCHIVOS A UN OBJETO DE BASE DE DATOS

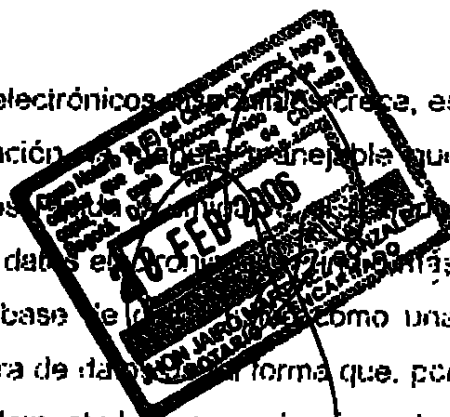
CAMPO TÉCNICO

[0001] El actual invento generalmente se relaciona con bases de datos y en particular con los sistemas y métodos que facilitan el almacenamiento de una instancia de tipos y/o datos de consulta.

ANTECEDENTES DEL INVENTO

[0002] Los avances en la tecnología de computadores (e.g., velocidad del microprocesador, capacidad de memoria, ancho de banda de transferencia de datos, funcionalidad del software, y similares) han contribuido en general al incremento de aplicaciones de computador en varias industrias. Aún los más potentes sistemas de servidores, los cuales frecuentemente están configurados como conjunto de servidores, se proveen generalmente, por ejemplo, a requerimientos de servicios originados de fuentes externas tales como la World Wide Web.

[0003] A medida que la cantidad de datos electrónicos disponibles crece, es mucho más importante almacenar dicha información de manera manejable que facilite las búsquedas y la recuperación de datos. Actualmente, el enfoque común es almacenar los datos en archivos o más bases de datos. En general, se denomina una base de datos como una colección organizada de información con estructura de datos en forma que, por ejemplo, un programa de computador pueda rápidamente buscar y seleccionar las porciones de datos que se quieren. Generalmente, los datos dentro de una base de datos están organizados via una o más tablas. Dichas tablas pueden estar organizadas con filas y columnas.



82

[0004] Las tablas pueden comprender un registro o más, mientras que un registro puede incluir una serie de campos. Los registros se indexan como filas dentro de una tabla y los campos de los registros se indexan comúnmente como columnas, de tal manera que un par de índices fila/columna pueda referenciar unos datos particulares dentro de una tabla. Por ejemplo, una fila puede almacenar un registro completo de datos que se refieren a una transacción de ventas, persona o proyecto. Asimismo, las columnas de la tabla pueden definir discretas porciones de las filas que tienen el mismo formato general de datos, mientras que las columnas pueden definir los campos de los registros.

[0005] Cada porción individual de datos, por sí sola, generalmente no es muy informativa. Las aplicaciones de bases de datos hacen que éstos sean más útiles porque le ayudan a los usuarios a que organicen y procesen los datos. La aplicación de base de datos permite al usuario comparar, escoger, ordenar, juntar, separar e interconectar los datos de tal manera que se pueda generar información útil de los datos. Empero, la capacidad y versatilidad de las bases de datos han crecido de manera tan increíble que permiten una capacidad de almacenamiento virtualmente sin fin utilizándolas. Además, los sistemas comunes de bases de datos ofrecen una limitada habilidad para consulta basados en el tiempo, la extensión del archivo, la locación y el tamaño. Por ejemplo, con el fin de buscar las enormes cantidades de datos asociados a una base de datos, una búsqueda normal se limita al nombre del archivo, el tamaño, fecha de creación y tales técnicas son deficientes e ineptas.

[0006] Con la continua e incrementada creación de datos, los problemas y dificultades alrededor de la creación, almacenamiento de dichos datos está llegando a su límite. Los usuarios finales escriben documentos, almacenan fotos, extraen música de discos compactos, reciben correo electrónico, guardan copias de correos electrónicos enviados, etc. Por ejemplo, en el simple proceso de crear un disco compacto de música el



usuario final puede crear megabytes de datos. La extracción de la música de un disco compacto, la conversión del archivo a un formato adecuado, la creación de la cobertura de la caja, el diseño de la etiqueta del disco, todo requiere la creación de datos.

[0007] Las complicaciones no solo están alrededor de los usuarios sino que los que desarrollan tienen asuntos similares con los datos. Los que desarrollan crean y escriben miríadas de aplicaciones que varían desde las personales hasta las empresariales altamente desarrolladas. Los desarrolladores frecuentemente, si no siempre, mientras están creando y/o desarrollando, están recolectando datos. A medida que obtienen dichos datos, éstos deben almacenarse. En otras palabras, los problemas y dificultades alrededor de la búsqueda, relación y almacenamiento de datos afectan tanto al desarrollador como al usuario final. En vista de lo anterior, existe la necesidad de mejorar y/o proveer sistemas y/o métodos que mitiguen las deficiencias asociadas con los sistemas y bases de datos convencionales.

RESUMEN DEL INVENTO

[0008] Lo siguiente presenta un resumen simplificado del invento con el fin de proporcionar un entendimiento básico de algunos de los aspectos de éste. Este resumen no es una extensa información general. No se propone ni identificar elementos claves o críticos ni delinear el alcance del invento. Más bien, se pretende presentar algunos conceptos del invento de manera que se pueda proporcionar una descripción más detallada que se presenta más adelante.

[0009] La materia del invento se relaciona con los sistemas y/o métodos que facilitan el mapeado de un modelo de datos a un objeto de bases de datos. Un sistema de almacenamiento de tipos puede proporcionar almacenamiento de una instancia de tipo y/o para consulta. El almacenamiento del tipo puede describir el

objeto de la base de datos y cómo los tipos de instancia se almacenan y/o se puede tener acceso a ellas. La consulta pueden encontrar por lo menos uno de los siguientes, un ítem, un documento y un contacto que cada uno satisfaga un cierto criterio. Además, el sistema de almacenamiento de tipos puede utilizar una interfase que reciba datos, mientras que los datos pueden incluir un esquema, un tipo, un criterio, una consulta de criterios, etc.

[0010] De acuerdo con un aspecto del objeto del invento, el sistema de almacenamiento de tipos puede incluir un componente de almacén que almacena por lo menos una instancia de tipo. El almacenamiento puede ser un mapeado del modelo de datos, donde el modelo de datos puede representar un sistema de almacenamiento de archivos. Además, el sistema de almacenamiento de tipos puede incluir también un componente de consulta. El componente de consulta puede suministrar por lo menos una consulta en el modelo de datos del sistema de almacenamiento de archivos para satisfacer un criterio. De acuerdo con otro aspecto del objeto del invento, el sistema de almacenamiento de tipos puede incluir un componente relacional que pueda utilizar un almacenamiento relacional y/o capacidad de consulta relacional. El componente relacional puede invocar un motor de bases de datos para que proporcione dichas técnicas relacionales, donde estas técnicas pueden facilitar el almacenamiento de instancias y/o consultas de tipo.

[0011] Además, el sistema de almacenamiento de tipos puede también incluir un componente de vista que puede proporcionar una vista que expone más las instancias de un tipo dado. El tipo puede ser una estructura o una estructura de legado. La vista se puede relacionar con un tipo base. En otro aspecto del invento, se proveen métodos que facilitan el mapeado de un modelo de datos a un objeto de datos.

[0012] La siguiente descripción y los dibujos anexos presentan en detalle ciertos aspectos ilustrativos del invento. Sin embargo, estos aspectos son indicativos de solo algunas de las varias maneras en que los principios del invento pueden emplearse y el asunto del invento tiene el propósito de incluir todos los dichos aspectos y sus equivalentes. Otras ventajas y características novedosas del invento se volverán aparentes de la siguiente descripción detallada del invento cuando se consideran en conjunto con los dibujos.

BREVE DESCRIPCIÓN DE LOS DIBUJOS

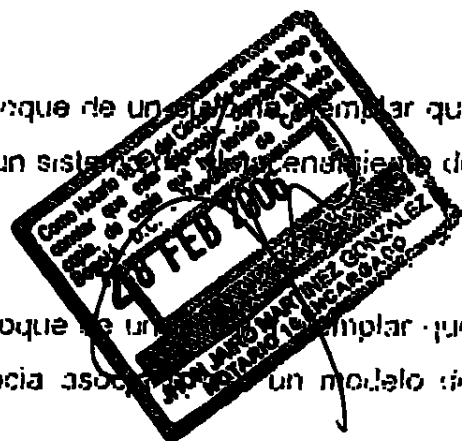
[0013] La Fig. 1 ilustra un diagrama en bloque de un sistema ejemplar que facilita el almacenamiento de un tipo de instancia asociado con el modelo de datos.

[0014] La Fig. 2 ilustra un diagrama de bloque de un sistema ejemplar que facilita el almacenamiento de un tipo de instancia y/o de consulta para encontrar por lo menos uno de los siguientes, un ítem, un documento y un contacto.

[0015] La Fig. 3 ilustra un diagrama de bloque de un sistema ejemplar que facilite la utilización de almacenamiento relacional y/o capacidades de consulta relacionales.

[0016] La Fig. 4 ilustra un diagrama en bloque de un sistema ejemplar que facilita el mapeado y/o la vista en conjunto con un sistema de almacenamiento de tipo.

[0017] La Fig. 5 ilustra un diagrama en bloque de un sistema ejemplar que facilita el almacenamiento de un tipo de instancia asociado con un modelo de datos.



[0018] La Fig. 6 ilustra un diagrama en bloque de una estructura de alto nivel de un almacén dentro de un modelo de datos de almacenamiento de archivos.

[0019] La Fig. 7 ilustra un diagrama en bloque de una instancia de tipo con tablas asociadas.

[0020] La Fig. 8 ilustra un diagrama en bloque de una jerarquía de tipo y su correspondiente proyección de vista.

[0021] La Fig. 9 ilustra una metodología ejemplar para el almacenamiento de un tipo de instancia que mapea a un modelo de datos y/o que suministra una consulta.

[0022] La Fig. 10 ilustra una metodología ejemplar para el almacenamiento de instancia de tipo que mapea a un modelo de datos y/o que suministra una consulta.

[0023] La Fig. 11 ilustra un ambiente ejemplar de operaciones en red, donde pueden ser empleados los aspectos novedosos del objeto del invento.

[0024] La Fig. 12 ilustra un ambiente operativo ejemplar que puede ser empleado de acuerdo con el objeto del invento

DESCRIPCIÓN DEL INVENTO

[0025] Tal como se utilizan en esta aplicación, los términos "componente", "sistema," "interfaz," "esquema," y similares, se refieren a una entidad relacionada con computación, ya sea hardware, software (e.g., en ejecución), y/o programación. Por ejemplo, un componente puede ser un proceso que se está ejecutando en

un procesador, un procesador, un objeto, un ejecutable, un programa y/o un computador. Para ilustración, tanto una aplicación que se ejecuta en un servidor y el servidor mismo pueden ser un componente. Uno o más componentes pueden residir dentro de un proceso y un componente puede estar localizado en un computador y/o distribuido entre dos o más computadores.

[0026] El objeto del invento se ha descrito con referencia a los dibujos, donde los numerales similares de referencia se utilizan para referenciar elementos similares por todo este. En la siguiente descripción, como explicación, se presentan numerosos detalles específicos con el fin de suministrar una comprensión completa del objeto del invento. Sin embargo, puede ser evidente que el objeto del invento puede practicarse sin estos detalles específicos. En otras instancias, se muestran en forma de diagramas en bloque las estructuras y aparatos bien conocidos con el fin de facilitar la descripción del objeto del invento.

[0027] Ahora, volviendo a las figuras, la Fig. 1 ilustra un sistema 100 que facilita el almacenamiento de una instancia de tipo asociada con un modelo de datos. El modelo de datos puede ser un modelo de datos de almacenamiento de archivo 102 que permite el almacenamiento, el hallazgo y relacionar la información. Por ejemplo, un tipo de información puede ser, pero no se limita a ser, un documento, una imagen, un video, un contacto, un mensaje, un correo electrónico, un clip de audio, etc. Los tipos informacionales pueden considerarse como unidades de información, los cuales pueden ser representados como instancias de tipos complejos que son parte de un sistema de datos que apoyan el legado. Se puede definir el legado como un conjunto de datos en el cual algunas características se pasan de un contexto a otro. En particular, la organización orientada hacia el objeto, los objetos heredan atributos/funcionalidades de otros objetos. Se debe apreciar que la herencia (legado) puede considerarse como una estructura y/o formato jerárquico.

[0028] Un sistema de almacenamiento de tipo 104 puede almacenar la instancia de tipo y consulta para encontrar eficiente y efectivamente por lo menos uno de los siguientes, ítems, documentos y/o contactos. El sistema de almacenamiento de tipo 104 puede recibir datos, donde éstos pueden incluir un tipo, un criterio, un esquema, un criterio de consulta, Específicamente, almacenar la instancia de tipo que se relaciona con un tipo informacional (e.g., un documento, una imagen, un video, un contacto un mensaje, un correo electrónico, un clip de audio) puede proporcionar por lo menos uno de los siguientes: 1) encontrar por lo menos un ítem en el modelo de datos de almacenamiento de archivo 102 que satisfaga un cierto criterio, 2) encontrar por lo menos un documento en el modelo de datos de almacenamiento de archivo 102 que satisfaga un criterio en particular; y 3) encontrar por lo menos un contacto (e.g., incluyendo un individuo, organización y grupo) que satisfaga un cierto criterio.

[0029] El sistema de almacenamiento de tipo 104 puede utilizar una técnica de base de datos relacional asociada con un almacenamiento relacional y una consulta relacional. Se debe reconocer que dichas capacidades pueden ser suministradas por un motor de bases de datos (no ilustrado). Las técnicas de base de datos relacionales pueden asociarse a las bases de datos relacionales, donde una base de datos relacional es una colección de ítems de datos organizada como una serie de tablas descritas formalmente. Se puede tener acceso a los datos dentro de la tabla y/o pueden ser reensamblados de varias maneras sin requerir reorganización de las tablas de bases de datos. Además, las bases de datos relacionales se pueden ampliar fácilmente, tal como añadiendo nuevas categorías sin modificaciones a aplicaciones y/o interfaces. Se debe apreciar que el objeto del invento no está limitado a bases de datos relacionales y/o a técnicas asociadas y que se puede utilizar cualquier técnica.

[0030] El sistema 100 además incluye un componente de interfase 106, el cual provee varios adaptadores, conectores, canales, vías de comunicación etc.

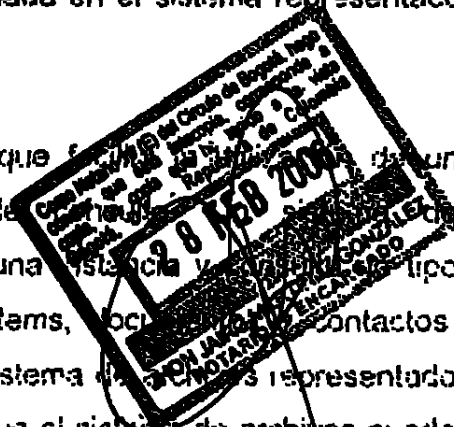
para integrar el sistema de almacenamiento de tipos 104, a virtualmente cualquier sistema operativo. Además, el componente de interfase 106 puede proveer varios adaptadores, conectores, canales, vías de comunicación, etc. que proporcionen una interacción con los datos y con el sistema de almacenamiento de tipos 104. Se debe apreciar que aunque el componente de interfase 106 está incorporado dentro del sistema de almacenamiento de tipos 104, dicha implementación no es tan limitada. Por ejemplo, el componente de interfase 106 puede ser un componente que esté solo para recibir o transmitir los datos con relación al sistema 100.

[0031] La Fig. 2 ilustra un sistema 200 que facilita el almacenamiento de una instancia de tipo y/o de consulta para encontrar un ítem, un documento, y un contacto asociado al modelo de datos 202. El modelo de datos 202 puede ser la representación de un modelo de un sistema de almacenamiento de archivos que utiliza una característica jerárquica y/o legado. El tipo puede incluir un documento, una imagen, un video, un contacto, un mensaje, un correo electrónico, un clip de audio, etc. Sin embargo, se debe apreciar que un tipo puede ser un tipo de información común que está almacenado en un sistema representado por el modelo de datos 202. Un sistema de almacenamiento de tipos 204 puede almacenar la instancia de tipo y suministrar consultas que puedan eficientemente por lo menos encontrar una de las siguientes: un ítem en el modelo de datos 202, un documento en el modelo de datos 202, y un contacto en el modelo de datos 202. El sistema de almacenamiento de tipo 204 puede suministrar una instancia de tipo, donde los datos pueden ser un criterio de consulta, un esquema, un modelo de datos, un tipo, El sistema de almacenamiento de tipos 204 puede emplear una interfase 206 para facilitar el suministro de adaptadores, conectores, canales, vías de comunicación, etc. para integrar el sistema de almacenamiento de tipos 204 en virtualmente cualquier sistema operativo y facilitar la comunicación.

[0032] El sistema de almacenamiento de tipos 204 puede incluir además un componente de almacenamiento 208 ("almacén 208") que puede almacenar las instancias de tipo. El almacenamiento puede ser un mapeado del modelo de datos 202, donde el mapeado del almacenamiento puede describir el objeto de la base de datos que está creado basado en una definición de esquema y cómo las instancias de tipos descritas en el esquema se almacenan y/o pueden ser accedidas. En otras palabras, el almacén 208 puede almacenar las instancias del tipo y por lo menos una regla asociada a mapear una declaración de tipo dentro del objeto de la base de datos.

[0033] El sistema de almacenamiento de tipo 204 puede incluir un componente de consulta 210 ("consulta 210") que proporciona consulta de datos. La consulta 210 puede obtener por lo menos uno de los siguientes: un ítem en el sistema representado por el modelo de datos 202; un documento en el sistema representado por el modelo de datos 202; y un contacto (e.g., incluyendo un individuo, una organización y un grupo) en el sistema representado por el modelo de dato 202. Se debe apreciar y comprender que la consulta puede estar basada en por lo menos ciertos criterios y/o criterios de consulta obtenidos vía la interfase 206. Además, la consulta 210 no está tan limitada al ítem, documento y contacto y cualquier tipo de información apropiada almacenada en el sistema representado por el modelo de datos 202 puede ser utilizado.

[0034] La Fig. 3 ilustra un sistema 300 que puede ser un sistema de almacenamiento relacional y, capacidades de almacenamiento de tipo 304 puede almacenar una instancia de tipo para eficiente y efectivamente se encuentran ítems, documentos y contactos donde dichos elementos están asociados a un sistema de archivos representado por un modelo de datos 302. Se debe apreciar que el sistema de archivos puede ser un sistema de almacenamiento de archivos que utiliza una estructura jerárquica y/o una característica de herencia (legado). El sistema de



almacenamiento de tipo 304 puede ser sustancialmente similar al sistema de almacenamiento de tipos 104, 204, como se muestra en las Figs. 1 y 2 respectivamente. El sistema de almacenamiento de tipos 304 puede invocar una interfase 306 para facilitar las comunicaciones y/o recibir datos para que sean utilizados más adelante de acuerdo con el objeto del invento.

[0035] El sistema de almacenamiento de tipos 304 puede incluir un componente de almacén 308 (almacén 308) y un componente de consulta 310 ("consulta 310"). El almacén 308 puede proporcionar cualquier técnica de almacenamiento adecuado para almacenar instancias de tipo. La consulta 310 puede suministrar una consulta que pueda eficiente y efectivamente obtener por lo menos una de las siguientes: por lo menos un ítem en un sistema que cumpla con un criterio; por lo menos un documento en un sistema que satisfaga un criterio; y por lo menos un contacto que cumpla con un criterio. Se debe apreciar y comprender que el almacén 308 y la consulta 310 pueden ser sustancialmente similares al almacén 208, y a la consulta 210 como se muestra en la Fig. 2.

[0036] El sistema de almacenamiento de tipos 304 puede también incluir un componente relacional 312 ("relacional 312"). El relacional 312 utiliza técnicas de bases de datos (e.g., utilizando un motor de bases de datos) para construir un almacenamiento relacional y/o suministrar una capacidad de consulta relacional para facilitar el almacenamiento de las instancias y/o de los tipos. El relacional 312 puede incorporar técnicas asociadas con bases de datos relacionales, mientras que la base de datos relacional puede ser una colección de datos organizados como una serie de tablas. Las técnicas asociadas con bases de datos relacionales se describen en detalle más arriba. Se debe apreciar que el presente invento no está limitado a la base de datos relacional y técnicas asociadas, sino que se puede utilizar cualquier técnica apropiada.

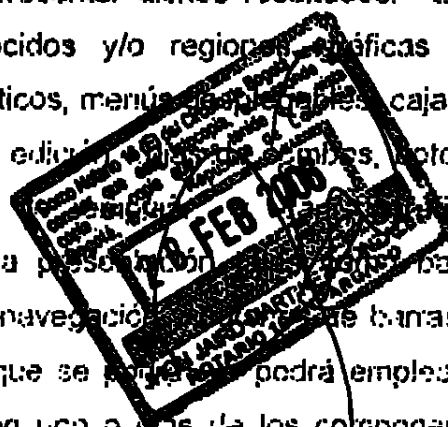
[0037] La Fig 4 ilustra un sistema 400 que facilita el mapeado y/o vista en conjunción con un modelo de datos 402. El modelo de datos 402 puede representar un sistema de almacenamiento de archivo que permite el almacenamiento, el hallazgo y el relacionar la información. Un tipo de información típica que pueda ser almacenado en el sistema puede incluir un documento, una imagen, música, un video, un contacto, un mensaje, etc. El tipo de información puede ser representado como instancias de tipos complejos que son parte de un sistema de tipos que apoyan la herencia. Un sistema de almacenamiento de tipos 404 puede almacenar las instancias de tipo y suministrar una consulta para encontrar ítems, documentos y contactos que satisfaga ciertos criterios. El sistema 400 puede también emplear una interfase 406 para facilitar la comunicación y/o recibir datos que pueden incluir criterios, tipos, esquemas, modelos y criterios de consulta.

[0038] El sistema de almacenamiento de tipos 404 puede también incluir un componente de mapa 408 para facilitar el almacenamiento de la instancia del tipo. El componente de mapa 408 suministra el mapeado de tipos descritos en un esquema para definir tipos y objetos de bases de datos. El componente de mapa 408 puede ser un mapeado de almacenamiento que describe por lo menos un objeto de base de datos, donde el objeto de base de datos puede ser creado basado en una definición de esquema y/o cómo las instancias de tipos descritos en el esquema se almacenan y/o se pueda tener acceso. Con otras palabras, las instancias de tipos pueden almacenarse y accederse usando reglas para mapear una declaración de tipo en el objeto de base de datos. El tipo en el esquema mapea a una clase (e.g., tiempo de ejecución de lenguaje común (common language runtime -CLR)) en un almacenamiento.

[0039] El sistema de almacenamiento de tipos 404 puede incluir un componente de vista 410 para proveer una proyección de vista. La proyección de vista puede exponer instancias del tipo que se está viendo. Se debe apreciar que

los tipos pueden estar en una estructura jerárquica utilizan lo por lo menos el legado. En otras palabras, cada tipo es parte de una jerarquía de tipo. La vista puede asociarse con un tipo dado y puede proyectar una sub serie de los respectivos tipos de la vista asociados con su tipo base. La vista puede proyectar instancias asociadas con el tipo particular. Por ejemplo, para un tipo "mensaje", solo las instancias en las cuales el mensaje es un padre pueden ser vistas basadas pro lo menos en la estructura de jerarquías.

[0040] El componente de vista 410 puede proporcionar varios tipos de interfases de usuario para facilitar la interacción entre un usuario y cualquier componente unido al sistema de almacenamiento de tipos 404. Como se demostró, el componente de vista 410 es una entidad separada que está incorporada dentro del sistema de almacenamiento de tipos 404. Sin embargo, se debe apreciar que el componente de vista 410 y/o similares componentes de vista pueden ser componentes separados del sistema de almacenamiento de tipos 404 y/o de una unidad que está sola. El componente de vista 410 puede proveer una o más interfases gráficas de usuario (graphical user interfaces GUI), interfases de línea de comando, y similares. Por ejemplo, se puede presentar una GUI que le suministra al usuario con una región o con los medios para cargar, importar, leer, etc. datos y puede incluir una región para presentar dichos resultados. Estas regiones pueden comprender textos conocidos y/o regiones gráficas que comprenden cajas de diálogos, controles estáticos, menús desplegables, cajas de listas, menús emergentes, como controles de edición, botones, botones de radio, cajas de chequeos, botones de selección, barras de desplazamiento, barras desplegadas, barras de herramientas para determinar si una región que se puede emplearse. Por ejemplo, el usuario puede interactuar con uno o más de los componentes unidos al sistema de almacenamiento de tipos 404.



[0041] El usuario también puede interactuar con las regiones para seleccionar y suministrar información vía varios aparatos, tales como por ejemplo, un ratón, un roller ball, un teclado numérico, un teclado, un puntero y/o activación de voz. Generalmente, un mecanismo como un botón para pulsar o la tecla de "enter" en el teclado puede emplearse luego de que se introduzca la información con el fin de iniciar la búsqueda. Sin embargo, se debe apreciar que el invento no es tan limitado. Por ejemplo, simplemente al resaltar una caja de chequeo puede iniciarse la transmisión de la información. En otro ejemplo, se pueda utilizar una interfase de línea de comando. Por ejemplo la interfase de línea de comando puede invitar al usuario (e.g., vía un mensaje de texto en una pantalla y un tono de audio) a que mande la información al proporcionarle un mensaje de texto. El usuario entonces puede suministrar la información adecuada, como una entrada alfa numérica correspondiente a una opción suministrada en el mensaje de la invitación de la interfase o una respuesta a una pregunta hecha en la invitación. Se debe apreciar que la interfase de línea de comando puede ser utilizada en conexión con una GUI y/o API. Adicionalmente, la interfase de línea de comando puede utilizarse en conexión con hardware (e.g., tarjeta de video) y/o despliegues (e.g., negro y blanco y EGA) con soporte gráfico limitado y/o canales de comunicación con ancho de banda bajo.

[0042] La Fig. 5 ilustra un sistema 500 que emplea una interfase para facilitar el almacenamiento de una instancia de tipo asociada con un modelo de datos 502. El sistema 500 incluye un sistema de almacenamiento 504, una interfase 506, y el modelo de datos 502, los cuales pueden ser sustancialmente similares a los componentes descritos en las figuras anteriores. La interfase 506 puede facilitar la comunicación asociada con los datos, lo cual puede incluir un criterio, un tipo, un esquema, un modelo de datos y un criterio de consulta. El sistema 500 puede proporcionar el almacenamiento de un tipo, consultar y/o suministrar una vista. Se debe apreciar y comprender que un motor

HS

la base de datos puede proveer un almacenamiento relacional y una consulta relacional al sistema 500.

[0043] El sistema 500 además incluye un componente inteligente 508. El componente inteligente 508 puede ser utilizado por el sistema de almacenamiento de tipos 504 para facilitar el almacenamiento y/o consulta para el sistema 500. Por ejemplo, el componente inteligente 510 puede ser utilizado para facilitar el determinar el tipo de usuario definido a ser almacenado. Los datos históricos, en conjunción con los perfiles del usuario pueden permitir al componente inteligente 508 el determinar el almacenamiento del tipo y/o consulta con cierto criterio.

[0044] Se debe entender que el componente inteligente 508 puede proporcionar razonamientos sobre los estados del sistema, el ambiente y/o del usuario o inferir sobre ellos desde una serie de observaciones que han sido capturadas vía eventos y/o datos. La inferencia puede emplearse para identificar un contexto específico o acción, o puede generar una distribución de probabilidades sobre los estados, por ejemplo. La inferencia puede ser probabilística - esto es, la computación de una distribución de probabilidades sobre los estados de interés basados en una consideración de datos y eventos. La inferencia puede también referirse a técnicas empleadas para componer eventos a alto nivel de una serie de eventos y/o datos. Dicha inferencia resulta en la construcción de nuevos eventos o acciones de una serie de eventos observados y/o datos de eventos almacenados, sean éstos correlacionados por proximidad temporal cercana o no y si los eventos y los datos son provenientes de varias fuentes de eventos o datos. Se pueden emplear técnicas de inferencia (entrenadas explícita y/o implícitamente) esquemas y/o sistemas (e.g., redes de soporte, redes neutras, sistemas expertos, redes de inferencia bayesianas (Bayesian belief networks), lógica difusa, motores de fusión de datos, etc.) en conexión con la acción automática desempeñada y/o inferida en conexión con el objeto del evento.

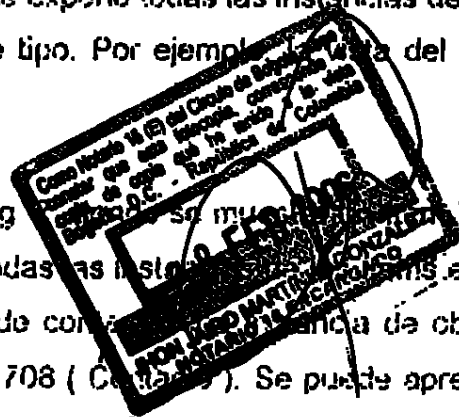
[0045] El clasificador es una función que mapea un vector de atributos de entrada, $x = (x_1, x_2, x_3, x_4, \dots, x_n)$, a una confianza que la entrada pertenece a una clase, que es, $f(x) = \text{confidence}(\text{class})$. Dicha clasificación puede utilizar análisis probabilística y/o basado en estadísticas (e.g., factorizando dentro de las utilidades y costos del análisis) para diagnosticar o inferir una acción que el usuario desea que se realice automáticamente. Una máquina vectorial de soporte (support vector machine - SVM) es un ejemplo de un clasificador que se puede utilizar. La SVM opera encontrando una hiper-superficie en el espacio de posibles entradas, la cual trata de dividir el criterio iniciador de los eventos no iniciadores. Intuitivamente, esto hace que la clasificación sea correcta para comprobar los datos que están cerca, pero no idéntica a los datos de entrenamiento. Otros enfoques de clasificación de modelos directos e indirectos incluyen, e.g., Bayes crédulas (naïve Bayes), redes de Bayes, árboles de decisiones (decision trees), redes neurales (neural networks), modelos de lógica difusa y modelos de clasificación probabilística que suministran diferentes patrones de independencia que se pueden utilizar. La clasificación como se utiliza en este también es inclusiva de regresión estadística que se utiliza para desarrollar modelos de prioridad.

