

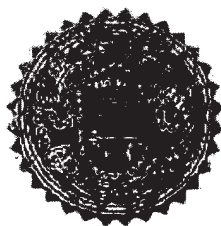
Delaware

PAGE 1

The First State

I, HARRIET SMITH WINDSOR, SECRETARY OF STATE OF THE STATE OF DELAWARE, DO HEREBY CERTIFY THE ATTACHED IS A TRUE AND CORRECT COPY OF THE CERTIFICATE OF AMENDMENT OF "SAVVIS ASSET HOLDINGS, INC.", CHANGING ITS NAME FROM "SAVVIS ASSET HOLDINGS, INC." TO "SAVVIS, INC.", FILED IN THIS OFFICE ON THE EIGHTH DAY OF MARCH, A.D. 2004, AT 11:39 O'CLOCK A.M.

A FILED COPY OF THIS CERTIFICATE HAS BEEN FORWARDED TO THE NEW CASTLE COUNTY RECORDER OF DEEDS.



3752675 8100

040171451

Harriet Smith Windsor

Harriet Smith Windsor, Secretary of State
AUTHENTICATION: 2974022

DATE: 03-08-04

**CERTIFICATE OF AMENDMENT
OF
CERTIFICATE OF INCORPORATION
OF**

SAVVIS ASSET HOLDINGS, INC.

It is hereby certified that:

FIRST: The name of the corporation is SAVVIS Asset Holdings, Inc., a corporation organized and existing under the applicable laws of the State of Delaware (hereinafter, the "Corporation").

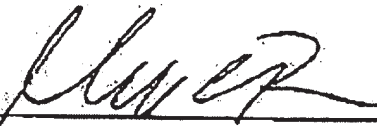
SECOND: The certificate of incorporation of the Corporation is hereby amended by deleting Article First thereof and substituting in lieu of said Article the following new Article:

"FIRST. The name of the corporation (hereinafter called the "Corporation") is SAVVIS, Inc."

THIRD: This Certificate of Amendment has been duly adopted and written consent has been given in accordance with the provisions of Section 228 and Section 242 of the General Corporation Law of the State of Delaware.

Dated: March 8, 2004

SAVVIS Asset Holdings, Inc., a Delaware corporation

By: 

Name: Grier C. Radlin
Title: Secretary

NAME CHANGE FROM:

DIGITAL ISLAND, INC.

TO

CABLE & WIRELESS INTERNET SERVICES, INC.

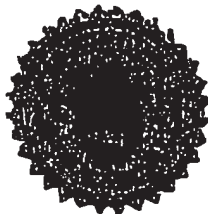
Delaware

PAGE 1

The First State

I, HARRIET SMITH WINDSOR, SECRETARY OF STATE OF THE STATE OF DELAWARE, DO HEREBY CERTIFY THE ATTACHED IS A TRUE AND CORRECT COPY OF THE CERTIFICATE OF AMENDMENT OF "DIGITAL ISLAND, INC.", CHANGING ITS NAME FROM "DIGITAL ISLAND, INC." TO "CABLE & WIRELESS INTERNET SERVICES, INC.", FILED IN THIS OFFICE ON THE TWELFTH DAY OF FEBRUARY, A.D. 2002, AT 9 O'CLOCK A.M.

Draft Only



3016554 8100

Harriet Smith Windsor
Harriet Smith Windsor, Secretary of State

AUTHENTICATION: 1936267

**CERTIFICATE OF AMENDMENT
OF THE
CERTIFICATE OF INCORPORATION
OF
DIGITAL ISLAND, INC.,
a Delaware corporation**

The undersigned William M. Austin and Howard Lasky hereby certify that:

ONE: They are the duly elected and acting Chief Executive Officer and Secretary, respectively, of said corporation.

TWO: The Amended and Restated Certificate of Incorporation of said corporation, last filed on July 2, 1999 shall be amended as set forth in this Certificate of Amendment.

THREE: Article First of the Certificate of Incorporation of said corporation is hereby amended to read in its entirety as follows:

"FIRST. The name of the corporation is Cable & Wireless Internet Services, Inc. (hereinafter sometimes called the "Corporation")."

FOUR: This amendment of the Certificate of Incorporation herein certified has been duly adopted in accordance with the provisions of Sections 228 and 242 of the General Corporation Law of the State of Delaware.

The undersigned further declare under penalty of perjury that the matters set forth in this Certificate of Amendment are true and correct of their own knowledge.

IN WITNESS WHEREOF, the undersigned have executed this Certificate of Amendment at San Francisco, California, on this 12th day of February, 2002.



William M. Austin, Chief Executive Officer



Howard Rasky, Secretary



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/987,723	11/15/2001	David A. Farber	P 283002 TN-Cont-2	7771

26633 7590 12/01/2003

HELLER EHRMAN WHITE & MCAULIFFE LLP
1666 K STREET,NW
SUITE 300
WASHINGTON, DC 20006

EXAMINER

HOMERE, JEAN RAYMOND

ART UNIT	PAPER NUMBER
----------	--------------

2177

DATE MAILED: 12/01/2003

12

Please find below and/or attached an Office communication concerning this application or proceeding.

Draft Only

Office Action Summary

Applicati n N .

09/987,723

Applicant(s)

FARBER ET AL.

Examiner

Jean R. Homere

Art Unit

2177

-- The MAILING DATE of this c mmunicati n appears on the cover sheet with the correspondence address --

Peri d for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133).
- Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

- 1) ☒ Responsive to communication(s) filed on 14 October 2003.
- 2a) ☐ This action is FINAL. 2b) ☒ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

- 4) ☒ Claim(s) 54-109 is/are pending in the application.
- 4a) Of the above claim(s) _____ is/are withdrawn from consideration.
- 5) ☐ Claim(s) _____ is/are allowed.
- 6) ☒ Claim(s) 54-109 is/are rejected.
- 7) ☐ Claim(s) _____ is/are objected to.
- 8) ☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

- 9) ☒ The specification is objected to by the Examiner.
- 10) ☐ The drawing(s) filed on _____ is/are: a) ☐ accepted or b) ☐ objected to by the Examiner.
- Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).
- Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. §§ 119 and 120

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) ☐ All b) ☐ Some * c) ☐ None of:
- ☐ Certified copies of the priority documents have been received.
 - ☐ Certified copies of the priority documents have been received in Application No. _____.
 - ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).
- * See the attached detailed Office action for a list of the certified copies not received.
- 13) ☐ Acknowledgment is made of a claim for domestic priority under 35 U.S.C. § 119(e) (to a provisional application) since a specific reference was included in the first sentence of the specification or in an Application Data Sheet. 37 CFR 1.78.
- a) ☐ The translation of the foreign language provisional application has been received.
- 14) ☐ Acknowledgment is made of a claim for domestic priority under 35 U.S.C. §§ 120 and/or 121 since a specific reference was included in the first sentence of the specification or in an Application Data Sheet. 37 CFR 1.78.

Attachment(s)

- 1) ☒ Notice of References Cited (PTO-892)
- 2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
- 3) ☒ Information Disclosure Statement(s) (PTO-1449) Paper No(s) 2,8,9.
- 4) ☐ Interview Summary (PTO-413) Paper No(s). _____.
- 5) ☐ Notice of Informal Patent Application (PTO-152)
- 6) ☐ Other: _____.

DETAILED ACTION

Information Disclosure Statement

1. The information disclosure statements (IDS) submitted on 09/03/03, 11/15/01 comply with the provisions of 37 CFR 1.97. Accordingly, the information disclosure statements are being considered by the examiner.
2. The 10 page IDS filed on 07/29/03 contains the same references cited in the IDS filed on 11/15/01. Accordingly, the information referred to therein has not been considered by the examiner. See attached IDS.

Related Applications

3. Applicants are advised to provide the serial numbers and corresponding statuses of all patent applications referenced at pages 1 of this application.

Drawings

4. This application has been filed with informal drawings, which are acceptable for examination purposes only. Formal drawings will be required when the application is allowed.

Specification

5. The title of the invention is not descriptive. A new title is required that is clearly indicative of the invention to which the claims are directed.

The abstract of the disclosure is objected to because it contains legal phraseologies (e.g. "comprising"). Correction is required. See MPEP § 608.01(b).

Art Unit: 2177

Applicant is reminded of the proper language and format for an abstract of the disclosure.

The abstract should be in narrative form and generally limited to a single paragraph on a separate sheet within the range of 50 to 150 words. It is important that the abstract not exceed 150 words in length since the space provided for the abstract on the computer tape used by the printer is limited. **The form and legal phraseology often used in patent claims, such as "means" and "said," should be avoided.** The abstract should describe the disclosure sufficiently to assist readers in deciding whether there is a need for consulting the full patent text for details.

The language should be clear and concise and should not repeat information given in the title. It should avoid using phrases which can be implied, such as, "The disclosure concerns," "The disclosure defined by this invention," "The disclosure describes," etc.

Double Patenting

6. The nonstatutory double patenting rejection is based on a judicially created doctrine grounded in public policy (a policy reflected in the statute) so as to prevent the unjustified or improper timewise extension of the "right to exclude" granted by a patent and to prevent possible harassment by multiple assignees. See *In re Goodman*, 11 F.3d 1046, 29 USPQ2d 2010 (Fed. Cir. 1993); *In re Longi*, 759 F.2d 887, 225 USPQ 645 (Fed. Cir. 1985); *In re Van Ornum*, 686 F.2d 937, 214 USPQ 761 (CCPA 1982); *In re Vogel*, 422 F.2d 438, 164 USPQ 619 (CCPA 1970); and, *In re Thorington*, 418 F.2d 528, 163 USPQ 644 (CCPA 1969).

A timely filed terminal disclaimer in compliance with 37 CFR 1.321(c) may be used to overcome an actual or provisional rejection based on a nonstatutory double patenting ground provided the conflicting application or patent is shown to be commonly owned with this application. See 37 CFR 1.130(b).

Effective January 1, 1994, a registered attorney or agent of record may sign a terminal disclaimer. A terminal disclaimer signed by the assignee must fully comply with 37 CFR 3.73(b).

7. Claims 54-109 are rejected under the judicially created doctrine of obviousness-type double patenting as being unpatentable over claims 1-55 of U.S. Patent No.

6,415,280. Although the conflicting claims are not identical, they are not patentably distinct from each other. Claims 54-109 do not recite the limitation of whereby at least some of the files being cached versions of files from a source server distinct from the

Art Unit: 2177

network of servers. It would have been obvious to one of ordinary skill in the art to modify the claims of the present application to achieve the delivery of the content of a file on the distributed network without having to use cache versions of some of the files. Such modification would simplify the delivery process of the files in question. Further, it has been held that omission of an element and its function in a combination where the remaining elements perform the same functions as before involves only routine skill in the art. *In re Karlson*, 136 USPQ 184 (CCPA 1963).

Claim Rejections - 35 USC § 103

8. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

This application currently names joint inventors. In considering patentability of the claims under 35 U.S.C. 103(a), the examiner presumes that the subject matter of the various claims was commonly owned at the time any inventions covered therein were made absent any evidence to the contrary. Applicant is advised of the obligation under 37 CFR 1.56 to point out the inventor and invention dates of each claim that was not commonly owned at the time a later invention was made in order for the examiner to consider the applicability of 35 U.S.C. 103© and potential 35 U.S.C. 102(f) or (g) prior art under 35 U.S.C. 103(a).

Art Unit: 2177

9. Claims 54, 56-58, 62-64, 66-94, 96-109 are rejected under 35 U.S.C. 103(a) as being unpatentable over Nelson et al. (Nelson), US. Patent No. 5,452,447, in view of Hamilton et al. (Hamilton), US Patent No. 5,640,564.

As to independent claims 54, 56-58, 62-64, 70, 74, 75, 83, 84, 92, 96, 98, 102-105, 107, Nelson substantially discloses the invention including a data processing system for caching a file server to thereby allow client users to request and retrieve files in a distributed computer system (abstract, lines 1-8 et seq.) In particular, Nelson discloses the use of a hash function on a data file to thereby quickly retrieve a copy of the data file from the cache upon user's request (col.17, lines 18-41 et seq.)

Nelson does not particularly detail the use of the hashing function on the data file to create an identifier, which can be utilized to retrieve the data file upon user's request. However, Hamilton discloses an analogous system wherein a hashing function is applied to a data item to thereby create an identifier, which a user can utilize to request and retrieve a corresponding data item (col. 6, lines 28-39 et seq.) It would have been obvious to one of ordinary skill in the art of data processing to combine the teachings of the cited references because Hamilton's teaching would allow the users of Nelson's system to expeditiously and dynamically retrieve a file as it is being updated.

As to claims 66-67, 69, 71-73, Nelson teaches a security system to provide access to requested files to authorized users only (col. 18, lines 33-46 et seq.)

As to claim 68, Nelson discloses the retrieval of a data item from another server, if it is not located on a given server (col. 15, lines 23-26; col. 15, lines 50-60 et seq.)

Art Unit: 2177

As to claim 79, Nelson teaches a substantially unique value of the data file (col. 17, lines 20-41 et seq.)

As to claims 80- 81, Nelson discloses that a plurality of data files can be distributed across the network as cached version of another server (col. 7, lines 29-35 et seq.)

As to claim 82, Nelson teaches that the name of the file will change when the file is modified (col. 22, lines 26-52 et seq.)

10. The limitations of claims 76-78, 85-91, 93-94, 97, 99, 100, 101, 106, 108, 109 have already been discussed in the rejection of claims 54, 56-58, 62-64, 66-75, 79-84, 92, 96, 98, 102-105, and 107 above . They are therefore rejected on similar grounds.

11. Claims 55, 59-61, 65 and 95 are rejected under 35 U.S.C. 103(a) as being unpatentable over Nelson et al. (Nelson), US. Patent No. 5,452,447, in view of Hamilton et al. (Hamilton), US Patent No. 5,640,564, further in view of Peter Deutsch ("Deutsch"), "Re: MD5 and LIFNS (was: Misc Comments)", supplied by applicants in IDS, paper no. 9.

As to claims 55, 59-61, 65 and 95, Nelson substantially discloses the invention including a data processing system for caching a file server to thereby allow client users to request and retrieve files in a distributed computer system (abstract, lines 1-8 et seq.) In particular, Nelson discloses the use of a hash function on a data file to thereby quickly

retrieve a copy of the data file from the cache upon user's request (col.17, lines 18-41 et seq.)

Nelson does not particularly detail the use of the hashing function on the data file to create an identifier, which can be utilized to retrieve the data file upon user's request. However, Hamilton discloses an analogous system wherein a hashing function is applied to a data item to thereby create an identifier, which a user can utilize to request and retrieve a corresponding data item (col. 6, lines 28-39 et seq.) It would have been obvious to one of ordinary skill in the art of data processing to combine the teachings of the cited references because Hamilton's teaching would allow the users of Nelson's system to expeditiously and dynamically retrieve a file as it is updated.

Nelson and Hamilton do not particularly teach that the use of an MD5 function to determine the name of a file. Deutsch, however, discloses that the use of MD5 is a suitable mechanism for determining a URN to thereby index said URN in the MD5 digest. It would have been obvious to one of ordinary skill in the art of data processing to combine the teachings of the cited references. Deutsch's teaching would provide users of the Nelson-Hamilton's system with an indexed name files stored in a digest to thereby quickly retrieve a requested file therefrom.

Conclusion

12. The prior art made of record and not relied upon is considered pertinent to applicant's disclosure. Please see attached PTO-892.

Art Unit: 2177

13. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Jean R. Homer whose telephone number is (703)-308-6647. The examiner can normally be reached on Monday-Friday from 09:30 a.m.-6:00 p.m.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, John Breen, can be reached on Monday-Friday from 8:00 a.m. to 3:30 p.m. at (703)-305-9790.

Any response to this action should be mailed to: Commissioner of Patents and Trademarks Washington, D.C. 20231, **or faxed to:** (703) 872-9306. Hand-delivered responses should be brought to Crystal Park II, 2121 Crystal Drive, Arlington, VA., Sixth Floor (Receptionist).

Any inquiry of a general nature or relating to the status of this application should be directed to the Group receptionist whose telephone number is (703) 305-3900.


Jean R. Homer
Primary Examiner, A.U. 2177
November 24, 2003

Notice of References Cited	Application/Control No. 09/987,723	Applicant(s)/Patent Under Reexamination FARBER ET AL.	
	Examiner Jean R. Homere	Art Unit 2177	Page 1 of 1

U.S. PATENT DOCUMENTS

*		Document Number Country Code-Number-Kind Code	Date MM-YYYY	Name	Classification
	A	US-5,781,629	07-1998	Haber et al.	713/177
	B	US-5,835,087	11-1998	Herz et al.	345/810
	C	US-			
	D	US-			
	E	US-			
	F	US-			
	G	US-			
	H	US-			
	I	US-			
	J	US-			
	K	US-			
	L	US-			
	M	US-			

FOREIGN PATENT DOCUMENTS

*		Document Number Country Code-Number-Kind Code	Date MM-YYYY	Country	Name	Classification
	N					
	O					
	P					
	Q					
	R					
	S					
	T					

NON-PATENT DOCUMENTS

*		Include as applicable: Author, Title Date, Publisher, Edition or Volume, Pertinent Pages)
	U	
	V	
	W	
	X	

*A copy of this reference is not being furnished with this Office action. (See MPEP § 707.05(a).)
Dates in MM-YYYY format are publication dates. Classifications may be US or foreign.