[0046] Además, el componente inteligente 508 puede utilizar un almacén de datos 510 para almacenar perfiles de usuarios y/o datos históricos. El almacén de datos 510 puede ser, por ejemplo, memoria volátil o no volátil, o puede incluir tanto una como la otra. Como ilustración, y no limitación, la memoria volátil puede incluir memoria de solo lectura (ROM), ROM borrable (PROM), ROM eléctricamente programable (EPROM), ROM eléctricamente programable borrable (EEPROM), o memoria flash. La memoria volátil puede incluir memoria de acceso aleatorio (RAM), la cual actúa como una memoria de acceso aleatorio. Como ilustración y no limitación, la RAM se encuentra disponible en muchas formas, tales como la RAM estática (SRAM), RAM dinámica (DRAM), DRAM sincronizada (SDRAM), SDRAM de tasa de doble dato (DDR SDRAM), (double data rate SDRAM), SDRAM avanzado (ESDRAM), (enhanced SDRAM), DRAM Synchlink (SLDRAM), RAM

directo de Rambus (RDRAM), RAM dinámico directo de Rambus (DRDRAM) y RAM dinámico de Rambus (RDRAM). El almacén de datos 510 de los sistemas y métodos del objeto intentan abarcar, sin limitarse a éstas y a cualquier otro tipo de memoria adecuada. Adicionalmente, se debe apreciar que el almacén de datos 510 puede ser un servidor y/o base de datos.

[0047] La Fig. 6 ilustra la estructura de alto nivel de un almacén. Se puede proveer un esquema donde varias instancias de tipos se pueden almacenar en tablas dependientes de su tipo. Los tipos pueden ser, pero no se limitan a, un ítem, la extensión de un ítem, el fragmento de un ítem y un vínculo. Cada tipo puede tener una tabla correspondiente que puede incluir columnas que tienen instancias de objeto en ellas. Por ejemplo, una columna en la tabla de ítems puede contener todas las instancias de ítems en el almacén. Para cada fila, la columna puede contener una representación serializada de la instancia de clase CLR que representa la instancia de tipo del ítem. Debe apreciarse que para la extensión del ítem, los fragmentos de ítem y las instancias de tipo vínculo pueden estar representadas en una estructura similar. Las instancias de tipo en línea pueden almacenarse dentro de una instancia de objeto padre más que almacenarse en tablas separadas y/o columnas. Para cada ítem, extensión de ítem, fragmento de ítem y vínculo, se puede generar una vista que expone todas las instancias de ese tipo. Cada tipo es parte de una jerarquía de tipo. Por ejemplo, la vista del ítem proyecta todos los ítems del almacén.

[0048] Volvamos brevemente a la Fig. 6. La tabla 704 del ítem 702 puede contener todas las instancias de objeto 706 (Documento") y una instancia de objeto 708 (Categoría). Se puede apreciar que cada tipo en un esquema mapea a la clase de CLR en el almacenamiento. La instancia de objeto 706 puede incluir un título, un resumen, un impreso, un autor y otros varios metadatos asociados con la instancia del objeto 706. De manera



similar, la instancia del objeto 708 puede incluir varios metadatos tales como un nombre, una dirección, un correo electrónico, etc.

[0049] La Fig. 8 ilustra una jerarquía de tipo 800 y su correspondiente proyección de vista 820. La jerarquía de tipo 800 puede incluir un ítem 802, un contacto 804, un documento 806, un mensaje 808, una persona 810, una organización 812, un correo electrónico 814, un fax 816, y una voz 818. Se debe apreciar que la jerarquía de tipo 800 es un ejemplo y cualquier jerarquía y/o tipos pueden ser utilizados de acuerdo con el objeto del invento. Como se ilustra, el ítem 802 se considera el padre de cada tipo incluido allí. Así, el contacto 804 es el padre de la persona 810 y de la organización 812, mientras que el mensaje 808 es el padre para el correo electrónico 814, fax 816, y voz 818. Además, la proyección de vista correspondiente 820 puede reflejar la jerarquía de tipo 800. La vista asociada con un tipo dado puede proyectar una sub serie de los ítems de la vista asociada con su tipo base. Por ejemplo, la vista del ítem proyecta todos los ítems en el almacén. La vista de contacto proyecta solamente los ítems que son del tipo Contacto. La vista de persona proyecta solo los contactos que tienen la mayoría de tipos como persona.

[0050] Volviendo a la Fig. 6, el mapeado de tipos puede administrarse al utilizar un algoritmo para hacer el mapeado del tipo al tipo. Se debe apreciar que el mapeado puede asociarse a cualquier estructura de almacenamiento de archivo (e.g., un modelo de datos basado en un sistema que maneja instancias complejas de tipos para describir y/o representar una instancia de información). Cada tipo declarado en un esquema está mapeado a un tipo CLR que soporta el contrato UDT de SQL. Se debe apreciar que aunque el SQL se utiliza en los siguientes ejemplos, cualquier sistema de manejo de bases de datos adecuado puede utilizarse. Los tipos pertenecen a un espacio de nombre cuyo nombre corresponde a un espacio de nombre del esquema con el sufijo anexo de

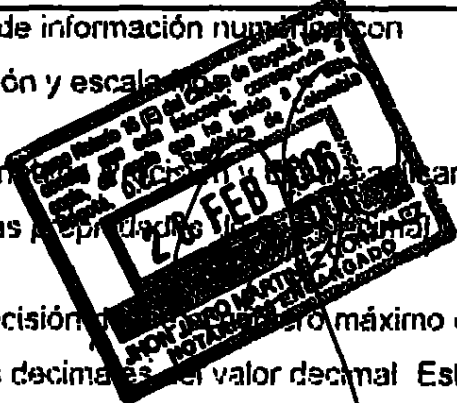
22

Almacén". Los tipos en un espacio de nombre dado se compilan en un ensamblaje único que se utiliza como unidad de despliegue del esquema. El nombre del tipo de CLR es igual al nombre declarado en el esquema. Por cada propiedad declarada del tipo, se agrega lo siguiente al correspondiente tipo de CLR: 1) un campo privado con el nombre igual al nombre especificado en el esquema, con el prefijo de "m_". Al campo se le atribuye con el atributo específico de UDT: System.Data.SqlTypes.SqlUdtField; 2) una propiedad pública con el nombre igual al nombre especificado en el esquema y que corresponde a declaraciones obtenidas/fijadas (get/set statements). A la propiedad se le atribuye el atributo específico de UDT System.Data.SqlTypes.SqlUdtProperty; y 3) el tipo de campo y la propiedad es el tipo de CLR que corresponde al tipo declarado en el esquema. Si el tipo es uno de aquellos escalares que está mapeado a uno de los tipos escalares SQL (tratados *infra*).

[0051] La siguiente tabla describe el mapeo de los tipos de escala del sistema de almacenamiento de archivos según los tipos de SQL administrados:

Tipo de Sistema de Almacenamiento de Archivo	Tipo de SQL Administrado	Descripción
String – Secuencia	SqlString	Información Unicode de longitud variable con una longitud máxima de 4000 caracteres. Se puede utilizar la clave utilizando la clave.
Binary - Binario	SqlBinary	Información Binaria de longitud variable con una longitud máxima de 2^32 bytes.

50

		La longitud se puede fijar de 1 - 3000 bytes o sin restricciones utilizando la clave
Boolean	SqlBoolean	Valor Boolean anulable
Byte	SqlByte	Un byte único sin firma
	SqlInt16	Información en enteros desde -2^15 (-32,768) hasta 2^15 - 1 (32,767).
Int32	SqlInt32	Información en enteros (número entero) desde -2^31 (-2,147,483,648) hasta 2^31 - 1 (2,147,483,647).
Int64	SqlInt64	Información en enteros (número entero) desde -2^63 (-9223372036854775808) hasta 2^63-1 (9223372036854775807).
Single - Sencillo	SqlSingle	Información en número de precisión flotante desde -3.40E + 38 hasta 3.40E + 38
Double - Doble	SqlDouble	Información en número de precisión flotante desde -1.79E + 308 hasta 1.79E + 308
Decimal	SqlDecimal	Tipos de información numérica con precisión y escala. Los atributos de precisión y escala se aplican para las propiedades de los tipos de datos. La Precisión es el número máximo de dígitos decimales del valor decimal. Esto incluye los dígitos a la izquierda y a la derecha del punto decimal. El valor del atributo de Precisión en un entero entre 1 y 28 

57

		La Escala define el número máximo de dígitos decimales a la derecha del punto decimal. El valor del atributo de Escala debe estar entre 0 y el valor del atributo de Precisión. Nota para el hito B: estos atributos no están respaldados en este hito. La Precisión se fija en 28 y la Escala en 14.
DateTime -- Fecha Hora	SqlDateTime	Fecha y Hora a partir de enero 1, 1753, a diciembre 31, 9999, con una precisión de trescientos partes de un segundo o 3.33 milisegundos.
Guid	SqlGuid	Un identificador unico global - globally unique identifier (GUID).
Xml	SqlString	XML se almacena actualmente como un campo de secuencia. El respaldo al tipo Xml Nativo está siendo considerado para post-Beta1
Stream	TBD	Se está planificando Respaldo para post-Beta1. En el caso de que se implemente, se reemplazarán las secuencias de secuencia (max) y binarias.

Tabla 1

[0052] Un objeto de base de datos creado en un almacén de sistema de almacenamiento de archivos puede ser almacenado en un nombre de esquema SQL derivado del nombre del esquema del sistema de almacenamiento de archivo. El sufijo "Almacén" - (Store) se adjunta al nombre del esquema del sistema de almacenamiento de archivo para generar el nombre del esquema SQL.

56

Por ejemplo, el esquema del sistema de almacenamiento de archivos genera objetos en el Esquema SQL [Sistema-Almacenamiento Almacén - System.Storage.Store] como un ítem [Sistema-Almacenamiento Almacén - System.Storage.Store]."

[0053] El contenido del sistema de almacenamiento de archivo se puede acceder por las vistas. Las vistas descritas a continuación son de sólo lectura, pero el sujeto del invento no se encuentra tan limitado, ya que las vistas se pueden autorizar con escritura. Cada tipo de ítem puede ser mapeado a un tipo de vista. Cada tipo de vista del ítem se puede identificar en un almacén del sistema de almacenamiento de archivo utilizando la convención para el nombre [nombre de esquema <schema name> Almacén Store].[<nombre de tipo de ítem - Item type name>] Un tipo de vista para el tipo T puede retornar todos los ítems que son de tipo T y todos los tipos que se deriven de T. La vista que corresponde al tipo de Sistema Almacenamiento Ítem - System.Storage.Item es [Sistema.Almacenamiento.Almacén].[Ítem]. Esta vista puede retornar todos los ítems a un almacén de sistema de almacenamiento de archivos. La siguiente table describe las columnas de un tipo de vista de ítem:

Nombre de Columna	Tipo	Descripción
Itemid - Identificación del ítem	[Sistema.Almacenamiento.Almacén / Itemid - Identificación del ítem]	Identificación
Typeid - Tipo de identificación	[Sistema.Almacenamiento.Almacén / typeid - Tipo de identificación]	Esquemático
NameSpaceName - Nombre espacio	nvarchar(255)	El nombre único del ítem

53

Nombre		
ContainerId – Contenedor de la Identificación	[Sistema.Almacenamiento.Almacén] ItemId - Identificación del Item	Identificación del Contenedor del Item
Item	Tipo de Item	Instancia del Tipo de Item
EntityState – Estado Entidad	[Sistema.Almacenamiento.Almacén]. Estado Entidad – EntityState	El estado de la información para un item
ObjectSize – Tamaño de Objeto	Ent Grande	Tamaño del Objeto Item (columna persistida)
ChangeInformation – Cambiar Información	[Sistema.Almacenamiento.Almacén]. Cambiar Información – ChangeInformation	Cambiar la Información de rastreo
ItemSyncMetadata – Sincronizar Metadata del Item	[Sistema.Almacenamiento.Almacén]. Sincronizar Metadata del Item – ItemSyncMetadata	Sincronizar metadata
PathHandle – Manija de Ruta	[Sistema.Almacenamiento.Almacén]. Manija de Ruta del Bin-BinPathHandle	Manija de la ruta del Item
PromotionStatus – Promoción Estado	Ent	El estado de la promoción del

Tabla 2

[0054] Cada tipo de enlace queda mapeado a una vista tipo. Cada tipo de vista se identifica en un almacén de sistema de almacenamiento de archivo utilizando la convención de nombre [-schema name - nombre de

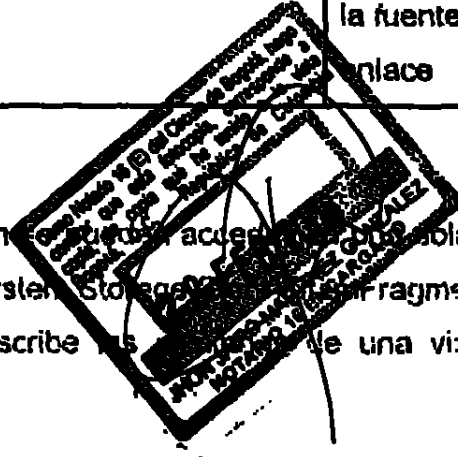
54

esquema>.Almacén - Store].[-enlace con tipo de nombre - link type name>]. La vista que corresponde al tipo de enlace de Sistema,Almacenamiento es [Sistema Almacenamiento Almacén - System Storage.Store].[Link - Enlace]. Esta vista puede incluir todos los enlaces en un almacén de sistema de almacenamiento de archivo. La tabla siguiente describe las columnas de un tipo de vista de enlace:

Nombre de Columna	Tipo	Descripción
SourceRef - Ref Fuente	[Sistema.Almacenamiento.Almacén]. Identificación del ítem	Identificación del ítem del ítem fuente
LinkId - Identificación del Enlace	[Sistema.Almacenamiento.Almacén]. Identificación del Enlace	Identificación del Enlace
TargetRef - Referencia Objetivo	[Sistema.Almacenamiento.Almacén]. Identificación del ítem	Identificación del ítem del ítem objetivo
TypeId - Tipo de Identificación	[Sistema.Almacenamiento.Almacén]. Tipo de Identificación	Tipo de Identificación del tipo de enlace de derivación
Link - Enlace	Tipo de Enlace	Instancia del tipo de enlace
EntityState - Estado	[Sistema.Almacenamiento.Almacén].	El estado de

de la Entidad	Estado de la Entidad	la información para el enlace
ObjectSize – Tamaño de Objeto	Ent Grande	Tamaño del objeto enlace (columna persistida)
ChangeInformation – Cambiar Información	[Sistema.Almacenamiento.Almacén]. Cambiar Información	Cambiar la información de rastreo
LinkSyncMetadata – Sincronizar Metadata del Enlace	[Sistema.Almacenamiento.Almacén]. Sincronizar Metadata del ítem	Sincronizar metadata
PathHandle – Manija de Ruta	[Sistema.Almacenamiento.Almacén]. Manija de Ruta del Bin-	La manija de la ruta al ítem que es la fuente del enlace

Tabla 3

[0055] Todos los fragmentos del ítem  pueden acceder a la vista [Sistema.Almacenamiento.Almacén - System Storage - Fragment - Fragmentodelítem]. La siguiente tabla describe las vistas de una vista de fragmento de ítem global:

Nombre de Columna	Tipo	Descripción
ItemId - Identificación	[Sistema Almacenamiento.Almacén].	Identificación

51

Idem	Identificación del Item	Idem del item propietario
SetId - Identificación del Conjunto	[Sistema.Almacenamiento.Almacén]. Identificación del Conjunto	La utilización del fragmento del item
FragmentId - Identificación del Fragmento	[Sistema.Almacenamiento.Almacén]. Identificación del Fragmento	La identificación de la instancia del fragmento
TypeId - Tipo de Identificación	[Sistema.Almacenamiento.Almacén]. Tipo de Identificación	Tipo de Identificación del tipo de instancia de Fragmento de Item de mayor derivación
ItemFragment - Fragmento del Item	[Sistema.Almacenamiento.Almacén]. Fragmento del Item	Instancia del tipo de Fragmento
EntityState - Estado de la Entidad	[Sistema.Almacenamiento.Almacén]. Estado de la Entidad	El estado de información para el enlace
ChangeInformation - Cambiar Información	[Sistema.Almacenamiento.Almacén]. Cambiar Información	Cambiar información



87

		de rastreo
PathHandle - Manija de Ruta	[Sistema.Almacenamiento.Almacén]. Manija de Ruta del Bin	La manija de la ruta al ítem que es la fuente del enlace

Tabla 4

[0056] Cada almacén suministra una vista de extensión global llamada [Sistema.Almacenamiento.Almacén - System.Storage Store].[Extensión de Ítem - ItemExtension]. Las instancias de todos los tipos de extensión se pueden acceder por esta vista. La siguiente tabla describe las columnas en la vista de extensión global:

Nombre de Columna	Tipo	Descripción
Itemid - Identificación del Ítem	[Sistema.Almacenamiento.Almacén]. Identificación del Ítem	Identificación del Ítem de la extensión
Typeid - Tipo de Identificación	[Sistema.Almacenamiento.Almacén]. Tipo de Identificación	Tipo de Identificación esquematizado
ItemExtension - Extensión del Ítem	[Sistema.Almacenamiento.Almacén]. Extensión del Ítem	Instancia del extensión
EntityState - Estado de la Entidad	[Sistema.Almacenamiento.Almacén]. Estado de la Entidad	Estado de la información para la extensión
ObjectSize -	Ent Grande	Tamaño de la

50

Tamaño de Objeto		extension del objeto (columna persistida)
ChangeInformation – Cambiar Información	[Sistema.Almacenamiento.Almacén]. Cambiar Información	Cambiar información de rastreo
PathHandle – Manija de Ruta	[Sistema Almacenamiento.Almacén]. Manija de Ruta	La manija de la ruta al ítem que es la propietaria de la extensión

Tabla 5

[0057] Cada tipo de extensión se mapea a un tipo de vista. Cada vista de extensión tipo se identifica en un almacén de sistema de almacenamiento de archivos utilizando la convención del nombre [<nombre de esquema - schema name>.Almacén. Store].<nombre del tipo de la extensión - extension type name >] La siguiente tabla describe las columnas de una vista de tipo de extensión.

Nombre de Columna	Tipo	
ItemId - Identificación del ítem	[Sistema.Almacenamiento.Almacén]. Identificación del ítem	Identificación
TypeId – Tipo de Identificación	[Sistema Almacenamiento.Almacén]. Tipo de Identificación	Identificación esquemático
ItemExtension – Extension del ítem	Tipo de extensión	Instancia del tipo de

54

		Extensión
ObjectSize – Tamaño de Objeto	Ent Grande	Tamaño de la extensión del objeto (columna persistida)
EntityState - Estado de la Entidad	[Sistema.Almacenamiento Almacén]. Estado de la Entidad	Estado de la información para la extensión
ChangeInformation – Cambiar Información	[Sistema.Almacenamiento.Almacén]. Cambiar Información	Cambiar información de rastreo
PathHandle – Manija de Ruta	[Sistema.Almacenamiento.Almacén]. Manija de Ruta	La manija de la ruta al ítem que es la propietaria de la extensión

Tabla 6

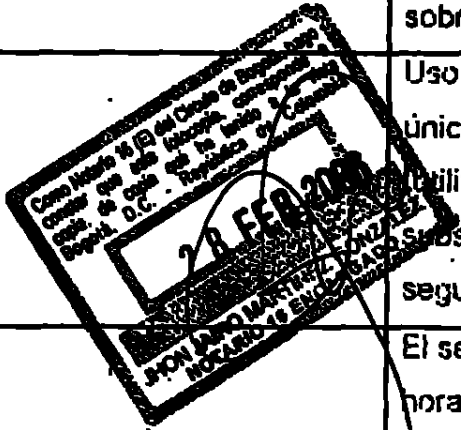
En un sistema de almacenamiento de archivos, se puede almacenar una instancia tipo incluida, en las instancias de tipo de entidad. Se debe comprender que se pueden acceder realizando una búsqueda y lista apropiada.

[0053] Todos los ítems quedan almacenados en la tabla de ítem llamada [Sistema Almacenamiento.Almacén - System.Storage.Store].[Tabla: Table Item - Item]. La única llave en la tabla siguiente es identificación del ítem - ItemId.



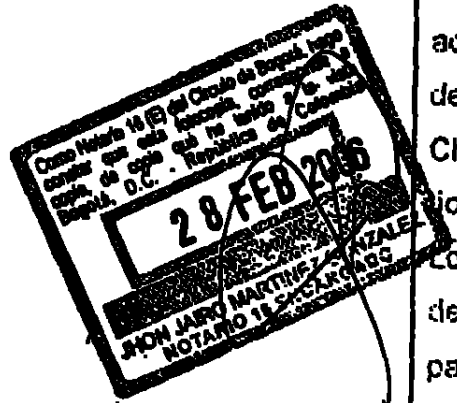
62

Nombre de Columna	Tipo	Descripción
ItemId - Identificación del Ítem	[Sistema.Almacenamiento.Almacén]. Identificación del Ítem	Identificación del Ítem
TypeId - Tipo de Identificación	[Sistema.Almacenamiento.Almacén]. Tipo de Identificación	Tipo de Identificación esquemático
NamespaceName - Nombre Espacio Nombre	Nvarchar(255)	El nombre único del ítem
ContainerId - Contenedor de la identificación	[Sistema.Almacenamiento.Almacén]. Identificación del Ítem	La identificación del contenedor del ítem
Item	Item type	El ítem
EntityState - Estado de la Entidad	[Sistema.Almacenamiento.Almacén]. Estado de la Entidad	El Estado de la Entidad udt que contiene meta información sobre el ítem.
SDId - identificación SD	Ent	Uso interno únicamente utilizado por el subsistema de seguridad)
SDLastUpdateLocal FS - Última actualización del FS Local SD	Ent Grande	El sello con la hora del último cambio de identificación



61

		SD. Utilizado para cambiar rastreo.
ChangeInformation - Cambiar Información	[Sistema.Almacenamiento.Almacén]. Cambiar Información	Cambiar información de rastreo
ItemSyncMetadata - Sincronizar Metadata de la Información	[Sistema.Almacenamiento.Almacén]. Sincronizar Metadata de la Información	Sincronizar metadata para identificaciones globales
TombstoneStatus - Estado Tombstone	Ent	Estado Tombstone
LastUpdateLocalTS - Última actualización del TS Local	Ent Grande	Uso interno únicamente (utilizado para indexar). Esta columna se mapea a Cambiar Información sobre Última actualización del TS Local - ChangeInformation.LastUpdateLocalTS. Se debe mapear para poder crearle un



62

		índice.
MaxOrd	Ent	Uso interno únicamente (usado para computar la Manija de Ruta para los nuevos ítems agregados a este contenedor de ítems)
TypePath -- Tipo de Ruta	Hierarchical_Type_Id -- Tipo de identificación jerárquica	Uso interno únicamente (columna persistida)
ObjectSize -- Tamaño de Objeto	Ent Grande	Tamaño del objeto de ítem (columna persistida)
PathHandle -- Manija de Ruta	[Sistema.Almacenamiento.Almacén] Manija de Ruta de Bin	La Manija de Ruta a este ítem
PromotionStatus -- Estado de Promoción	Ent	El estado de promoción para el ítem
LastAccessTime -- Última Hora de Acceso	Fecha Hora	La última hora de acceso para la secuencia de

		archivo asociada al ítem.
StreamSize – Tamaño de la Secuencia	Ent Grande	El tamaño de la secuencia asociada con el ítem
AllocationSize – Tamaño de la Asignación	Ent Grande	El tamaño asignado para la secuencia asociada con el ítem

Tabla 7

[0069] Los índices en la tabla de ítems se describen a continuación:

Columnas	Unico	Agrupado	Columnas Incluidas
TypePath, ItemId – Tipo de Ruta, Identificación del ítem	Si	Si	
ItemId - Identificación del ítem	Si	No	TombstoneStatus – Estado TypePath – PathHandle – SDId – Identificación de SD
PathHandle - Manija de la Ruta	Si	No	Identificación de SD, TombstoneStatus – Estado Tombstone
ContainerId, NamespaceName –	Si	No	SDId – Identificación de SD, TombstoneStatus – Estado

64

Identificador del Contenedor, Nombre Espacio Nombre			Tombstone
LastUpdateLocalTS - Última actualización del TS Local	No	No	SDId – Identificación de SD, TombstoneStatus – Estado Tombstone, PathHandle – Manja de Ruta

Tabla 3

[0060] La totalidad de los enlaces se almacenan en la tabla de enlaces, la cual se denomina [Sistema.Almacenamiento Almacén - System.Storage.Store].[Tabla! - Table! Enlace - Link]. La única llave en la tabla siguiente es Identificación del ítem - ItemId, Identificación del Enlace - LinkId:

Nombre de Columna	Tipo	Descripción
SourceRef – Ref de la Fuente	[Sistema.Almacenamiento.Almacén]. Identificación del ítem	Identificación del ítem del ítem fuente
LinkId – Identificación del Enlace	[Sistema.Almacenamiento.Almacén]. Identificación del Enlace	Identificación del Enlace
TargetRef – Ref del Objetivo	[Sistema.Almacenamiento.Almacén]. Identificación del ítem	Identificación del ítem del ítem destino
TypeId – Tipo de identificación	[Sistema.Almacenamiento.Almacén]. Tipo de identificación	Identificación esquematizada
TypePath – Tipo	Tipo de identificación jerárquica	Uso interno

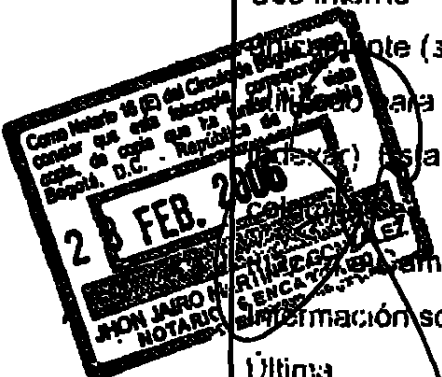
Ruta			Tombstone
TargetRef -- Ref Objetivo	No	No	Identificación de SD, Estado Tombstone
LastUpdateLocalTS - Última actualización del TS Local	No	No	Identificación de SD, Estado Tombstone , Manija de Ruta

Table 10

[0062] La totalidad de las Extensiones de Entidad - EntityExtensions se almacenan en una única tabla denominada: [Sistema.Almacenamiento Almacén].[Tabla] - Table! Extensión de Ítem - ItemExtension]. La siguiente tabla describe la tabla de extensión de ítem:

Nombre de Columna	Tipo	Descripción
ItemId - Identificación del ítem	[Sistema.Almacenamiento.Almacén]. Identificación del ítem	Identificación del ítem de la extensión
TypeId -- Tipo de identificación	[Sistema.Almacenamiento Almacén]. Tipo de identificación	Tipo de identificación esquematizado
TypePath -- Tipo de Ruta	Tipo de identificación jerárquica	Uso interno únicamente (columna oculta)
ItemExtension -- Extensión del ítem	[Sistema.Almacenamiento Almacén]. Extensión del ítem	Objeto de Extensión del ítem
EntityState --	[Sistema.Almacenamiento Almacén]	Estado de la

Estado de la Entidad	Estado de la Entidad	Entidad útil que contiene meta información sobre la Extensión de la Entidad
SDId - Identificación de SD	Ent	Uso interno únicamente
TombstoneStatus - Tombstone	Ent	Estado Tombstone
ObjectSize - Tamaño de Objeto	Ent Granda	Tamaño del Objeto de la Extensión de Entidad (columna persistida)
ChangeInformation - Cambiar Información	[Sistema.Almacenamiento.Almacén]. Cambiar Información	Cambiar la información de rastreo
LastUpdateLocal TS - Última actualización del TS Local	Ent Granda	Uso interno únicamente (sólo para cambiar información sobre Última actualización del TS Local -



		ChangeInformatio n.LastUpdateLoca ITS. Se debe mapear para poder crearle un indice.
PathHandle -- Manija de Ruta	[Sistema.Almacenamiento.Almacén]. Manija de Ruta de Bin	La Manija de Ruta a este ítem que es la fuente del enlace Extensión de la Entidad

Tabla 11

[0063] Los índices de la tabla de Extensión de Ítems se describen a continuación:

Columnas	Unico	Agrupado	Columnas Incluidas
TypePath – Tipo de Ruta, Identificación de ítem - itemId, TypeId - Tipo de identificación	Si	Si	
ItemId – Identificación de ítem, TypeId – Tipo de identificación	No	No	Identificación de SD, Estado Tipo de Ruta
PathHandle - Manija de Ruta	Si	No	Identificación de SD, Estado
LastUpdateLocal TS - Última actualización del TS Local	No	No	Identificación de SD, Estado Manija de Ruta

22

Tabla 12

[0064] La totalidad de las instancias de tipo de Fragmento de ítem se almacenan en una única tabla denominada [Sistema.Almacenamiento.Almacén].[Tabla - Table] Fragmento de ítem - ItemFragment). La siguiente tabla ilustra el tipo de Fragmento de ítem:

Nombre de Columna	Tipo	Descripción
ItemId - Identificación de ítem	[Sistema.Almacenamiento.Almacén]. Identificación de ítem	Identificación de ítem del ítem propietario
SetId - Identificación del conjunto	[Sistema.Almacenamiento.Almacén]. Identificación del conjunto	Utilización de fragmento de ítem
FragmentId - Identificación del fragmento	[Sistema.Almacenamiento.Almacén]. Identificación del fragmento	Identificación de la instancia del fragmento
ItemFragment - Fragmento del ítem	[Sistema.Almacenamiento.Almacén]. Fragmento del ítem	El objeto de Fragmento del ítem
TypeId - Tipo de Identificación	[Sistema.Almacenamiento.Almacén]. Tipo de Identificación	Identificación de tipo de fragmento (columna persistente)
TypePath - Tipo de Ruta	Tipo de identificación para ruta	(columna persistente)
EntityState -	[Sistema.Almacenamiento Almacén].	Estado de la

[Handwritten signature]

Estado de la Entidad	Estado de la Entidad	Entidad unit que contiene meta información sobre el fragmento
SDId - Identificación de SD	Ent	Uso interno únicamente
TombstoneStatus - Estado Tombstone	Ent	Estado Tombstone
ObjectSize - Tamaño del Objeto	Ent Grande	Tamaño del Objeto de Fragmento de item (columna persistida)
ChangeInformation - Cambiar Información	[Sistema.Almacenamiento.Almacén] Cambiar Información	Cambiar la información de rastreo
LastUpdateLocal TS - Última actualización del TS Local	Ent Grande	Uso interno únicamente (sólo utilizado para Esta Esta Cambiar actualización del TS Local - ChangeInformatio



72

		n LastUpdateLocal ITS. Se debe mapear para poder crearle un índice.
PathHandle - Manija de Ruta	[Sistema.Almacenamiento.Almacén]. Manija de Ruta de Bin	La Manija de Ruta al ítem propietario

Tabla 13

[0065] Los índices de la tabla de Fragmento de Ítem se describen a continuación:

Columnas	Unico	Agrupado	Columnas Incluidas
TypePath – Tipo de Ruta, ItemId – Identificación de Ítem, SetId – Identificación del Conjunto, FragmentId – Identificación del Fragmento	Sí	Sí	
ItemId - Identificación de Ítem, SetId – Identificación del Conjunto, FragmentId – Identificación del Fragmento	Sí	No	Identificación de SD, Estado Tombstone, Manija de Ruta
PathHandle - Manija de Ruta	Sí	No	Identificación de SD, Estado Tombstone, Manija de Ruta
LastUpdateLocal ITS - Última actualización del TS Local	No	No	Identificación de SD, Estado Tombstone, Manija de Ruta

73

Tabla 14

[0066] Las Figs 9-10 ilustran metodologías de acuerdo con el sujeto del invento. Para simplicidad en la explicación, las metodologías se ilustran y se describen como una serie de actos. Se entiende que el sujeto del invento no se encuentra limitado por los actos ilustrados y/o por el orden de los mismos, por ejemplo, los actos pueden ocurrir en múltiple orden y/o de manera concurrente, junto con otros actos no descritos en este documento. Adicionalmente, no todos los actos ilustrados pueden ser requeridos para implementar las metodologías según el sujeto del invento. Además, aquellos conocedores en la materia apreciarán que las metodologías pueden ser representados de manera alternativa como una serie de estados interrelacionados vía un diagrama de estado o eventos.

[0067] La Fig. 9 ilustra una metodología 900 que facilita el almacenamiento de una instancia de tipo que mapea a un modelo de información y/o suministra una búsqueda. El modelo de información puede ser un modelo de almacenamiento de información de archivos que permite el almacenamiento, la búsqueda y el racionamiento de la información. Un tipo de información puede ser, pero no se limita a, un documento, una imagen, un video, un contacto, un mensaje, un correo electrónico, un clip de audio, etc. Estos tipos de información (ej., unidades de información) se pueden representar como instancias de tipos complejos que son parte de un sistema tipo que herencia, la cual permite a los objetos heredar atributos y/o comportamientos de otros tipos. También se debe tener en cuenta que dicha herencia puede ser representada en una estructura y/o formato jerárquico.

[0068] En el numeral de referencia 902, la información se puede recibir, la información puede incluir un tipo, un criterio, un esquema, un criterio de búsqueda. La instancia tipo se puede almacenar para mapear un modelo de información

un modelo de información de almacenamiento de archivos) en el numeral de referencia 904. Por ejemplo, el modelo puede mapear tipos descritos en el esquema a los tipos definidos por el usuario y objetos de la base de datos. El mapeo de almacenamiento puede describir los objetos de la base de datos que se generan basados en una definición de esquema y en la forma cómo las instancias tipo descritos en el esquema se almacenan y/o acceden. En un ejemplo, la estructura de la base de datos puede ser diseñada de manera tal que las instancias tipo y las reglas para mapear las declaraciones tipo en objetos de base de datos se puedan proporcionar.

[0069] En el numeral de referencia 906, se puede invocar una búsqueda para satisfacer por lo menos un criterio para encontrar los ítems, documentos y contactos de manera eficiente y efectiva. Por ejemplo, el almacenamiento de la instancia tipo que se relaciona con los tipos de información puede utilizar una búsqueda para suministrar por lo menos uno de entre estos: 1) encontrar por lo menos un ítem en el modelo de información de almacenamiento de archivos 102 que cumpla con ciertos criterios; 2) encontrar por lo menos un documento en modelo de información de almacenamiento de archivos 102 que cumpla con un criterio particular; y 3) encontrar por lo menos un contacto (ej., incluyendo un individuo, una organización y un grupo) que cumpla con ciertos criterios.

[0070] La Fig. 10 ilustra una metodología 1000 que facilita el almacenamiento de una instancia tipo y/o proporcionar información. En el numeral de referencia 1002, la información puede incluir un tipo, un esquema, un entorno de búsqueda, etcétera. Dicha información puede incluir un tipo, un esquema, un entorno de búsqueda, etcétera. En el 1004, un motor de base de datos se puede utilizar para proporcionar un mecanismo para almacenar las instancias tipo y/o proporcionar búsquedas. Por ejemplo, un motor de base de datos relacional se puede utilizar para proporcionar un almacenamiento relacional y una capacidad de búsqueda de relación. El motor de la base de datos relacional puede utilizar múltiples ítems de información.

28

de cómputo que opera en un computador local y/o remoto, aquellos conocedores del tema apreciarán que el invento también puede ser implementado en conjunto con otros módulos de programa. En general, los módulos de programa incluyen rutinas, programas, componentes, estructuras de información, etc., que realicen tareas específicas y/o implementan tipos de información abstractos particulares.

[0073] Adicionalmente, aquellos conocedores del tema apreciarán que los métodos inventivos se pueden realizar con otras configuraciones del sistema de cómputo, incluyendo sistemas de cómputo de procesador único o de procesador múltiple, mini-computadores, procesadores centrales, así como computadores personales, instrumentos manuales, artículos electrónicos a base de microprocesador y/o programables, entre otros, cada uno de los cuales puede comunicar de manera operativa con uno o más instrumentos asociados. Los aspectos ilustrados del invento también se pueden realizar en ambientes de cómputo distribuidos en los que ciertas tareas se efectúan por medio de instrumentos de procesamiento remoto que están conectados en una red de comunicaciones. Sin embargo, algunos, si no la totalidad de los aspectos del invento se pueden efectuar en computadores independientes. En un ambiente de cómputo distribuido, los módulos de programa se pueden ubicar en herramientas de almacenamiento de memoria ya sea locales y/o remotos.

[0074] La Fig. 11 es un diagrama esquemático de un ejemplo de ambiente de cómputo 1100 con el que el sujeto del invento puede interactuar. El sistema 1100 incluye uno o más cliente (s) 1110. Los clientes 1110 pueden ser hardware y/o software (ej., secuencias, procesos, instrumentos de cómputo). El sistema 1100 también incluye uno o más servidores 1120. Los servidores 1120 pueden ser hardware y/o software (ej., secuencias, procesos, instrumentos de cómputo). Los servidores 1120 pueden albergar secuencias para realizar transformaciones utilizando el sujeto del invento, por ejemplo.

22

[0075] Una posible comunicación entre un cliente 1110 y un servidor 1120 puede ser en la forma de un paquete de información adaptado para ser transmitido entre dos o más procesos de cómputo. El sistema 1100 incluye una estructura de comunicación 1140 que puede ser utilizada para facilitar las comunicaciones entre los clientes 1110 y los servidores 1120. Los clientes 1110 se conectan de forma operativa con uno o más almacenes de información de cliente 1150 que pueden ser empleados para almacenar la información local de los clientes 1110. De manera similar, los servidores 1120 están conectados de manera operativa a uno o más almacenes de información del servidor 1130 que puede ser utilizados para almacenar información local a los servidores 1140.

[0076] En la Fig. 12, se hace referencia a un ejemplo de ambiente 1200 para implementar varios aspectos del invento, que incluye un computador 1212. Dicho computador 1212 incluye una unidad de procesamiento 1214, un sistema de memoria 1216, y un sistema bus 1218. Dicho sistema bus 1218 acopla componentes del sistema, incluyendo pero sin limitarse al sistema de memoria 1216 a la unidad de procesamiento 1214. Dicha unidad de procesamiento 1214 puede ser cualquiera de múltiples procesadores disponibles. Los microprocesadores duales y otras arquitecturas de multiprocesador también pueden ser empleados como unidad de procesamiento 1214.

[0077] El sistema bus 1218 puede ser cualquiera de los siguientes tipos de estructura incluyendo la memoria bus o controlador de bus, bus periférico o externo, y/o bus local utilizando cualquier una de las siguientes arquitecturas bus, incluyendo pero sin limitarse a Arquitectura de Sistema Industrial Standard Architecture (ISA), Arquitectura de Micro Channel Architecture (MSA), ISA Extensiva - Extended ISA, Instrumentos Electrónicos Inteligentes - Intelligent Drive Electronics (IDE), Bus Local VESA - VESA Local Bus (VLB), Componente Periférico de Interconexión - Peripheral Component Interconnect (PCI), Tarjeta Bus - Card Bus, Bus Serial Universal -

78

Universal Serial Bus (USB), Puerto Avanzado de Gráficos - Advanced Graphics Port (AGP), Asociación Bus de Tarjetas de Memoria de Computadores Personales - Personal Computer Memory Card International Association bus (PCMCIA), Firewire (IEEE 1394), e Interfase de Sistemas de Pequeños Computadores - Small Computer Systems Interface (SCSI).

[0078] El sistema de memoria 1216 incluye una memoria volátil 1220 y una memoria no volátil 1222. El sistema básico de entrada/salida - basic input/output system (BIOS), que contiene las rutinas básicas para transferir información entre elementos en el computador 1212, como durante el inicio, es almacenado en la memoria no volátil 1222. A manera de ilustración y no de limitante, la memoria no volátil 1222 puede incluir memoria de solo lectura - read only memory (ROM), ROM programable - programmable ROM (PROM), ROM programable electrónicamente - electrically programmable ROM (EPROM), ROM programable borrrable electrónicamente - electrically erasable programmable ROM (EEPROM), o memoria flash. La memoria volátil 1220 incluye memoria de acceso aleatoria - random access memory (RAM), la cual actúa como memoria caché externa. A manera de ilustración y no de limitante, la RAM está disponible en múltiples formas tales como RAM estática - static RAM (SRAM), RAM dinámica - dynamic RAM (DRAM), DRAM sincronizada - synchronous DRAM (SDRAM), SDRAM de velocidad de información doble - double data rate SDRAM (DDR SDRAM), SDRAM mejorada - enhanced SDRAM (ESDRAM), Synchlink DRAM (SLDRAM), Rambus direct RAM (RDRAM), RAM directo Rambus Dynamic RAM (DRDRAM), y RAM Rambus dinámico (RDRAM).

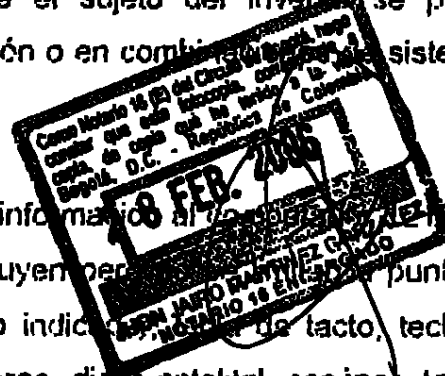
[0079] El computador 1212 también incluye memoria de almacenamiento de cómputo removibles/no-removibles, volátiles/no-volátiles. La Fig. 12 ilustra, por ejemplo, un almacenamiento en disco 1224. Dicho almacenamiento en disco 1224 incluye pero no se limita a instrumentos como la unidad de disco magnético,



unidad de disco flexible, unidad de cinta, unidad Jaz, unidad Zip, unidad LS-100, tarjeta de memoria flash, o tarjeta de memoria. Adicionalmente, el almacenamiento en disco 1224 puede incluir medios de almacenamiento de manera separada o en conjunto con otros medios de almacenamiento que incluyen, pero no se limitan a una unidad de disco óptico tales como una unidad de disco compacto ROM (CD-ROM), una unidad de disco en la que se puede grabar (CD-R Drive), una unidad de CD reescribible (CD-RW Drive) o una unidad de disco versátil digital ROM (DVD-ROM). Para facilitar la conexión de las unidades de almacenamiento en disco 1224 al sistema bus 1213, típicamente se utiliza una interfase removible o no-removible, como una interfase 1226.

[0080] Se puede apreciar que la Fig. 12 describe un software que actúa como un intermediario entre los usuarios y los recursos básicos de cómputo descritos en el ambiente de operación adecuado 1200. Este software incluye un sistema de operación 1228. Dicho sistema de operación 1228, el cual puede ser almacenado en un almacenamiento en disco 1224, actúa para controlar y asignar recursos del sistema de cómputo 1212. Las aplicaciones de sistema 1230 aprovechan la administración de recursos por el sistema de operación 1228 a través de módulos de programa 1232 y la información de programa 1234 almacenada ya sea en el sistema de memoria 1216 o en el almacenamiento en disco 1224. Se debe tener en cuenta que el sujeto del invento se puede implementar en múltiples sistemas de operación o en combinación de sistemas de operación.

[0081] Un usuario ingresa comandos o información al sistema 1200 por los instrumentos de entrada 1236. Estos incluyen, pero no se limitan a, punteros tales como el ratón, bola de mando, puntero indicador, puntero de tacto, teclado, micrófono, palanca de juegos, teclado de juegos, disco satelital, escáner, tarjeta de sintonización de TV, cámara digital, cámara digital de video, cámara web, entre otros. Estos y otros instrumentos de entrada se conectan a la unidad 1212.



procesamiento 1214 por el sistema bus 1218 vía puertos de interfase 1238. Los puertos de interfase 1238 incluyen, por ejemplo, un puerto serial, un puerto paralelo, un puerto de juegos, y un bus serial universal - universal serial bus (USB). Los instrumentos de salida 1240 utilizan algunos de los mismos tipos de puertos que los instrumentos de entrada 1236. En consecuencia, por ejemplo, el puerto USB puede ser utilizado para proveer entrada al computador 1212, y para sacar información de dicho computador 1212 a un instrumento de salida 1240. Un adaptador de salida 1242 se proporciona para ilustrar que existen algunos instrumentos de salida 1240 tales como monitores, parlantes e impresoras, entre otros instrumentos de salida 1240, los cuales requieren de adaptadores especiales. Los adaptadores de salida 1242 incluyen, a manera de ilustración pero no de limitante, tarjetas de sonido y de vídeo que pueden proporcionar un medio de conexión entre el instrumento de salida 1240 y el sistema bus 1218. Se debe tener en cuenta que otros instrumentos y/o sistema de instrumentos proporcionan capacidades tanto de entrada como salida, tales como computadores remotos 1244.

[0082] El computador 1212 puede operar en un ambiente de red utilizando conexiones lógicas a uno o más computadores remotos, tales como los computadores remotos 1244. Estos computadores remotos 1244 pueden ser un computador personal, un servidor, un enrutador, una red de PC, una estación de trabajo, un electrodoméstico basado en un microprocesador, un instrumento de par u otro nodo de red común, entre otros, y por lo general, uno o más de los cuales o la totalidad de los elementos descritos relativos al computador 1212, no se describen en breves, sólo se ilustra un instrumento de almacenamiento de datos 1244, tal como un computador remoto 1244. Dicho computador remoto 1244 está lógicamente al computador 1212 a través de la interfase de red 1248. La interfase de red 1248 incluye redes de comunicación alámbricas y/o inalámbricas, tales como redes de área local - local-area networks (LAN) y redes de área extensas - wide-

area networks (WAN) Las tecnologías LAN incluyen una Interfase de Información Distribuida en Fibra - Fiber Distributed Data Interface (FDDI), una Interfase de Información Distribuida en Cobre - Copper Distributed Data Interface (CDDI), Ethernet, Token Ring, entre otros. Las tecnologías WAN incluyen pero no se limitan a enlaces punto a punto, redes de cambio de circuito como Redes Digitales de Servicios Integrados - Integrated Services Digital Networks (ISDN) y sus variaciones, redes de cambio de paquetes y Líneas de Suscriptor Digital - Digital Subscriber Lines (DSL).

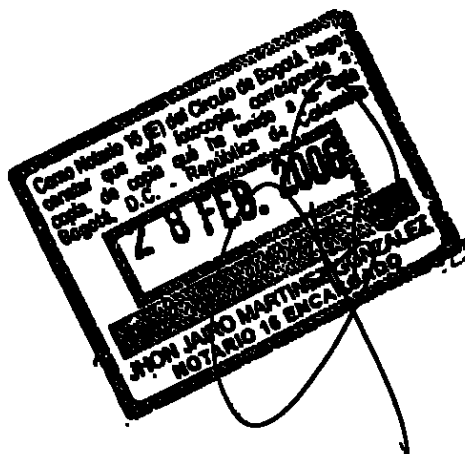
[0083] Las conexiones de comunicación 1250 hacen referencia al hardware/software utilizado para conectar la interfase de red 1248 al bus 1218. Mientras la conexión de comunicación 1250 se exhibe para dar claridad en la ilustración adentro del computador 1212, también puede ser externo al computador 1212. El hardware/software necesario para la conexión a la interfase de red 1248 incluye, únicamente para fines de mostrarlo como ejemplo, tecnologías internas y externas tales como módems, incluyendo módems telefónicos corrientes, cable módems y módems DSL, adaptadores ISDN y tarjetas Ethernet.

[0084] Lo previamente descrito incluye ejemplos del sujeto del invento. Obviamente no es posible describir todas las combinaciones que existen de componentes o de metodologías para describir dicho invento, pero una persona con conocimiento corriente en el tema puede reconocer que existen muchas otras combinaciones y permutaciones del sujeto de la invención. Tanto, se entiende que el invento intenta cubrir todas estas combinaciones y variaciones que recaen en el espíritu y el alcance de la invención.

[0085] Específicamente y con respecto a las modificaciones de los componentes, instrumentos, circuitos, sistemas, entre otros, previamente descritos, los términos (incluyendo la referencia a "medios") utilizados para describir tales

componentes deben corresponder, salvo que se indique lo contrario, a cualquier componente que realice la función especificada del componente descrito (ej., un equivalente funcional), a pesar de que no sea equivalente estructuralmente a la estructura mencionada, que realiza la función en los ejemplos aquí ilustrados como aspectos del invento. En cuanto a esto, también se debe tener en cuenta que el invento incluye un sistema y un medio reconocible por el computador con instrucciones ejecutables por el computador para la realización de las acciones y/o eventos de los múltiples métodos del invento.

[0086] Adicionalmente, aunque se haya mencionado una característica específica del invento con respecto a sólo una de múltiples implementaciones, dicha característica puede ser combinada con una o más características de las otras implementaciones como se desee y en la manera en que sea ventajoso para cualquier aplicación dada o específica. Adicionalmente, en la medida en que el término "incluye" e "incluyendo" y sus variaciones en la descripción detallada o en las reivindicaciones, es la intención que dicho término sea inclusive, de manera similar al término "comprende".



6. El sistema de la reivindicación 5, en el que el componente relacional utiliza una técnica de base de datos relacional que es una colección de ítems de información organizada como un conjunto de tablas descritas formalmente, en las cuales la información puede ser por lo menos uno accedido sin reorganización y reensamblada sin reorganización.

7. El sistema de la reivindicación 1, en el que el mapeo describe por lo menos uno de los objetos de la base de datos generados en el esquema y cómo la instancia de tipo descrita en dicho esquema se almacena y se accede.

8. El sistema de la reivindicación 1, en el que el tipo en el esquema mapea a una clase de tiempo de operación común - common language runtime (CLR) en un almacenamiento del modelo de información de almacenamiento de archivo.

9. El sistema de la reivindicación 8, en el que el tipo es por lo menos uno de los siguientes: un ítem; una extensión de ítem; un fragmento de ítem; y un enlace.

10. El sistema de la reivindicación 1, que además incluye una tabla que contiene por lo menos uno de los siguientes: 1) una columna que contiene por lo menos una instancia del tipo respectivo; y 2) una fila que contiene una representación serializada de la instancia de clase de CLR, que presenta la instancia tipo.

11. El sistema de la reivindicación 1, que además incluye una vista de vista que genera una vista que exhibe por lo menos una instancia de tipo, en la que la vista proyecta un subconjunto del respectivo tipo, asociada con el tipo base.

12. El sistema de la reivindicación 1, en el que el tipo contiene por lo menos uno de los siguientes: un tipo de jerarquía y un legado.



13. Un medio reconocible por el computador que ha almacenado los componentes del sistema de la reivindicación 1

14. Un método implementado en el computador que facilita el mapeo de un modelo de información que incluye:

obtener un modelo de información de almacenamiento de archivo;

recibir por lo menos uno de entre un esquema, un tipo, un criterio y una información de búsqueda;

mapear un objeto de esquema a un objeto de base de datos basado en el esquema asociado al modelo de información de almacenamiento de archivo; y

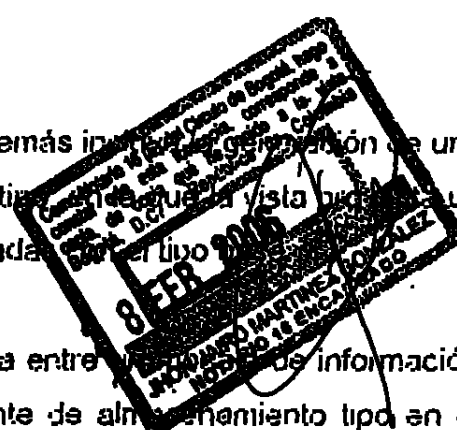
almacenar una instancia de tipo del esquema.

15. El método de la reivindicación 14, que además incluye una búsqueda para encontrar por lo menos uno de los siguientes: por lo menos un ítem en el sistema que cumpla con ciertos criterios; por lo menos un documento en el sistema que cumpla con ciertos criterios; y por lo menos un contacto que cumpla con ciertos criterios.

16. El método de la reivindicación 14, que además incluye la utilización de un motor de base de datos relacional para proporcionar un almacenamiento relacional y una capacidad de búsqueda relacional.

17. El método de la reivindicación 14, que además incluye la generación de una vista que exhibe por lo menos una instancia de tipo, donde la vista proporciona un subconjunto del tipo respectivo de la vista asociada.

18. Un paquete de información que comunica entre un componente de almacenamiento de archivo y un componente de almacenamiento tipo en el cual el paquete de información facilita el método de la reivindicación 14



66

19. Un sistema implementado en el computador que facilita el mapeo de un modelo de información que incluye:

medios para recibir por lo menos uno de los siguientes: un esquema, un tipo, un criterio y un criterio de búsqueda; y

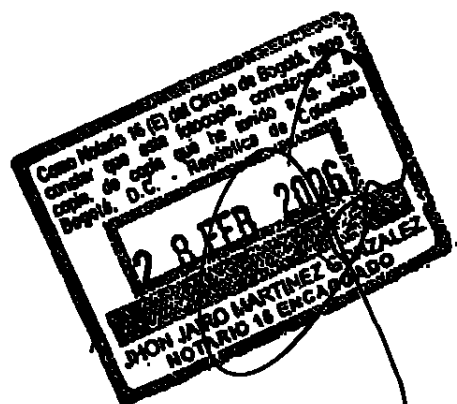
medios para mapear un objeto de esquema a un objeto de base de datos basado en un esquema asociado con el modelo de información de almacenamiento de archivo, en el que se almacena una instancia tipo.

20. El sistema de la reivindicación 11, en el que el componente de vista invoca a una interfase de usuario a interactuar entre un usuario y el componente de almacenamiento tipo.



COMPENDIO

El sujeto del invento proporciona un sistema y/o método que facilita mapear un modelo de base de datos a un objeto de base de datos. Un sistema de almacenamiento tipo puede utilizar un mapeo de almacenamiento de un modelo de información de almacenamiento de archivo. El mapeo puede describir el objeto de la base de datos creado basado al menos parcialmente en un esquema y en cómo las instancias de tipo descritas en el esquema se almacenan y/o se acceden. Adicionalmente, se puede proporcionar una búsqueda para encontrar por lo menos uno de entre un ítem, un documento, y/o un contacto que cumplan con por lo menos un criterio. El sistema de almacenamiento tipo puede recibir información, por el cual la información es por lo menos uno de entre un esquema, un modelo de información, un tipo, una búsqueda, un criterio de búsqueda vía una interfase para suministrar el almacenamiento y la búsqueda. Adicionalmente, el sistema de almacenamiento tipo puede generar una vista que exhibe por lo menos una instancia de tipo.



66

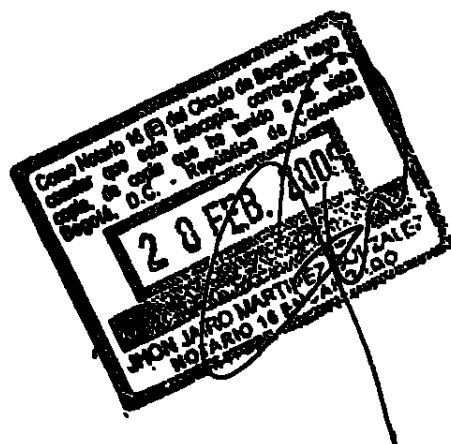
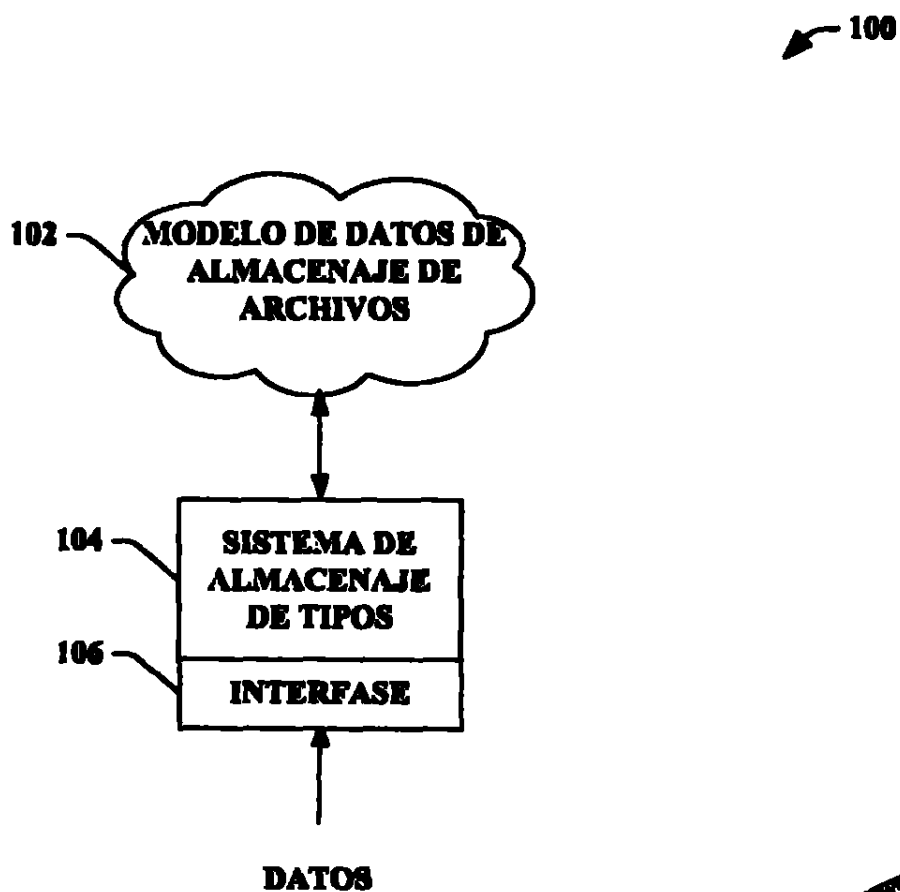


FIG. 1

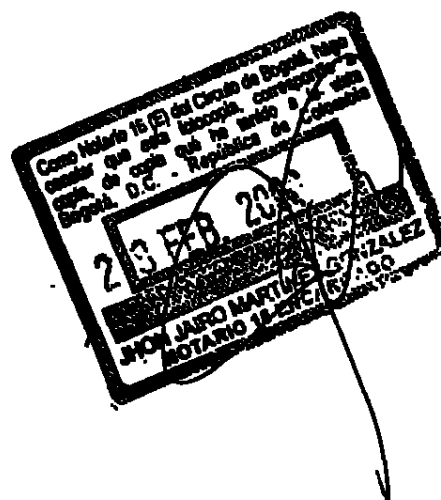
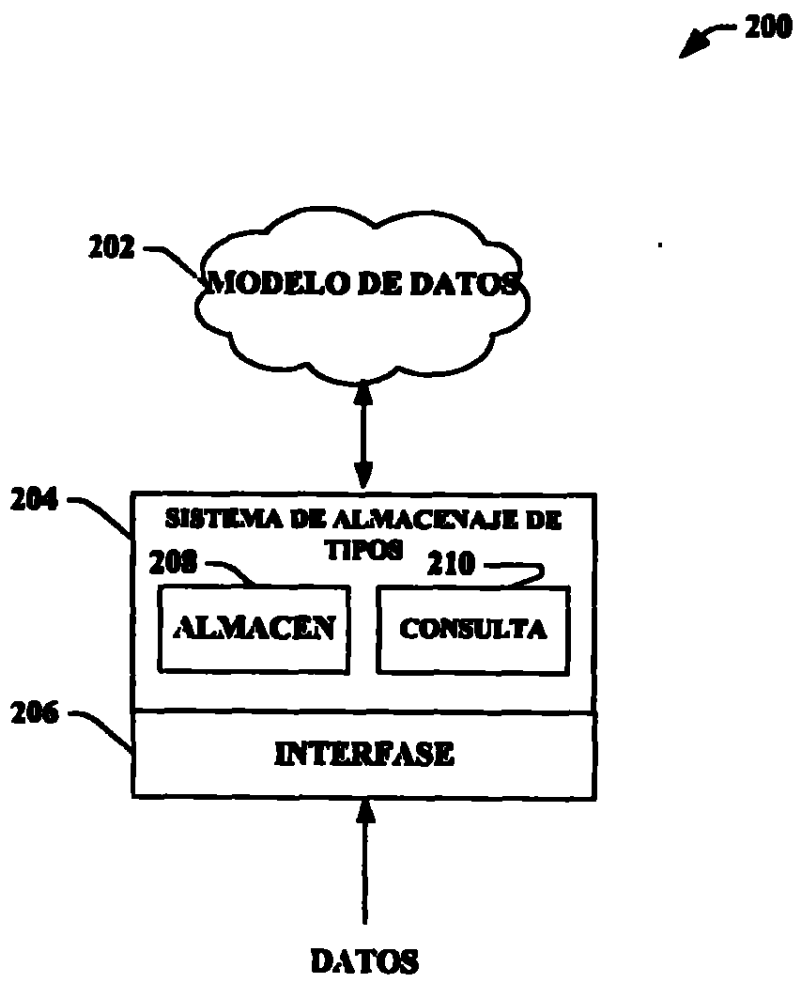