Atty.
Dkt. No.

M#

Client Ref.

#2
1-3-02

283002

Applicant: FARBER et al.

Appln. No.: Cont. of 09/283,160

Filing Date: November 15, 2001

Examiner: Jean Homere

Group Art Unit: 2177

INFORMATION DISCLOSURE STATEMENT BY APPLICANT

Date: November 15, 2001

Page

1

of

7

U.S. PATENT DOCUMENTS

Examiner's Initials*	Document Number	Date MM/YYYY	Name (Family Name of First Inventor)	Class	Sub Class	Filing Date (if appropriate)
JRH	AR 5,638,443	06/1997	STEFIK et al.	705	54	11/23/94
	BR 5,802,291	09/1998	BALICK et al.	709	202	3/30/95
	CR 5,542,087	07/1996	NEIMAT et al.	707	10	10/15/93
	DR 5,287,499	02/1994	NEMES	707	2	5/16/91
	ER 4,922,417	05/1990	CHURM et al.	707	1	10/24/86
	FR 5,341,477	08/1994	PITKIN et al.	709	226	8/6/93
	GR 5,907,704	05/1999	GUDMUNDSON et al.			4/3/95
	HR 6,134,603	10/2000	Jones et al.	709	330	
	IR 6,006,018	12/1999	Burnett et al.	395	200.49	
	JR 5,809,494	09/1998	Nguyen	707	1	
	KR 5,640,564	06/1997	Hamilton et al.	709	303	
	LR 5,452,447	09/1995	Nelson et al.	707	205	
V	MR 5,202,982	04/1993	Gramlich et al.	707	2	

FOREIGN PATENT DOCUMENTS

Document Number	Date MM/YYYY	Country	Inventor Name	English Abstract	Translation Readily Available
				Enclosed	No
NR 05162529	06/1993	JP	HOROSHI	x	x

OTHER (Including in this order Author, Title, Periodical Name, Date, Pertinent Pages, etc.)

JRH	OR	Gwertzman, James, et al. "The Case for Geographical Push-Caching." Technical Report HU TR 34-94 (excerpt), Harvard University, DAS, Cambridge, MA 02138, 1994, 2 pgs.			
	PR	Grigni, Michelangelo, et al. "Tight Bounds on Minimum Broadcasts Networks." SIAM Journal of Discrete Mathematics, Vol. 4, No. 2, May 1991, pgs. 207-222.			
	QR	Devine, Robert. "Design and Implementation of DDH: A Distributed Dynamic Hashing Algorithm." In Proceedings of 4th International Conference on Foundations of Data Organizations and Algorithms, 1993, pgs. 101-114.			
	RR	Deering, Stephen, et al. "Multicast Routing in Datagram Internetworks and Extended LANs." ACM Transactions on Computer Systems, Vol. 8, No. 2, May 1990, pgs. 85-110.			
	SR	Cormen, Thomas H., et al. Introduction to Algorithms, The MIT Press, Cambridge, Massachusetts, 1994, pgs. 219-243, 991-993.			
V	TR	Naor, Moni, et al. "The Load, Capacity and Availability of Quorum Systems." In Proceedings of the 35th IEEE Symposium on Foundations of Computer Science, November 1994, pgs. 214-225.			

Examiner Jean R Homere

Date Considered: 5-01-03

*EXAMINER: Initial if citation considered, whether or not citation is in conformance with MPEP § 609. Draw line through citation if not in conformance and not considered. Include copy of this form with next communication to Applicant.

FORM PTO-1449 (modified)
To: U.S. Department of Commerce
(PW FORM PAT-1449)
Patent and Trademark Office

Atty.
Dkt. No.

M#

Client Ref.

283002

INFORMATION DISCLOSURE STATEMENT BY APPLICANT

Applicant: FARBER et al.

Appl. No.: Cont. of 09/283,160

Filing Date: November 15, 2001

Date: November 15, 2001

Page

2

of

7

Examiner: Jean Homere

Group Art Unit: 2177

U.S. PATENT DOCUMENTS

Examiner's Initials*	Document Number	Date MM/YYYY	Name (Family Name of First Inventor)	Class	Sub Class	Filing Date (if appropriate)
-------------------------	--------------------	-----------------	---	-------	--------------	------------------------------------

FOREIGN PATENT DOCUMENTS

Document Number	Date MM/YYYY	Country	Inventor Name	English Abstract	Translation Readily Available
				Enclosed	No
				Enclose	No

OTHER (Including in this order Author, Title, Periodical Name, Date, Pertinent Pages, etc.)

JRH	AR	Nisan, Noam. "Pseudorandom Generators for Space-Bounded Computation." In Proceedings of the Twenty-Second Annual ACM Symposium on Theory of Computing, May 1990, pgs. 204-212.			
	BR	Palmer, Mark, et al. "Fido: A Cache that Learns to Fetch." In Proceedings of the 17th International Conference on Very Large Data Bases, September 1991, pgs. 255-264.			
	CR	Peleg, David, et al. "The Availability of Quorum Systems." Information and Computation 123, 1995, 210-223.			
	DR	Rabin, Michael. "Efficient Dispersal of Information for Security, Load Balancing, and Fault Tolerance." Journal of the ACM, Vol. 36, No. 2, April 1989, pgs. 335-348			
	ER	Ravi, R., "Rapid Rumor Ramification: Approximating the Minimum Broadcast Time." In Proceedings of the 35th IEEE Symposium on Foundation of Computer Science, November 1994, pgs. 202-213.			
	FR	Schmidt, Jeanette, et al. "Chernoff-Hoeffding Bounds for Applications with Limited Independence." In Proceedings of the 4th ACS-SIAM Symposium on Discrete Algorithms, 1993, pgs. 331-340.			
	GR	Tarjan, Robert Endre, et al. "Storing a Sparse Table." Communications of the ACM, Vol. 22, No. 11, November 1979, pgs. 606-611.			
	HR	Wegman, Mark, et al. "New Hash Functions and Their Use in Authentication and Set Equality." Journal of Computer and System Sciences Vol. 22, June 1981, pgs. 265-279.			
	IR	Vitter, Jeffrey Scott, et al. "Optimal Prefetching via Data Compression." In Proceedings of 32nd IEEE Symposium on Foundations of Computer Science, November 1991, pgs. 121-130.			
	JR	Fredman, Michael, et al. "Storing a Sparse Table with O(1) Worst Case Access Time." Journal of the Association for Computing Machinery, Vol. 31, No. 3, July 1984, pgs. 538-544.			
	KR	Yao, Andrew Chi-Chih. "Should Tables be Sorted?" Journal of the Association for Computing Machinery, Vol. 28, No. 3, July 1981, pgs. 615-628.			
	LR	Floyd, Sally, et al. "A reliable Multicast Framework for Light-Weight Sessions and Application Level Framing." In Proceeding of ACM SIGCOMM '95, pgs. 342-356.			
	MR	Feeley, Michael, et al. "Implementing Global Memory Management in a Workstation Cluster." In Proceedings of the 15th ACM Symposium on Operating Systems Principles, 1995, pgs. 201-212.			

Examiner

Jean R Homere

Date Considered:

5-01-03

*EXAMINER: Initial if citation considered, whether or not citation is in conformance with MPEP § 609. Draw line through citation if not in conformance and not considered. Include copy of this form with next communication to Applicant.

FORM PTO-1449 (modified)
To: U.S. Department of Commerce
(PW FORM PAT-1449)
Patent and Trademark Office

Atty.
Dkt. No.

M#

Client Ref.

283002

**INFORMATION DISCLOSURE STATEMENT
BY APPLICANT**

Applicant: FARBER et al.

Appln. No.: Cont. of 09/283,160

Filing Date: November 15, 2001

Date: November 15, 2001

Page

3

of

7

Examiner: Jean Homere

Group Art Unit: 2177

U.S. PATENT DOCUMENTS

Examiner's Initials*	Document Number	Date MM/YYYY	Name (Family Name of First Inventor)	Class	Sub Class	Filing Date (if appropriate)
-------------------------	--------------------	-----------------	---	-------	--------------	------------------------------------

FOREIGN PATENT DOCUMENTS

Document Number	Date MM/YYYY	Country	Inventor Name	English Abstract	Translation Readily Available		
				Enclosed	No	Enclose	No

OTHER (Including in this order Author, Title, Periodical Name, Date, Pertinent Pages, etc.):

JRH	AR	Carter, J. Lawrence, et al. "Universal Classes of Hash Functions." Journal of Computer and System Sciences, Vol. 18, No. 2, April 1979, pgs. 143-154.
	BR	PATENT ABSTRACTS OF JAPAN, "Electronic Mail Multiplexing System and Communication Control Method in The System." 06/30/1993, JP 05162529
	CR	KIM et al., "Experiences with Tripwire: Using Integrity Checkers For Intrusion Detection", COAST Labs. Dept. of Computer Sciences Purdue University, February 22, 1995, pages 1-12
JAH	DR	KIM et al., "The Design and Implementation of Tripwire: A file System Integrity Checker", COAST Labs. Dept. of Computer Sciences Purdue University, November 19, 1993, pages 1-21
	ER	
	FR	
	GR	
	HR	
	IR	
	JR	
	KR	
	LR	
	MR	

Examiner

Jean R Homere

Date Considered: 5-01-03

*EXAMINER: Initial if citation considered, whether or not citation is in conformance with MPEP § 609. Draw line through citation if not in conformance and not considered. Include copy of this form with next communication to Applicant.

Atty.
Dkt. No.

M#

Client Ref.

283002

**INFORMATION DISCLOSURE STATEMENT
BY APPLICANT**

Applicant: FARBER et al.

Appln. No.: Cont. of 09/283,160

Filing Date: November 15, 2001

Date: November 15, 2001

Page 4 of 7

Examiner: Jean Homere

Group Art Unit: 2177

U.S. PATENT DOCUMENTS

Examiner's Initials*		Document Number	Date MM/YYYY	Name (Family Name of First Inventor)	Class	Sub Class	Filing Date (if appropriate)
JRH	AR	4,888,681	12/1989	BARNES			
	BR	4,972,367	11/1990	BURKE			
	CR	5,050,212	09/1991	DYSON			
	DR	5,202,982	04/1993	GRAMLICH			
	ER	5,208,858	05/1993	VOLLERT			
	FR	5,301,316	05/1994	HAMILTON			
	GR	5,343,527	08/1994	MOORE			
	HR	4,571,700	02/1996	EMRY, JR.			
	IR	4,675,810	06/1987	GRUNER			
	JR	5,050,074	09/1991	MARCA			
	KR	5,276,901	01/1994	HOWELL			
	LR	5,384,565	01/1995	CANNON			
	MR	5,202,982	04/1993	GRAMLICH			
	NR	5,357,623	10/1994	MEGORY-COHEN			

FOREIGN PATENT DOCUMENTS

		Document Number	Date MM/YYYY	Country	Inventor Name	Class	SubClass	English Abstract	Translation Readily Available
								Enclosed	No
	OR								
	PR								
	QR								
	RR								
	SR								
	TR								
	UR								

OTHER (Including in this order Author, Title, Periodical Name, Date, Pertinent Pages, etc.)

JRH	VR	Bert dem Boer et al., Collisions for the compression function of MD _s , pp. 292-304							
	WR	Sakti Pramanik et al., Multi-Directory Hasing, 1993, Info. Sys., Vol. 18, No. 1, pp. 63-74							
	XR	Murlidhar Koushik, Dynamic Hashing with Distributed Overflow Space: A File Organization with Good Insertion Performance, 1993, Info. Sys., Vol. 18, No. 5, pp. 299-317							
	YR	Witold Litwin et al., LH*-Linear Hashing for Distributed Files, HP Labs Tech. Report No. HPL-93-21, June 1993, pp. 1-22							
	ZR	Yuliang Zheng et al., HAVAL - A One-Way Hashing Algorithm with Variable Length of Output (Extended Abstract), pp. 83-105							
	AAR	Chris Charnes and Josef Pieprzky, Linear Nonequivalence versus Nonlinearity, Pieprzky, pp. 156-164							

Examiner Jean R Homere

Date Considered: 5-01-03

*EXAMINER: Initial if citation considered, whether or not citation is in conformance with MPEP § 609. Draw line through citation if not in conformance and not considered. Include copy of this form with next communication to Applicant.

Atty.
Dkt. No.

M#

Client Ref.

283002

**INFORMATION DISCLOSURE STATEMENT
BY APPLICANT**

Applicant: FARBER et al.

Appln. No.: Cont. of 09/283,160

Filing Date: November 15, 2001

Date: November 15, 2001

Page

5

of

7

Examiner: Jean Homere

Group Art Unit: 2177

U.S. PATENT DOCUMENTS

Examiner's Initials*		Document Number	Date MM/YYYY	Name (Family Name of First Inventor)	Class	Sub Class	Filing Date (if appropriate)
JRH	AR	3,668,647	06/1992	EVANGELISTI			
	BR	4,215,402	07/1980	MITCHELL			
	CR	4,290,105	09/1981	CICHELLI			
	DR	4,376,299	03/1983	RIVEST			
	ER	4,405,829	09/1983	RIVEST			
	FR	4,412,285	10/1983	NECHES			
	GR	4,414,624	11/1983	SUMMER, JR.			
	HR	4,441,155	04/1984	FLETCHER			
	IR	4,464,713	08/1984	BENHASE			
	JR	5,772,933	03/1986	MATICK			
	KR	4,642,793	02/1987	MEADEN			
	LR	4,691,2999	09/1987	RIVEST			
	MR	4,725,945	02/1988	KRONSTADT			
	NR	4,773,039	09/1988	ZAMORA			

FOREIGN PATENT DOCUMENTS

ORIGIN: FREE DOCUMENT											
		Document Number	Date MM/YYYY	Country	Inventor Name	Class	SubClass	Abstract		Readily Available	
								Enclosed	No	Enclose	No
	OR										
	PR										
	QR										
	RR										
	SR										
	TR										
	UR										

OTHER (Including in this order Author, Title, Periodical Name, Date, Pertinent Pages, etc.)

JRH	VR	Witold Litwin et al., Linear Hashing for Distributed Files, ACM SIGMOD, May 1993, pp. 327-336				
	WR	Ming-Ling Lo et al., On Optimal Processor Allocation to Support Pipelined Hash Joins, ACM SIGMOD, pp. 69-78, May 1993				
	XR	Thomas A. Berson, Differential Cryptanalysis Mod 2^{32} with Applications to MD5, pp. 69-81				
	YR	William Perrizo et al., Distributed Join Processing Performance Evaluation, Twenty-Seventh Hawaii International Conference on System Sciences, Vol. II, pp. 236-244				
	ZR	Vijay Kumar, A Concurrency Control Mechanism Based on Extendible Hashing for Main Memory Database Systems, ACM, Vol. 3, 1989, pp. 109-113				
	AAR	Birgit Pfizman, Sorting Out Signature Schemes, November 1993, 1 st Conf. Computer & Comm. Security '93, p. 74-85				

Examiner

Jean R Homere

Date Considered:

5-01-03

*EXAMINER: Initial if citation considered, whether or not citation is in conformance with MPEP § 609. Draw line through citation if not in conformance and not considered. Include copy of this form with next communication to Applicant.

Atty.
Dkt. No.

M#

Client Ref.

283002

**INFORMATION DISCLOSURE STATEMENT
BY APPLICANT**

Applicant: FARBER et al.

Appln. No.: Cont. of 09/283,160

Filing Date: November 15, 2001

Date: November 15, 2001

Page

6

of

7

Examiner: Jean Homere

Group Art Unit: 2177

U.S. PATENT DOCUMENTS

Examiner's Initials*		Document Number	Date MM/YYYY	Name (Family Name of First Inventor)	Class	Sub Class	Filing Date (if appropriate)
JRH	AR	4,887,235	12/1989	HOLLOWAY			
	BR	4,888,681	12/1989	BARNES			
	CR	4,490,782	12/1984	DIXON			
	DR	4,972,367	11/1990	BURKE			
	ER	4,922,414	05/1990	HOLLOWAY			
	FR	5,057,837	10/1991	COLWELL			
	GR	5,007,658	12/1991	BENDERT			
	HR	5,025,421	06/1991	CHO			
	IR	5,129,081	07/1992	KOBAYASHI			
	JR	5,129,082	07/1992	TIRFING			
	KR	5,144,667	09/1992	POGUE, JR			
	LR	5,179,680	01/1993	COLWELL			
	MR	5,301,286	04/1994	RAJANI			
	NR	5,404,508	04/1995	KONRAD			

FOREIGN PATENT DOCUMENTS

ORIGINAL PAPER DOCUMENT											
		Document Number	Date MM/YYYY	Country	Inventor Name	Class	SubClass	Abstract		Readily Available	
								Enclosed	No	Enclose	No
	OR										
	PR										
	QR										
	RR										

OTHER (Including in this order Author, Title, Periodical Name, Date, Pertinent Pages, etc.)