FIG. 2

20

3/12

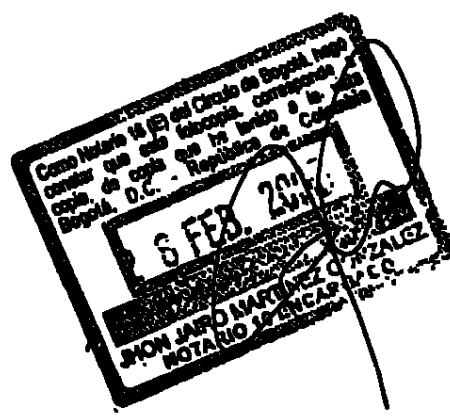
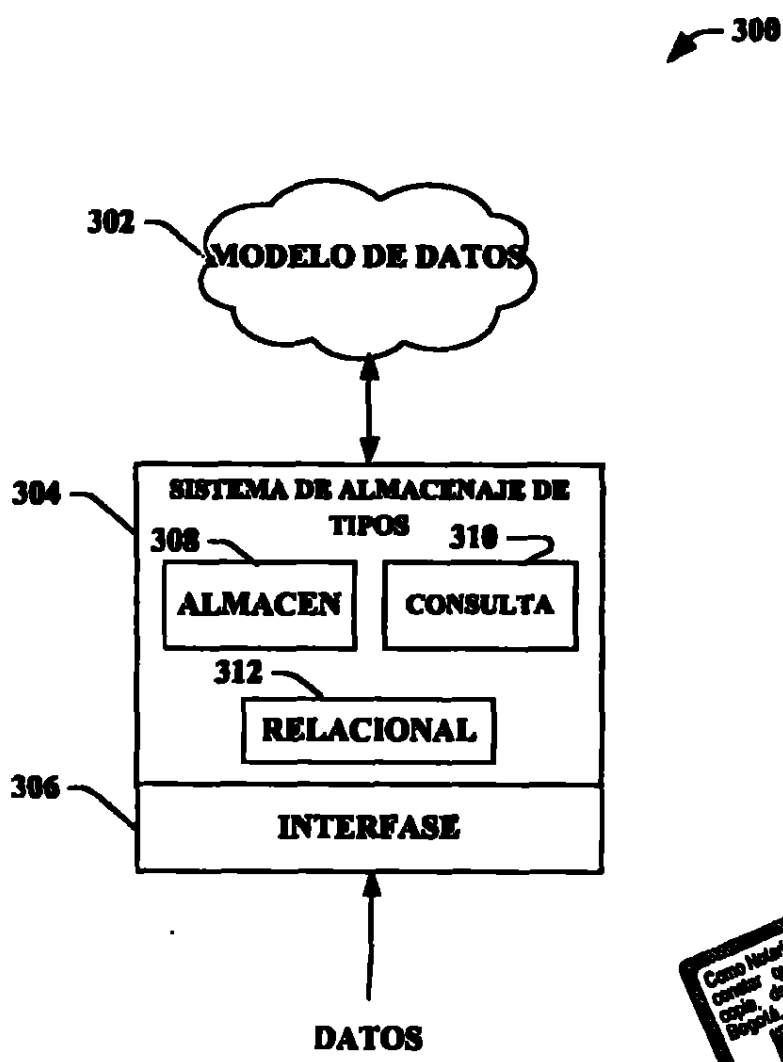


FIG. 3

et

4/12

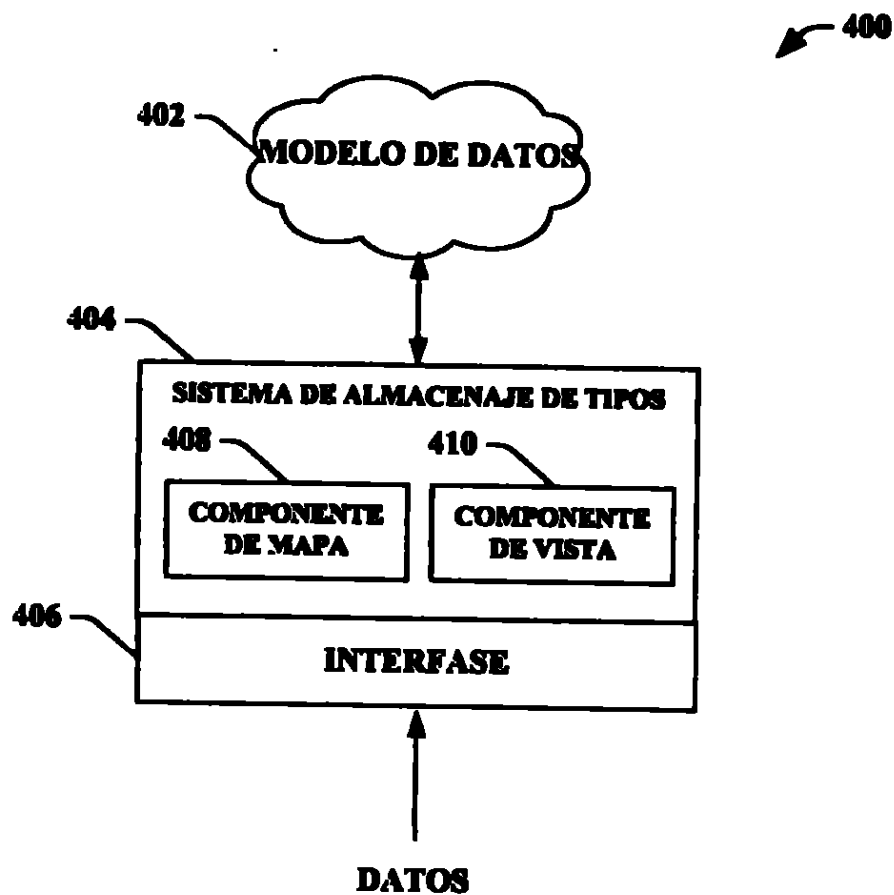
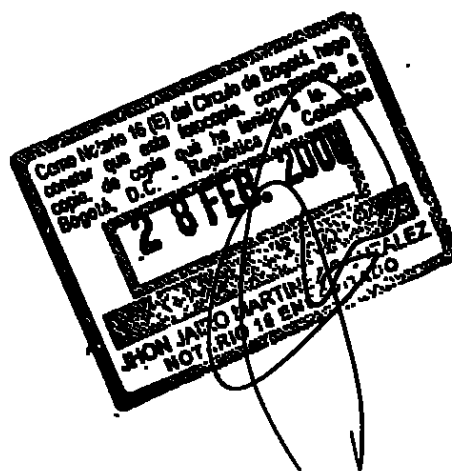


FIG. 4



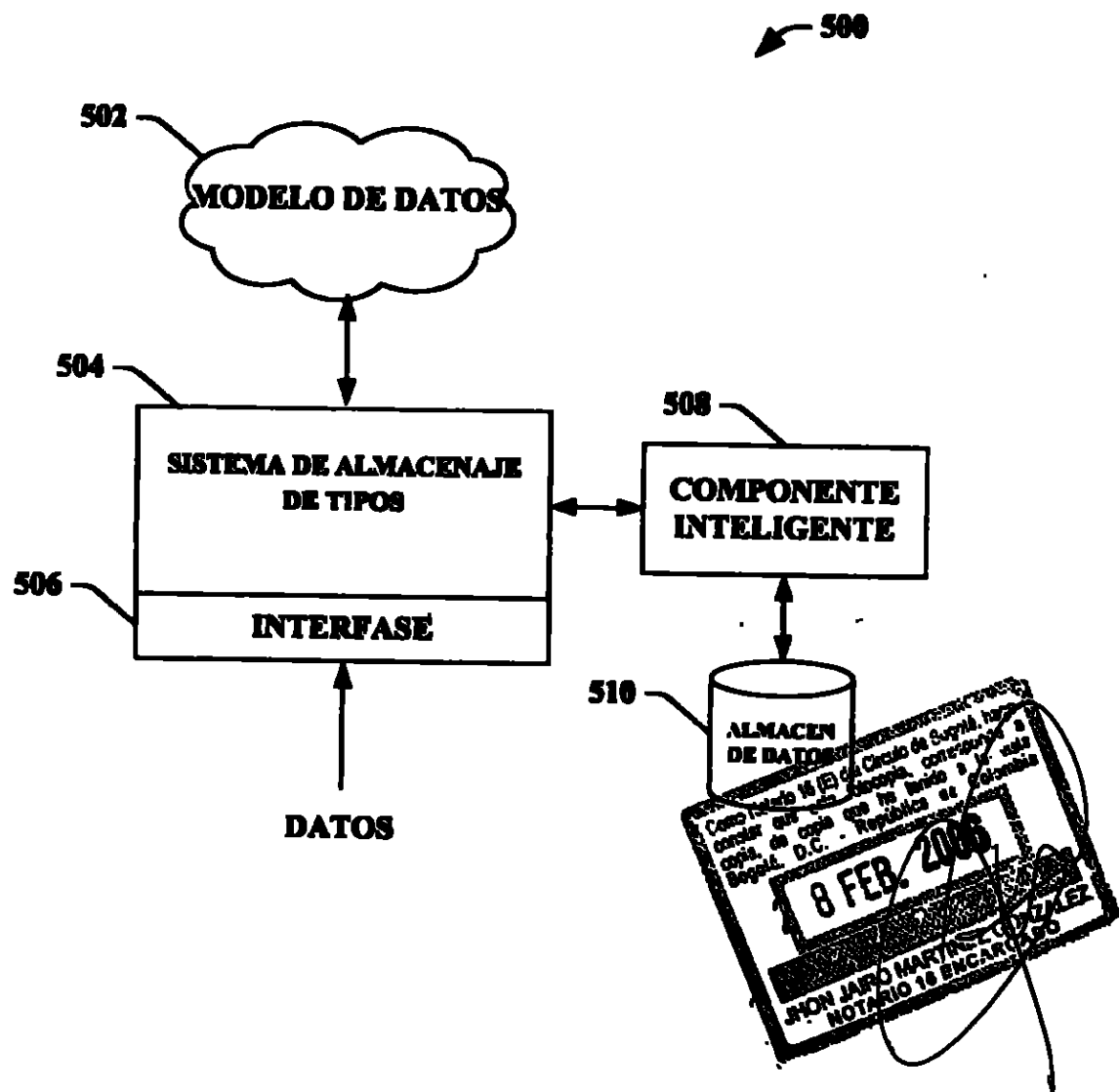


FIG. 5

[Handwritten signature]

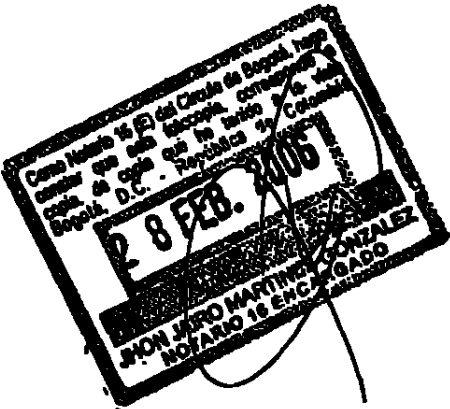
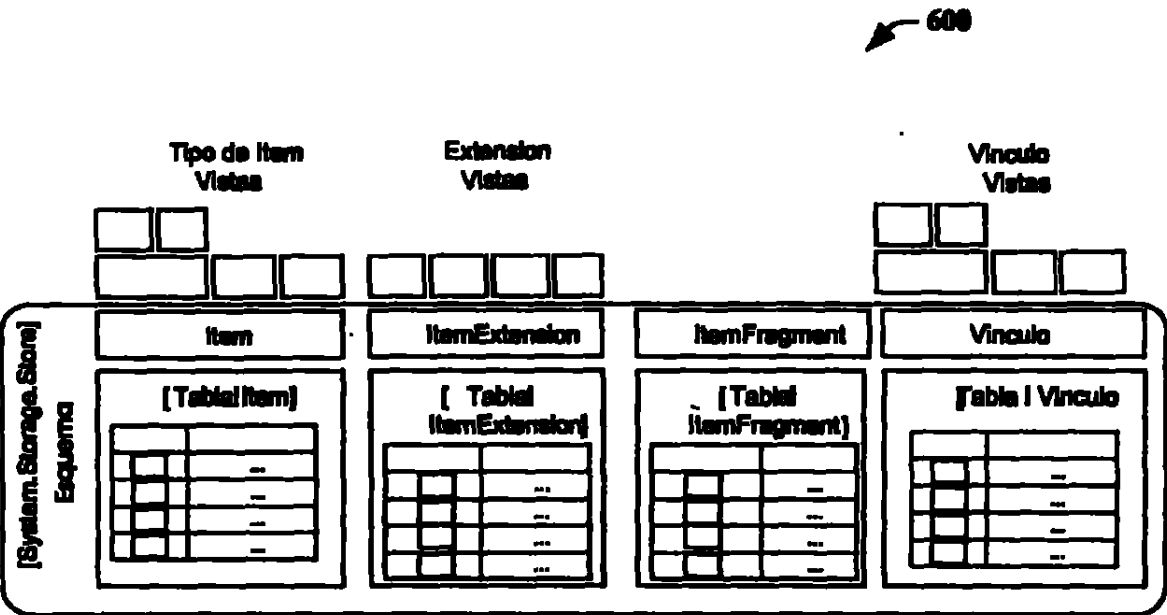


FIG. 6

94

7/12

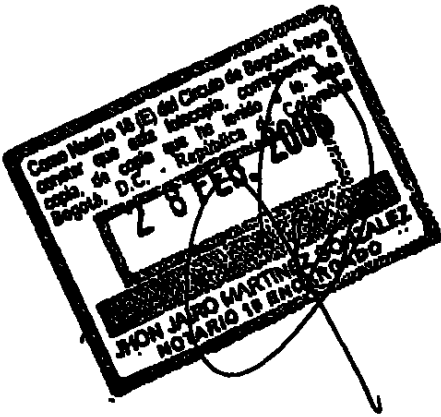
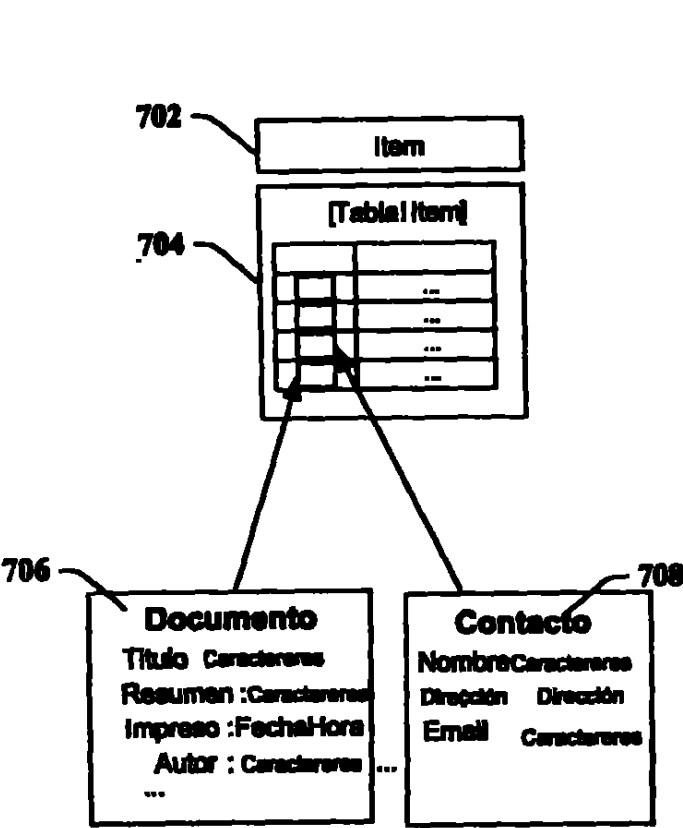


FIG. 7

TS

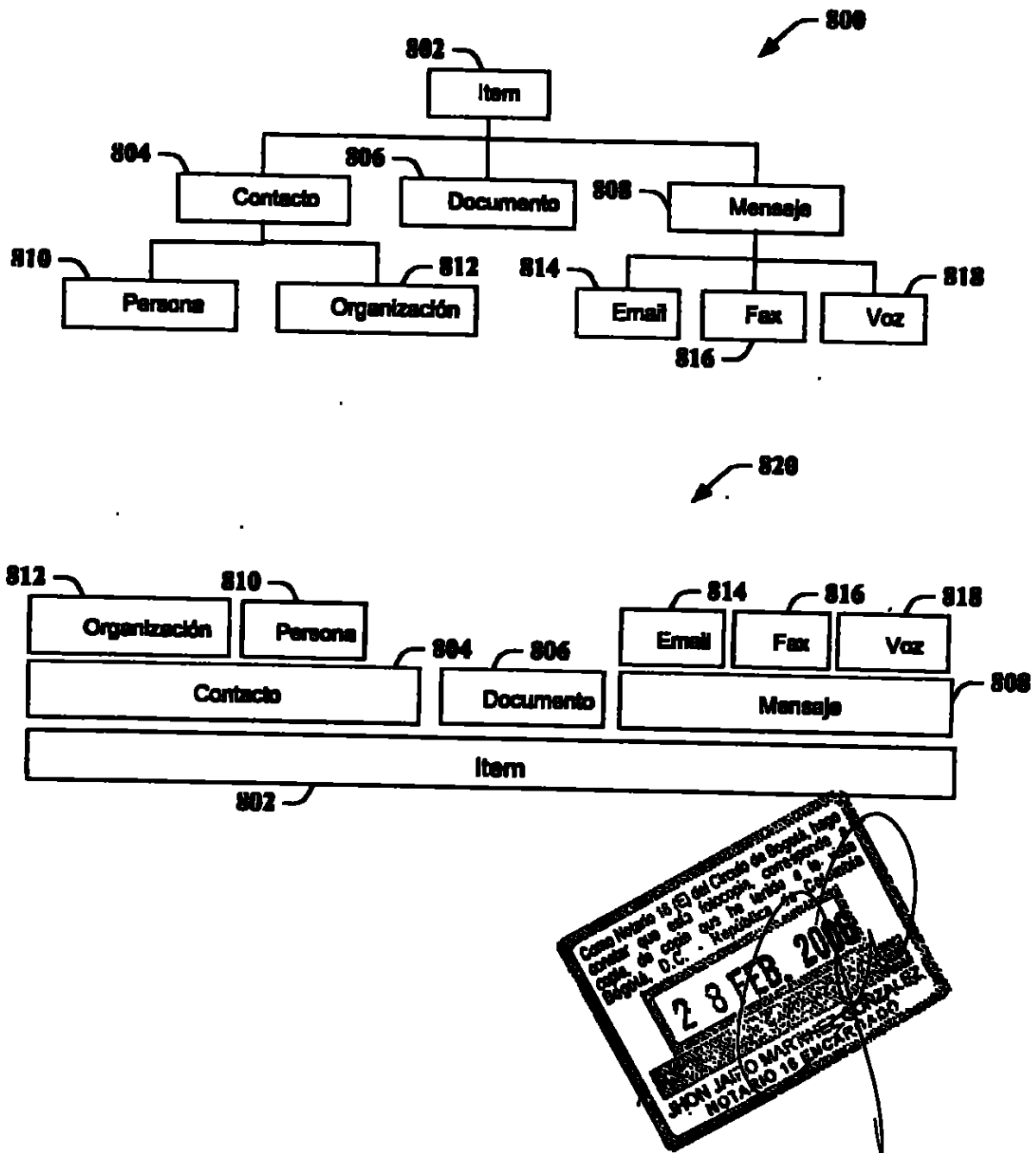


FIG. 8

9/12

900

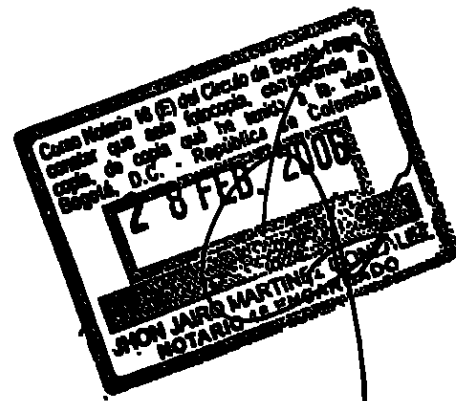
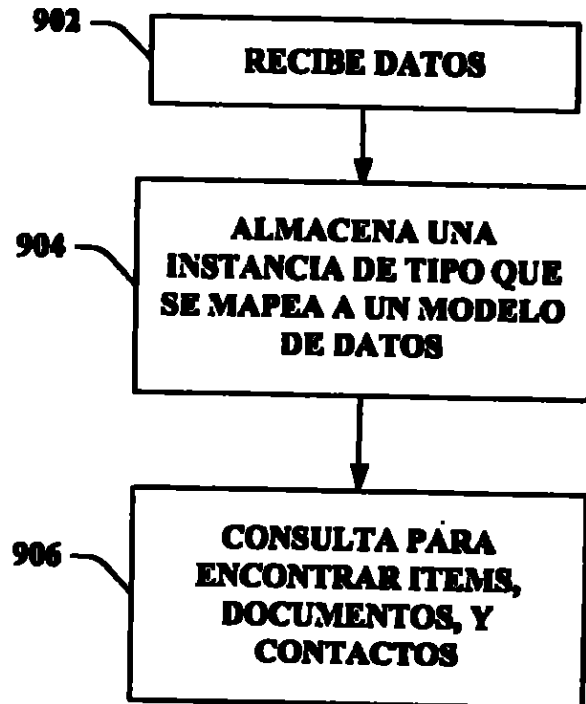


FIG. 9

92

1000

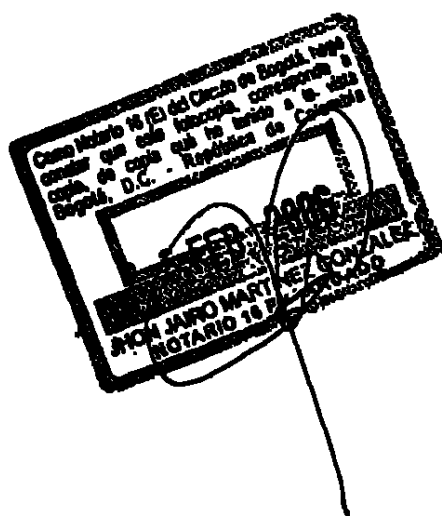
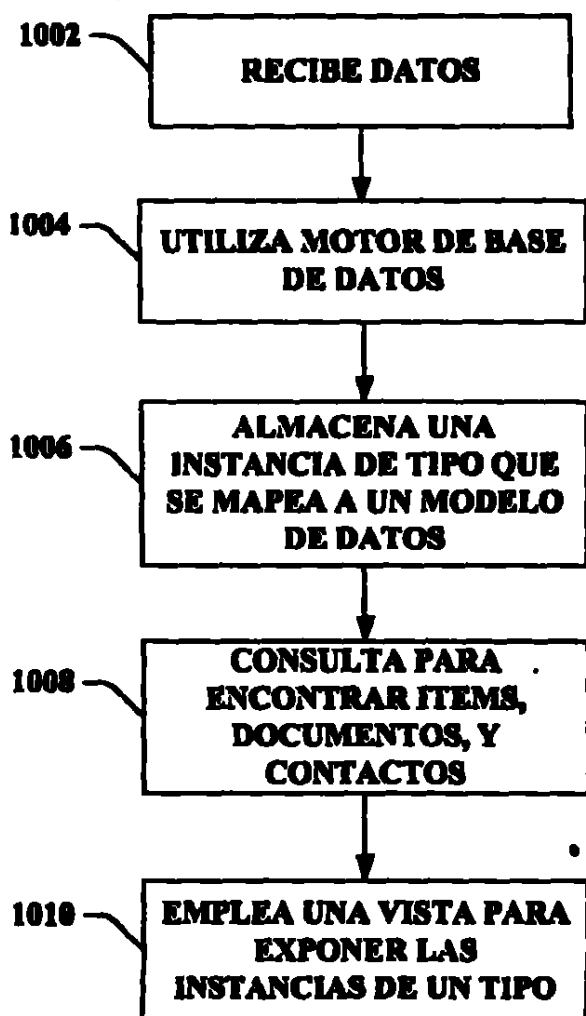


FIG. 10

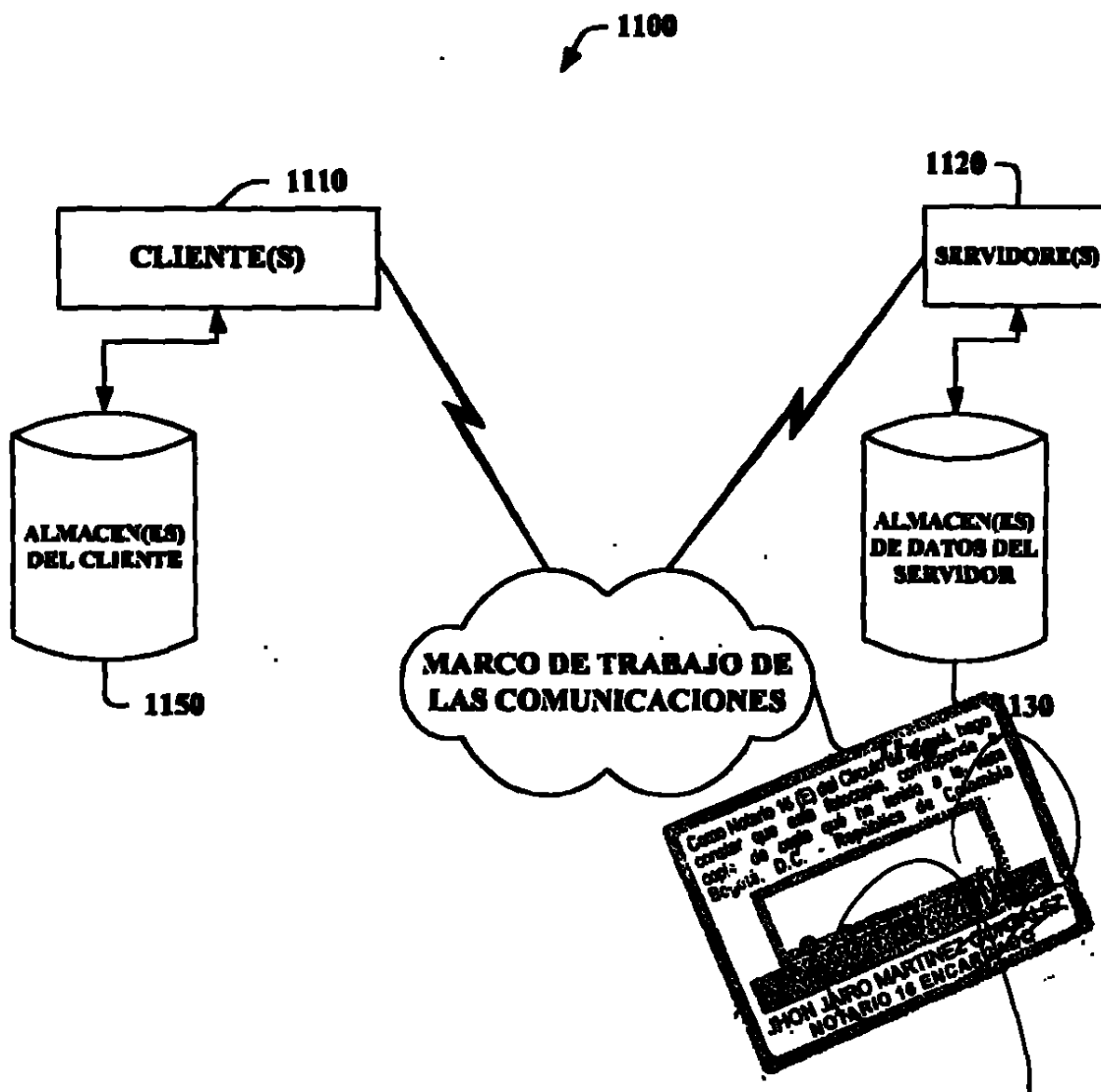


FIG. 11

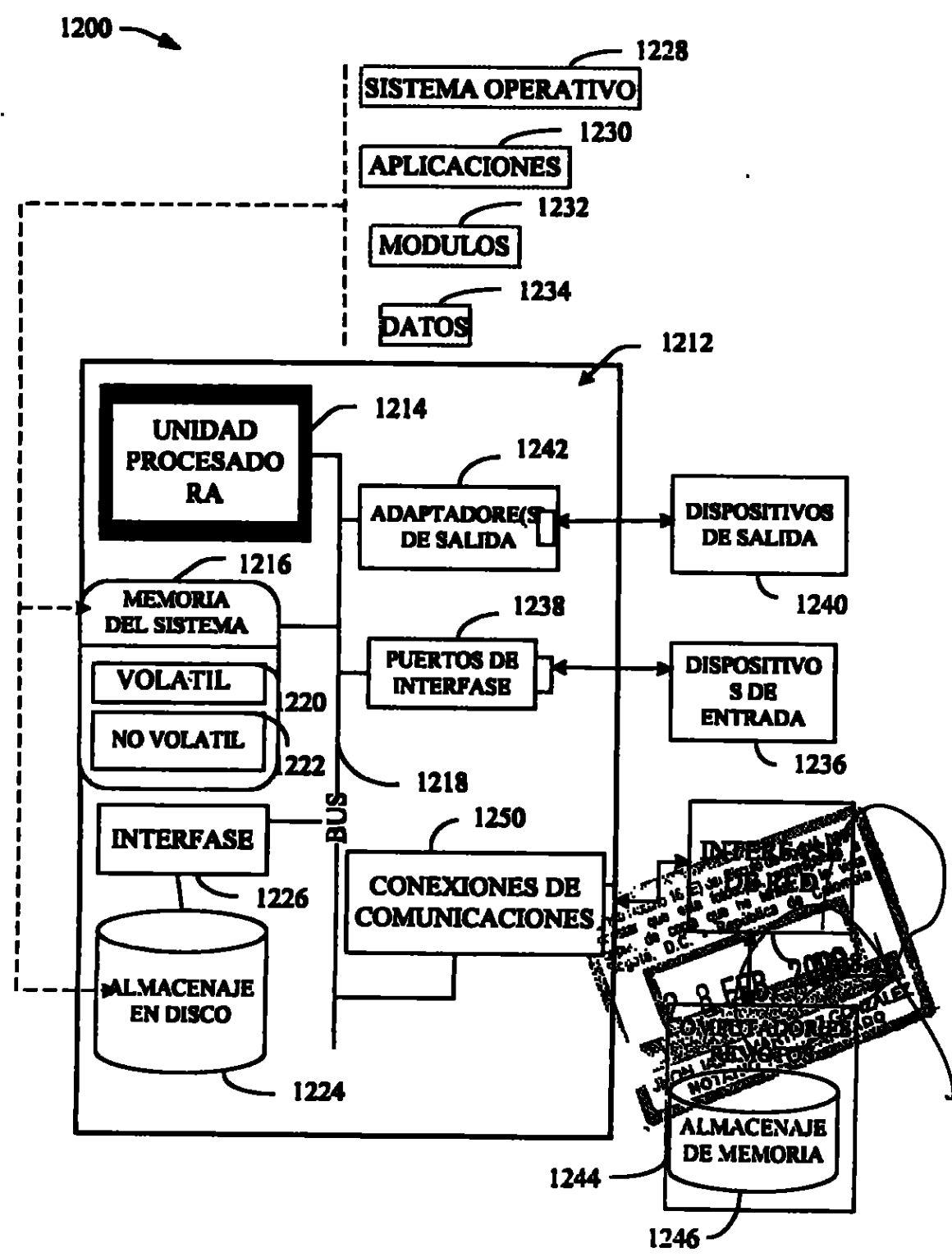


FIG. 12

PM 1406792



THE UNITED STATES OF AMERICA

TO ALL TO WHOM THESE PRESENTS SHALL COME:

UNITED STATES DEPARTMENT OF COMMERCE

United States Patent and Trademark Office

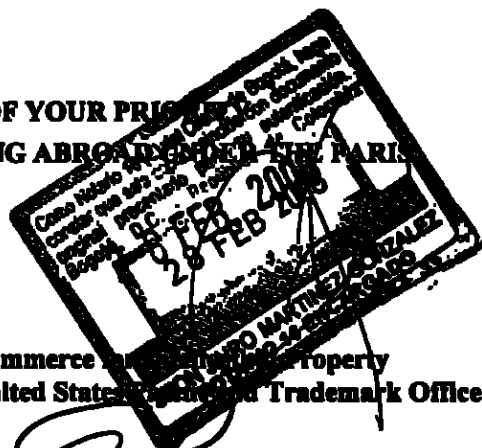
December 22, 2005

THIS IS TO CERTIFY THAT ANNEXED HERETO IS A TRUE COPY FROM THE RECORDS OF THE UNITED STATES PATENT AND TRADEMARK OFFICE OF THOSE PAPERS OF THE BELOW IDENTIFIED PATENT APPLICATION THAT MET THE REQUIREMENTS TO BE GRANTED A FILING DATE UNDER 35 USC 111.