JRH	SR	Zhiyu Tian et al., A New Hashing Function: Statistical Behaviour and Algorithm, pp. 3-13									
	TR	G. L. Friedman, Digital Camera with Apparatu for Authentication of Images Produced from an Image File, NASA CASE NO. NPO-19108-1-CU, Serial No. 08/159,980, November 24, 1993									
	UR	H. Goodman, Ada, Object-Oriented Techniques, and Concurrency in Teaching Data Structures and File Management REPORT DOCUMENTATION PAGE AD-A275 385 - 94-04277									
	VR	Advances in Cryptology-EUROCRYPT'93, Workshop on the Theory and Application of Cryptographic Techniques Lofthus, Norway, May 23-27, 1993 Proceedings									
	WR	Proceedings of the 1993 ACM SIGMOD International Conference on Management of Data, Vol. 22, Issue 2, June 1993									
	XR	Advances in Cryptology-AUSCRYPT '92 - Workshop on the Theory and Application of Cryptographic Techniques Gold Coast, Queensland, Australia, December 13-16, 1992 Proceedings									

Examiner Jean R Homere

Date Considered: 5-01-03

*EXAMINER: Initial if citation considered, whether or not citation is in conformance with MPEP § 609. Draw line through citation if not in conformance and not considered. Include copy of this form with next communication to Applicant.

Atty.
Dkt. No.

M#

Client Ref.

283002

Applicant: FARBER et al.

Appl. No.: Cont. of 09/283,160

Filing Date: November 15, 2001

Examiner: Jean Homere

Group Art Unit: 2177

**INFORMATION DISCLOSURE STATEMENT
BY APPLICANT**

Date: November 15, 2001

Page

7

of

7

U.S. PATENT DOCUMENTS

Examiner's Initials*		Document Number	Date MM/YYYY	Name (Family Name of First Inventor)	Class	Sub Class	Filing Date (if appropriate)
	AR						
	BR						
	CR						
	DR						
	ER						
	FR						
	GR						
	HR						
	IR						
	JR						
	KR						

FOREIGN PATENT DOCUMENTS

		Document Number	Date MM/YYYY	Country	Inventor Name	English Abstract		Translation Readily Available	
						Enclosed	No	Enclose	No
	LR								
	MR								
	NR								
	OR								
	PR								
	QR								
	RR								
	SR								
	TR								
	UR								

OTHER (Including in this order Author, Title, Periodical Name, Date, Pertinent Pages, etc.)

VR	KIM et al., "Experiences with Tripwire: Using Integrity Checkers For Intrusion Detection", COAST Labs. Dept. of Computer Sciences Purdue University, February 22, 1995, pages 1-12			
WR	KIM et al., "The Design and Implementation of Tripwire: A file System Integrity Checker", COAST Labs. Dept. of Computer Sciences Purdue University, November 19, 1993, pages 1-21			

Examiner Jean R Homere

Date Considered: 5-01-03

*EXAMINER: Initial if citation considered, whether or not citation is in conformance with MPEP § 609. Draw line through citation if not in conformance and not considered. Include copy of this form with next communication to Applicant.

Please type a plus sign (+) inside this box → +



PTO/SB/08A (08-00)

Approved for use through 10/31/2002. OMB 0651-0031
U.S. Patent and Trademark Office: U.S. DEPARTMENT OF COMMERCE

Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it contains a valid OMB control number.

Substitute for form 1449A/PTO

INFORMATION DISCLOSURE STATEMENT BY APPLICANT

(use as many sheets as necessary)

Sheet 1 of 10

Complete if Known

Application Number	09/987,723
Filing Date	November 15, 2001
First Named Inventor	David A. FARBER et al.
Group Art Unit	2177
Examiner Name	Unassigned
Attorney Docket Number	39685-0036

U.S. PATENT DOCUMENTS

Examiner Initials*	Cite No. ¹	U.S. Patent Document		Name of Patentee or Applicant of Cited Document	Date of Publication of Cited Document MM-DD-YYYY	Pages, Columns, Lines, Where Relevant Passages or Relevant Figures Appear
		Number	Kind Code ² (if known)			
	A01	4,888,681		BARNES	12/1989	
	A02	4,972,367		BURKE	11/1990	
	A03	5,050,212		DYSON	9/1991	
	A04	5,202,982		GRAMLICH	4/1993	
	A05	5,208,858		VOLLERT	5/1993	
	A06	5,301,316		HAMILTON	5/1994	
	A07	5,343,527		MOORE	8/1994	
	A08	4,571,700		EMRY, JR.	2/1996	
	A09	4,675,810		GRUNER	6/1987	
	A10	5,050,074		MARCA	9/1991	
	A11	5,276,901		HOWELL	1/1994	

RECEIVED

JUL 30 2003

Technology Center 2100

FOREIGN PATENT DOCUMENTS

Examiner Initials*	Cite No. ¹	Foreign Patent Document			Name of Patentee or Applicant of Cited Document	Date of Publication of Cited Document MM-DD-YYYY	Pages, Columns, Lines, Where Relevant Passages or Relevant Figures Appear	T ₆
		Office ³	Number ⁴	Kind Code ⁵ (if known)				
	A12	JP	05102529		HOROSHI	6/1993		

Examiner
Signature

Date
Considered

*EXAMINER: Initial if reference considered, whether or not citation is in conformance with MPEP 609. Draw line through citation if not in conformance and not considered. Include copy of this form with next communication to applicant.

¹ Unique citation designation number. ² See attached Kinds of U.S. Patent Documents. ³ Enter Office that issued the document, by the two-letter code (WIPO Standard ST.3). ⁴ For Japanese patent documents, the indication of the year of the reign of the Emperor must precede the serial number of the patent document. ⁵ Kind of document by the appropriate symbols as indicated on the document under WIPO Standard ST. 16 if possible. ⁶ Applicant is to place a check mark here if English language Translation is attached.

Burden Hour Statement: This form is estimated to take 2.0 hours to complete. Time will vary depending upon the needs of the individual case. Any comments on the amount of time you are required to complete this form should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, Washington, DC 20231. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. SEND TO: Assistant Commissioner for Patents, Washington, DC

A circular stamp from the Office of Intellectual Property (OIP). The text "OIP" is at the top, "JUL 29 2003" is in the center, and "OFFICE OF INTELLECTUAL PROPERTY" is around the bottom edge. Below the stamp, the text "no payments are required" is visible.

Approved for use through 10/31/2002. OMB 0651-0031

Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it contains a valid OMB control number.

Complete if Known

Application Number	09/987,723
Filing Date	November 15, 2001
First Named Inventor	David A. FARBER et al.
Group Art Unit	2177
Examiner Name	Unassigned
Attorney Docket Number	39685-0036

Sheet	2	of	10
-------	---	----	----

U.S. PATENT DOCUMENTS						
Examiner Initials *	Cite No. ¹	U.S. Patent Document		Name of Patentee or Applicant of Cited Document	Date of Publication of Cited Document MM-DD-YYYY	Pages, Columns, Lines, Where Relevant Passages or Relevant Figures Appear
		Number	Kind Code ² (if known)			
	A13	5,384,565		CANNON	1/1995	<div>RECEIVED</div> <div>JUL 30 2003</div> <div>Technology Center 2100</div>
	A14	5,357,623		MEGORY-COHEN	10/1994	
	A15	3,668,647		EVANGELISTI	6/1992	
	A16	4,215,402		MITCHELL	7/1980	
	A17	4,290,105		CICHELLI	9/1981	
	A18	4,376,299		RIVEST	3/1983	
	A19	4,405,829		RIVEST	9/1983	
	A20	4,412,285		NECHES	10/1983	
	A21	4,414,624		SUMMER, JR.	11/1983	
	A22	4,441,155		FLETCHER	4/1984	
	A23	4,464,713		BENHASE	8/1984	

[illegible]Date
Considered

¹ Unique citation designation number. ² See attached Kinds of U.S. Patent Documents. ³ Enter Office that issued the document, by the two-letter code (WIPO Standard ST.3). ⁴ For Japanese patent documents, the indication of the year of the reign of the Emperor must precede the serial number of the patent document. ⁵ Kind of document by the appropriate symbols as indicated on the document under WIPO Standard ST. 16 if possible. ⁶ Applicant is to place a check mark here if English language Translation is attached.

Burden Hour Statement: This form is estimated to take 2.0 hours to complete. Time will vary depending upon the needs of the individual case. Any comments on the amount of time you are required to complete this form should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, Washington, DC 20231. **DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. SEND TO:** Assistant Commissioner for Patents, Washington, DC

+

+



Substitute for form 1449A/PTO

Complete if Known

(use as many sheets as necessary)

Sheet	3	of	10
-------	---	----	----

Application Number	09/987,723
--------------------	------------

Filing Date	November 15, 2001
-------------	-------------------

First Named Inventor	David A. FARBER et al.
----------------------	------------------------

Group Art Unit	2177
----------------	------

Examiner Name	Unassigned
---------------	------------

Attorney Docket Number	39685-0036
------------------------	------------

U.S. PATENT DOCUMENTS					
Examiner Initials *	Cite No. ¹	U.S. Patent Document	Name of Patentee or Applicant of Cited Document	Date of Publication of Cited Document MM-DD-YYYY	Pages, Columns, Lines, Where Relevant Passages or Relevant Figures Appear
		Number Kind Code ² (if known)			
	A24	4,577,233	MATICK	3/1986	
	A25	4,642,793	MEADEN	2/1987	
	A26	4,691,299	RIVEST	9/1987	
	A27	4,725,945	KRONSTADT	2/1988	
	A28	4,773,039	ZAMORA	9/1988	
	A29	4,887,235	HOLLOWAY	12/1989	
	A30	4,490,782	DIXON	12/1984	
	A31	4,922,414	HOLLOWAY	5/1990	
	A32	5,057,837	COLWELL	1/1991	
	A33	5,007,658	BENDERT	12/1991	
	A34	5,025,421	CHO	6/1991	

[illegible]Date
Considered

¹ Unique citation designation number. ² See attached Kinds of U.S. Patent Documents. ³ Enter Office that issued the document, by the two-letter code (WIPO Standard ST.3). ⁴ For Japanese patent documents, the indication of the year of the reign of the Emperor must precede the serial number of the patent document. ⁵ Kind of document by the appropriate symbols as indicated on the document under WIPO Standard ST. 16 if possible. ⁶ Applicant is to place a check mark here if English language Translation is attached.

+

+

Approved for use through 10/31/2002. OMB 0651-0031

Patent and Trademark Office: U.S. DEPARTMENT OF COMMERCE

Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it contains a valid OMB control number.

Complete if Known

Application Number	09/987,723
Filing Date	November 15, 2001
First Named Inventor	David A. FARBER et al.
Group Art Unit	2177
Examiner Name	Unassigned
Attorney Docket Number	39685-0036

Sheet	4	of	10
-------	---	----	----

U.S. PATENT DOCUMENTS						
Examiner Initials *	Cite No. ¹	U.S. Patent Document		Name of Patentee or Applicant of Cited Document	Date of Publication of Cited Document MM-DD-YYYY	Pages, Columns, Lines, Where Relevant Passages or Relevant Figures Appear
		Number	Kind Code ² (if known)			
	A35	5,129,081		KOBAYASHI	7/1992	
	A36	5,129,082		TIRFING	7/1992	
	A37	5,144,667		POGUE, JR.	9/1992	
	A38	5,179,680		COLWELL	1/1993	
	A39	5,301,286		RAJANI	4/1994	
	A40	5,404,508		KONRAD	4/1995	
	A41	5,638,443		STEFIK et al.	6/1997	
	A42	5,802,291		BALICK et al.	9/1998	
	A43	5,542,087		NEIMAT et al.	7/1996	
	A44	5,287,499		NEMES	2/1994	
	A45	4,922,417		CHURM et al.	5/1990	

[illegible]Date
Considered

Burden Hour Statement: This form is estimated to take 2.0 hours to complete. Time will vary depending upon the needs of the individual case. Any comments on the amount of time you are required to complete this form should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, Washington, DC 20231. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. SEND TO: Assistant Commissioner for Patents, Washington, DC

Please type a plus sign (+) inside this box →



PTO/SB/08A (08-00)

Approved for use through 10/31/2002. OMB 0651-0031

U.S. Patent and Trademark Office: U.S. DEPARTMENT OF COMMERCE

Under the Paperwork Reduction Act of 1995, no person shall be required to respond to a collection of information unless it contains a valid OMB control number

Substitute for form 1449A/PTO

INFORMATION DISCLOSURE STATEMENT BY APPLICANT

(use as many sheets as necessary)

Sheet 6 of 10

Complete if Known

Application Number	09/987,723
Filing Date	November 15, 2001
First Named Inventor	David A. FARBER et al.
Group Art Unit	2177
Examiner Name	Unassigned
Attorney Docket Number	39685-0036

RECEIVED

JUL 30 2003

Technology Center 2100

OTHER PRIOR ART -- NON PATENT LITERATURE DOCUMENTS

Examiner Initials *	Cite No.	Include name of the author (in CAPITAL LETTERS), title of the article (when appropriate), title of the item (book, magazine, journal, serial, symposium, catalog, etc.), date, page(s), volume-issue number(s), publisher, city and/or country where published.	T ²
	A54	BERT DEM BOER et al., Collisions for the compression function of MD5, pp. 292-304	
	A55	SAKTI PRAMANIK et al., Multi-Directory Hashing, 1993, Info. Sys., Vol. 18, No. 1, pp. 63-74	
	A56	MURLIDHAR KOUSHIK, Dynamic Hashing with Distributed Overflow Space: A File Organization with Good Insertion Performance, 1993, Info. Sys., Vol. 18, No. 5, pp. 299-317	
	A57	WITOLD LITWIN et al., LH*-Linear Hashing for Distributed Files, HP Labs Tech. Report No. HPL-93-21, June 1993, pp. 1-22	
	A58	YULIANG ZHENG et al., HAVAL-A One-Way Hashing Algorithm with Variable Length of Output (Extended Abstract), 1993, pp. 83-105	
	A59	CHRIS CHARNES and JOSEF PIEPRZKY, Linear Nonequivalence versus Nonlinearity, Pieprzky, 1993, pp. 156-164	
	A60	WITOLD LITWIN et al., Linear Hashing for Distributed Files, ACM SIGMOD, May 1993, pp. 327-336	
	A61	MING-LING LO et al., On Optimal Processor Allocation to Support Pipelined Hash Joins, ACM SIGMOD, pp. 69-78, May 1993	
	A62	THOMAS A. BERSON, Differential Cryptanalysis Mod 2 ³² with Applications to MD5, pp. 69-81	
	A63	WILLIAM PERRIZO et al., Distributed Join Processing Performance Evaluation, Twenty-Seventh Hawaii International Conference on System Sciences, Vol. II, pp. 236-244	

Examiner
Signature

Date
Considered

*EXAMINER: Initial if reference considered, whether or not citation is in conformance with MPEP 609. Draw line through citation if not in conformance and not considered. Include copy of this form with next communication to applicant.

¹ Unique citation designation number. ² Applicant is to place a check mark here if English language Translation is attached.

Burden Hour Statement: This form is estimated to take 2.0 hours to complete. Time will vary depending upon the needs of the individual case. Any comments on the amount of time you are required to complete this form should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, Washington, DC 20231. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. SEND TO: Assistant Commissioner for Patents, Washington, DC 20231.