APPLICATION NUMBER: 11/091,079

FILING DATE: March 28, 2005

THE COUNTRY CODE AND NUMBER OF YOUR PRIOR APPLICATION, TO BE USED FOR FILING ABROAD UNDER THE PARIS CONVENTION, IS US11/091,079



**By Authority of the
Under Secretary of Commerce and Director of the United States Patent and Trademark Office**



E. J. Burnett
E. BORNETT
Certifying Officer

20427 USPTO

PTO/SB/05 (04-04)

Approved for use through 07/31/2008. CMS 0001-0002
U.S. Patent and Trademark Office, U.S. DEPARTMENT OF COMMERCE

Under the Privacy Protection Act of 1974, no person or organization is required to furnish information to a collection of information unless it displays a valid OMB control number.

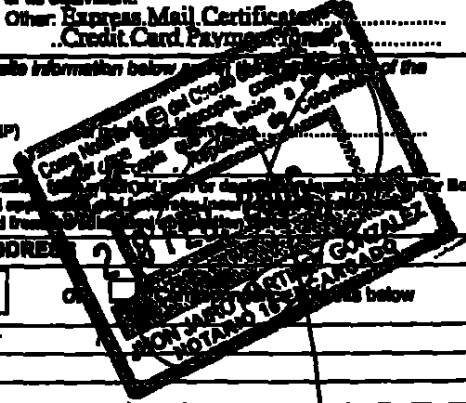
UTILITY PATENT APPLICATION TRANSMITTAL

(Only for new nonprovisional applications under 37 CFR 1.53(a))

Attorney Docket No.	MS311235.01/MSFTP852US
First Inventor	Nigel R. Ellis, et al.
Title	MAPPING OF A FILE...
Express Mail Label No.	EV628834791US

1132 USPTO
11/09/07

APPLICATION ELEMENTS See MPEP chapter 609 concerning utility patent application contents.	ADDRESS TO: Commissioner for Patents P.O. Box 1480 Alexandria VA 22313-1480
1. ☒ Fee Transmittal Form (e.g., PTO/SB/17) (Submit an original and a duplicate for fee processing) 2. ☐ Applicant claims small entity status. See 37 CFR 1.27. 3. ☒ Specification [Total Pages <u>35</u>] (preferred arrangement set forth below) - Descriptive title of the invention - Cross References to Related Applications - Statement Regarding Fed sponsored R & D - Reference to sequence listing, a table, or a computer program listing appendix - Background of the invention - Brief Summary of the invention - Brief Description of the Drawings (if filed) - Detailed Description - Claim(s) - Abstract of the Disclosure 4. ☒ Drawing(s) (35 U.S.C. 113) [Total Sheets <u>12</u>] 5. Oath or Declaration [Total Sheets <u>3</u>] a. ☒ Newly executed (original or copy) b. ☐ Copy from a prior application (37 CFR 1.63(d)) (for continuations/divisional with Box 18 completed) 6. ☐ DELETION OF INVENTOR(S) Signed statement attached deleting inventor(s) named in the prior application, see 37 CFR 1.53(d)(2) and 1.53(b). 7. ☐ CD-ROM or CD-R in duplicate, large table or Computer Program (Appendix) 8. Nucleotide and/or Amino Acid Sequence Submission (if applicable, all necessary) a. ☐ Computer Readable Form (CRF) b. Specification Sequence Listing or: i. ☐ CD-ROM or CD-R (2 copies); or ii. ☐ Paper c. ☐ Statements verifying identity of above copies 9. ☒ Application Data Sheet. See 37 CFR 1.78 10. ☐ Assignment Papers (cover sheet & document(s)) 11. ☐ 37 CFR 3.73(b) Statement (when there is an assignee) ☒ Power of Attorney 12. ☐ English Translation Document (if applicable) 13. ☐ Information Disclosure Statement (IDS)/PTO-1449 ☐ Copies of IDS Citations 14. ☐ Preliminary Amendment 15. ☒ Return Receipt Postcard (MPEP 503) (Should be specifically itemized) 16. ☐ Certified Copy of Priority Document(s) (if foreign priority is claimed) 17. ☒ Nonpublication Request under 35 U.S.C. 122 (b)(2)(B)(i). Applicant must attach form PTO/SB/35 or its equivalent. 18. ☒ Other: Express Mail Certificate... Credit Card Payment Form... 19. If a CONTINUING APPLICATION, check appropriate box, and supply the requisite information below as part of the disclosure of the specification following the title, or in an Application Data Sheet under 37 CFR 1.78: ☐ Continuation ☐ Divisional ☐ Continuation-in-part (CIP) Prior application information: Examiner _____ For CONTINUATION OR DIVISIONAL APPS only: The entire disclosure of the prior application, including any claims, drawings, and abstract, is incorporated by reference into this application. The incorporation can only be relied upon when a portion has been inadvertently omitted from this application. 19. CORRESPONDENCE ADDRESS ☒ Customer Number: <u>27195</u> Name <u>Amin & Turocy, LLP</u> Address <u>1900 E. 9th Street</u> <u>National City Center, 24th Floor</u> City <u>Cleveland</u> State <u>OH</u> Zip Code <u>44114</u> Country <u>USA</u> Telephone <u>216-696-8730</u> Fax <u>216-696-8731</u> Name (Print/Type) <u>Himanshu S. Amin</u> Registration No. (Attorney/Agent) <u>40,894</u> Signature <u>[Signature]</u> Date <u>March 28, 2005</u>	



This collection of information is required by 37 CFR 1.53(b). The information is required to obtain or retain a benefit by the public which is to file (and by the USPTO to process) an application. Confidentiality is governed by 35 U.S.C. 122 and 37 CFR 1.14. This collection is estimated to take 12 minutes to complete, including gathering, preparing, and submitting the completed application form to the USPTO. Time will vary depending upon the individual case. Any comments on the amount of time you require to complete this form and/or suggestions for reducing this burden, should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, U.S. Department of Commerce, P.O. Box 1480, Alexandria, VA 22313-1480. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. SEND TO: Commissioner for Patents, P.O. Box 1480, Alexandria, VA 22313-1480.

If you need assistance in completing the form, call 1-800-PTO-9199 and select option 2.

702

PTO/BRN17 (12-04)

Approved for use through 07/31/2008. OMB 0851-0032
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE

(Under the Patent and Trademark Act of 1980, no revenue can be received in payment for a collection of information unless it displays a valid OMB control number)

Effective on 12/08/2004.
Fees pursuant to the Consolidated Appropriations Act, 2005 (P.L. 108-447).**FEE TRANSMITTAL**
For FY 2005☐ Applicant claims small entity status. See 37 CFR 1.27TOTAL AMOUNT OF PAYMENT (\$)**1,000.00****Complete if Known**

Application Number	
Filing Date	Herewith
First Named Inventor	Nigel R. Ellis, et al.
Examiner Name	
Art Unit	
Attorney Docket No.	MS311235.01/MSFTP852US

METHOD OF PAYMENT (check all that apply)

☐ Check ☒ Credit Card ☐ Money Order ☐ None ☐ Other (please identify): _____

☒ Deposit Account Deposit Account Number: **50-1063** Deposit Account Name: **Amin & Turocy, LLP**

For the above-identified deposit account, the Director is hereby authorized to: (check all that apply)

☐ Charge fee(s) indicated below ☐ Charge fee(s) indicated below, except for the filing fee

☒ Charge any additional fee(s) or underpayments of fee(s) under 37 CFR 1.16 and 1.17 ☒ Credit any overpayments

WARNING: Information on this form may become public. Credit card information should not be included on this form. Provide credit card information and authorization on PTO-582B.

FEE CALCULATION**1. BASIC FILING, SEARCH, AND EXAMINATION FEES**

Application Type	FILING FEES		SEARCH FEES		EXAMINATION FEES		Fees Paid (\$)
	Fee (\$)	Small Entity Fee (\$)	Fee (\$)	Small Entity Fee (\$)	Fee (\$)	Small Entity Fee (\$)	
Utility	300	150	500	250	200	100	1,000
Design	200	100	100	50	130	65	0
Plant	200	100	300	150	160	80	0
Reissue	300	150	500	250	600	300	0
Provisional	200	100	0	0	0	0	0

2. EXCESS CLAIM FEES

Fee Description	Fee (\$)	Small Entity Fee (\$)
Each claim over 20 or, for Reissues, each claim over 20 and more than in the original patent	50	25
Each independent claim over 3 or, for Reissues, each independent claim more than in the original patent	200	100
Multiple dependent claims	360	180

Total Claims **Extra Claims** **Fee (\$)** **Fees Paid (\$)**

20 - 20 or HP = 0 x 50 = 0

HP = Highest number of total claims paid for, if greater than 20

Indep. Claims **Extra Claims** **Fee (\$)** **Fees Paid (\$)**

3 - 3 or HP = 0 x 200 = 0

HP = Highest number of independent claims paid for, if greater than 3

Multiple Dependent Claims

Fee (\$) **Fees Paid (\$)**

360 0

3. APPLICATION SIZE FEE

If the specification and drawings exceed 100 sheets of paper, the application size fee due is \$250.00 for each additional 50 sheets or fraction thereof. See 35 U.S.C. 41(a)(1)(G) and 37 CFR 1.101(c)(2). This fee is in addition to the filing fee.

Total Sheets **Extra Sheets** **Number of each additional 50 or fraction thereof** **Fees Paid (\$)**

47 - 100 = 0 / 50 = 0 (round up to a whole number)

4. OTHER FEE(S)

Non-English Specification, \$130 fee (no small entity discount)

Other: _____

SUBMITTED BY

Signature		Registration No. (Attorney/Agent)	40,894
Name (Print/Type)	Himanshu S. Amin	Phone	(216) 696-8730
		Date	March 28, 2005

This collection of information is required by 37 CFR 1.136. The information is required to obtain or retain a benefit by the public which is to file (and by the USPTO to process) an application. Confidentiality is governed by 35 U.S.C. 122 and 37 CFR 1.14. This collection is estimated to take 30 minutes to complete, including gathering, preparing, and submitting the completed application form to the USPTO. Time will vary depending upon the individual case. Any comments on the amount of time you require to complete this form and/or suggestions for reducing this burden, should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, U.S. Department of Commerce, P.O. Box 1480, Alexandria, VA 22313-1480. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. SEND TO: Commissioner for Patents, P.O. Box 1480, Alexandria, VA 22313-1480.

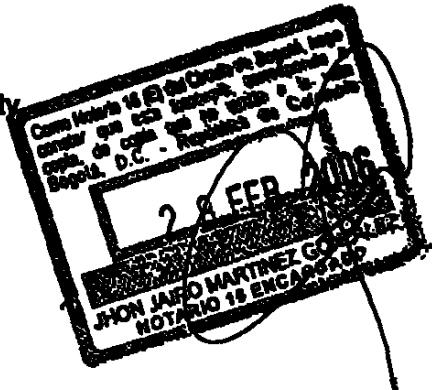
If you need assistance in completing the form, call 1-800-PTO-8199 and select option 2.

Application Data Sheet**Application Information**

Filing Date:: 03/28/05
Application Type:: Regular
Subject Matter:: Utility
Suggested classification::
Suggested Group Art Unit::
Computer Readable Form (CRF)?:: NO
Title:: MAPPING OF A FILE SYSTEM MODEL TO A
 DATABASE OBJECT
Attorney Docket Number:: MS311235.01/MSFTP852US
Request for Early Publication?:: NO
Request for Non-Publication?:: YES
Suggested Drawing Figure:: 1
Total Drawing Sheets:: 12
Small Entity?:: NO
Petition Included?:: NO
Secrecy Order in Parent Appl.?:: NO

Applicant Information

Applicant Authority Type:: Inventor
Primary Citizenship Country:: US
Status:: Full Capacity
Given Name:: Nigel
Middle Name:: R.
Family Name:: Ellie
Name Suffix::
City of Residence:: Redmond
State or Province of Residence:: WA



1 Initial 03/28/05

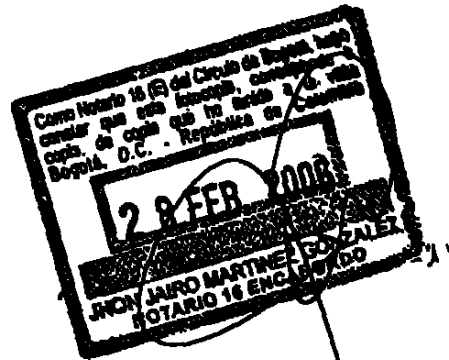
Country of Residence:: US
 Street of mailing address:: 12125 194th Avenue NE
 City of mailing address:: Redmond
 State or Province of mailing address:: WA
 Country of mailing address:: US
 Postal or Zip Code of mailing address:: 98053

Applicant Authority Type:: Inventor
 Primary Citizenship Country:: US
 Status:: Full Capacity
 Given Name:: Gregory
 Middle Name:: S.
 Family Name:: Friedman
 Name Suffix::

City of Residence:: Redmond
 State or Province of Residence:: WA
 Country of Residence:: US
 Street of mailing address:: 9812 173rd Court NE

City of mailing address:: Redmond
 State or Province of mailing address:: WA
 Country of mailing address:: US
 Postal or Zip Code of mailing address:: 98052

Applicant Authority Type:: Inventor
 Primary Citizenship Country:: Macedonia
 Status:: Full Capacity
 Given Name:: Bekim
 Middle Name::

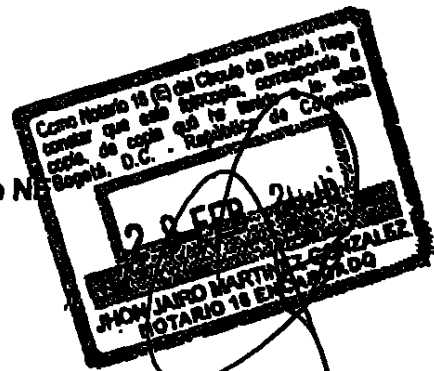


2 Initial 03/28/05

705

Family Name:: Demiroski
Name Suffix::
City of Residence:: Redmond
State or Province of Residence:: WA
Country of Residence:: US
Street of mailing address:: 6005 142nd Court NE
City of mailing address:: Redmond
State or Province of mailing address:: WA
Country of mailing address:: US
Postal or Zip Code of mailing address:: 98052

Applicant Authority Type:: Inventor
Primary Citizenship Country:: US
Status:: Full Capacity
Given Name:: Anil
Middle Name:: Kumar
Family Name:: Nori
Name Suffix::
City of Residence:: Redmond
State or Province of Residence:: WA
Country of Residence:: US
Street of mailing address:: 7421 204th Drive NE
City of mailing address:: Redmond
State or Province of mailing address:: WA
Country of mailing address:: US
Postal or Zip Code of mailing address:: 98053



Applicant Authority Type:: Inventor
Primary Citizenship Country:: India
Status:: Full Capacity
Given Name:: Amit
Middle Name::
Family Name:: Shukla
Name Suffix::
City of Residence:: Redmond
State or Province of Residence:: WA
Country of Residence:: US
Street of mailing address:: 10835 170th Court NE
City of mailing address:: Redmond
State or Province of mailing address:: WA
Country of mailing address:: US
Postal or Zip Code of mailing address:: 98052

Applicant Authority Type:: Inventor
Primary Citizenship Country:: US
Status:: Full Capacity
Given Name:: Srinivasmurthy
Middle Name:: P.
Family Name:: Acharya
Name Suffix::
City of Residence:: Sammamish
State or Province of Residence:: WA
Country of Residence:: US
Street of mailing address:: 2304 239th Place
City of mailing address:: Sammamish
State or Province of mailing address:: WA

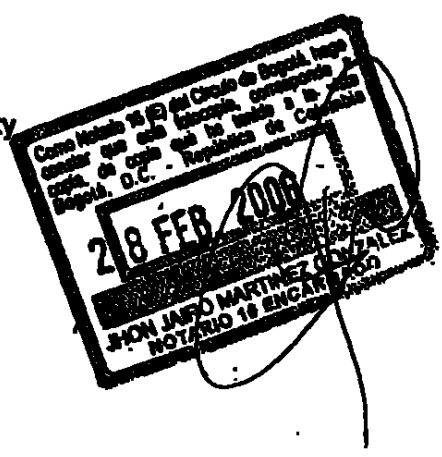


4 Initial 03/28/05

Country of mailing address:: US
Postal or Zip Code of mailing address:: 98074

Applicant Authority Type:: Inventor
Primary Citizenship Country:: US
Status:: Full Capacity
Given Name:: Jeffray
Middle Name:: T.
Family Name:: Pearce
Name Suffix::
City of Residence:: Sammamish
State or Province of Residence:: WA
Country of Residence:: US
Street of mailing address:: 437 237th Avenue SE
City of mailing address:: Sammamish
State or Province of mailing address:: WA
Country of mailing address:: US
Postal or Zip Code of mailing address:: 98074

Applicant Authority Type:: Inventor
Primary Citizenship Country:: US
Status:: Full Capacity
Given Name:: Michael
Middle Name:: J.
Family Name:: Newman
Name Suffix::
City of Residence:: Redmond
State or Province of Residence:: WA
Country of Residence:: US



5 Initial 03/28/05

Street of mailing address:: 18685 NE 56th Court
 City of mailing address:: Redmond
 State or Province of mailing address:: WA
 Country of mailing address:: US
 Postal or Zip Code of mailing address:: 98052

Applicant Authority Type:: Inventor
 Primary Citizenship Country:: US
 Status:: Full Capacity
 Given Name:: Jason
 Middle Name:: T.
 Family Name:: Hunter
 Name Suffix::
 City of Residence:: Redmond
 State or Province of Residence:: WA
 Country of Residence:: US
 Street of mailing address:: 4832 167th Court NE
 City of mailing address:: Redmond
 State or Province of mailing address:: WA
 Country of mailing address:: US
 Postal or Zip Code of mailing address:: 98052

Correspondence Information

Correspondence Customer Number:: 27195

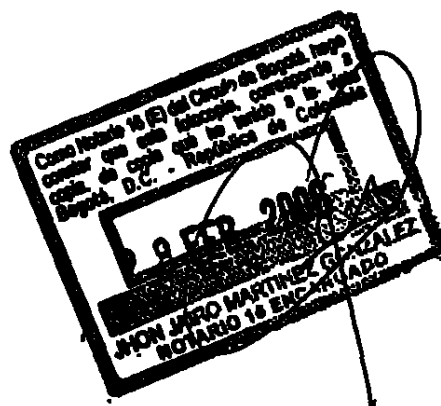
Representative Information

Representative Customer Number:: 27195



Assignee Information

Assignee name:: Microsoft Corporation
Street of mailing address:: One Microsoft Way
City of mailing address:: Redmond
State or Province of mailing address:: WA
Country of mailing address:: US
Postal or Zip Code of mailing address:: 98052



110

Atty. Docket No. MS311235.01/MSFTP852US

**MAPPING OF A FILE SYSTEM MODEL
TO A DATABASE OBJECT**

by

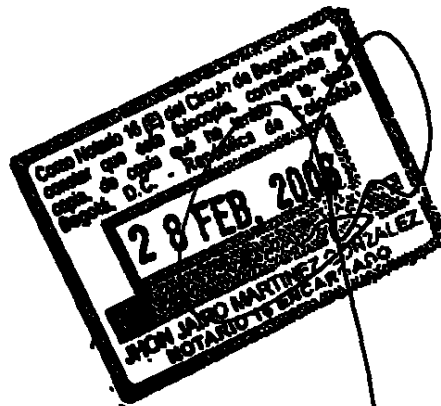
**Nigel R. Ellis, Gregory S. Friedman, Bekim Demiroski,
Anil Kumar Nori, Amit Shukla, Srinivasmurthy P. Acharya,
Jeffrey T. Pearce, Michael J. Newman, and Jason T. Hunter**

MAIL CERTIFICATION

I hereby certify that the attached patent application (along with any other paper referred to as being attached or enclosed) is being deposited with the United States Postal Service on this date March 28, 2005, in an envelope as "Express Mail Post Office to Addressee" Mailing Label Number EV62834791US addressed to Commissioner for Patents, P.O. Box 1450, Alexandria, Virginia 22313-1450.



Himanshu S. Amin



Title: MAPPING OF A FILE SYSTEM MODEL TO A DATABASE OBJECT**TECHNICAL FIELD**

[0001] The present invention generally relates to databases, and more particularly to systems and/or methods that facilitate storing a type instance and/or querying data.

BACKGROUND OF THE INVENTION

[0002] Advances in computer technology (e.g., microprocessor speed, memory capacity, data transfer bandwidth, software functionality, and the like) have generally contributed to increased computer application in various industries. Ever more powerful server systems, which are often configured as an array of servers, are commonly provided to service requests originating from external sources such as the World Wide Web, for example.

[0003] As the amount of available electronic data grows, it becomes more important to store such data in a manageable manner that facilitates user friendly and quick data searches and retrieval. Today, a common approach is to store electronic data in one or more databases. In general, a typical database can be referred to as an organized collection of information with data structured such that a computer program can quickly search and select desired pieces of data, for example. Commonly, data within a database is organized via one or more tables. Such tables can be arranged with rows and columns.

[0004] The tables can comprise one or more records, wherein a record can include a set of fields. Records are commonly indexed as rows within a table, and the record fields are typically indexed as columns, such that a row-column pair of indices can reference a particular datum within a table. For example, one may store a complete data record relating to a sales transaction, a person, or a product. Likewise, columns of the table can define discrete portions of the rows according to the same general data format, wherein the columns can define fields of the records.

[0005] Each individual piece of data, standing alone, is typically not very informative. Database applications make data more useful because they help users organize and process the data. The database application allows the user to compare, sort,

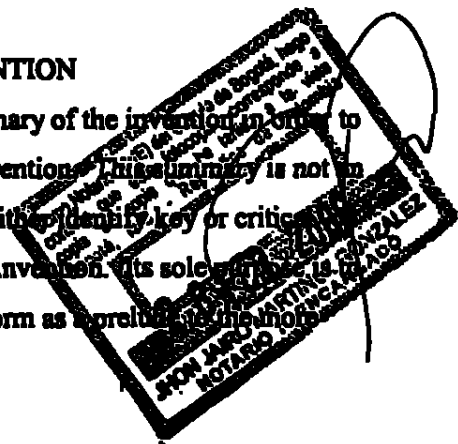
order, merge, separate and interconnect the data, so that useful information can be generated from the data. Yet, the capacity and versatility of databases have grown to an incredible amount to allow a virtually endless storage capacity utilizing databases. Moreover, typical database systems offer limited query-ability based upon time, file extension, location, and size. For example, in order to search the vast amounts of data associated to a database, a typical search is limited to a file name, a file size, a date of creation, wherein such techniques are deficient and inept.

[0006] With a continuing and increasing creation of data from end-users, the problems and difficulties surrounding finding, relating, and storing such data is reaching its peak. End-users write documents, store photos, rip music from compact discs, receive email, retain copies of sent email, etc. For example, in the simple process of creating a music compact disc, the end-user can create megabytes of data. Ripping the music from the compact disc, converting the file to a suitable format, creating a jewel case cover, designing a compact disc label, all require the creation of data.

[0007] Not only are the complications surrounding users, developers have similar issues with data. Developers create and write myriad of applications varying from personal applications to highly developed enterprise applications. While creating and/or developing, developers frequently, if not always, gather data. While obtaining such data, the data needs to be stored. In other words, the problems and difficulties surrounding finding, relating, and storing data affect both the developer and the end user. In view of the above, there is a need to improve upon and/or provide systems and/or methods that mitigate deficiencies associated with conventional systems and databases.

SUMMARY OF THE INVENTION

[0008] The following presents a simplified summary of the invention in order to provide a basic understanding of some aspects of the invention. This summary is not an extensive overview of the invention. It is intended to neither identify key or critical elements of the invention nor delineate the scope of the invention. Its sole purpose is to present some concepts of the invention in a simplified form as a prelude to the more detailed description that is presented later.



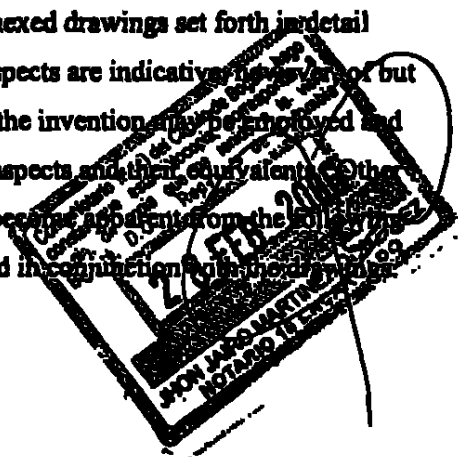
113

[0009] The subject invention relates to systems and/or methods that facilitate mapping of a data model to a database object. A type storage system can provide storing of a type instance and/or querying. The storing of the type can describe the database object and how instances of types are stored and/or accessed. The querying can find at least one of an item, a document, and a contact that each satisfies a certain criteria. Moreover, the type storage system can utilize an interface to receive data, wherein the data can include a schema, a type, a criteria, a query criteria, etc.

[0010] In accordance with one aspect of the subject invention, the type storage system can include a store component that stores at least one type instance. The storage can be a mapping of the data model, wherein the data model can represent a file storage system. Furthermore, the type storage system can also include a query component. The query component can provide at least one query in the file storage system data model to satisfy a criterion. In accordance with another aspect of the subject invention, the type storage system can include a relational component that can utilize a relational storage and/or a relational query capability. The relational component can invoke a database engine to provide such relational techniques, wherein these techniques can facilitate storing the type instance and/or querying.

[0011] Moreover, the type storage system can further include a view component that can provide a view that exposes all instances of a given type. The type can be a hierarchical structure and/or an inheritance structure. The view can be related to a particular type that projects a subset of the types with a base type. In other aspects of the subject invention, methods are provided that facilitate mapping of a data model to a database object.

[0012] The following description and the annexed drawings set forth in detail certain illustrative aspects of the invention. These aspects are indicative of various ways in which the principles of the invention may be employed and the subject invention is intended to include all such aspects and their equivalents. Other advantages and novel features of the invention will become apparent from the following detailed description of the invention when considered in conjunction with the drawings.

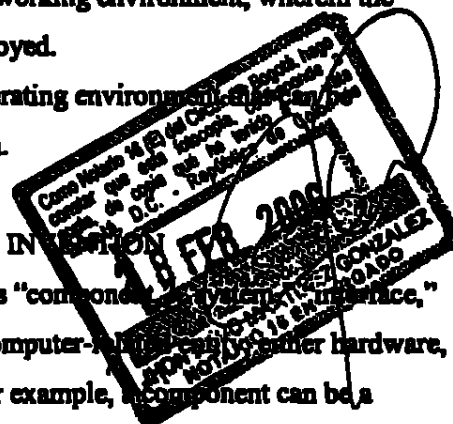


BRIEF DESCRIPTION OF THE DRAWINGS

- [0013] Fig. 1 illustrates a block diagram of an exemplary system that facilitates storing a type instance associated with a data model.
- [0014] Fig. 2 illustrates a block diagram of an exemplary system that facilitates storing a type instance and/or querying to find at least one of an item, a document, and a contact.
- [0015] Fig. 3 illustrates a block diagram of an exemplary system that facilitates utilizing a relational storage and/or relational query capabilities.
- [0016] Fig. 4 illustrates a block diagram of an exemplary system that facilitates mapping and/or viewing in conjunction with a type storage system.
- [0017] Fig. 5 illustrates a block diagram of an exemplary system that facilitates storing a type instance associated with a data model.
- [0018] Fig. 6 illustrates a block diagram of a high-level structure of a store within a file storage data model.
- [0019] Fig. 7 illustrates a block diagram of a type instance with associated tables.
- [0020] Fig. 8 illustrates block diagram of a type hierarchy and a corresponding view projection.
- [0021] Fig. 9 illustrates an exemplary methodology for storing a type instance that maps to a data model and/or providing a query.
- [0022] Fig. 10 illustrates an exemplary methodology for storing a type instance that maps to a data model and/or providing a query.
- [0023] Fig. 11 illustrates an exemplary networking environment, wherein the novel aspects of the subject invention can be employed.
- [0024] Fig. 12 illustrates an exemplary operating environment, wherein the novel aspects of the subject invention can be employed in accordance with the subject invention.

DESCRIPTION OF THE INVENTION

- [0025] As utilized in this application, terms "component," "module," "schema," and the like are intended to refer to a computer-related entity, either hardware, software (e.g., in execution), and/or firmware. For example, a component can be a process running on a processor, a processor, an object, an executable, a program, and/or a



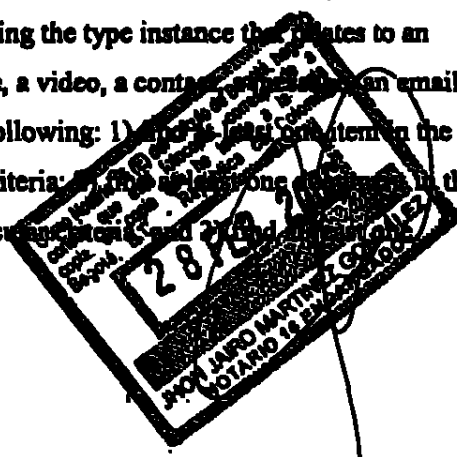
HLS

computer. By way of illustration, both an application running on a server and the server can be a component. One or more components can reside within a process and a component can be localized on one computer and/or distributed between two or more computers.

[0026] The subject invention is described with reference to the drawings, wherein like reference numerals are used to refer to like elements throughout. In the following description, for purposes of explanation, numerous specific details are set forth in order to provide a thorough understanding of the subject invention. It may be evident, however, that the subject invention may be practiced without these specific details. In other instances, well-known structures and devices are shown in block diagram form in order to facilitate describing the subject invention.

[0027] Now turning to the figures, Fig. 1 illustrates a system 100 that facilitates storing a type instance associated with a data model. The data model can be a file storage data model 102 that enables storing, finding, and relating information. For example, an information type can be, but is not limited to, a document, an image, a video, a contact, a message, an email, an audio clip, etc. The informational types can be considered units of information, which can be represented as instances of complex types that are part of a type system that supports inheritance. Inheritance can be defined as a situation in which certain characteristics are passed on from one context to another. In particular to object-oriented programming, objects inherit attributes and/or behaviors from other objects. It is to be appreciated that inheritance can be considered a hierarchical structure and/or format.

[0028] A type storage system 104 can store the type instance and query to efficiently and effectively find at least one of items, documents, and/or contacts. The type storage system 104 can receive data, wherein the data can include a type, a criteria, a schema, a query criteria, Specifically, storing the type instance that relates to an informational type (e.g., a document, an image, a video, a contact, a message, an email, an audio clip) can provide at least one of the following: 1) find at least one item in the file storage data model 102 that satisfy a certain criteria; 2) find at least one item in the file storage data model 102 that satisfy a particular criteria; and 3) find at least one item in the file storage data model 102 that satisfy a particular criteria.

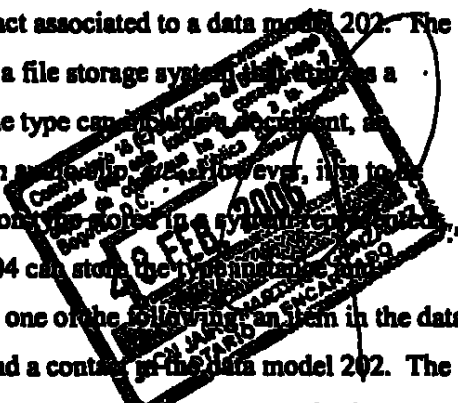


contact (e.g., including an individual, organization, and group) that satisfy a certain criteria.

[0029] The type storage system 104 can utilize a relational database technique associated to a relational storage and a relational query. It is to be appreciated that such capabilities can be provided by a database engine (not shown). The relational database techniques can be associated to relational databases, wherein a relational database is a collection of data items organized as a set of formally-described tables. The data within the tables can be accessed and/or reassembled in various ways without the requirement of reorganizing the database tables. Furthermore, the relational database can be easily extended, such as adding new categories without modifications to existing applications and/or data. It is to be appreciated that the subject invention is not limited to relational databases and/or associated techniques and that any suitable technique can be utilized.

[0030] The system 100 further includes an interface component 106, which provides various adapters, connectors, channels, communication paths, etc. to integrate the type storage system 104 into virtually any operating system. In addition, the interface component 106 can provide various adapters, connectors, channels, communication paths, etc. that provide for interaction with data and the type storage system 104. It is to be appreciated that although the interface component 106 is incorporated into the type storage system 104, such implementation is not so limited. For instance, the interface component 106 can be a stand-alone component to receive or transmit the data in relation to the system 100.

[0031] Fig. 2 illustrates a system 200 that facilitates storing a type instance and/or querying to find an item, a document, and a contact associated to a data model 202. The data model 202 can be a model representation of a file storage system that has a hierarchical characteristic and/or inheritance. The type can be a file, a document, an image, a video, a contact, a message, an email, an attachment, etc. However, it is to be appreciated that a type can be a typical information stored in a system represented by the data model 202. A type storage system 204 can store the type instance and provide querying that can efficiently find at least one of the following: an item in the data model 202, a document in the data model 202, and a contact in the data model 202. The type storage system 204 can receive data, wherein the data can be a query criteria, a



TTT

schema, a criteria, a schema definition, a data model, a type, The system 200 can further employ an interface 206 to facilitate providing various adapters, connectors, channels, communication paths, *etc.* to integrate the type storage system 204 into virtually any operating system and facilitate communication.

[0032] The type storage system 204 can further include a store component 208 ("store 208") that can store the type instances. The storage can be a mapping of the data model 202, wherein the storage mapping can describe the database object that is created based on a schema definition and how instances of types described in the schema are stored and/or accessed. In other words, the store 208 can store the instances of the type and at least one rule associated to mapping a type declaration into the database object.

[0033] The type storage system 204 can include a query component 210 ("query 210") that provides a querying of data. The query 210 can obtain at least one of the following: an item in the system represented by data model 202; a document in the system represented by data model 202; and a contact (*e.g.*, including an individual, an organization, and a group) in the system represented by the data model 202. It is to be appreciated and understood that the query can be based at least upon a certain criteria and/or query criteria obtained *via* the interface 206. Furthermore, the query 210 is not so limited to the item, document, and contact, and any suitable information type stored in the system represented by the data model 202 can be utilized.

[0034] Fig. 3 illustrates a system 300 that facilitates utilizing a relational storage and relational query capabilities. A type storage system 304 can store a type instance and query to efficiently and effectively find items, documents, and contacts, wherein such elements are associated to a file system represented by a data model 302. It is to be appreciated that the file system can be a storage file system that utilizes a hierarchical structure and/or an inheritance characteristic. The type storage system 304 can be substantially similar to the type storage system 104 as depicted in Fig. 1 and 2 respectively. The type storage system 304 can involve an interface 306 that facilitates communication and/or receiving data to be further utilized in accordance with the subject invention.

[0035] The type storage system 304 can include a store component 308 (store 308) and a query component 310 ("query 310"). The store 308 can provide any suitable

storing technique to store the type instance. The query 310 can provide a query that can efficiently and effectively obtain at least one of the following: at least one item in a system that meets a criteria; at least one document in a system that satisfies a criteria; and at least one contact that meets a criteria. It is to be appreciated and understood that the store 308 and the query 310 can be substantially similar to the store 208, and the query 210 as depicted in Fig. 2.

[0036] The type storage system 304 can further include a relational component 312 ("relational 312"). The relational 312 utilizes database techniques (e.g., utilizing a database engine) to build a relational storage and/or provide a relational query capability to facilitate storing type instances and/or querying. The relational 312 can incorporate techniques associated to a relational database, wherein the relational database is a collection of data items organized as a set of formally-described tables as described in detail above. It is to be appreciated that the subject invention is not limited to relational database and associated techniques and that any suitable technique can be utilized.

[0037] Fig. 4 illustrates a system 400 that facilitates mapping and/or viewing in conjunction with a data model 402. The data model 402 can represent a file storage system that enables storing, finding, and relating information. A typical information type that can be stored in the system can include a document, an image, music, a video, a contact, a message, etc. The information type can be represented as instances of complex types that are part of a type system that supports inheritance. A type storage system 404 can store the type instances and provide a query to find items, documents, and contacts that satisfy a certain criteria. The system 400 can further employ an interface 406 to facilitate communication and/or receiving data that can include criteria, types, schemas, models, and query criteria.

[0038] The type storage system 404 can further include a map component 408 to facilitate storing the type instance. The map component 408 provides for mapping of types described in a schema to defined types and database objects. The map component 408 can be a storage mapping that describes at least one database object wherein the database object can be created based on a schema definition and/or wherein instances of types described in the schema are stored and/or accessed. In other embodiments, instances of types can be stored and rules can be utilized to map a type declaration into the database object.

Each type in the schema maps to a class (e.g., common language runtime (CLR)) in a storage.

[0039] The type storage system 404 can include a view component 410 to provide a view projection. The view projection can expose instances of the type that is viewed. It is to be appreciated that the types can be in a hierarchical structure utilizing at least inheritance. In other words, each type is a part of a type hierarchy. The view can be associated with a given type and can project a subset of the respective types of the view associated with its base type. The view can project instances associated to the particular type. For example, for a type "message," only the instances of which message is a parent can be viewed based at least upon the hierarchy structure.

[0040] The view component 410 can provide various types of user interfaces to facilitate interaction between a user and any component coupled to type storage system 404. As depicted, the view component 410 is a separate entity that is incorporated into the type storage system 404. However, it is to be appreciated that the view component 410 and/or similar view components can be a separate component from the type storage system 404 and/or a stand-alone unit. The view component 410 can provide one or more graphical user interfaces (GUIs), command line interfaces, and the like. For example, a GUI can be rendered that provides a user with a region or means to load, import, read, etc. data, and can include a region to present the results of such. These regions can comprise known text and/or graphic regions comprising dialogue boxes, static controls, drop-down-menus, list boxes, pop-up menus, as edit controls, combo boxes, radio buttons, check boxes, push buttons, and graphic boxes. In addition, utilities to facilitate the presentation such vertical and/or horizontal scroll bars for navigation and push buttons to determine whether a region will be viewable can be employed. For example, the user can interact with one or more of the components of the type storage system 404.

[0041] The user can also interact with the regions to select and provide information via various devices such as a mouse, a rollerball, a touch pad, a pen and/or voice activation, for example. Typically, a mouse push button or the enter key on the keyboard can be employed subsequently entering the information in order to initiate the search. However, it is to be appreciated that the invention is not so

limited. For example, merely highlighting a check box can initiate information conveyance. In another example, a command line interface can be employed. For example, the command line interface can prompt (e.g., via a text message on a display and an audio tone) the user for information via providing a text message. The user can then provide suitable information, such as alpha-numeric input corresponding to an option provided in the interface prompt or an answer to a question posed in the prompt. It is to be appreciated that the command line interface can be employed in connection with a GUI and/or API. In addition, the command line interface can be employed in connection with hardware (e.g., video cards) and/or displays (e.g., black and white, and EGA) with limited graphic support, and/or low bandwidth communication channels.

[0042] Fig. 5 illustrates a system 500 that employs intelligence to facilitate storing a type instance associated with a data model 502. The system 500 includes a type storage system 504, an interface 506, and the data model 502, which can be substantially similar to the components depicted in earlier figures. The interface 506 can facilitate communication associated to data, which can include a criteria, a type, a schema, a data model, and a query criteria. The system 500 can provide storing a type, querying, and/or providing a view. It is to be appreciated and understood that a database engine can provide a relational storage and a relational query to the system 500.

[0043] The system 500 further includes an intelligent component 508. The intelligent component 508 can be utilized by the type storage system 504 to facilitate storing and/or querying for the system 500. For example, the intelligent component 510 can be utilized to facilitate determining a user defined type to be stored. Historic data in conjunction with user profiles can allow the intelligent component 508 to determine storing the type and/or querying with certain criteria.

[0044] It is to be understood that the intelligent component 508 can provide for reasoning about or infer states of the system, environment, and/or user from a set of observations as captured via events and/or data. Inference can be employed to identify a specific context or action, or can generate a probability distribution over states of interest based on a consideration of data and events. Inference can also refer to techniques employed for composing higher-level events from a set of

events and/or data. Such inference results in the construction of new events or actions from a set of observed events and/or stored event data, whether or not the events are correlated in close temporal proximity, and whether the events and data come from one or several event and data sources. Various classification (explicitly and/or implicitly trained) schemes and/or systems (e.g., support vector machines, neural networks, expert systems, Bayesian belief networks, fuzzy logic, data fusion engines...) can be employed in connection with performing automatic and/or inferred action in connection with the subject invention.

[0045] A classifier is a function that maps an input attribute vector, $x = (x_1, x_2, x_3, x_4, \dots, x_n)$, to a confidence that the input belongs to a class, that is, $f(x) = \text{confidence}(\text{class})$. Such classification can employ a probabilistic and/or statistical-based analysis (e.g., factoring into the analysis utilities and costs) to prognose or infer an action that a user desires to be automatically performed. A support vector machine (SVM) is an example of a classifier that can be employed. The SVM operates by finding a hypersurface in the space of possible inputs, which hypersurface attempts to split the triggering criteria from the non-triggering events. Intuitively, this makes the classification correct for testing data that is near, but not identical to training data. Other directed and undirected model classification approaches include, e.g., naive Bayes, Bayesian networks, decision trees, neural networks, fuzzy logic models, and probabilistic classification models providing different patterns of independence can be employed. Classification as used herein also is inclusive of statistical regression that is utilized to develop models of priority.

[0046] Furthermore, the intelligent component 508 can utilize a data store 510 to store user profiles and/or historic data. The data store 510 can be, for example, either volatile memory or nonvolatile memory, or can include both volatile and nonvolatile memory. By way of illustration, and not limitation, nonvolatile memory can include read only memory (ROM), programmable ROM (PROM), erasable programmable ROM (EPROM), electrically erasable programmable ROM (EEPROM), or flash memory. Volatile memory can include random access memory (RAM), which can include cache memory. By way of illustration and not limitation, RAM can include many forms such as static RAM (SRAM), dynamic RAM (DRAM), and synchronous DRAM (SDRAM).

(SDRAM), double data rate SDRAM (DDR SDRAM), enhanced SDRAM (ESDRAM), Synchlink DRAM (SLDRAM), Rambus direct RAM (RDRAM), direct Rambus dynamic RAM (DRDRAM), and Rambus dynamic RAM (RDRAM). The data store 510 of the subject systems and methods is intended to comprise, without being limited to, these and any other suitable types of memory. In addition, it is to be appreciated that the data store 510 can be a server and/or database.

[0047] Fig. 6 illustrates a high-level structure of a store. A schema can be provided, wherein various instances of types can be stored in tables dependent upon their type. The types can be, but are not limited to, an item, an item extension, an item fragment, and a link. Each type can have a corresponding table that can include columns having object instances therein. For instance, one column in the item table can contain all instances of items in the store. For each row, the column can contain a serialized representation of the CLR class instance that represents the item type instance. It is to be appreciated that for the item extension, item fragment and the link type instances can be represented in a similar structure. Inline type instances can be stored inside a parent object instance rather than storing in separate tables and/or columns. For each item, item extension, item fragment, and link a view can be generated that exposes all the instances of that type. Each type is a part of a type hierarchy. For instance, the item view projects all items in the store.

[0048] Briefly turning to Fig. 7, an item 702 is depicted. A table 704 of the item 702 can contain all instances of the items in the store (not shown). The table 704 can contain an object instance 706 ("Document") and an object instance 708 ("Contact"). It is to be appreciated that each type in a schema maps to a CLR class in the storage. The object instance 706 can include a title, an abstract, a printed, an author, and/or other metadata associated with the object instance 706. Similarly, the object instance 708 can include various metadata such as name, address, email, etc.

[0049] Fig. 8 illustrates a type hierarchy 800. The type hierarchy 800 can include an item 802, a contact 804, a document 806, a message 808, a person 810, an organization 812, an email 814, a phone number 816, and a voice 818. It is to be appreciated that the type hierarchy 800 is an example and any suitable hierarchy and/or types can be utilized in accordance with the subject invention. As

illustrated, item 802 is considered a parent of every type included therein. Thus, contact 804 is the parent for person 810 and organization 812, while message 808 is the parent for email 814, fax 816, and voice 818. Furthermore, the corresponding view projection 820 can reflect the type hierarchy 800. The view associated with a given type can project a subset of the items of the view associated with its base type. For example, the item view projects all the items in the store. The contact view projects only the items that are of type Contact. The person view projects only the contacts that have the most derived type as person.

[0050] Turning back to Fig. 6, type mapping can be provided utilizing an algorithm for mapping the type to the type used to describe the corresponding storage structure in the store. It is to be appreciated that the mapping can be associated to any file storage system (e.g., a data model based on a system that utilizes complex instances of types to describe and/or represent a unit of information). Each type declared in a schema is mapped to a CLR class that supports the SQL UDT contract. It is to be appreciated that although SQL is utilized in the following examples, any suitable database management system can be employed. The types belong to a namespace whose name corresponds to the namespace of the schema with ".Store" suffix appended. The types in a given namespace are compiled into a single assembly that is utilized as a unit of deployment of the schema. The name of the CLR type is equal to the name declared in the schema. For each declared property of the type, the following is added to the corresponding CLR type: 1) a private field with the name equal to the name specified in the schema, pre-fixed by "m_". The field is attributed with the UDT specific attribute: System.Data.SqlTypes.SqlUdtField; 2) a public property with the name equal to the name specified in the schema and corresponding get/set statements. The property is attributed with the UDT specific attribute System.Data.SqlTypes.SqlUdtProperty, and the type of the field and the property is the CLR type that corresponds to the type declared in the schema. If the type is one of the scalar types that it is mapped to a SQL scalar type (discussed *infra*).

[0051] The following table describes the mapping of the file storage system scalar types to the corresponding SQL managed types:

File Storage System Type	Managed SQL Type	Description
String	SqlString	Variable-length Unicode data with a maximum length of length of 2^{31} characters. Length can be fixed from 1 – 4000 characters or unconstrained using the "max" keyword.
Binary	SqlBinary	Variable-length binary data with a maximum length of length of 2^{32} bytes. Length can be fixed from 1 – 8000 bytes or unconstrained using the "max" keyword.
Boolean	SqlBoolean	Null-able Boolean value
Byte	SqlByte	A single unsigned byte
Int16	SqlInt16	Integer data from -2^{15} (-32,768) through $2^{15} - 1$ (32,767).
Int32	SqlInt32	Integer (whole number) data from -2^{31} (-2,147,483,648) through $2^{31} - 1$ (2,147,483,647).
Int64	SqlInt64	Integer (whole number) data from -2^{63} (-9223372036854775808) through $2^{63} - 1$ (9223372036854775807).
Single	SqlSingle	Floating precision number data from $-3.40E + 38$ through $3.40E + 38$
Double	SqlDouble	Floating precision number data from $-1.79E + 308$ through $1.79E + 308$
Decimal	SqlDecimal	Numeric data types with fixed precision and scale. Precision and Scale attributes apply for properties of type Decimal. Precision defines the maximum number of decimal digits that decimal value. This includes the digits to the left and to the right of the decimal point. The value of the Precision attribute is an integer between 1 and 28. Scale defines the maximum number of decimal digits to the right of the decimal point. The value of the Scale attribute must be between 0 and the value of the Precision attribute. Note for milestone B: the value of the Scale attribute is not supported in this milestone. The value of the Scale attribute is fixed to 14.
DateTime	SqlDateTime	Date and time data from January 1, 1975 through December 31, 9999, with accuracy up to three-hundredths of one second (3.00 milliseconds).
Guid	SqlGuid	A globally unique identifier (GUID).
Xml	SqlString	XML is currently stored in a string field. Native Xml

		type support is being considered for post-Beta1.
Stream	TBD	Support planned for post-Beta1. If supported it may replace string(max) and binary(max) declarations.

Table 1

[0052] A database object created in a file storage system store can be stored in a SQL schema name derived from the file storage system schema name. The suffix ".Store" is appended to the file storage system schema name to produce the SQL schema name. For example, the file storage system.storage schema produces objects in the "[System.Storage.Store]" SQL schema such as "[System.Storage.Store].Item."

[0053] Content in the file storage system can be accessed through views. The views depicted below are read-only, but the subject invention is not so limited since the views can be write-enabled. Each item type can be mapped to a typed view. Each typed item view can be identified in a file storage system store using the naming convention [<schema name>.Store]. [<Item type name>]. A type view for type T can return all items that are type T and all types that derive from T. The view that corresponds to the System.Storage.Item type is [System.Storage.Store]. [Item]. This view can return all the items in a file storage system store. The following table describes the columns of an item type view:

Column name	Type	Description
ItemId	[System.Storage.Store].ItemId	ItemId of the item
TypeId	[System.Storage.Store].TypeId	Schematized type id
NamespaceName	nvarchar(255)	The unique name of the item
ContainerId	[System.Storage.Store].ItemId	Id of the container of the item
Item	Item type	Instance of the item
EntityState	[System.Storage.Store].	The state information for the item
ObjectSize	Bigint	Size of the item object (persisted column)
ChangeInformation	[System.Storage.Store].ChangeInformation	Change tracking information

ItemSyncMetadata	[System.Storage.Store].ItemSyncMetadata	Sync metadata
PathHandle	[System.Storage.Store].BinPathHandle	The handle of the path to the item
PromotionStatus	Int	The status of the promotion for the item

Table 2

[0054] Each link type is mapped to a typed view. Each typed link view is identified in a file storage system store using the naming convention [<schema name>.Store]. [<link type name>]. The view that corresponds to the System.Storage.Link type is [System.Storage.Store].[Link]. This view can contain all the links in a file storage system store. The following table describes the columns of a link type view:

Column name	Type	Description
SourceRef	[System.Storage.Store].ItemId	ItemId of the source item
LinkId	[System.Storage.Store].LinkId	Id of the link
TargetRef	[System.Storage.Store].ItemId	ItemId of the target item
TypeId	[System.Storage.Store].TypeId	TypeId of the most derived type of the link instance
Link	Link type	Instance of the link type
EntityState	[System.Storage.Store].EntityState	The state information for the link
ObjectSize	Bigint	Size of the link object (persisted column)
ChangeInformation	[System.Storage.Store].ChangeInformation	Change tracking information
LinkSyncMetadata	[System.Storage.Store].ItemSyncMetadata	Link sync metadata
PathHandle	[System.Storage.Store].BinPathHandle	The handle of the path to the item that is the source of the link

Table 3

122

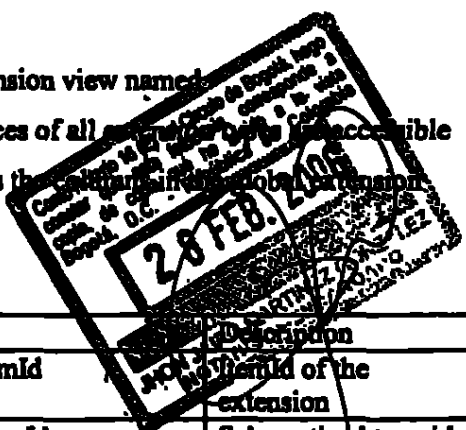
[0055] All item fragments can be accessible through single view [System.Storage.Store].[ItemFragment]. The following table describes the columns of global item fragment view:

Column name	Type	Description
ItemId	[System.Storage.Store].ItemId	ItemId of the owning item
SetId	[System.Storage.Store].SetId	The usage of the item fragment
FragmentId	[System.Storage.Store].FragmentId	The Id of the fragment instance
TypeId	[System.Storage.Store].TypeId	TypeId of the most derived type of the ItemFragment instance
ItemFragment	[System.Storage.Store].[ItemFragment]	Instance of the ItemFragment type
EntityState	[System.Storage.Store].EntityState	The state information for the link
ChangeInformation	[System.Storage.Store].ChangeInformation	Change tracking information
PathHandle	[System.Storage.Store].BinPathHandle	The handle of the path to the item that is the source of the link

Table 4

[0056] Each store provides a global extension view named [System.Storage.Store].[ItemExtension]. Instances of all extensions are accessible through this view. The following table describes the columns of the global extension view:

Column name	Type	Description
ItemId	[System.Storage.Store].ItemId	ItemId of the extension
TypeId	[System.Storage.Store].TypeId	Schematized type id
ItemExtension	[System.Storage.Store].ItemExtension	Extension type instance
EntityState	[System.Storage.Store].EntityState	The state



178

		information for the extension
ObjectSize	Bigint	Size of the extension object (persisted column)
ChangeInformation	[System.Storage.Store].ChangeInformation	Change tracking information
PathHandle	[System.Storage.Store].BinPathHandle	The handle of the path to the item that owns the extension

Table 5

[0057] Each extension type is mapped to a typed view. Each typed extension view is identified in a file storage system store using the naming convention [<schema name>.Store].[<extension type name>]. The following table describes the columns of an extension type view:

Column name	Type	Description
ItemId	[System.Storage.Store].ItemId	ItemId of the extension
TypeId	[System.Storage.Store].TypeId	Schematized type id
ItemExtension	Extension type	Extension type instance
ObjectSize	Bigint	Size of the extension object (persisted column)
EntityState	[System.Storage.Store].EntityState	The state information for the extension
ChangeInformation	[System.Storage.Store].ChangeInformation	Change tracking information
PathHandle	[System.Storage.Store].BinPathHandle	The handle of the path to the item that owns the extension

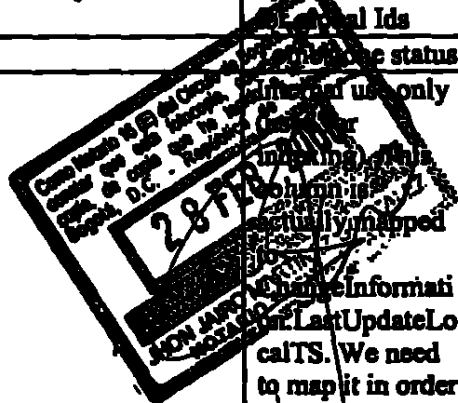
Table 6

In the file storage system, an inline type instance can be stored within entity type instances. It is to be appreciated and understood that they are accessed by querying the appropriate search view.

129

[0058] All Items are stored in a single item table, called [System.Storage.Store].[TableItem]. The unique key in the following table is ItemId:

Column name	Type	Description
ItemId	[System.Storage.Store].ItemId	ItemId of the item
TypeId	[System.Storage.Store].TypeId	Schematized type id
NamespaceName	Nvarchar(255)	The unique name of the item
ContainerId	[System.Storage.Store].ItemId	The id of the container item
Item	Item type	The item
EntityState	[System.Storage.Store].EntityState	EntityState udt which contains meta information about the item.
SDId	Int	Internal use only (used by security sub-system)
SDLastUpdateLocalTS	Bigint	The timestamp of the last change of the SDId. Used for change tracking.
ChangeInformation	[System.Storage.Store].ChangeInformation	Change tracking information
ItemSyncMetadata	[System.Storage.Store].ItemSyncMetadata	Sync metadata
TombstoneStatus	Int	Tombstone status
LastUpdateLocalTS	Bigint	Internal use only (used by security sub-system) for indexing. This column is actually mapped to ChangeInformation.LastUpdateLocalTS. We need to map it in order to create an index on it.
MaxOrd	int	Internal use only (used to compute



		PathHandle for new items added to this container item)
TypePath	Hierarchical_Type_Id	Internal use only (persisted column)
ObjectSize	Bigint	Size of the item object (persisted column)
PathHandle	[System.Storage.Store].BinPathHandle	The path handle to this item
PromotionStatus	Int	The promotion status for the item
LastAccessTime	DateTime	The last access time for the file stream associated with the item.
StreamSize	Bigint	The size of the stream associated with the item
AllocationSize	Bigint	The allocated size for the stream associated with the item

Table 7

[0059] The indexes on the item table are described in the following table:

Columns	Unique	Clustered	Included Columns
TypePath, ItemId	Yes	Yes	
ItemId	Yes	No	TypePath, PromotionStatus, PathHandle, LastAccessTime, StreamSize, AllocationSize
PathHandle	Yes	No	ItemId, PromotionStatus, LastAccessTime, StreamSize, AllocationSize
ContainerId, NamespaceName	Yes	No	ItemId, PromotionStatus, PathHandle, LastAccessTime, StreamSize, AllocationSize
LastUpdateLocalTS	No	No	ItemId, PromotionStatus, PathHandle, LastAccessTime, StreamSize, AllocationSize

Table 8

[0060] All links will be stored in the link table, which is named [System.Storage.Store].[Table!Link]. The unique key in the following table is ItemId, LinkId:

Column name	Type	Description
SourceRef	[System.Storage.Store].ItemId	ItemId of the source item
LinkId	[System.Storage.Store].LinkId	Id of the link
TargetRef	[System.Storage.Store].ItemId	ItemId of the target item
TypeId	[System.Storage.Store].TypeId	Schematized type id
TypePath	Hierarchical_Type_Id	Internal use only (persisted column)
Link	Link type	The link object
EntityState	[System.Storage.Store].EntityState	State information about the link
SDId	Int	Internal use only
TombstoneStatus	Int	Tombstone status
ObjectSize	Bigint	Size of the link object (persisted column)
ChangeInformation	[System.Storage.Store].ChangeInformation	Change tracking information
LinkSyncMetadata	[System.Storage.Store].LinkSyncMetadata	Sync metadata
LastUpdateLocalTS	Bigint	Internal use only (used for indexing). This column is used to track change information. It is used to update LocalTS. It is used to create an index.
PathHandle	[System.Storage.Store].BinPathHandle	Bin path handle to the link object that is the root of the link

Table 9

[0061] The indexes on the Link table are described in the following table:

126

Columns	Unique	Clustered	Included Columns
SourceRef, LinkId	Yes	Yes	
TypePath	No	No	SDId, TombstoneStatus
SourceRef, TypePath	No	No	SDId, TombstoneStatus
PathHandle	No	No	SDId, TombstoneStatus
TargetRef	No	No	SDId, TombstoneStatus
LastUpdateLocalTS	No	No	SDId, TombstoneStatus, PathHandle

Table 10

[0062] All EntityExtensions are stored in a single table called: [System.Storage.Store].[Table!ItemExtension]. The following table describes item extension table:

Column name	Type	Description
ItemId	[System.Storage.Store].ItemId	ItemId of the extension
TypeId	[System.Storage.Store].TypeId	Schematized type id
TypePath	Hierarchical_Type_Id	Internal use only (persisted column)
ItemExtension	[System.Storage.Store].ItemExtension	The ItemExtension object
EntityState	[System.Storage.Store].EntityState	EntityState udt which contains meta information about the EntityExtension.
SDId	Int	Internal use only
TombstoneStatus	Int	Tombstone status
ObjectSize	Bigint	Size of the EntityExtension object (persisted column)
ChangeInformation	[System.Storage.Store].ChangeInformation	Change tracking information
LastUpdateLocalTS	Bigint	Internal use only (used for indexing). This column is actually mapped to ChangeInformation. LastUpdateLocalTS. We need to map it in order to create an

		index on it.
PathHandle	[System.Storage.Store].BinPathHandle	The path handle to the item owning this EntityExtension

Table 11

[0063] The indexes on the ItemExtension table are described in the following table:

Columns	Unique	Clustered	Included Columns
TypePath, ItemId, TypeId	Yes	Yes	
ItemId, TypeId	No	No	SDId, TombstoneStatus, PathHandle
PathHandle	Yes	No	SDId, TombstoneStatus
LastUpdateLocalTS	No	No	SDId, TombstoneStatus, PathHandle

Table 12

[0064] All ItemFragment type instances are stored in a single table named [System.Storage.Store].[Table!ItemFragment]. The following table depicts the ItemFragment type:

Column name	Type	Description
ItemId	[System.Storage.Store].ItemId	ItemId of the owning item
SetId	[System.Storage.Store].SetId	The usage of the item fragment
FragmentId	[System.Storage.Store].FragmentId	Id of the fragment instance
ItemFragment	[System.Storage.Store].ItemFragment	The ItemFragment object
TypeId	[System.Storage.Store].TypeId	Serialized type id
TypePath	Hierarchical_Type_Id	Internal use only (persisted column)
EntityState	[System.Storage.Store].EntityState	EntityState udt which contains meta information about the fragment.
SDId	int	Internal use only
TombstoneStatus	int	Tombstone status

ObjectSize	bigint	Size of the ItemFragment object (persisted column)
ChangeInformation	[System.Storage.Store].ChangeInformation	Change tracking information
LastUpdateLocalTS	bigint	Internal use only (used for indexing). This column is actually mapped to ChangeInformation.LastUpdateLocalTS. We need to map it in order to create an index on it.
PathHandle	[System.Storage.Store].BinPathHandle	The path handle to the owning item

Table 13

[0065] The indexes on the ItemFragment table are described in the following table:

Column name	Unique	Clustered	Included Columns
TypePath, ItemId, SetId, FragmentId	Yes	Yes	
ItemId, SetId, FragmentId	Yes	No	SDId, TombstoneStatus, PathHandle
PathHandle	Yes	No	SDId, TombstoneStatus
LastUpdateLocalTS	No	No	SDId, TombstoneStatus, PathHandle

Table 14

[0066] Figs. 9-10 illustrate methodologies in accordance with the subject invention. For simplicity of explanation, the methodologies are depicted and described as a series of acts. It is to be understood and appreciated that the subject invention is not limited by the acts illustrated and/or by the order of the acts. Example acts can occur in various orders and/or concurrently, and with other acts not presented and described herein. Furthermore, not all illustrated acts may be required to implement the methodologies in accordance with the subject invention. In addition, those skilled in the

1789

art will understand and appreciate that the methodologies could alternatively be represented as a series of interrelated states *via* a state diagram or events.