Please type a plus sign (+) inside this box



PTO/SB/08A (08-00)

Approved for use through 10/31/2002. OMB 0651-0031

Under the Paperwork Reduction Act of 1995, no person is required to respond to a collection of information unless it contains a valid OMB control number

Substitute for form 1449A/PTO

INFORMATION DISCLOSURE STATEMENT BY APPLICANT

(use as many sheets as necessary)

Sheet 7 of 10

Complete if Known

Application Number	09/987,723
Filing Date	November 15, 2001
First Named Inventor	David A. FARBER et al.
Group Art Unit	2177
Examiner Name	Unassigned
Attorney Docket Number	39685-0036

RECEIVED

JUL 30 2003

OTHER PRIOR ART -- NON PATENT LITERATURE DOCUMENTS

Technology Center 2100

Examiner Initials *	Cite No.	Include name of the author (in CAPITAL LETTERS), title of the article (when appropriate), title of the item (book, magazine, journal, serial, symposium, catalog, etc.), date, page(s), volume-issue number(s), publisher, city and/or country where published.	T ²
	A64	VIJAY KUMAR, A Concurrency Control Mechanism Based on Extendible Hashing for Main Memory Database Systems, ACM, Vol. 3, 1989, pp. 109-113	
	A65	BIRGIT PFITZMAN, Sorting Out Signature Schemes, November 1993, 1 st Conf. Computer & Comm. Security '93, p. 74-85	
	A66	ZHIYU TIAN et al., A New Hashing Function: Statistical Behaviour and Algorithm, Spring 1993, pp. 3-13	
	A67	G.L. FRIEDMAN, Digital Camera with Apparatus for Authentication of Images Produced from an Image File, NASA CASE NO. NPO-19108-1-CU, Serial No. 08/159,980, November 24, 1993	
	A68	H. GOODMAN, ADA, Object-Oriented Techniques, and Concurrency in Teaching Data Structures and File Management REPORT DOCUMENTATION PAGE AD-A275 385 - 94-04277	
	A69	Advances in Cryptology-EUROCRYPT '93, Workshop on the Theory and Application of Cryptographic Techniques Lofthus, Norway, May 23-27, 1993, Proceedings	
	A70	Proceedings on the 1993 ACM SIGMOD International Conference on Management of Data, Vol. 22, Issue 2, June 1993	
	A71	Advances of Cryptology-AUSCRYPT '92 - Workshop on the Theory and Application of Cryptographic Techniques Gold Coast, Queensland, Australia, December 13-16, 1992 Proceedings	
	A72	GWERTZMAN, JAMES et al., "The Case for Geographical Push-Caching." Technical Report HU TR 34-94 (excerpt), Harvard University, DAS, Cambridge, MA 02138, 1994, 2 pgs.	
	A73	GRIGNI, MICHELANGELO, et al., "Tight Bounds on Minimum Broadcasts Networks", SIAM Journal of Discrete Mathematics, Vol. 4, No. 2, May 1991, pgs. 207-222	

Examiner Signature		Date Considered	
--------------------	--	-----------------	--

*EXAMINER: Initial if reference considered, whether or not citation is in conformance with MPEP 609. Draw line through citation if not in conformance and not considered. Include copy of this form with next communication to applicant.

¹ Unique citation designation number. ² Applicant is to place a check mark here if English language Translation is attached.

Burden Hour Statement: This form is estimated to take 2.0 hours to complete. Time will vary depending upon the needs of the individual case. Any comments on the amount of time you are required to complete this form should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, Washington, DC 20231. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. SEND TO: Assistant Commissioner for Patents, Washington, DC 20231.

Please type a plus sign (+) inside this box



PTO/SB/08A (08-00)

Approved for use through 10/31/2002. OMB 0651-0031

U.S. Patent and Trademark Office: U.S. DEPARTMENT OF COMMERCE

Under the Paperwork Reduction Act of 1995, no person is required to respond to a collection of information unless it contains a valid OMB control number

Substitute for form 1449A/PTO

INFORMATION DISCLOSURE STATEMENT BY APPLICANT

(use as many sheets as necessary)

Sheet 8 of 10

Complete if Known

Application Number	09/987,723
Filing Date	November 15, 2001
First Named Inventor	David A. FARBER et al.
Group Art Unit	2177
Examiner Name	Unassigned
Attorney Docket Number	39685-0036

RECEIVED

JUL 30 2003

OTHER PRIOR ART -- NON PATENT LITERATURE DOCUMENTS

Technology Center 2100

Examiner Initials *	Cite No. ¹	Include name of the author (in CAPITAL LETTERS), title of the article (when appropriate), title of the item (book, magazine, journal, serial, symposium, catalog, etc.), date, page(s), volume-issue number(s), publisher, city and/or country where published.	T ²
	A74	DEVINE, ROBERT, "Design and Implementation of DDH: A Distributed Dynamic Hashing Algorithm", In Proceedings of 4th International Conference on Foundations of Data Organizations and Algorithms, 1993, pp. 101-114	
	A75	DEERING, STEPHEN et al., "Multicast Routing in Datagram Internet Works and Extended LANs," ACM Transactions on Computer Systems, Vol. 8, No. 2, May 1990, pp. 85-110	
	A76	CORMEN, THOMAS H., et al., "Introduction to Algorithms, The MIT Press, Cambridge, Massachusetts, 1994, pp. 219-243, 991-993	
	A77	NAOR, MONI et al., "The Load, Capacity and Availability of Quorum Systems", In Proceedings of the 35th IEEE Symposium on Foundations of Computer Science, November 1994, pp. 214-225	
	A78	NISAN, NOAM, "Pseudorandom Generators for Space-Bounded Computation", In Proceedings of the Twenty-Second Annual ACM Symposium on Theory of Computing, May 1990, pp. 204-212	
	A79	PALMER, MARK et al., "Fido: A Cache that Learns to Fetch", In Proceedings of the 17th International Conference on Very Large Data Bases, September 1991, pp. 255-264	
	A80	PELEG, DAVID et al., "The Availability of Quorum Systems", Information and Computation 123, 1995, 210-223	
	A81	RABIN, MICHAEL, "Efficient Dispersal of Information for Security, Load Balancing, and Fault Tolerance", Journal of the ACM, Vol. 36, No. 2, April 1989, pp. 335-348	
	A82	RAVI, R., "Rapid Rumor Ramification: Approximating the Minimum Broadcast Time", In Proceedings of the 35th IEEE Symposium on Foundation of Computer Science, November 1994, pp. 202-213	
	A83	SCHMIDT, JEANETTE, et al., "Chernoff-Hoeffding Bounds for Applications with Limited Independence", In Proceedings of the 4th ACS-SIAM Symposium on Discrete Algorithms, 1993, p. 331-340	

Examiner
Signature

Date
Considered

*EXAMINER: Initial if reference considered, whether or not citation is in conformance with MPEP 609. Draw line through citation if not in conformance and not considered. Include copy of this form with next communication to applicant.

¹ Unique citation designation number. ² Applicant is to place a check mark here if English language Translation is attached.

Burden Hour Statement: This form is estimated to take 2.0 hours to complete. Time will vary depending upon the needs of the individual case. Any comments on the amount of time you are required to complete this form should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, Washington, DC 20231. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. SEND TO: Assistant Commissioner for Patents, Washington, DC 20231.

Please type a plus sign (+) inside this box



JUL 29 2003

PTO/SB/08A (08-00)

Approved for use through 10/31/2002. OMB 0651-0031

U.S. Patent and Trademark Office: U.S. DEPARTMENT OF COMMERCE

Under the Paperwork Reduction Act of 1995, no person is required to respond to a collection of information unless it contains a valid OMB control number

Substitute for form 1449A/PTO

INFORMATION DISCLOSURE STATEMENT BY APPLICANT

(use as many sheets as necessary)

Sheet 9 of 10

Complete if Known

Application Number	09/987,723
Filing Date	November 15, 2001
First Named Inventor	David A. FARBER et al.
Group Art Unit	2177
Examiner Name	Unassigned
Attorney Docket Number	39685-0036

RECEIVED

JUL 30 2003

OTHER PRIOR ART -- NON PATENT LITERATURE DOCUMENTS

Technology Center 2100

Examiner Initials *	Cite No. ¹	Include name of the author (in CAPITAL LETTERS), title of the article (when appropriate), title of the item (book, magazine, journal, serial, symposium, catalog, etc.), date, page(s), volume-issue number(s), publisher, city and/or country where published.	T ²
	A84	TARJAN, ROBERT ENDRE, et al., "Storing a Sparse Table", Communications of the ACM, Vol. 22, No. 11, November 1979, pp. 606-611	
	A85	WEGMAN, MARK et al., "New Hash Functions and Their Use in Authentication and Set Equality", Journal of Computer and System Sciences, Vol. 22, June 1981, pp. 265-279	
	A86	VITTER, JEFFREY SCOTT et al., "Optimal Prefetching via Data Compression", In Proceedings of 32nd IEEE Symposium on Foundations of Computer Science, November 1991, p. 124-130	
	A87	FRIEDMAN, MICHAEL et al., "Storing a Sparse Table with O(1) Worst Case Access Time", Journal of the Association for Computing Machinery, Vol. 31, No. 3, July 1984, pp. 538-544	
	A88	YAO, ANDREW CHI-CHIH, "Should Tables be Sorted?", Journal of the Association for Computing Machinery, Vol. 28, No. 3, July 1981, pp. 615-628	
	A89	FLOYD, SALLY et al., "A Reliable Multicast Framework for Light-Weight Sessions and Application Level Framing", In Proceeding of ACM SIGCOMM '95, pp. 342-356	
	A90	FEELEY, MICHAEL, et al., "Implementing Global Memory Management in a Workstation Cluster", In Proceedings of the 15th ACM Symposium on Operating Systems Principles, 1995, pp. 201-212	
	A91	CARTER, J. LAWRENCE, et al., "Universal Classes of Hash Functions", Journal of Computer and System Sciences, Vol. 18, No. 2, April 1979, pp. 143-154	
	A92	PATENT ABSTRACTS OF JAPAN, "Electronic Mail Multiplexing System and Communication Control Method in The System", 06/30/1993, JP 05162529	
	A93	KIM et al., "Experiences with Tripwire: Using Integrity Checkers For Intrusion Detection", COAST Labs. Dept. of Computer Sciences Purdue University, February 22, 1995, pp. 1-12	

Examiner
Signature

Date
Considered

*EXAMINER: Initial if reference considered, whether or not citation is in conformance with MPEP 609. Draw line through citation if not in conformance and not considered. Include copy of this form with next communication to applicant.

¹ Unique citation designation number. ² Applicant is to place a check mark here if English language Translation is attached.

Burden Hour Statement: This form is estimated to take 2.0 hours to complete. Time will vary depending upon the needs of the individual case. Any comments on the amount of time you are required to complete this form should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, Washington, DC 20231. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. SEND TO: Assistant Commissioner for Patents, Washington, DC 20231.

+

JUL 29 2003

+

U.S. Patent and Trademark Office: U.S. DEPARTMENT OF COMMERCE

Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it contains a valid OMB control number

Complete if Known

Application Number	09/987,723
Filing Date	November 15, 2001
First Named Inventor	David A. FARBER et al.
Group Art Unit	2177
Examiner Name	Unassigned
Attorney Docket Number	39685-0036

Sheet	10	Of	10
-------	----	----	----

Examiner Initials *	Cite No. ¹	Include name of the author (in CAPITAL LETTERS), title of the article (when appropriate), title of the item (book, magazine, journal, serial, symposium, catalog, etc.), date, page(s), volume-issue number(s), publisher, city and/or country where published.	T ²
	A94	KIM et al., "The Design and Implementation of Tripwire: A File System Integrity Checker", COAST Labs. Dept. of Computer Sciences Purdue University, November 19, 1993, pp. 1-21	

RECEIVED

JUL 30 2003

Technology Center 2100

RECEIVED

JUL 30 2003

Technology Center 2100

Examiner
Signature

Date Considered

*EXAMINER: Initial if reference considered, whether or not citation is in conformance with MPEP 609. Draw line through citation if not in conformance and not considered. Include copy of this form with next communication to applicant.

¹ Unique citation designation number. ² Applicant is to place a check mark here if English language Translation is attached.

Burden Hour Statement: This form is estimated to take 2.0 hours to complete. Time will vary depending upon the needs of the individual case. Any comments on the amount of time you are required to complete this form should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, Washington, DC 20231. **DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. SEND TO: Assistant Commissioner for Patents, Washington, DC 20231.**



Please type a plus sign (+) inside this box → ☐

PTO/SB/08A (08-00)

Approved for use through 10/31/2002. OMB 0651-0031
U.S. Patent and Trademark Office: U.S. DEPARTMENT OF COMMERCE

Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it contains a valid OMB control number

Substitute for form 1449A/PTO

INFORMATION DISCLOSURE STATEMENT BY APPLICANT

(use as many sheets as necessary)

Sheet 1 of 1

Complete if Known

Application Number	09/987,723
Filing Date	November 15, 2001
First Named Inventor	David A. FARBER et al.
Group Art Unit	2177
Examiner Name	Unassigned
Attorney Docket Number	39685-0036

OTHER PRIOR ART -- NON PATENT LITERATURE DOCUMENTS

Examiner Initials *	Cite No. ¹	Include name of the author (in CAPITAL LETTERS), title of the article (when appropriate), title of the item (book, magazine, journal, serial, symposium, catalog, etc.), date, page(s), volume-issue number(s), publisher, city and/or country where published.	T ²
JRH	B01	PETER DEUTSCH (peterd@bunyip.com), "Re: MD5 and LIFNs (was: Misc Comments)", www.acl.lanl.gov/URI/archive/uri-94q2.messages/0106.html, April 26, 1994	
	B02	ALEXANDER DUPUY (dupuy@smarts.com), "RE: MD5 and LIFNs (was: Misc Comments)", www.acl.lanl.gov/URI/archive/uri-94q2.messages/0113.html, April 26, 1994	
	B03	ALEXANDER DUPUY (dupuy@smarts.com), "MD5 and LIFNs (was: Misc Comments)", www.acl.lanl.gov/URI/archive/uri-94q2.messages/0081.html, April 17, 1994	
	B04	ALBERT LANGER (cmf851@anu.oz.au), http://groups.google.com/groups?selm=1991Aug7.225159.786%40newshost.anu.edu.au&oe=UTF-8&output=gplain, August 7, 1991	
	B05	CLIFFORD LYNCH (Calur@uccmvsa.bitnet), "ietf uri/uri overview draft paper (long)", www.acl.lanl.gov/URI/archive/uri-93q1.messages/0015.html, March 25, 1993	
	B06	K. SOLLINS and L. MASINTER, "Functional Requirements for Uniform Resource Names", www.w3.org/Addressing/rfc1737.txt, December 1994, pp. 1-7	
	B07	W3C:ID, HTTP: A protocol for networked information, "Basic HTTP as defined in 1992", www.w3.org/Protocols/HTTP2.html, 1992	

RECEIVED

SEP 04 2003

Technology Center 2100

Examiner Signature	Jean R. Homere	Date Considered	11-20-03
--------------------	----------------	-----------------	----------

*EXAMINER: Initial if reference considered, whether or not citation is in conformance with MPEP 609. Draw line through citation if not in conformance and not considered. Include copy of this form with next communication to applicant.

¹ Unique citation designation number. ² Applicant is to place a check mark here if English language Translation is attached.

Burden Hour Statement: This form is estimated to take 2.0 hours to complete. Time will vary depending upon the needs of the individual case. Any comments on the amount of time you are required to complete this form should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, Washington, DC 20231. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. SEND TO: Assistant Commissioner for Patents, Washington, DC 20231.

WEST Search History

DATE: Monday, November 24, 2003

Set Name Query
side by side

Hit Count Set Name
result set

*DB=USPT,PGPB,JPAB,EPAB,DWPI,TDBD; THES=ASSIGNEE;
PLUR=YES; OP=ADJ*

L8	l5 not l1	39	L8
L7	5978791.uref.	22	L7
L6	L1 not l2	31	L6
L5	L4 same (request\$ near3 file\$) same (retriev\$ or locat\$ or fetch\$)	40	L5
L4	L3 same (network or (server same client))	593	L4
L3	file near3 (name or id)same (md5 or function)	4057	L3
L2	L1 same (network or (server same client))	10	L2
L1	file near3 (name or id)same md5	41	L1

END OF SEARCH HISTORY

WEST☐ **Generate Collection** **Print**

L8: Entry 29 of 39

File: USPT

Nov 10, 1998

US-PAT-NO: 5835087

DOCUMENT-IDENTIFIER: US 5835087 A

TITLE: System for generation of object profiles for a system for customized electronic identification of desirable objects

DATE-ISSUED: November 10, 1998

INVENTOR-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY
Herz; Frederick S. M.	Davis	WV	26260	
Eisner; Jason M.	Philadelphia	PA	19107	
Ungar; Lyle H.	Philadelphia	PA	19103	

APPL-NO: 08/ 551201 [PALM]

DATE FILED: October 31, 1995

PARENT-CASE:

CROSS-REFERENCE TO RELATED APPLICATIONS This patent application is a continuation-in-part of U.S. patent application Ser. No. 08/346,425, filed Nov. 29, 1994 and titled "SYSTEM AND METHOD FOR SCHEDULING BROADCAST OF AND ACCESS TO VIDEO PROGRAMS AND OTHER DATA USING CUSTOMER PROFILES", now U.S. Pat. No. 5,758,257, which application is assigned to the same assignee as the present application.