[0067] Fig. 9 illustrates a methodology 900 that facilitates storing a type instance that maps to a data model and/or provides a query. The data model can be a file storage data model that enables storing, finding, and relating information. An information type can be, but is not limited to, a document, an image, a video, a contact, a message, an email, an audio clip, *etc.* These informational types (*e.g.*, units of information) can be represented as instances of complex types that are part of a type system that supports inheritance, wherein inheritance allows objects to inherit attributes and/or behaviors from other objects. It is to be appreciated that inheritance can be considered a hierarchical structure and/or format.

[0068] At reference numeral 902, data can be received, wherein the data can include a type, a criteria, a schema, a query criteria, The type instance can be stored to map to a data model (*e.g.*, a file storage data model) at reference numeral 904. For instance, the model can map types described in the schema to user defined types and database objects. The storage mapping can describe the database objects that are created based on a schema definition and how instances of types described in the schema are stored and/or accessed. In one example, a database structure can be designed such that the instances of types and the rules for mapping of type declarations into database objects can be provided.

[0069] At reference numeral 906, a query can be invoked to satisfy at least one criterion to efficiently and effectively find items, documents, and contacts. For instance, storing the type instance that relates to information types can allow the query to provide at least one of the following: 1) find at least one instance in the file storage data model 102 that satisfy a certain criteria; 2) find at least one object in the file storage data model 102 that satisfy a particular criteria; and 3) find at least one contact (including an individual, organization, and group) that satisfy a certain criteria.

[0070] Fig. 10 illustrates a methodology 1000 that facilitates storing a type instance and/or providing a query. At reference numeral 1002, data is obtained and/or received. The data can include a type, a criteria, a schema, a query criteria, *etc.* At 1004, a database engine can be utilized to provide at least one mechanism for storing the type

128

instances and/or for querying. For instance, a relational database engine can be utilized to provide a relational storage and a relation query capability. The relational database engine can utilize a collection of data items organized as a set of formally described tables. The data within the tables can be accessed and/or reassembled in various ways without the requirement of reorganizing the database tables. Furthermore, the relational database can be easily extended, such as adding new categories without modifications to existing applications and/or data.

[0071] At reference numeral 1006, a type instance is stored that maps to a data model, wherein the data model can be a file storage system. The storage mapping describes the database objects that are created based on the schema and how instances of types described in the schema are stored and/or accessed. The mapping can be of types described in the schema, wherein such mapping is to user defined types and database objects. At reference numeral 1008, a query can be provided to find at least one of an item, a document, and a contact. The query can be utilized to search based at least in part upon a criterion. Moreover, at reference numeral 1010, a view can be employed to expose the instances of the type. For instance, the type system can be hierarchical in structure, wherein a view can be generated to expose any instance of the particular type. In other words, because of the hierarchy, the view is associated with a given type that projects a subset of the particular types of the view associated with its base type.

[0072] In order to provide additional context for implementing various aspects of the subject invention, Figs. 11-12 and the following discussion is intended to provide a brief, general description of a suitable computing environment in which the various aspects of the subject invention may be implemented. While the invention has been described above in the general context of computer-executable instructions of a computer program that runs on a local computer and/or remote computer, those skilled in the art will recognize that the invention also may be implemented in combination with other program modules. Generally, program modules include routines, programs, components, data structures, etc., that perform particular tasks and/or implement particular abstract data types.

[0073] Moreover, those skilled in the art will appreciate that the inventive methods may be practiced with other computer system configurations, including single-

processor or multi-processor computer systems, minicomputers, mainframe computers, as well as personal computers, hand-held computing devices, microprocessor-based and/or programmable consumer electronics, and the like, each of which may operatively communicate with one or more associated devices. The illustrated aspects of the invention may also be practiced in distributed computing environments where certain tasks are performed by remote processing devices that are linked through a communications network. However, some, if not all, aspects of the invention may be practiced on stand-alone computers. In a distributed computing environment, program modules may be located in local and/or remote memory storage devices.

[0074] Fig. 11 is a schematic block diagram of a sample-computing environment 1100 with which the subject invention can interact. The system 1100 includes one or more client(s) 1110. The client(s) 1110 can be hardware and/or software (e.g., threads, processes, computing devices). The system 1100 also includes one or more server(s) 1120. The server(s) 1120 can be hardware and/or software (e.g., threads, processes, computing devices). The servers 1120 can house threads to perform transformations by employing the subject invention, for example.

[0075] One possible communication between a client 1110 and a server 1120 can be in the form of a data packet adapted to be transmitted between two or more computer processes. The system 1100 includes a communication framework 1140 that can be employed to facilitate communications between the client(s) 1110 and the server(s) 1120. The client(s) 1110 are operably connected to one or more client data store(s) 1130 that can be employed to store information local to the client(s) 1110. Similarly, the server(s) 1120 are operably connected to one or more server data store(s) 1150 that can be employed to store information local to the servers 1120.

[0076] With reference to Fig. 12, an exemplary computer 1212 for implementing various aspects of the invention includes a processing unit 1214, a system memory 1216, and a system bus 1218. The computer 1212 includes a processing unit 1214, a system memory 1216, and a system bus 1218. The system bus 1218 couples system components including, but not limited to, the system memory 1216 to the processing unit 1214. The processing unit 1214 can be any of various available processors. Dual microprocessors and other multiprocessor architectures also can be employed as the processing unit 1214.

[0077] The system bus 1218 can be any of several types of bus structure(s) including the memory bus or memory controller, a peripheral bus or external bus, and/or a local bus using any variety of available bus architectures including, but not limited to, Industrial Standard Architecture (ISA), Micro-Channel Architecture (MSA), Extended ISA (EISA), Intelligent Drive Electronics (IDE), VESA Local Bus (VLB), Peripheral Component Interconnect (PCI), Card Bus, Universal Serial Bus (USB), Advanced Graphics Port (AGP), Personal Computer Memory Card International Association bus (PCMCIA), Firewire (IEEE 1394), and Small Computer Systems Interface (SCSI).

[0078] The system memory 1216 includes volatile memory 1220 and nonvolatile memory 1222. The basic input/output system (BIOS), containing the basic routines to transfer information between elements within the computer 1212, such as during start-up, is stored in nonvolatile memory 1222. By way of illustration, and not limitation, nonvolatile memory 1222 can include read only memory (ROM), programmable ROM (PROM), electrically programmable ROM (EPROM), electrically erasable programmable ROM (EEPROM), or flash memory. Volatile memory 1220 includes random access memory (RAM), which acts as external cache memory. By way of illustration and not limitation, RAM is available in many forms such as static RAM (SRAM), dynamic RAM (DRAM), synchronous DRAM (SDRAM), double data rate SDRAM (DDR SDRAM), enhanced SDRAM (ESDRAM), Synchlink DRAM (SLDRAM), Rambus direct RAM (RDRAM), direct Rambus dynamic RAM (DRDRAM), and Rambus dynamic RAM (RDRAM).

[0079] Computer 1212 also includes removable/non-removable, volatile/non-volatile computer storage media. Fig. 12 illustrates, for example, disk storage 1224. Disk storage 1224 includes, but is not limited to, devices and/or media such as magnetic disk drive, floppy disk drive, tape drive, Jaz drive, Zip drive, NS-300 drive, flash memory card, or memory stick. In addition, disk storage 1224 can include storage media separately or in combination with other storage media including, but not limited to, optical disk drive such as a compact disk ROM device (CD-ROM), CD recordable drive (CD-R Drive), CD rewritable drive (CD-RW Drive) or a digital versatile disk (DVD-ROM). To facilitate connection of the disk storage devices 1224 to the system bus 1218, a removable or non-removable interface is typically used such as interface 1226.

[0080] It is to be appreciated that Fig. 12 describes software that acts as an intermediary between users and the basic computer resources described in the suitable operating environment 1200. Such software includes an operating system 1228. Operating system 1228, which can be stored on disk storage 1224, acts to control and allocate resources of the computer system 1212. System applications 1230 take advantage of the management of resources by operating system 1228 through program modules 1232 and program data 1234 stored either in system memory 1216 or on disk storage 1224. It is to be appreciated that the subject invention can be implemented with various operating systems or combinations of operating systems.

[0081] A user enters commands or information into the computer 1212 through input device(s) 1236. Input devices 1236 include, but are not limited to, a pointing device such as a mouse, trackball, stylus, touch pad, keyboard, microphone, joystick, game pad, satellite dish, scanner, TV tuner card, digital camera, digital video camera, web camera, and the like. These and other input devices connect to the processing unit 1214 through the system bus 1218 via interface port(s) 1238. Interface port(s) 1238 include, for example, a serial port, a parallel port, a game port, and a universal serial bus (USB). Output device(s) 1240 use some of the same type of ports as input device(s) 1236. Thus, for example, a USB port may be used to provide input to computer 1212, and to output information from computer 1212 to an output device 1240. Output adapter 1242 is provided to illustrate that there are some output devices 1240 like monitors, speakers, and printers, among other output devices 1240, which require special adapters. The output adapters 1242 include, by way of illustration and not limitation, video and sound cards that provide a means of connection between the output device 1240 and the system bus 1218. It should be noted that other devices and/or system bus devices provide both input and output capabilities such as remote computer(s) 1244.

[0082] Computer 1212 can operate in a networked environment using logical connections to one or more remote computers, such as remote computer(s) 1244. The remote computer(s) 1244 can be a personal computer, a server, a network PC, a workstation, a microprocessor based appliance, a peer device or other common network node and the like, and typically includes many or all of the elements described relative to computer 1212. For purposes of brevity, only a memory storage device 1246 is

illustrated with remote computer(s) 1244. Remote computer(s) 1244 is logically connected to computer 1212 through a network interface 1248 and then physically connected via communication connection 1250. Network interface 1248 encompasses wire and/or wireless communication networks such as local-area networks (LAN) and wide-area networks (WAN). LAN technologies include Fiber Distributed Data Interface (FDDI), Copper Distributed Data Interface (CDDI), Ethernet, Token Ring and the like. WAN technologies include, but are not limited to, point-to-point links, circuit switching networks like Integrated Services Digital Networks (ISDN) and variations thereon, packet switching networks, and Digital Subscriber Lines (DSL).

[0083] Communication connection(s) 1250 refers to the hardware/software employed to connect the network interface 1248 to the bus 1218. While communication connection 1250 is shown for illustrative clarity inside computer 1212, it can also be external to computer 1212. The hardware/software necessary for connection to the network interface 1248 includes, for exemplary purposes only, internal and external technologies such as, modems including regular telephone grade modems, cable modems and DSL modems, ISDN adapters, and Ethernet cards.

[0084] What has been described above includes examples of the subject invention. It is, of course, not possible to describe every conceivable combination of components or methodologies for purposes of describing the subject invention, but one of ordinary skill in the art may recognize that many further combinations and permutations of the subject invention are possible. Accordingly, the subject invention is intended to embrace all such alterations, modifications, and variations that fall within the spirit and scope of the appended claims.

[0085] In particular and in regard to the various functions described by the above described components, devices, circuits, systems and the like, the term (including a reference to a "means") used to describe such components is intended to correspond, unless otherwise indicated, to any component which performs the specified function of the described component (e.g., a functional equivalent), even though not structurally equivalent to the disclosed structure, which performs the function in the herein illustrated exemplary aspects of the invention. In this regard, it will also be recognized that the invention includes a system as well as a computer-readable medium having computer-

executable instructions for performing the acts and/or events of the various methods of the invention.

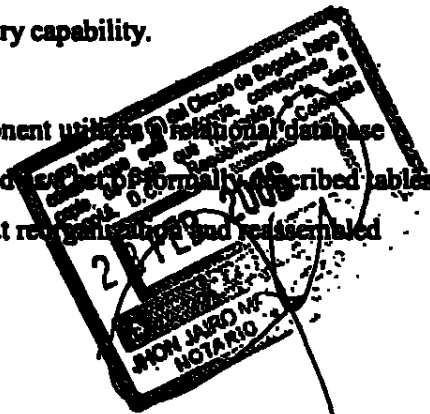
[0086] In addition, while a particular feature of the invention may have been disclosed with respect to only one of several implementations, such feature may be combined with one or more other features of the other implementations as may be desired and advantageous for any given or particular application. Furthermore, to the extent that the terms “includes,” and “including” and variants thereof are used in either the detailed description or the claims, these terms are intended to be inclusive in a manner similar to the term “comprising.”



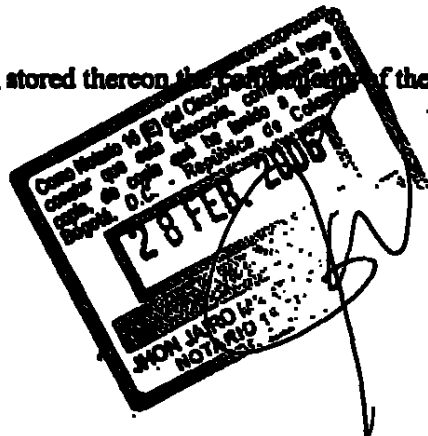
CLAIMS

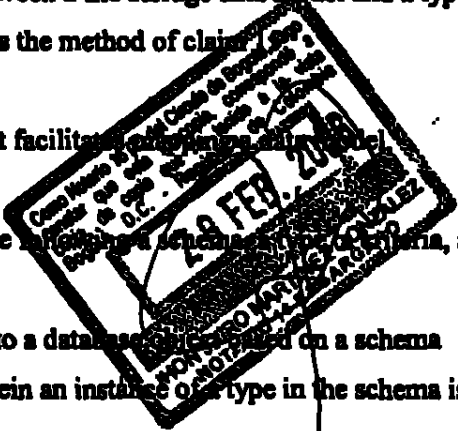
What is claimed is:

1. A system that facilitates mapping a data model, comprising:
a file storage data model that is utilized to define a mapping schema; and
a type storage component that maps a schema object to a database object based on the associated mapping schema, wherein an instance of a type in the mapping schema is stored.
2. The system of claim 1, the instance of the type is at least one of: a document; an image; music; a video; a contact; a message; a person; an organization; an email; a fax; a voice; and an audio clip.
3. The system of claim 1, the type storage component further provides a query to find at least one of the following: at least one item in the system that satisfies a respective criteria; at least one document in the system that satisfies a respective criteria; and at least one contact that satisfies a respective criteria.
4. The system of claim 3, the contact can include at least one of the following: an individual; an organization; and a group.
5. The system of claim 1, further comprising a relational component that utilizes at least one of a relational storage and a relational query capability.
6. The system of claim 5, the relational component utilizes a relational database technique that is a collection of data items organized as one or more tables wherein data can be at least one of accessed without reorganization and reassembled without reorganization.



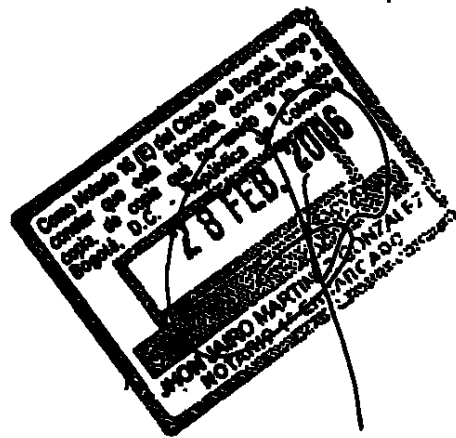
7. The system of claim 1, the mapping describes at least one of the database objects created based on the schema and how the instance of the type described in the schema is stored and accessed.
8. The system of claim 1, the type in the schema maps to a common language runtime (CLR) class in a storage of the file storage data model.
9. The system of claim 8, the type is at least one of the following: an item; an item extension; an item fragment; and a link.
10. The system of claim 1, further comprising a table that has at least one of the following: 1) a column that contains at least one instance of the respective type; and 2) a row that contains a serialized representation of the CLR class instance that represents the type instance.
11. The system of claim 1, further comprising a view component that generates a view that exposes at least one instance of the type, wherein the view projects a subset of the respective type of the view associated with a base type.
12. The system of claim 1, the type contains at least one of the following a type hierarchy and an inheritance.
14. A computer readable medium having stored thereon the system of claim 1.



15. A computer-implemented method that facilitates mapping a data model, comprising:
- obtaining a file storage data model;
 - receiving at least one of a schema, a type, a criteria, and a query information;
 - mapping a schema object to a database object based on the schema associated to the file storage data model; and
 - storing an instance of the type from the schema.
16. The method of claim 15, further comprising querying to find at least one of the following: at least one item in the system that satisfies a respective criteria; at least one document in the system that satisfies a respective criteria; and at least one contact that satisfies a respective criteria.
17. The method of claim 15, further comprising utilizing a relational database engine to provide a relational storage and a relational query capability.
18. The method of claim 15, further comprising generating a view that exposes at least one instance of the type, wherein the view projects a subset of the respective type of the view associated with a base type.
19. A data packet that communicates between a file storage data model and a type storage component, the data packet facilitates the method of claim 15.
20. A computer-implemented system that facilitates mapping a data model, comprising:
- means for receiving at least one of the schema, a type, a criteria, and a query criteria; and
 - means for mapping a schema object to a database object based on a schema associated to a file storage data model, wherein an instance of a type in the schema is stored.
- 

ABSTRACT

The subject invention provides a system and/or a method that facilitates mapping a data base model to a database object. A type storage system can utilize a storage mapping of a file storage data model. The mapping can describe the database object created based at least in part upon a schema and how instances of the type described in the schema are stored and/or accessed. Furthermore, a query can be provided to find at least one of an item, a document, and/or a contact that satisfy at least one criterion. The type storage system can receive data, wherein the data is at least one of a schema, a data model, a type, a query, and query criteria via an interface to provide the storing and querying. Additionally, the type storage system can generate a view that exposes at least one instance of the type.



[Handwritten signature]

PATENT APPLICATION

DECLARATION AND POWER OF ATTORNEY

ATTORNEY DOCKET NO. MSPT642US

MS DOCKET NO. MS311235.51

I, a below named inventor, I hereby declare that my residence/post office address and citizenship are as stated below next to my names. I believe I am the original, first and sole inventor (if only one name is listed below) or an original, first and joint inventor (if plural names are listed below) of the subject matter which is claimed and for which a patent is sought on the invention entitled: MAPPING OF A FILE SYSTEM MODEL TO A DATABASE OBJECT, the specification of which is filed herewith unless the following box is checked:

() was filed on _____ as US Application Serial No. or PCT International Application Number _____ and was amended on _____ (if applicable).

I hereby state that I have reviewed and understood the contents of the above-identified specification, including the claims, as amended by any amendment(s) referred to above. I acknowledge the duty to disclose all information which is material to patentability as defined in 37 CFR 1.56.

Foreign Application(s) and/or Claims of Foreign Priority

I hereby claim foreign priority benefits under Title 35, United States Code Section 119 of any foreign application(s) for patent or invention(s) certificate listed below and have also identified below any foreign application for patent or invention(s) certificate having a filing date before that of the application on which priority is claimed:

COUNTRY	APPLICATION NUMBER	DATE FILED	PRIORITY CLAIMED UNDER 35 U.S.C. 119
			YES NO
			YES NO

POWER OF ATTORNEY:

As a named inventor, I hereby appoint the following attorney(s) and/or agent(s) associated with

Customer No. 27195

to prosecute this application and transact all business in the Patent and Trademark Office connected therewith.

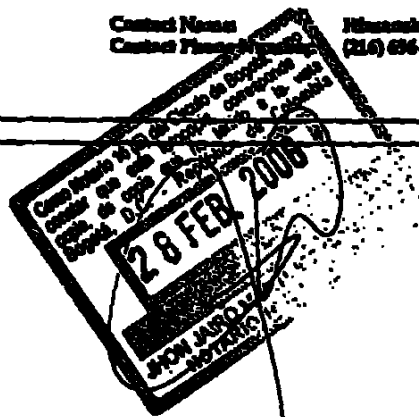
David Bentley Eppensner Reg. No. 28,499	Martha L. Stibely Reg. No. 28,523	Ronald O. Zink Reg. No. 28,794	Patricia E. Bennett Reg. No. 27,888
Jeffrey L. Banch Reg. No. 28,399	Michael W. Rodanowski Reg. No. 28,693	John Wursch Reg. No. 28,392	Hanantha S. Amla Reg. No. 48,894
Gangway Turvey Reg. No. 24,923			

Send Correspondence to:

Contact Name: Hanantha S. Amla
Firm Name: Amla & Turvey, LLP
Firm Address: 1900 E. 9th Street, National City Center - 24th Floor
City, State and Zip: Cleveland, OH 44114

Direct Telephone Calls To:

Contact Name: Hanantha S. Amla
Contact Phone: (216) 694-8720



BEST AVAILABLE COPY

1-108

32

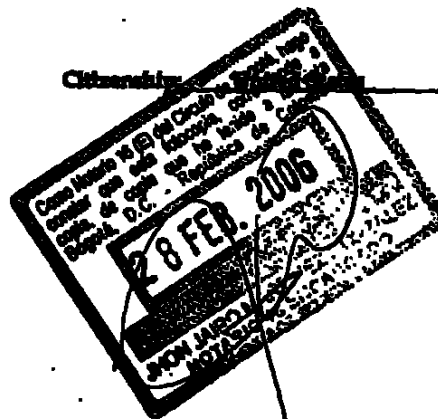
I hereby declare that all statements made herein of my own knowledge are true and that all statements made on information
 belief are believed to be true; and further that these statements were made with the knowledge that willful false
 statements and the like so made are punishable by fine or imprisonment, or both, under Section 1001 of Title 18 of the United
 States Code and that such willful false statements may jeopardize the validity of the application or any patent issued thereon.

Full Name of Inventor: Srinivasanthy P. Acharya Citizenship: United States
 Residence: Sammamish, Washington
 Post Office Address: 2304 23rd Place NE
Sammanish, Washington 98074
[Signature] 3/24/2005
 Inventor's Signature Date

Full Name of Inventor: Jeffrey T. Fagan Citizenship: United States
 Residence: Sammanish, Washington
 Post Office Address: 437 23rd Avenue SE
Sammanish, Washington 98074
[Signature] 3/24/2005
 Inventor's Signature Date

Full Name of Inventor: Michael J. Newman Citizenship: United States
 Residence: Redmond, Washington
 Post Office Address: 18000 NE 50th Court
Redmond, Washington 98073
[Signature] 3/24/05
 Inventor's Signature Date

Full Name of Inventor: Jason T. Hunter Citizenship: United States
 Residence: Redmond, Washington
 Post Office Address: 4635 167th Court NE
Redmond, Washington 98073
[Signature] 3/24/05
 Inventor's Signature Date



154

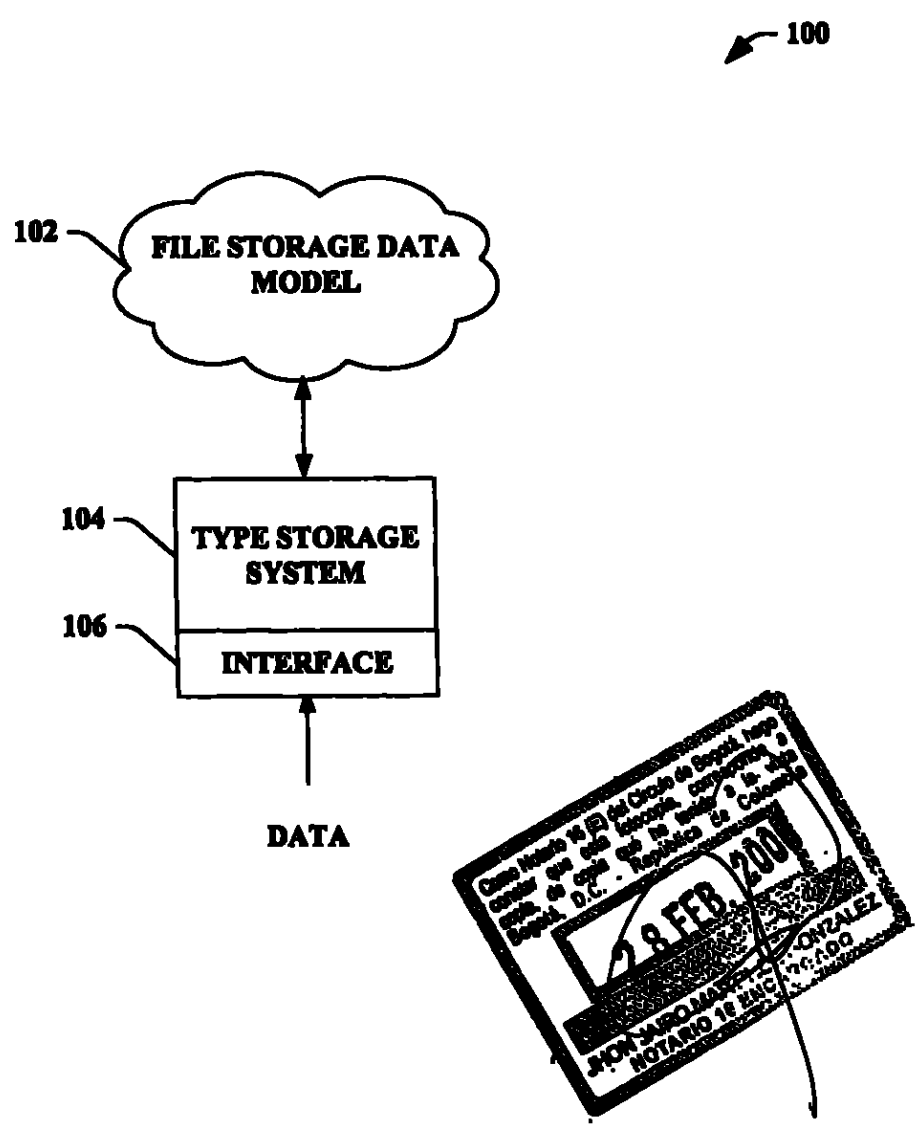


FIG. 1

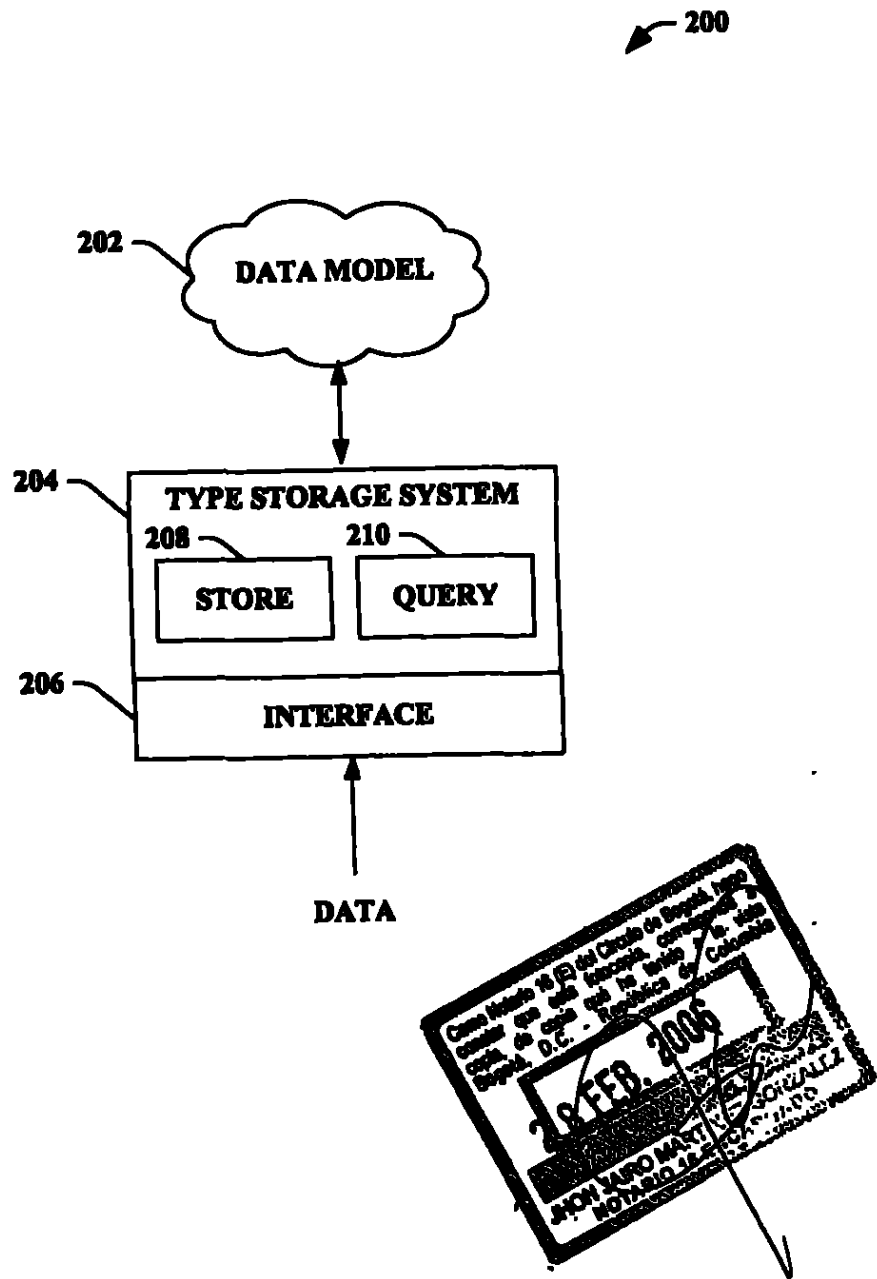


FIG. 2

3/12

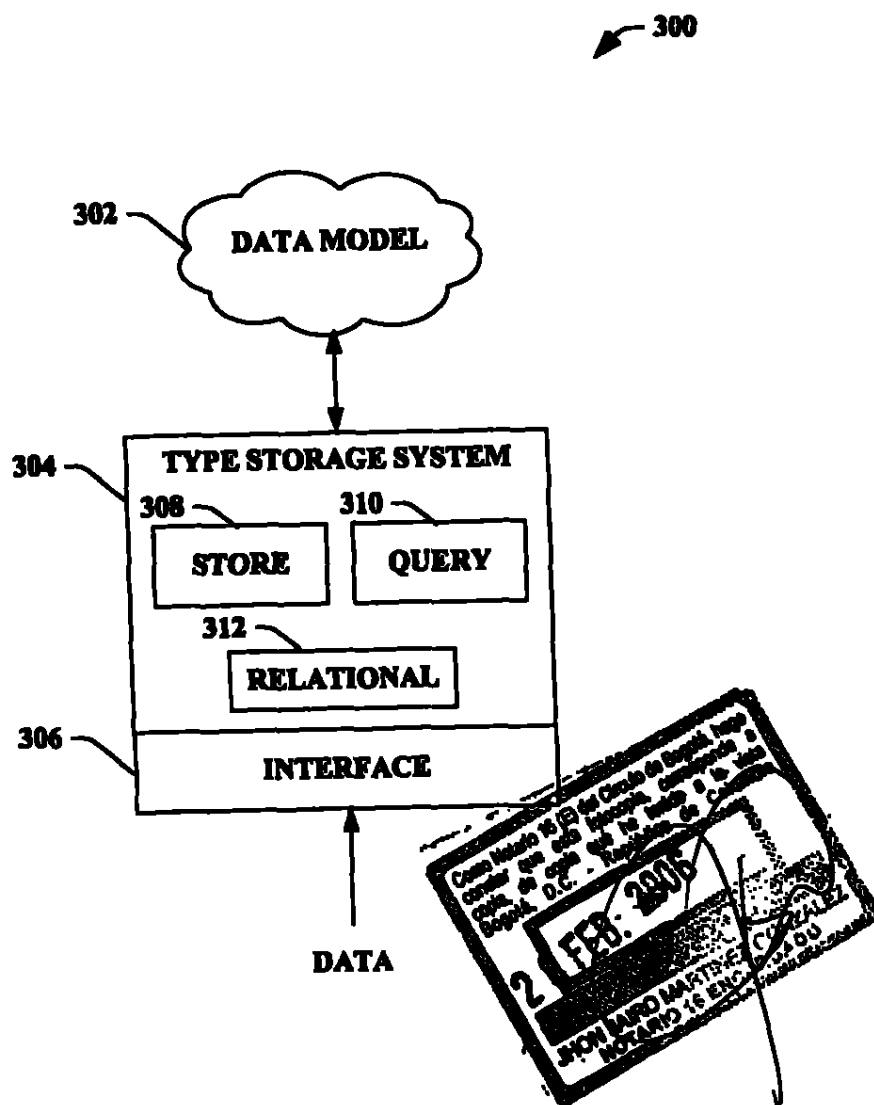


FIG. 3

152

4/12

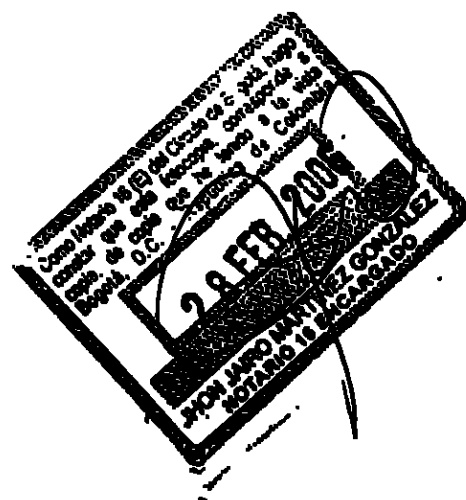
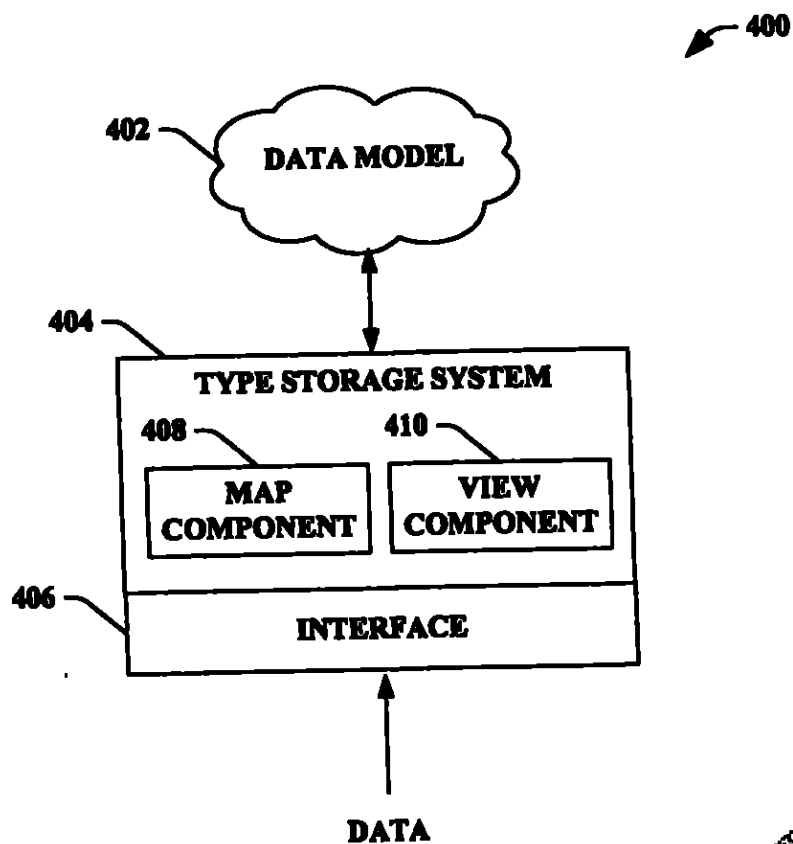