INT-CL: [06] H04 N 7/14

US-CL-ISSUED: 345/327; 348/1, 348/7, 348/10, 348/12, 348/13, 455/2, 455/4.2, 455/5.1

US-CL-CURRENT: 345/810; 725/14, 725/35, 725/46

FIELD-OF-SEARCH: 348/1, 348/2, 348/6, 348/7, 348/10, 348/12, 348/13, 348/906, 455/2, 455/3.1, 455/4.1, 455/4.2, 455/5.1, 455/6.1, 455/6.2, 455/6.3

PRIOR-ART-DISCLOSED:

U.S. PATENT DOCUMENTS

Search Selected**Search ALL**

	PAT-NO	ISSUE-DATE	PATENTEE-NAME	US-CL
<input type="checkbox"/>	<u>4529870</u>	July 1985	Chaum	235/380
<input type="checkbox"/>	<u>4706080</u>	November 1987	Sincoskie	340/825
<input type="checkbox"/>	<u>4759063</u>	July 1988	Chaum	380/30
<input type="checkbox"/>	<u>4914698</u>	April 1990	Chaum	380/30
<input type="checkbox"/>	<u>4926480</u>	May 1990	Chaum	380/23
<input type="checkbox"/>	<u>4947430</u>	August 1990	Chaum	380/25
<input type="checkbox"/>	<u>4987593</u>	January 1991	Chaum	380/3
<input type="checkbox"/>	<u>5131039</u>	July 1992	Chaum	380/23
<input type="checkbox"/>	<u>5136501</u>	August 1992	Silverman et al.	364/408
<input type="checkbox"/>	<u>5245656</u>	September 1993	Loeb et al.	380/23
<input type="checkbox"/>	<u>5251324</u>	October 1993	McMullan, Jr.	455/2
<input type="checkbox"/>	<u>5276736</u>	January 1994	Chaum	380/24
<input type="checkbox"/>	<u>5301109</u>	April 1994	Landauer et al.	364/419
<input type="checkbox"/>	<u>5321833</u>	June 1994	Chang et al.	395/600
<input type="checkbox"/>	<u>5331554</u>	July 1994	Graham	364/419.07
<input type="checkbox"/>	<u>5331556</u>	July 1994	Black, Jr. et al.	364/419.01
<input type="checkbox"/>	<u>5373558</u>	December 1994	Chaum	380/23
<input type="checkbox"/>	<u>5410344</u>	April 1995	Graves et al.	348/1
<input type="checkbox"/>	<u>5469206</u>	November 1995	Strubbe et al.	348/7
<input type="checkbox"/>	<u>5483278</u>	January 1996	Strubbe et al.	348/7
<input type="checkbox"/>	<u>5534911</u>	July 1996	Levitan	348/1
<input type="checkbox"/>	<u>5541638</u>	July 1996	Story	348/7
<input type="checkbox"/>	<u>5600364</u>	February 1997	Hendricks et al.	348/1

OTHER PUBLICATIONS

Tak W. Yan & Hector Garcia-Molina, SIFT -A Tool for Wide-Area Information Dissemination, 1995 USENIX Technical Conference, New Orleans, LA., Jan. 16-20, pp. 177-186.

Masahiro Morita & Yoichi Shinoda, Information Filtering Based on User Behavior Analysis and Best Match Text Retrieval, Proceedings of the Seventeenth Annual International ACM-SIGIR Conference on Research and Development in Information Retrieval, Dublin, Jul. 3-6, 1994, Pages Title Page (272)-281.

Jim Binkley & Leslie Young, Rama: An Architecture for Internet Information Filtering, Journal of Intelligent Information Systems: Integrating Artificial Intelligence and Database Technologies, vol. 5, No. 2, Sep. 1995, pp. 81-99.

Sincoskie, W.D. & Cotton C.J. "Extended Bridge Algorithms for Large Networks", IEEE Network, Jan. 1988 -vol. 2, No. 1, pp. 16-24.

Deering, S.; Estrin, D.; Farinacci, D.; Jacobson, V.; Liu, C.; Wei, L; "An Architecture for Wide-Area Multicast Routing", Computer Communication Review, vol. 24, No. 4, Oct. 1994, Proceedings of SIGCOMM 94, pp. 126-135.

Bolot, J.; Turetti, T. & Wakeman, I.; "Scalable Feedback Control for Multicast Video Distribution In the Internet", Computer Communication Review, vol. 24, No. 4 Oct. 1994, Proceedings of SIGCOMM 94, pp. 58-67.

Rivest, R.L.; Shamir, A. & Adleman, L.; "A Method for Obtaining Digital Signatures and Public-Key Cryptosystems", Communications of the ACM, Feb. 1978, vol. 21, No. 2, pp. 120-126.

Chaum, D. "Security without Identification: Card Computers to Make Big Brother Obsolete", Communications of the ACM, 28(10), Oct. 1985, pp. 1030-1044.

Chaum, D. "Untraceable Electronic Mail, Return Addresses, and Digital Pseudonyms", Communications of the ACM, vol. 24, No. 2, Feb., 1981.

Sheth, B. & Maes, P. "Evolving Agents For Personalized Information Filtering", Proc.

9th IEEE Conference, 1993 pp. 345-352.

Cutting, D.R.; Karger, D.R.; Pedersen, J.O. & Tukey, J.W. "Scatter/Gather: A Cluster-based Approach to Browsing Large Document Collections", 15 Ann. Int'l SIGIR '92, ACM, 1992, pp. 318-329.

Rose, D.E.; Mander, R.; Oren, T., Ponceleon, D.B.; Salomon, G. & Wong, Y.Y. "Content Awareness in a File System Interface Implementing the `Pile` Metaphor for Organizing Information", 16 Ann. Int'l SIGIR '93, ACM, pp. 260-269.

Damashek, M., Gauging Similarity via N-Grams: Language-Independent Sorting, Categorization, and Retrieval of Text, pp. 1-11, Jan. 24, 1995.

Salton, G., "Developments in Automatic Text Retrieval", Science, vol. 253, pp. 974-980, Aug. 30, 1991.

Chaum, D., "Achieving Electronic Privacy", Scientific American, Aug. 1992, pp. 96-101.

Loeb, S., "Architecting Personalized Delivery of Multimedia Information", Communications of the ACM, Dec. 1992, vol. 35, No. 12, pp. 39-50.

Foltz, P.W., Dumais, S.T., "Personalized Information Delivery: An Analysis Of Information Filtering Methods", Communications of the ACM, Dec. 1992, vol. 35, No. 12, pp. 51-60.

Belkin, N.J., Croft, W.B., "Information Filtering And Information Retrieval: Two Sides of the Same Coin?", Communications of the ACM, Dec. 1992, vol. 35, No. 12, pp. 29-38.

Chalmers, M., Chitson, P., "Bead: Explorations In Information Visualization", 15th Ann. Int'l SIGIR 92/Denmark -Jun. 1992, pp. 330-337.

Willett, P., "Recent Trends in Hierarchic Document Clustering: A Critical Review", Information Processing & Management, vol. 24, No. 5, pp. 557-597, 1988.

ART-UNIT: 271

PRIMARY-EXAMINER: Faile; Andrew I.

ASSISTANT-EXAMINER: Miller; John W.

ATTY-AGENT-FIRM: Duft, Graziano & Forest, P.C.

ABSTRACT:

This invention relates to customized electronic identification of desirable objects, such as news articles, in an electronic media environment, and in particular to a system that automatically constructs both a "target profile" for each target object in the electronic media based, for example, on the frequency with which each word appears in an article relative to its overall frequency of use in all articles, as well as a "target profile interest summary" for each user, which target profile interest summary describes the user's interest level in various types of target objects. The system then evaluates the target profiles against the users' target profile interest summaries to generate a user-customized rank ordered listing of target objects most likely to be of interest to each user so that the user can select from among these potentially relevant target objects, which were automatically selected by this system from the plethora of target objects that are profiled on the electronic media. Users' target profile interest summaries can be used to efficiently organize the distribution of information in a large scale system consisting of many users interconnected by means of a communication network. Additionally, a cryptographically-based pseudonym proxy server is provided to ensure the privacy of a user's target profile interest summary, by giving the user control over the ability of third parties to access this summary and to identify or contact the user.

24 Claims, 17 Drawing figures

WEST☐ **Generate Collection** **Print**

L8: Entry 33 of 39

File: USPT

Nov 19, 1996

US-PAT-NO: 5577210

DOCUMENT-IDENTIFIER: US 5577210 A

TITLE: Remote booting of an operating system by a network

DATE-ISSUED: November 19, 1996

INVENTOR-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY
Abdous; Arave	Massy			FR
Demortain; Stephane	Palaiseau			FR
Dalongville; Didier	Gometz Le Chatel			FR

ASSIGNEE-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY	TYPE CODE
Bull S.A.	Paris			FR	03

APPL-NO: 08/ 075578 [PALM]

DATE FILED: June 16, 1993

FOREIGN-APPL-PRIORITY-DATA:

COUNTRY	APPL-NO	APPL-DATE
FR	91 12837	October 17, 1991

PCT-DATA:

APPL-NO	DATE-FILED	PUB-NO	PUB-DATE	371-DATE	102 (E) -DATE
PCT/FR92/00977	October 15, 1992	WO93/08527	Apr 29, 1993	Jun 16, 1993	Jun 16, 1993

INT-CL: [06] G06 F 9/445

US-CL-ISSUED: 395/200.1; 395/200.2, 395/438, 395/894, 364/280.2, 364/975.2

US-CL-CURRENT: 709/219; 709/222, 710/74, 711/111

FIELD-OF-SEARCH: 395/200, 395/400, 395/200.09, 395/200.10, 395/200.11, 395/200.2, 395/404, 395/438, 395/439, 395/894, 364/280.23, 364/248.1, 364/280.2, 364/280.3, 364/280.6, 364/975.2

PRIOR-ART-DISCLOSED:

U.S. PATENT DOCUMENTS

Search Selected**Search ALL**

	PAT-NO	ISSUE-DATE	PATENTEE-NAME	US-CL
<input type="checkbox"/>	<u>4752870</u>	June 1988	Matsumura	364/DIG.1
<input type="checkbox"/>	<u>4958278</u>	September 1990	Meguro	364/200
<input type="checkbox"/>	<u>5109484</u>	April 1992	Hughes et al.	364/280.2
<input type="checkbox"/>	<u>5230052</u>	June 1993	Dayan et al.	395/700
<input type="checkbox"/>	<u>5280627</u>	January 1994	Flaherty et al.	395/700
<input type="checkbox"/>	<u>5305457</u>	April 1994	Takida et al.	395/700
<input type="checkbox"/>	<u>5367688</u>	November 1994	Croll	395/700
<input type="checkbox"/>	<u>5421009</u>	May 1995	Platt	395/200.1
<input type="checkbox"/>	<u>5444850</u>	August 1995	Chang	395/200.1

FOREIGN PATENT DOCUMENTS

FOREIGN-PAT-NO	PUBN-DATE	COUNTRY	US-CL
0106213	April 1984	EP	
0290924	November 1988	EP	
0358292	March 1990	EP	
2231180	November 1990	GB	

OTHER PUBLICATIONS

IBM Technical Disclosure Bulletin; "Shared Folders Between a Host and PC"; Jan. 1989; vol. 31, No. 8; pp. 428-429.

ART-UNIT: 232

PRIMARY-EXAMINER: An; Meng-ai T.

ASSISTANT-EXAMINER: Rinehart; Mark H.

ATTY-AGENT-FIRM: Kerkam, Stowell, Kondracki & Clarke, P.C. Kondracki; Edward J.

ABSTRACT:

A method for remote booting by a server of at least one terminal, including a volatile memory (RAM), at least one processor, a telecommunications card enabling connection to the server by a network, said server being provided with telecommunication device and memory of sufficient capacity to store the communications protocol, the operating system of the terminals recopied into an image file, its own operating system, and the applications programs, wherein the method includes the remote loading of a startup program by transforming the first interruption produced by a terminal following its being powered up into a request for reading an image file memorized in the server.

13 Claims, 3 Drawing figures

WEST

End of Result Set



Generate Collection

Print

L2: Entry 10 of 10

File: USPT

Jun 13, 2000

US-PAT-NO: 6076111

DOCUMENT-IDENTIFIER: US 6076111 A

TITLE: Methods and apparatuses for transferring data between data processing systems which transfer a representation of the data before transferring the data

DATE-ISSUED: June 13, 2000

INVENTOR-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY
Chiu; Chan	Sunnyvale	CA		
Morris; Steve	Palo Alto	CA		
Wang; Wu	Los Altos	CA		

ASSIGNEE-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY	TYPE CODE
Pictra, Inc.	Sunnyvale	CA			02

APPL-NO: 08/ 957219 [PALM]

DATE FILED: October 24, 1997

INT-CL: [07] G06 F 15/16

US-CL-ISSUED: 709/232; 709/203

US-CL-CURRENT: 709/232; 709/203

FIELD-OF-SEARCH: 709/232, 709/203

PRIOR-ART-DISCLOSED:

U.S. PATENT DOCUMENTS

Search Selected

Search ALL

	PAT-NO	ISSUE-DATE	PATENTEE-NAME	US-CL
<input type="checkbox"/>	<u>5442749</u>	August 1995	Northcutt et al.	709/219
<input type="checkbox"/>	<u>5771355</u>	June 1998	Kuzma	709/232
<input type="checkbox"/>	<u>5790789</u>	August 1998	Suarez	709/202
<input type="checkbox"/>	<u>5793966</u>	August 1998	Amstein	709/203
<input type="checkbox"/>	<u>5860068</u>	January 1999	Cook	705/26
<input type="checkbox"/>	<u>5926623</u>	July 1999	Tsukakoshi et al.	707/1

ART-UNIT: 278

PRIMARY-EXAMINER: Lim; Krisna

ATTY-AGENT-FIRM: Blakely, Sokoloff, Taylor & Zafman

ABSTRACT:

A method for transferring data between processing systems and systems for transferring data. A method in one example creates a first representation of a first digital media which is stored on a first digital processing system. The first representation is compared to a plurality of representations of a corresponding plurality of digital media. Typically, but not necessarily, the plurality of representations are stored on a second digital processing system and the first representation is transmitted to the second digital processing system. The first digital media is transmitted to the second digital processing system if the first representation does not match any of the plurality of representations.

28 Claims, 9 Drawing figures

Draft Only

WEST

☐

L2: Entry 9 of 10

File: USPT

Jul 4, 2000

US-PAT-NO: 6085249

DOCUMENT-IDENTIFIER: US 6085249 A

TITLE: Method and apparatuses for transferring data for multiple applications through a single communication link in response to authentication information

DATE-ISSUED: July 4, 2000

INVENTOR-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY
Wang; Wu	Los Altos	CA		
Kongara; Prasad	San Jose	CA		

ASSIGNEE-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY	TYPE CODE
Pictra, Inc.	Sunnyvale	CA			02

APPL-NO: 08/ 985746 [PALM]

DATE FILED: December 5, 1997

PARENT-CASE:

CROSS-REFERENCE TO RELATED APPLICATIONS This is a continuation-in-part of application Ser. No. 08/957,219 filed Oct. 24, 1997.

INT-CL: [07] G06 F 15/16

US-CL-ISSUED: 709/229; 709/227, 709/203, 709/218

US-CL-CURRENT: 709/229; 709/203, 709/218, 709/227

FIELD-OF-SEARCH: 713/201, 713/202, 709/229, 709/300, 709/218, 709/217

PRIOR-ART-DISCLOSED:

U.S. PATENT DOCUMENTS

	PAT-NO	ISSUE-DATE	PATENTEE-NAME	US-CL
<input type="checkbox"/>	<u>5560008</u>	September 1996	Johnson et al.	709/229
<input type="checkbox"/>	<u>5708780</u>	January 1998	Levergood et al.	709/229
<input type="checkbox"/>	<u>5764890</u>	June 1998	Glasser et al.	713/202
<input type="checkbox"/>	<u>5815665</u>	September 1998	Teper et al.	709/229
<input type="checkbox"/>	<u>5845070</u>	December 1998	Ikudome	709/229
<input type="checkbox"/>	<u>5872915</u>	February 1999	Dykes et al.	713/201
<input type="checkbox"/>	<u>5875296</u>	February 1999	Shi et al.	713/201

ART-UNIT: 278

PRIMARY-EXAMINER: Maung; Zarni

ASSISTANT-EXAMINER: Tran; Philip B.

ATTY-AGENT-FIRM: Blakely, Sokoloff, Taylor & Zafman

ABSTRACT:

A method and apparatus for transferring data in response to authentication information. Authentication information is transmitted from a client computer to a server computer to authorize access to restricted data. Invocation information is transmitted from the server computer to the client computer in response to the authentication information. A first application is invoked in the client computer using the invocation information. The restricted data is then transmitted from the server computer to the first application program in the client computer based on the access authorized by the authentication information.

20 Claims, 11 Drawing figures

Draft Only

WEST☐ **Generate Collection** **Print**

L6: Entry 30 of 31

File: USPT

Jul 14, 1998

US-PAT-NO: 5781629

DOCUMENT-IDENTIFIER: US 5781629 A

TITLE: Digital document authentication system

DATE-ISSUED: July 14, 1998

INVENTOR-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY
Haber; Stuart A.	New York	NY		
Stornetta, Jr.; W. Scott	Morriston, Jr.	NJ		

ASSIGNEE-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY	TYPE CODE
Surety Technologies, Inc.	Florham Park	NJ			02

APPL-NO: 08/ 803326 [PALM]

DATE FILED: February 21, 1997

PARENT-CASE:

This is a continuation of application Ser. No. 08/330,459, filed Oct. 28, 1994, abandoned.

INT-CL: [06] H04 L 9/32

US-CL-ISSUED: 380/23; 380/28

US-CL-CURRENT: 713/177; 380/28, 713/175

FIELD-OF-SEARCH: 380/23, 380/28, 380/30

PRIOR-ART-DISCLOSED:

U.S. PATENT DOCUMENTS

Search Selected**Search ALL**

PAT-NO	ISSUE-DATE	PATENTEE-NAME	US-CL
<input type="checkbox"/> <u>Re34954</u>	May 1995	Haber et al.	380/49
<input type="checkbox"/> <u>4309569</u>	January 1982	Merkle	
<input type="checkbox"/> <u>4405829</u>	September 1983	Rivest et al.	
<input type="checkbox"/> <u>4881264</u>	November 1989	Meakle	380/25
<input type="checkbox"/> <u>5136646</u>	August 1992	Haber et al.	
<input type="checkbox"/> <u>5136647</u>	August 1992	Haber et al.	
<input type="checkbox"/> <u>5373561</u>	December 1994	Haber et al.	
<input type="checkbox"/> <u>5432852</u>	July 1995	Leighton et al.	380/30

OTHER PUBLICATIONS

Bayes, et al., "Improving the Efficiency and the Reliability of Digital Time-Stamping", sequences II methods in communication, security, and computer science, editors R.M. Capocelli, et al. pp. 329-334, (Springer-Verlag, New York 1993).

Haber et al., "How to Time-Stamp a Digital Document", Journal of Cryphotography, vol. 3, No. 2, pp. 99-111 (1991).

ART-UNIT: 222

PRIMARY-EXAMINER: Barron, Jr.; Gilberto

ATTY-AGENT-FIRM: Meltzer, Lippe, Goldstein, et al.

ABSTRACT:

A process for time-stamping a digital document is provided. The process provides a certificate which not only allows for the authentication of a document at a later time but which includes a name or nickname which allows for the unique identification of the document at a later time. The name or nickname provided in accordance with the present invention is not only simple and concise but allows for the self-authentication of the document which it refers to. The name can be used when two independent parties desire to refer to the same unique document in a quick and simple way.

74 Claims, 10 Drawing figures

Draft Only



#11C
11/5/03
AW

UNITED STATES PATENT AND TRADEMARK OFFICE

Attorney Docket No.: 40055-0006

Applicant(s): David Farber *et al.*

Confirmation No.: 7771

Appl. No.: 09/987,723

Examiner: Jean Raymond Homere

Filing Date: November 15, 2001

Group Art Unit: 2177

Title: IDENTIFYING DATA IN NETWORK USING IDENTIFIERS WHICH ARE
BASED ON CONTENTS OF DATA (as amended)

RECEIVED

REPLY AND PRELIMINARY AMENDMENT

OCT 21 2003

Technology Center 2100

Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:

Applicants herein respond the Office Action mailed May 14, 2003 (Paper No. 6), for the above-captioned application. Applicants also Petition for a four-month Extension of Time and provide the requisite fees herewith. While applicants believe that the correct payment for the extension of time fee is attached, the Commissioner is authorized to credit any overpayment or charge any deficiency to Deposit Account No. 08-1641.

Please also amend the Application as set forth in the Amendments below.

10/17/2003 EAREGAY1 00000106 09987723

01 FC:1201
02 FC:1202

946.00 OP
720.00 OP

AMENDMENTS

IN THE TITLE:

Please replace the title with

--IDENTIFYING DATA IN NETWORK USING IDENTIFIERS WHICH ARE BASED ON
C1 CONTENTS OF DATA—

Draft Only

IN THE CLAIMS:

Claims 1-45 (Cancelled).

46. (Withdrawn) A method of eliminating a data item at a given location in a data processing system when said data item can be obtained from another location in the system, the method comprising the steps of:

determining a substantially unique identifier for the data, said identifier depending on all of the data in the data item and only on the data in the data item;

making and maintaining a source association between the data identifier and at least one location at which said data item is known to be present; and

based on said source association, if said data item is present at said other location, removing the data item from the given location.

47. (Withdrawn) A method of deleting a data item from a location in a data processing system, the method comprising the steps of:

for each of a plurality of data items in the system:

determining a substantially unique identifier for the data, said identifier depending on all of the data in the data item and only on the data in the data item; and

making and maintaining, an association between each of the data items and the unique identifier of the data items; and

for a given data item:

determining a substantially unique identifier for the data, said identifier depending on all of the data in the data item and only on the data in the data item; and

determining whether a contextual identifier or a compound data item or a remote processor in the system refers to the unique identifier of the data item, and based on said determining, deleting said data item and its association if no other contextual identifier or compound data item or remote processor refers to said data item.

48. (Withdrawn) The method of claim 47, wherein said determining is based on a use count for the data item, and wherein said data item is deleted only if said use count indicates that no other contextual identifier or compound data item or remote processor in the system refers to the data item.

49. (Withdrawn) A method of substantially synchronizing data items at a client location in a data processing system after a period of independent changes on the client and another location in the system, given a context, the method comprising the steps of:

making and maintaining a list of changes to the context association between each context name of a data item and the identifier of said data item, in the given context and during the period of independent change;

obtaining the list of changes from the other location for the given context; and, for each context name in the list of changes

updating the context identifier associations at the client whenever it is determined that the context association of the given context name changed either only at the client or only at the other location during the period of independent changes; and

performing a conflict-resolution task such as notifying an operator of the client location, whenever it is determined that the context association changed at both the client and the other location.

Claims 50-53 (Cancelled)

C2
54. (Currently Amended) ~~A content delivery method, in~~ In a system in which a plurality of files are distributed across a plurality of computers network of servers, ~~at least some of the files being cached versions of files from a source server distinct from the network of servers,~~ the content delivery a method comprising:

~~for a particular file, determining~~ obtaining a name for a data file, the name being based at least in part on using a given function of the data, said data being the data which comprises the contents of the particular data file; and

in response to a request for the a particular data file, the request including at least the name of the particular file, causing a copy of the particular file to be provided from a given one of the ~~servers of the network of servers~~ plurality of computers.

55. (Currently Amended) A ~~content delivery~~ method, in a system in which a plurality of files are distributed across a plurality of computers ~~network of servers~~, at ~~least some of the files being cached versions of files from a source server distinct from the servers in the network~~, wherein data in a file in the system may represent a digital message, a digital image, a video signal or an audio signal, the ~~content delivery~~ method comprising:

obtaining ~~determining~~ a name for a particular data file, the name being having been determined using an MD5 function of the data, ~~said data being the data which~~ comprises the contents of the particular data file; and

in response to a request for the particular data file, the request including at least the name of the particular data file, providing the particular data file from a given one of the ~~servers of the network of servers~~ plurality of computers, said providing being based at least in part on the determined name.

56. (Currently Amended) A ~~content delivery~~ method, in a system in which a plurality of files are distributed across a ~~network of servers~~ plurality of computers, wherein some of the computers ~~processors in the network~~ communicate with each other using a TCP/IP communication protocol, the ~~content delivery~~ method comprising:

obtaining a name for a particular data file, the contents of said data file representing a digital image, ~~determining a name for the particular file, wherein the name is being~~ determined using at least a given function of the data which comprises the contents of the particular data file; and

in response to a request for the particular data file, the request including at least the name of the particular data file, providing the particular file from a given one of the ~~servers of the network of servers~~ plurality of computers.

57. (Currently Amended) A ~~content delivery~~ method comprising:

causing a plurality of files to be distributed across a ~~network of servers~~ plurality of computers, ~~at least some of the files being cached versions of files from a source server which is distinct from the network of servers;~~

obtaining a name for a particular file, ~~determining a name~~, the name being having been determined using a given function of the data, ~~said data used by said function being data~~ which comprises the contents of the particular file; and

in response to a request for the particular file, the request including at least the name of the particular file, causing the particular file to be provided from a given one of the ~~servers of the network of servers~~ plurality of processors.

58. (Currently Amended) A ~~content delivery~~ method, in a system in which a plurality of files are distributed across a plurality of computers ~~network of servers~~, ~~at least some of the files being cached versions of files from a source server which is distinct from the network of servers;~~ ~~the content delivery~~ the method comprising:

obtaining ~~determining~~ a name for a particular file, the name being determined using a given function of the data which comprises the contents of the particular file; and

in response to a request for the particular file, the request including at least the name of the particular file, providing the particular file from a given one of the ~~servers of the network of servers~~ computers, wherein the contents of the particular file may represent a digital message, a digital image, a video signal or an audio signal.

59. (Currently Amended) A method, in a network comprising a plurality of computers ~~processors~~, some of the computers ~~processors~~ functioning as servers and some of the computers ~~processors~~ functioning as clients, wherein some computers ~~processors~~ in the network communicate with each other using a TCP/IP

communication protocol, wherein a key is required to identify a file on the network, the method comprising:

storing some files on a first computer server in the network and storing copies of some of the files from the first computer server on a set of computers ~~cache servers~~ distinct from the first computer server;

for a particular file, determining a different cache key from an ordinarily used ~~cache~~ key for the file, the different ~~cache~~ key being determined at least in part using a message function MD5 of the data, ~~wherein said data~~ which comprises the contents of the particular file; and

responsive to a ~~client~~ request for the particular file, the request including the different ~~cache~~ key for the file, causing the particular file to be provided to the ~~client~~ requestor,

wherein the contents of the file may represent: a page in memory, a digital message, a digital image, a video signal or an audio signal.

C2
60. (Currently Amended) A ~~content delivery~~ method comprising:

distributing a set of files from a first computer server across a network of computers ~~servers~~ distinct from the first computer server;

applying an MD5 function to the contents of a particular file to obtain a True Name for the file;

in response to a request for the particular file, the request including at least the True Name of the particular file, causing the particular file to be provided from a given one of the computers ~~servers of the network of servers~~, wherein the request for the particular file is resolved based, at least in part, on a measure of availability of at least one of the computers ~~servers~~.

61. (Currently Amended) A method as in claim 60 wherein the measure of availability for a computer server is based on at least one of:

(a) a measurement of bandwidth to the computer server;

- (b) a measurement of a cost of a connection to the computer server, and
- (c) a measurement of reliability of a connection to the computer server.

62. (Currently Amended) A content delivery method comprising:
distributing a plurality of files across a network of computers servers, ~~at least some of the files being cached versions of files from a source server distinct from the servers in the network;~~
for a particular file, determining a True Name using at least a given function of the data which comprises the contents of the particular file;
obtaining a request for the particular file, the request including at least the True Name of the particular file; and
responsive to the request, causing the particular file to be provided from one of the servers of the network of ~~servers~~ computers.

63. (Previously presented) A content delivery method, comprising:
distributing files across a network of servers;
for a particular file having a contextual name specifying a location in the network at which the file may be located, determining another name for the particular file, the other name including a data identifier determined using a given function of the data, where said data used by the given function comprises the contents of the particular file;
obtaining a request for the particular file, the request including the contextual name and the other name of the particular file,
responsive to the request, providing the particular file from one of the servers of the network of servers, said providing being based on the other name of the particular item.

64. (Previously presented) A content delivery method, comprising:
distributing a set of files across a network of servers;

for a particular file representing a digital image, the file having a contextual name specifying a location in the network at which the file may be located, determining another name for the particular file, the other name including a True Name for the file which was determined using a message digest function of the data, where said data used by the given function comprises the contents of the particular file;

obtaining a request for the particular file, the request including the contextual name and the True Name of the particular file; and

responsive to the request, providing the particular file from one of the servers of the network of servers, said providing being based on the True Name of the particular item.

C2 65. (Previously presented) A method comprising:
applying an MD5 function to the contents of an image file containing data representing a digital image to obtain a True Name for the file
distributing copies of the image file from a first server across a network of servers distinct from the first server;
obtaining a request for the image file, the request including at least the True Name of the file; and
responsive to the request, causing a copy of the image file to be provided from one of the servers of the network of servers.

NEW CLAIMS

66. (New) A method as in claim 54, wherein the requested file is only provided to authorized or licensed parties.

67. (New) A method as in claim 54, wherein the requested file is not provided to unlicensed or unauthorized parties.

68. (New) A method as in claim 54 further comprising:
determining, using at least the name, whether a copy of the data file is present on a particular one of said computers.

69. (New) A method as in claim 54 further comprising:
determining, using at least the name, whether an authorized or licensed copy of the data file is present on a particular one of said computers.

70. (New) A method, in a system in which a plurality of files are distributed across a plurality of computers, the method comprising:
obtaining a name for a data file, the name being based at least in part on a given function of the data which comprises the contents of the particular file;
determining, using at least the name, whether a copy of the data file is present on a at least one of said computers.

71. (New) A method as in claim 70 further comprising:
determining whether a copy of the data file that is present on a at least one of said computers is an authorized or licensed copy of the file.

72. (New) A method as in claim 71 further comprising:

allowing the file to be provided from one of the computers having an authorized or licensed copy of the file.

73. (New) A method as in claim 70 wherein at least some of the plurality of computers comprise a peer-to-peer network.

74. (New) A method, in a system in which a plurality of files are distributed across a plurality of computers which form a peer-to-peer network, the method comprising:

obtaining a TrueName for a data file, the TrueName being based at least in part on a given function of the data which comprises the contents of the particular file; and
determining, using at least the name, whether a licensed or authorized copy of the data file is present on a particular computer.

75. (New) A method comprising:

obtaining a name for a data file, the name being based at least in part on a function of the data which comprise the contents of the file; and

in response to a request for the data file, the request including at least the obtained name of the data file, causing the contents of the data file to be provided from a computer having a licensed or authorized copy of the data file.

76. (New) A method as in claim 75 wherein the function is a message digest function or a hash function.

77. (New) A method as in claim 75 wherein the function is selected from the functions: MD4, MD5, and SHA.

78. (New) A method as in claim 75 wherein the given function randomly distributes its outputs.

79. (New) A method as in claim 75 wherein the function produces a substantially unique value based on the data comprising the data file.

80. (New) A method as in claim 75 wherein a data file may comprise a file, a portion of a file, a page in memory, a digital message, a digital image, a video signal or an audio signal.

81. (New) A method as in claim 75 wherein certain processors in the network communicate with each other using a TCP/IP communication protocol.

82. (New) A method as in claim 75 wherein said name for said data file, as determined using said function, will change when the data file is modified.

C3 83. (New) A method, in a system in which a plurality of files are distributed across a plurality of computers, the method comprising:

obtaining a name for a data file, the name being based at least in part on an MD5 function of the data which comprises the contents of the particular file;

determining, using at least the obtained name, whether an authorized or licensed copy of the data file is present on a at least one of said computers.

84. (New) A method comprising:

obtaining a list of file names for each of a plurality of files, each of said file names having been determined, at least in part, by applying a function to the contents of the corresponding file;

using at least said list to determine whether unauthorized or unlicensed copies of some of the plurality of data files are present on a particular computer.

85. (New) A method as in claim 84 further comprising:

in response to a request for a particular data file, allowing the contents of the data file to be provided from a computer determined to have a licensed or authorized copy of the data file.

86. (New) A method as in claim 84 wherein the particular computer is part of a peer-to-peer network of computers.

87. (New) A method as in claim 84 further comprising:
if the computer is found to have a file that it is not authorized or licensed to have, recording information about the computer and about the file.

88. (New) A method as in claim 84 wherein the function is a message digest function or a hash function.

89. (New) A method as in claim 84 wherein the function is selected from the functions: MD4, MD5, and SHA.

90. (New) A method as in claim 84 wherein the given function randomly distributes its outputs.

91. (New) A method as in claim 84 wherein the function produces a substantially unique value based on the data comprising the data file.

92. (New) A method comprising:
obtaining a list of True Names for each of a plurality of files;
for at least some computers that make up part of a peer-to-peer network of computers, comparing at least some of the contents of the computers to the list of True Names to determine whether unauthorized or unlicensed copies of some of the plurality of data files are present on those computers; and

based at least in part on said comparing, if a computer is found to have content that it is not authorized or licensed to have, recording information about the computer and about the unauthorized or unlicensed content.

93. (New) A method as in claim 92 wherein the True Names are determined using a message digest function or a hash function.

94. (New) A method as in claim 92 wherein the function is selected from the functions: MD4, MD5, and SHA.

95. (New) A method comprising:
obtaining a list of True Names for each of a plurality of files, wherein, for each of the files, the True Name for that file is determined using an MD5 function of the contents of the file;

C3 comparing at least some of the contents of a computers to the list of True Names to determine whether unauthorized or unlicensed copies of some of the plurality of data files are present on that computer; and

based at least in part on said comparing, if a computer is found to have content that it is not authorized or licensed to have, recording information about the computer and about the unauthorized or unlicensed content.

96. (New) A method comprising:
distributing a set of files from a first computer across a network of computers distinct from the first computer;
in response to a request for the particular file, causing the particular file to be provided from a given one of the computers, wherein the request for the particular file is resolved based, at least in part, on a measure of availability of at least one of the computers, and wherein the measure of availability for a computer is based on at least one of:

- (a) a measurement of bandwidth to the computer;
- (b) a measurement of a cost of a connection to the computer, and
- (c) a measurement of reliability of a connection to the computer.

97. (New) A method as in claim 91 wherein the request for the particular file includes a name determined as a function of the contents of the file.