FIG. 4

5/12

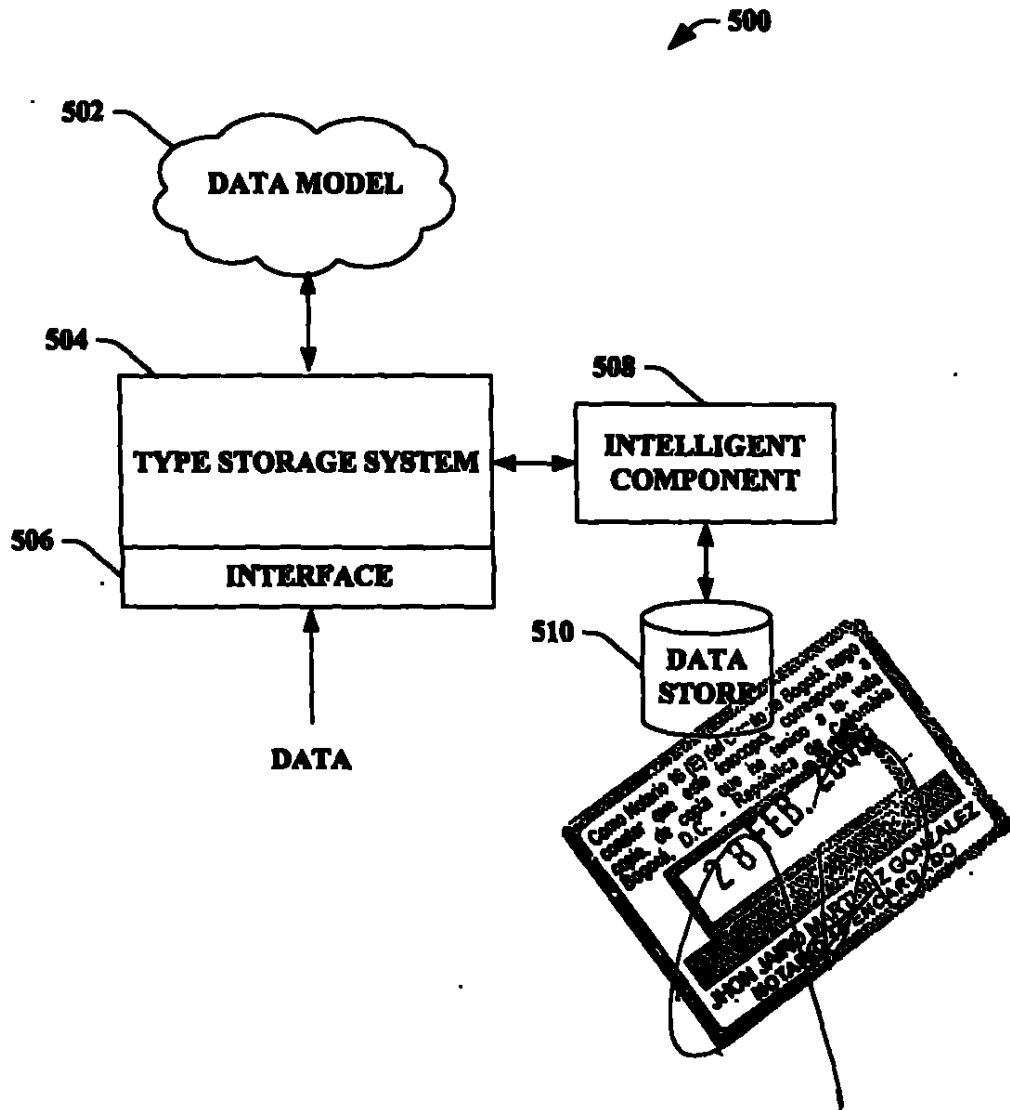


FIG. 5

154

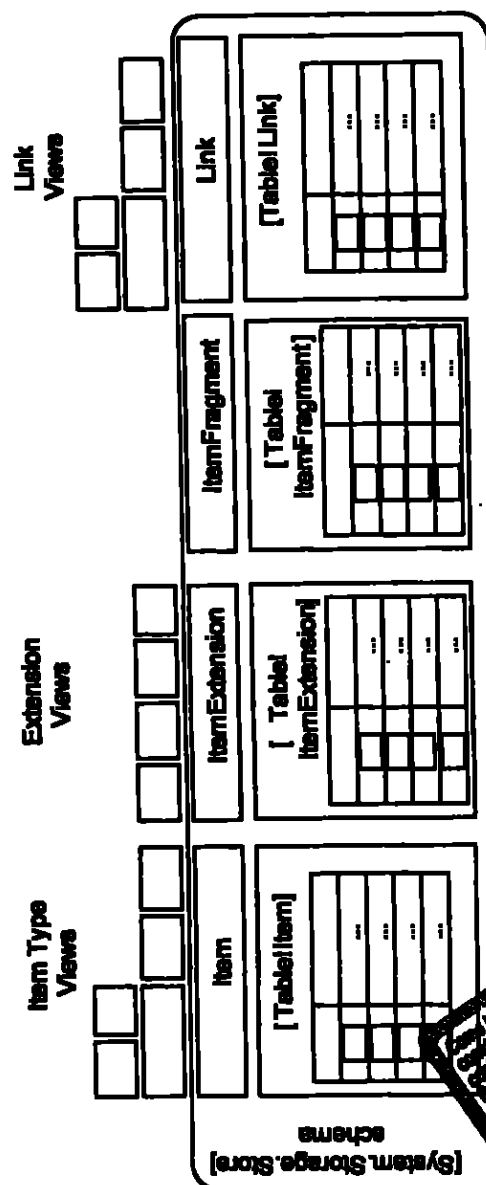
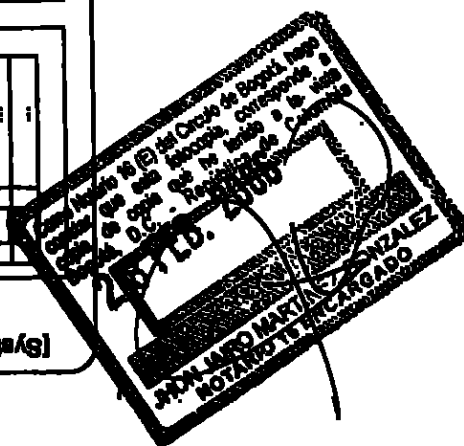


FIG. 6



7/12

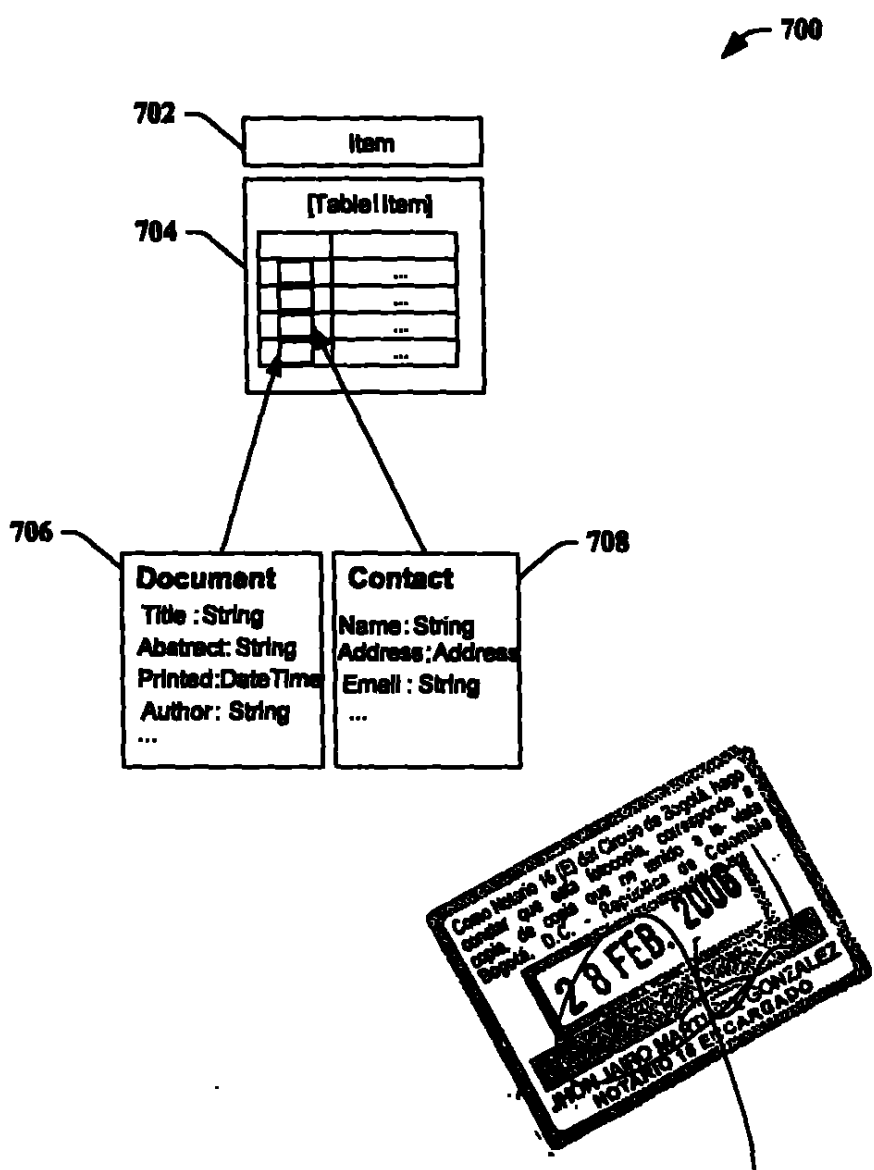


FIG. 7

155

8/12

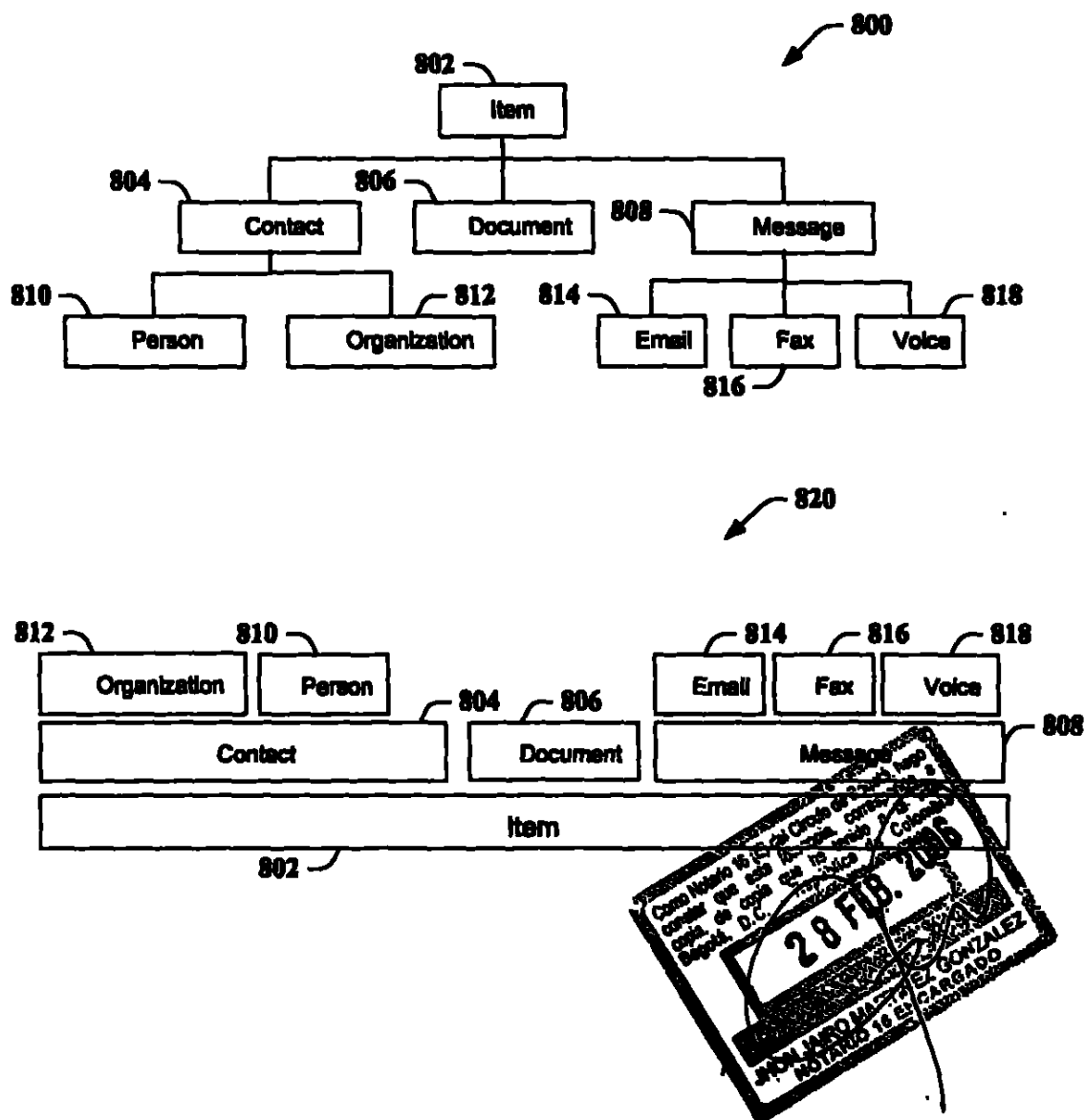


FIG. 8

158

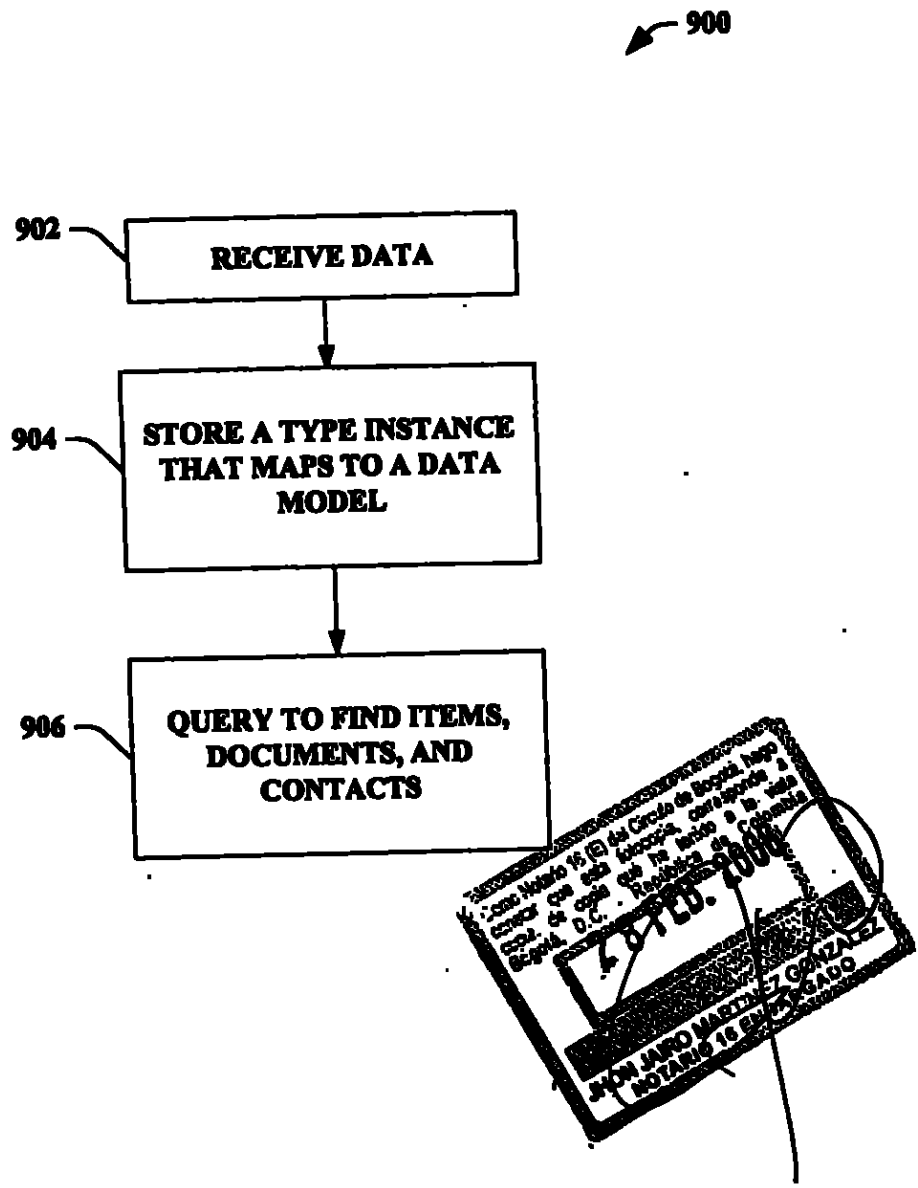


FIG. 9

158

10/12

1000

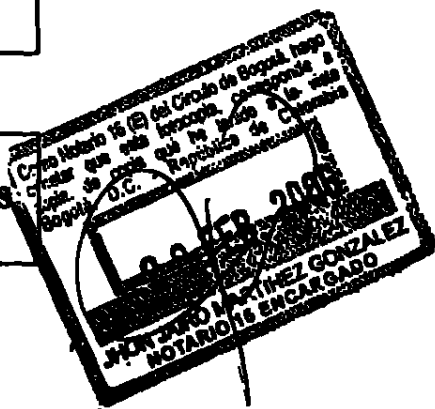
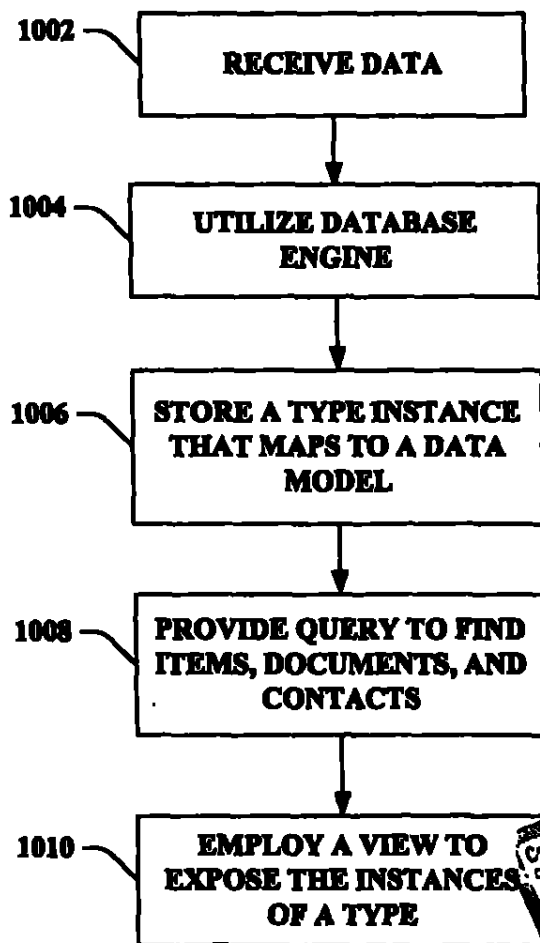


FIG. 10

11/12

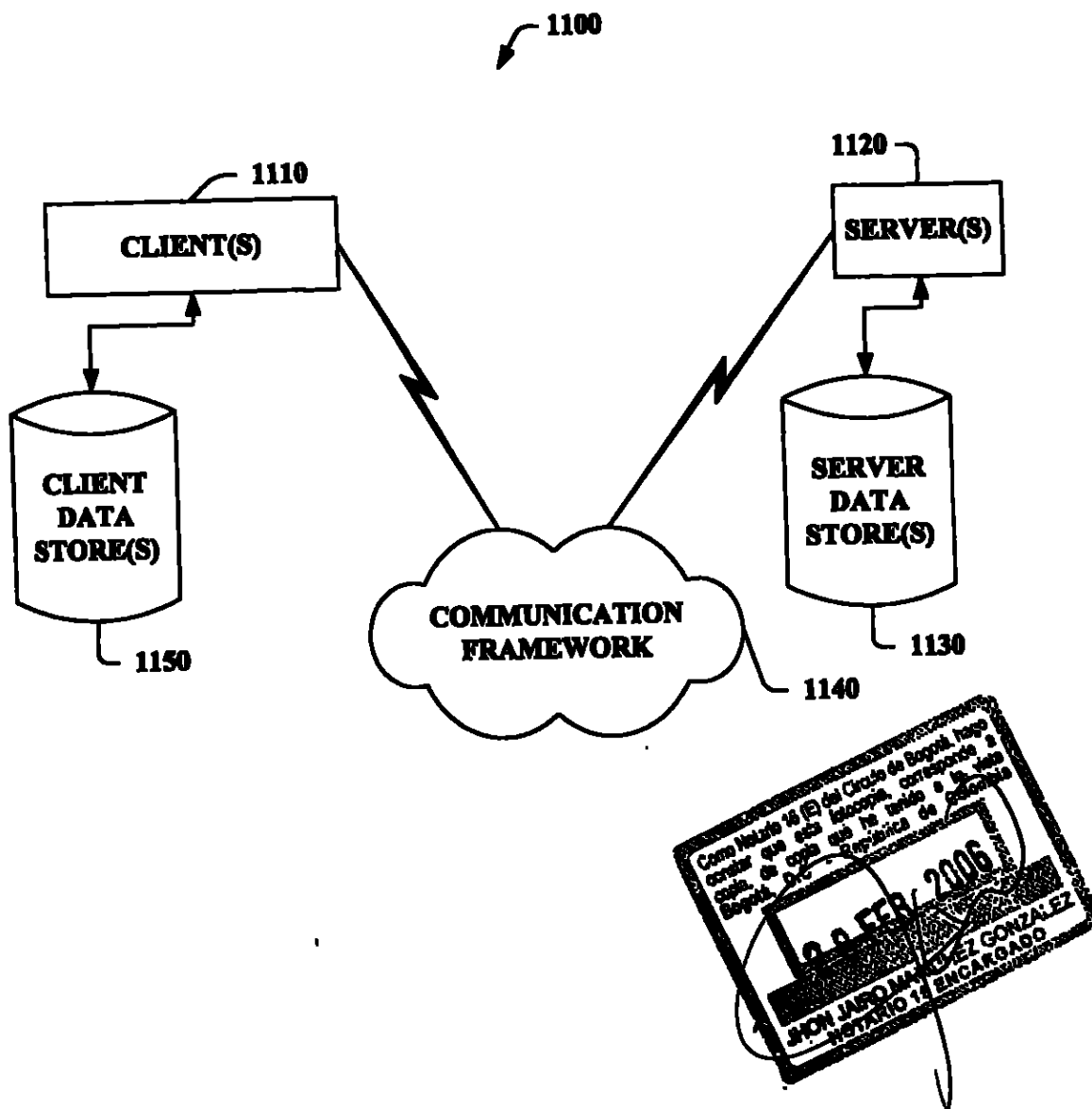


FIG. 11

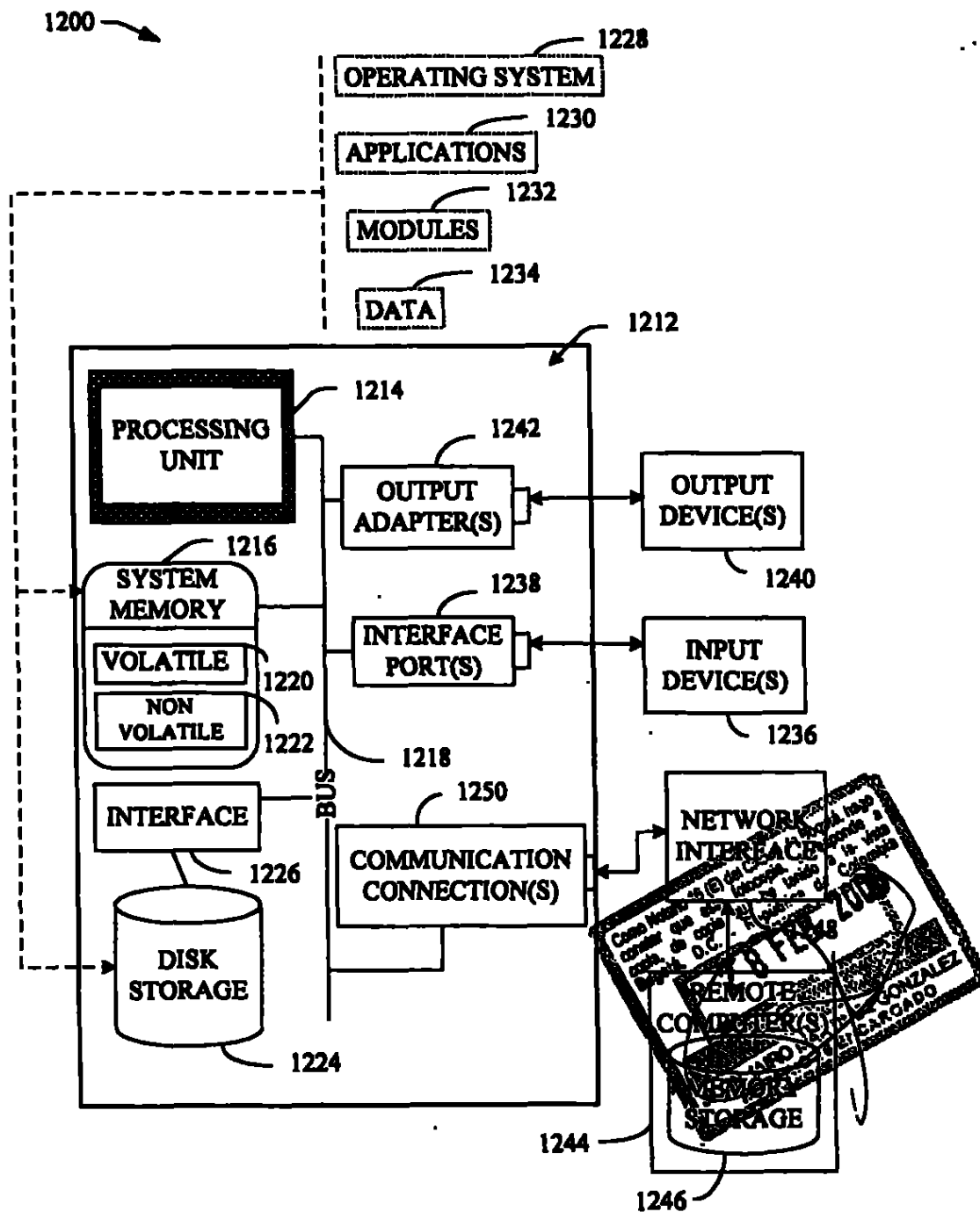


FIG. 12