98. (New) In a system in which a data file is distributed across a plurality of computers, a method comprising:

obtaining a name for the data file, the name being based at least in part on a given function of the data which comprise the contents of the data file;

determining, using at least the name, whether an authorized or licensed copy of the data file is present on a particular one of said computers.

99. (New) A method as in claim 94, further comprising:

in response to a request for the data file, allowing a copy of the file to be provided from a given one of the plurality of computers having an authorized or licensed copy of the file.

100. (New) A method as in claim 54, further comprising:
maintaining accounting information relating to the data files.

101. (New) A method as in claim 96, wherein the maintaining of accounting information includes at least some of:

- (a) tracking which files have been stored on a computer; and
- (b) tracking which files have been transmitted from a computer.

102. (New) Computer-readable media tangibly embodying a program of instructions executable by at least one computer, the program comprising code to:

obtain a name for a data file, the name being based at least in part on a given function of the data which comprises the contents of the data file; and

in response to a request for the a data file, the request including at least the name of the particular file, cause a copy of the file to be provided from a given one of the plurality of computers.

103. (New) Computer-readable media tangibly embodying a program of instructions executable by at least one computer, the program comprising code to:

obtain a TrueName for a data file, the TrueName being based at least in part on a given function of the data which comprises the contents of the particular file; and

determine, using at least the name, whether a licensed or authorized copy of the data file is present on a particular computer.

C3 104. (New) Computer-readable media tangibly embodying a program of instructions executable by at least one computer, the program comprising code to:

obtain a list of file names for each of a plurality of files, each of said file names having been determined, at least in part, by applying a function to the contents of the corresponding file; and

determine, using at least said list, whether unauthorized or unlicensed copies of some of the plurality of data files are present on a particular computer.

105. (New) Computer-readable media tangibly embodying a program of instructions executable by at least one computer, the program comprising code to:

obtain a name for the data file, the name being based at least in part on a given function of the data which comprise the contents of the data file; and

determine, using at least the name, whether an authorized or licensed copy of the data file is present on a particular one of said computers.

106. (New) Media as in claim 105 wherein the given function is a message digest function or a hash function.

107. (New) A computer system programmed to:
obtain a list of file names for each of a plurality of files, each of said file names having been determined, at least in part, by applying a function to the contents of the corresponding file; and
determine, using at least said list, whether unauthorized or unlicensed copies of some of the plurality of data files are present on a particular computer.

CB 108. (New) A computer system as in claim 107 wherein the function is a message digest function or a hash function.

109. (New) A computer system as in claim 107 wherein the function is an MD5 function.

REMARKS

Applicants hereby elect **Group I**, claims 54-65.

Applicants, of course, reserve the right to file one or more divisional applications covering subject matter of the non-elected claims.

New claims 66-109 have been added and the title has been amended.

Claims 46-49 and 54-109 are pending, claims 46-49 having been withdrawn from consideration. No new matter is added with the amendments, which are fully supported by the specification.

It is respectfully submitted that the application is in condition for examination, and an early action on the merits is courteously requested.

The Examiner is invited to contact Brian Siritzky, the undersigned attorney for applicants, at 202-912-2196 for any reason related to the advancement of this case.

Respectfully submitted,

Date: 10/13/2003

By 

HELLER EHRMAN WHITE & MCAULIFFE
1666 K Street, NW, Suite 300
Washington, D.C. 20006
Phone: (202) 912-2000
Fax: (202) 912-2020

Brian Siritzky
Attorney for Applicant
Reg. No.: 37,497

Customer No. 26633

#10
11/5/03
A.W.

PETITION FOR EXTENSION OF TIME UNDER 37 CFR 1.136(a)

Docket Number (Optional)
40055-0006



In re Application of David A. FARBER et al.

Application Number 09/987,723

Filed November 15, 2001

For IDENTIFYING AND REQUESTING DATA IN NETWORK
USING IDENTIFIERS WHICH ARE BASED ON CONTENTS OF
DATA

Group Art Unit
2177

Examiner
HOMERE, Jean Raymond

This is a request under the provisions of 37 CFR 1.136(a) to extend the period for filing a response in the above identified application.

The requested extension and appropriate non-small-entity fee are as follows
(check time period desired):

- ☐ One month (37 CFR 1.17(a)(1)) \$ _____
- ☐ Two months (37 CFR 1.17(a)(2)) \$ _____
- ☐ Three months (37 CFR 1.17(a)(3)) \$ _____
- ☒ Four months (37 CFR 1.17(a)(4)) \$1480
- ☐ Five months (37 CFR 1.17(a)(5)) \$ _____

- ☐ Applicant claims small entity status. See 37 CFR 1.27. Therefore, the fee amount shown above is reduced by one-half, and the resulting fee is: \$ _____.
- ☒ A check in the amount of the fee is enclosed.
- ☐ Payment by credit card. Form PTO-2038 is attached.
- ☐ The Commissioner has already been authorized to charge fees in this application to a Deposit Account.
- ☒ The Commissioner is hereby authorized to charge any fees which may be required, or credit any overpayment, to Deposit Account Number 08-1641.
I have enclosed a duplicate copy of this sheet.

I am the ☐ applicant/inventor.

☐ assignee of record of the entire interest. See 37 CFR 3.71

Statement under 37 CFR 3.73(b) is enclosed. (Form PTO/SB/96).

☒ attorney or agent of record.

☐ attorney or agent under 37 CFR 1.34(a).

Registration number if acting under 37 CFR 1.34(a). _____.

RECEIVED

OCT 21 2003

Technology Center 2100

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-2038.

10/13/2003
Date

B. Sirtzky
Signature

Brian Sirtzky, Reg. No. 37,497
Typed or printed name

NOTE: Signatures of all the inventors or assignees of record of the entire interest or their representative(s) are required. Submit multiple forms if more than one signature is required, see below*.

☒ *Total of 1 forms are submitted.

10/17/2003 EAREGAY1 00000106 09987723

03 FC:1254

1480.00 OP



2177/14

PTO/SB/21 (08-00)

Please type a plus sign (+) inside this box → ☐

Approved for use through 10/31/2002. OMB 0651-0031

U.S. Patent and Trademark Office: U.S. DEPARTMENT OF COMMERCE

Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it displays a valid OMB control number.

TRANSMITTAL FORM <i>(to be used for all correspondence after initial filing)</i>	Application Number	09/987,723	
	Filing Date	November 15, 2001	
	First Named Inventor	David A. FARBER et al.	
	Group Art Unit	2177	
	Examiner Name	HOMERE, Jean Raymond	
Total Number of Pages in This Submission	21	Attorney Docket Number	40055-0006

ENCLOSURES (check all that apply)		
<input checked="" type="checkbox"/> Fee Transmittal Form	<input type="checkbox"/> Assignment Papers <i>(for an Application)</i>	<input type="checkbox"/> After Allowance Communication to Group
<input checked="" type="checkbox"/> Fee Attached	<input type="checkbox"/> Drawing(s)	<input type="checkbox"/> Appeal Communication to Board of Appeals and Interferences
<input checked="" type="checkbox"/> Amendment / Response	<input type="checkbox"/> Licensing-related Papers	<input type="checkbox"/> Appeal Communication to Group <i>(Appeal Notice, Brief, Reply Brief)</i>
<input type="checkbox"/> After Final	<input type="checkbox"/> Petition	<input type="checkbox"/> Proprietary Information
<input type="checkbox"/> Affidavits/declaration(s)	<input type="checkbox"/> Petition to Convert to a Provisional Application	<input type="checkbox"/> Status Letter
<input checked="" type="checkbox"/> Extension of Time Request	<input type="checkbox"/> Power of Attorney, Revocation Change of Correspondence Address	<input type="checkbox"/> Other Enclosure(s) <i>(please identify below):</i>
<input type="checkbox"/> Express Abandonment Request	<input type="checkbox"/> Terminal Disclaimer	RECEIVED OCT 21 2003 Technology Center 2100
<input type="checkbox"/> Information Disclosure Statement	<input type="checkbox"/> Request for Refund	
<input type="checkbox"/> Certified Copy of Priority Document(s)	<input type="checkbox"/> CD, Number of CD(s)	
<input type="checkbox"/> Response to Missing Parts/ Incomplete Application		
<input type="checkbox"/> Response to Missing Parts under 37 CFR 1.52 or 1.53	Remarks	

SIGNATURE OF APPLICANT, ATTORNEY, OR AGENT	
Firm or Individual name	Brian Siritzky, Reg. No. 37,497, HELLER EHRMAN WHITE & MCAULIFFE LLP Customer No. 26633
Signature	
Date	10/13/2003

CERTIFICATE OF MAILING			
I hereby certify that this correspondence is being deposited with the United States Postal Service as first class mail in an envelope addressed to: Assistant Commissioner for Patents, Washington, D.C. 20231 on this date: <input type="text"/>			
Typed or printed name			
Signature		Date	

**FEE TRANSMITTAL
for FY 2003**

Effective 01/01/2003. Patent fees are subject to annual revision.

☐ Applicant claims small entity status. See 37 CFR 1.27

TOTAL AMOUNT OF PAYMENT (\$) 3146

Complete if Known

Application Number	09/987,723
Filing Date	November 15, 2001
First Named Inventor	David A. FARBER et al.
Examiner Name	HOMERE, Jean Raymond
Art Unit	2177
Attorney Docket No.	40055-0006

METHOD OF PAYMENT (check one)

☒ Check ☐ Credit card ☐ Money Order ☐ Other ☐ None☐ Deposit Account:

Deposit Account Number 08-1641 (Docket No. 40055-0006)

Deposit Account Name Heller Ehrman White & McAuliffe LLP

The Commissioner is authorized to: (check all that apply)

☐ Charge fee(s) indicated below ☒ Credit any overpayments☒ Charge any additional fee(s) during the pendency of this application☐ Charge fee(s) indicated below, except for the filing fee to the above-identified deposit account.

FEE CALCULATION

1. BASIC FILING FEE

Large Fee Code	Entity Fee (\$)	Small Fee Code	Entity Fee (\$)	Fee Description	Fee Paid
1001	770	2001	385	Utility filing fee	
1002	340	2002	170	Design filing fee	
1003	530	2003	265	Plant filing fee	
1004	770	2004	385	Reissue filing fee	
1005	160	2005	80	Provisional filing fee	

SUBTOTAL (1) (\$) 720

2. EXTRA CLAIM FEES FOR UTILITY AND REISSUE

Total Claims	Extra Claims	Fee from below	Fee Paid
60	-20** = 40	x 18 = 720	
Independent Claims	25	-14** = 11	x 86 = 946

Multiple Dependent

Large Fee Code	Entity Fee (\$)	Small Fee Code	Entity Fee (\$)	Fee Description
1202	18	2202	9	Claims in excess of 20
1201	86	2201	43	Independent claims in excess of 3
1203	290	2203	145	Multiple dependent claim, if not paid
1204	86	2204	43	**Reissue independent claims over original patent
1205	18	2205	9	**Reissue claims in excess of 20 and over original patent

SUBTOTAL (2) (\$) 1666

**or number previously paid, if greater; For Reissues, see above

FEE CALCULATION (continuation)

3. ADDITIONAL FEES

Large Fee Code	Entity Fee (\$)	Small Fee Code	Entity Fee (\$)	Fee Description	Fee Paid
1051	130	2051	65	Surcharge - late filing fee or oath	
1052	50	2052	25	Surcharge - late provisional filing fee or cover sheet	
1053	130	1053	130	Non-English specification	
1812	2,520	1812	2,520	For filing a request for <i>ex parte</i> reexamination	
1804	920*	1804	920*	Requesting publication of SIR prior to Examiner action	
1805	1,840*	1805	1,840*	Requesting publication of SIR after Examiner action	
1251	110	2251	55	Extension for reply within first month	
1252	420	2252	210	Extension for reply within second month	
1253	950	2253	475	Extension for reply within third month	
1254	1,480	2254	740	Extension for reply within fourth month	1480
1255	2,010	2255	1,005	Extension for reply within fifth month	
1401	330	2401	165	Notice of Appeal	
1402	330	2402	165	Filing a brief in support of an appeal	
1403	290	2403	145	Request for oral hearing	
1451	1,510	1451	1,510	Petition to institute a public use proceeding	
1452	110	2452	55	Petition to revive - unavoidable	
1453	1,330	2453	665	Petition to revive - unintentional	
1501	1,330	2501	665	Utility issue fee (or reissue)	
1502	480	2502	240	Design issue fee	
1503	640	2503	320	Plant issue fee	
1460	130	1460	130	Petitions to the Commissioner	
1807	50	1807	50	Processing fee under 37 CFR 1.17(q)	
1806	180	1806	180	Submission of Information Disclosure Stmt	
8021	40	8021	40	Recording each patent assignment per property (times number of properties)	
1809	770	2809	385	Filing a submission after final rejection (37 CFR 1.129(a))	
1810	770	2810	385	For each additional invention to be examined (37 CFR 1.129(b))	
1801	770	2801	385	Request for Continued Examination (RCE)	
1802	900	1802	900	Request for expedited examination of a design application	

Other fee (specify)

* Reduced by Basic Filing Fee Paid

SUBTOTAL (3) (\$) 1480

SUBMITTED BY

Name (Print/Type) Brian Siritzky

Registration No. 37,497
(Attorney/Agent)

Complete (if applicable)

Telephone (202) 912-2000

Signature

Date

Customer No. 26633

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-2038.

This collection of information is required by 37 CFR 1.17 and 1.27. 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, Washington, DC 20231. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. SEND TO: Mail Stop 3, Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.

If you need assistance in completing the form, call 1-800-PTO-9199 (1-800-786-9199) and select option 3.

GOOG-1002



#9
2177

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Attorney Docket No. 39685-0036

In re patent application of:
David A. FARBER et al.

Confirmation No. 7771

Serial No.: 09/987,723

Art Unit: 2177

Filed: November 15, 2001

Examiner: Homere, J.

For: IDENTIFYING AND REQUESTING DATA IN NETWORK USING IDENTIFIERS
WHICH ARE BASED ON CONTENTS OF DATA

**INFORMATION DISCLOSURE STATEMENT
UNDER 37 CFR §1.56 and 37 CFR §1.97**

RECEIVED

SEP 04 2003

Commissioner for Patents
P. O. Box 1450
Alexandria, VA 22313-1450

Technology Center 2100

Sir:

Submitted herewith on Form PTO/SB/08A is a listing of documents known to applicants in order to comply with applicants' duty of disclosure pursuant to 37 C.F.R. §1.56.

The submission of any document herewith, which is not a statutory bar, is not intended as an admission that such document constitutes prior art against the claims of the present application or is considered to be material to patentability as defined in 37 C.F.R. §1.56(b). Applicants do not waive any rights to take any action which would be appropriate to antedate or otherwise remove as a competent reference any document which is determined to be a *prima facie* prior art reference against the claims of the present application.

The documents submitted herewith on Form PTO/SB/08A, lists the documents cited by or submitted to the Patent Office in parent application Serial No. 09/283,160 filed April 1, 1999 and grandparent Serial No. 08/960,079 filed October 24, 1997.

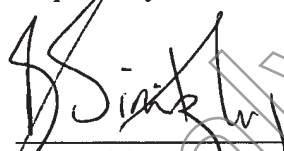
The documents were previously cited by or submitted to the Patent Office in parent application Serial No. 09/283,160 filed April 1, 1999 and grandparent Serial No. 08/960,079 filed October 24, 1997.

The instant Information Disclosure Statement is being filed in compliance with 37 CFR §1.97(b) prior to the mailing date of the first official action, therefore, no fee is required in connection with its filing. However, the Commissioner is hereby authorized to charge any deficiency or to credit any overpayment to Deposit Account No. 08-1641.

Applicants respectfully request that the listed documents be considered by the Examiner and be made of record in the present application and that an initialed copy of Form PTO-SB/08A be returned in accordance with M.P.E.P. §609.

9/3/2003
Date

Respectfully submitted,


Brian Siritzky
Reg. No. 37,497

Customer ID No. 26633
HELLER EHRMAN WHITE & McAULIFFE LLP
1666 K Street, N.W., Suite 300
Washington, DC 20006
Telephone: (202) 912-2000
Facsimile: (202) 912-2020



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
 United States Patent and Trademark Office
 Address: COMMISSIONER OF PATENTS AND TRADEMARKS
 P.O. Box 1450
 Alexandria, Virginia 22313-1450
 www.uspto.gov

APPLICATION NUMBER	FILING OR 371 (c) DATE	FIRST NAMED APPLICANT	ATTY. DOCKET NO./TITLE
09/987,723	11/15/2001	David A. Farber	P 283002 TN-Cont-2

CONFIRMATION NO. 7771

26633
 HELLER EHRMAN WHITE & MCAULIFFE LLP
 1666 K STREET, NW
 SUITE 300
 WASHINGTON, DC 20006



OC000000010623454

Date Mailed: 08/01/2003

NOTICE OF ACCEPTANCE OF POWER OF ATTORNEY

This is in response to the Power of Attorney filed 07/29/2003.

The Power of Attorney in this application is accepted. Correspondence in this application will be mailed to the above address as provided by 37 CFR 1.33.

Draft Only

ANGELA S WHITE
 2100 (703) 308-8264

OFFICE COPY



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
 United States Patent and Trademark Office
 Address: COMMISSIONER OF PATENTS AND TRADEMARKS
 P.O. Box 1450
 Alexandria, Virginia 22313-1450
 www.uspto.gov

APPLICATION NUMBER	FILING OR 371 (c) DATE	FIRST NAMED APPLICANT	ATTY. DOCKET NO./TITLE
09/987,723	11/15/2001	David A. Farber	P 283002 TN-Cont-2

CONFIRMATION NO. 7771

PILLSBURY WINTHROP
 1600 TYSONS BOULEVARD
 MCLEAN, VA 22102



OC000000010623435

Date Mailed: 08/01/2003

NOTICE REGARDING CHANGE OF POWER OF ATTORNEY

This is in response to the Power of Attorney filed 07/29/2003.

- The Power of Attorney to you in this application has been revoked by the assignee who has intervened as provided by 37 CFR 3.71. Future correspondence will be mailed to the new address of record(37 CFR 1.33).

ANGELA S WHITE
 2100 (703) 308-8264

OFFICE COPY



#8
8/1/03
Aw.

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re patent application of:
David A. FARBER et al.

Attorney Docket No. 39685-0036
Confirmation No. 7771

Serial No.: 09/987,723

Art Unit: 2177

Filed: November 15, 2001

Examiner: Homere, J.

For: IDENTIFYING AND REQUESTING DATA IN NETWORK USING IDENTIFIERS
WHICH ARE BASED ON CONTENTS OF DATA

**INFORMATION DISCLOSURE STATEMENT
UNDER 37 CFR §1.56 and 37 CFR §1.97**

RECEIVED

Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

JUL 30 2003

Technology Center 2100

Sir:

Submitted herewith on Form PTO/SB/08A is a listing of documents known to applicants in order to comply with applicants' duty of disclosure pursuant to 37 C.F.R. §1.56.

The submission of any document herewith, which is not a statutory bar, is not intended as an admission that such document constitutes prior art against the claims of the present application or is considered to be material to patentability as defined in 37 C.F.R. §1.56(b). Applicants do not waive any rights to take any action which would be appropriate to antedate or otherwise remove as a competent reference any document which is determined to be a *prima facie* prior art reference against the claims of the present application.

The documents submitted herewith on Form PTO/SB/08A, lists the documents cited by or submitted to the Patent Office in parent application Serial No. 09/283,160 filed April 1, 1999 and grandparent Serial No. 08/960,079 filed October 24, 1997.

As provided in 37 C.F.R. §1.98(d), copies of the documents are not being provided since they were previously cited by or submitted to the Patent Office in parent application Serial No. 09/283,160 filed April 1, 1999 and grandparent Serial No. 08/960,079 filed October 24, 1997.

The instant Information Disclosure Statement is being filed in compliance with 37 CFR §1.97(b) prior to the mailing date of the first official action, therefore, no fee is required in connection with its filing. However, the Commissioner is hereby authorized to charge any deficiency or to credit any overpayment to Deposit Account No. 08-1641.

Applicants respectfully request that the listed documents be considered by the Examiner and be made of record in the present application and that an initialed copy of Form PTO-SB/08A be returned in accordance with M.P.E.P. §609.

Respectfully submitted,

7/29/2003
Date

Brian Siritzky
Brian Siritzky
Reg. No. 37,497

Customer ID No. 26633
HELLER EHRMAN WHITE & McAULIFFE LLP
1666 K Street, N.W., Suite 300
Washington, DC 20006
Telephone: (202) 912-2000
Facsimile: (202) 912-2020



Please type a plus sign (+) inside this box



PTO/SB/82 (10-00)

Approved for use through 10/31/2002. OMB 0651-0035

Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE

Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it displays a valid OMB control number.

REVOCATION OF POWER OF ATTORNEY OR AUTHORIZATION OF AGENT	Application Number	09/987,723
	Filing Date	November 15, 2001
	First Named Inventor	Farber
	Group Art Unit	2177
	Examiner Name	Homere, Jean
	Attorney Docket Number	39685-0036

I hereby revoke all previous powers of attorney or authorizations of agent given in the above-identified application:

☒ A Power of Attorney or Authorization of Agent is submitted herewith.

OR

☐ Please change the correspondence address for the above-identified application to:

☐ Customer Number



Place Customer
Number Bar Code
Label here

OR

☐ Firm or
Individual Name

Address

Address

City

Country

State

ZIP

Telephone

Fax

I am the:

☐ Applicant/Inventor.

☒ Assignee of record of undivided part interest.
Certificate under 37 CFR 3.73(b) is enclosed.

SIGNATURE of Applicant or Assignee of Record

Name

KINETECH, INC. by EZRA GOLDMAN, V.P.

Signature

Ezra Gold

Date

July 16, 2003

NOTE: Signatures of all the inventors or assignees of record of the entire interest or their representative(s) are required. Submit multiple forms if more than one signature is required, see below*.

☒ *Total of 2 forms are submitted.

#7
8/1/03
A.W.

RECEIVED

JUL 30 2003

Technology Center 2100



Please type a plus sign (+) inside this box



PTO/SB/82 (10-00)

Approved for use through 10/31/2002. OMB 0651-0035

Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE

Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it displays a valid OMB control number.

REVOCATION OF POWER OF ATTORNEY OR AUTHORIZATION OF AGENT	Application Number	09/987,723
	Filing Date	November 15, 2001
	First Named Inventor	Farber
	Group Art Unit	2177
	Examiner Name	Homere, Jean
	Attorney Docket Number	39685-0036

I hereby revoke all previous powers of attorney or authorizations of agent given in the above-identified application:

☒ A Power of Attorney or Authorization of Agent is submitted herewith.

OR

☐ Please change the correspondence address for the above-identified application to:

☐ Customer Number

Place Customer
Number Bar Code
Label here

OR

<input type="checkbox"/> Firm or Individual Name					
Address					
Address					
City					
Country		State		ZIP	
Telephone		Fax			

I am the:

☐ Applicant/Inventor.

☒ Assignee of record of undivided part interest.
Certificate under 37 CFR 3.73(b) is enclosed.

JUL 30 2003

Technology Center 2100

SIGNATURE of Applicant or Assignee of Record

Name	Eric I. Schwartz, Senior Corporate Counsel, Cable! Wireless Internet Services
Signature	
Date	5/28/03

NOTE: Signatures of all the inventors or assignees of record of the entire interest or their representative(s) are required. Submit multiple forms if more than one signature is required, see below*.

☒ *Total of 2 forms are submitted.



Please type a plus sign (+) inside this box



PTO/SB/81 (02-01)
Approved for use through 10/31/2002. OMB 0651-0035
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE

Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it displays a valid OMB control number.

**POWER OF ATTORNEY OR
AUTHORIZATION OF AGENT**

Application Number	09/987,723
Filing Date	November 15, 2001
First Named Inventor	Farber
Group Art Unit	2177
Examiner Name	Homere, Jean
Attorney Docket Number	39685-0036

I hereby appoint:

☒ Practitioners at Customer Number

26633

OR

☐ Practitioner(s) named below:

Name	Registration Number

Place Customer
Number Bar Code
Label here

RECEIVED

JUL 30 2003

Technology Center 2100

as my/our attorney(s) or agent(s) to prosecute the application identified above, and to transact all business in the Patent and Trademark Office connected therewith.

Please change the correspondence address for the above-identified application to:

☐ The above-mentioned Customer Number.

OR

☒ Practitioners at Customer Number

26633

OR

Place Customer
Number Bar Code
Label here

☐ Firm or
Individual Name

Address

Address

City

State

ZIP

Country

Telephone

Fax

I am the:

☐ Applicant/Inventor.

☒ Assignee of record of an undivided part interest.

Certificate under 37 CFR 3.73(b) is enclosed. (Form PTO/SB/96).

SIGNATURE of Applicant or Assignee of Record

Name	Eric J. Schwartz, Senior Corporate Counsel (able to work)
Signature	<i>[Signature]</i> Internet Services, Inc
Date	5/28/03

NOTE: Signatures of all the inventors or assignees of record of the entire interest or their representative(s) are required. Submit multiple forms if more than one signature is required, see below*.

☒ *Total of 2 forms are submitted.

Please type a plus sign (+) inside this box: ☐

PTO/SB/81 (02-01)

Approved for use through 10/31/2002. OMB 0651-0035

U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE

Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it displays a valid OMB control number.

POWER OF ATTORNEY OR AUTHORIZATION OF AGENT	Application Number	09/987,723
	Filing Date	November 15, 2001
	First Named Inventor	Farber
	Group Art Unit	2177
	Examiner Name	Homere, Jean
	Attorney Docket Number	39685-0036

I hereby appoint:

☒ Practitioners at Customer Number Place Customer Number Bar Code Label here

OR

☐ Practitioner(s) named below:

Name	Registration Number

as my/our attorney(s) or agent(s) to prosecute the application identified above, and to transact all business in the Patent and Trademark Office connected therewith.

Please change the correspondence address for the above-identified application to:

☐ The above-mentioned Customer Number.

OR

☒ Practitioners at Customer Number Place Customer Number Bar Code Label here

OR

☐ Firm or Individual Name

Address

Address

City

State

ZIP

Country

Telephone

Fax

I am the:

☐ Applicant/Inventor.

☒ Assignee of record of an undivided part interest.

Certificate under 37 CFR 3.73(b) is enclosed. (Form PTO/SB/96).

SIGNATURE of Applicant or Assignee of Record

Name	KINETECH, INC. by EZRA GOLDMAN, V.P.
Signature	Ezra Goldman
Date	July 16, 2003

NOTE: Signatures of all the inventors or assignees of record of the entire interest or their representative(s) are required. Submit multiple forms if more than one signature is required, see below.

☒ *Total of 2 forms are submitted.

**STATEMENT UNDER 37 CFR 3.73(b)**Applicant/Patent Owners: Cable & Wireless Internet Services, Inc. and Kinetech, Inc.Application No./Patent No.: 09/987,723Filed/Issue Date: November 15, 2001Entitled: Identifying Data in a Data Processing SystemCable & Wireless Internet Services, Inc., a Corporation

(Name of Assignee)

(Type of Assignee, e.g., corporation, partnership, university, government agency, etc.)

states that it is:

1. ☐ the assignee of the entire right, title, and interest; or
2. ☒ an assignee of an undivided part interest

in the patent application/patent identified above by virtue of either:

- A. ☐ An assignment from the inventor(s) of the patent application/patent identified above. The assignment was recorded in the Patent and Trademark Office at Reel _____, Frame _____, or for which a copy thereof is attached.

OR

- B. ☒ A chain of title from the inventor(s), of the patent application/patent identified above, to the current assignee as shown below:

1. From: InventorsTo: Kinetech, Inc.The document was recorded in the United States Patent and Trademark Office at Reel 012313, Frame 0446, or for which a copy thereof is attached.2. From: Kinetech, Inc.To: Digital Island, Inc.The document was recorded in the United States Patent and Trademark Office at Reel 013306, Frame 0790, or for which a copy thereof is attached.3. From: Digital Island, Inc.To: Cable & Wireless Internet Services, Inc. (Name change)

The document was recorded in the United States Patent and Trademark Office at Reel _____, Frame _____, or for which a copy thereof is attached.

☐ Additional documents in the chain of title are listed on a supplemental sheet.

- ☒ Copies of assignments or other documents in the chain of title are attached.

[NOTE] A separate copy (i.e., the original assignment document or a true copy of the original document) must be submitted to Assignment Division in accordance with 37 CFR Part 3, if the assignment is to be recorded in the records of the USPTO. See MPEP 302.8]

The undersigned (whose title is supplied below) is empowered to sign this statement on behalf of the assignee.

5/28/03

Date

[Signature]
SignatureEric I. Schwartz

Typed or printed name

Senior Corporate Counsel

Title

RECEIVED

JUL 30 2003

Technology Center 2100



PTO/SB/98 (08-00)

Approved for use through 10/11/2002. OMB 0651-0031

U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE

Under the Paperwork Reduction Project, no persons are required to respond to a collection of information unless it displays a valid OMB control number.

STATEMENT UNDER 37 CFR 3.73(b)

Applicant/Patent Owners: Kinetech, Inc. and

Cable & Wireless Internet Services, Inc.

Application No./Patent No.: 09/987,723

Filed/Issue Date: November 15, 2001

Entitled: Identifying Data in a Data Processing System

Kinetech, Inc., a Corporation

(Name of Assignee)

(Type of Assignee, e.g., corporation, partnership, university, government agency, etc.)

states that it is:

1. ☐ the assignee of the entire right, title, and interest; or
2. ☒ an assignee of an undivided part interest

in the patent application/patent identified above by virtue of either:

- A. ☒ An assignment from the inventor(s) of the patent application/patent identified above. The assignment was recorded in the Patent and Trademark Office at Reel 012313, Frame 0446, or for which a copy thereof is attached.

OR

- B. ☐ A chain of title from the inventor(s), of the patent application/patent identified above, to the current assignee as shown below:

1. From: _____ To: _____
The document was recorded in the United States Patent and Trademark Office at Reel _____, Frame _____, or for which a copy thereof is attached.

2. From: _____ To: _____
The document was recorded in the United States Patent and Trademark Office at Reel _____, Frame _____, or for which a copy thereof is attached.

3. From: _____ To: _____
The document was recorded in the United States Patent and Trademark Office at Reel _____, Frame _____, or for which a copy thereof is attached.

☐ Additional documents in the chain of title are listed on a supplemental sheet.

- ☒ Copies of assignments or other documents in the chain of title are attached.

(NOTE: A separate copy (i.e., the original assignment document or a true copy of the original document) must be submitted to Assignment Division in accordance with 37 CFR Part 3, if the assignment is to be recorded in the records of the USPTO. See MPEP 302.8)

The undersigned (whose title is supplied below) is empowered to sign this statement on behalf of the assignee.

July 16, 2003
Date

Ezra Goldman
Signature

EZRA GOLDMAN
Typed or printed name

VICE PRESIDENT
Title

RECEIVED

JUL 30 2003

Technology Center 2100

RECEIPT FROM PTO FOR INDICATED ITEMS

(Do NOT Use for New or Continuing Applications of Any Kind)
 Use 2 postcards for all New Applns. (cont/Div/CIP, too)
Use this sheet when filing CPA

Appln. No: 09/987,723	Atty: Brian Sirtzky
First Inventor: FARBER et al	Date: September 13, 2002
	Matter No: 0000003
	Client No: 018404

ENCLOSED:

- ☐ Response/Amendment ☐ Appendix ☐ Cover Sheet ☐ Cited/Listed Documents
☐ Completion Request for R 53(f)/PCT Nat.

- # No. of Pages Abstract
 # No. of Pages Spec and Claims
 # No. of Numbered Claims Only
 # No. of Sheets of Drawings (Figs

- ☐ 1 Set Formal ☐ 1 Set Informal ☐ Cover Letter

- ☐ Declaration ☒ # of pages
☐ Assignment ☐ Cover Sheet
☐ Small Entity Declaration
☐ Extension Petition (PAT-111)
 # ☐ No. of Priority Documents
☐ IDS Letter ☒ Cited Appln(s) ☐ Foreign Sch Rep/OA
☐ PTO-1449 ☐ Cited Documents
☐ Issue Fee Transmittal Form PTOL-85(b) in duplicate

\$ 40.00 Amount Requested be Charged to our Dep. Acct. No. 03-3975

OTHER:

CORRECTIVE ASSIGNMENT DOCUMENT, ORIGINAL COVERSHEET

Current DUE DATE:

(Submit Singl Copy Only)

RECEIVED

JUL 30 2003

Technology Center 2100

**RECORDATION FORM/COVER SHEET
PATENT APPLICATIONS & PATENTS ONLY**

TO THE ASSISTANT COMMISSIONER OF PATENTS AND TRADEMARKS:

SIR: PLEASE RECORD THE ATTACHED ORIGINAL DOCUMENTS OR COPY THEREOF.

1. NAME OF CONVEYING PARTY(IES) (ASSIGNOR(S))

1. Kinetech, Inc.

3.

5.

7.

ADDITIONAL NAME(S) OF CONVEYING PARTY(IES) ATTACHED? ☐ YES ☒ NO

2. PARTY(IES) (ASSIGNEE(S)) RECEIVING INTEREST:

NAME: Digital Island, Inc.

ADDRESS: 45 Fremont Street, San Francisco, CA 94150

ADDITIONAL NAME(S) & ADDRESS(ES) ATTACHED? ☐ YES ☒ NO**RECEIVED**

JUL 30 2003

Technology Center 2100

3. NATURE OF CONVEYANCE (DOCUMENT):

(Submit herewith only one document for recordation—multiple copies of same Assignment signed by different inventors is one document)☐ ASSIGNMENT OF☐ WHOLE☐ PART INTEREST☐ ORIGINAL☐ FACSIMILE/PHOTOCOPY☐ CHANGE OF NAME☐ VERIFIED TRANSLATION☐ SECURITY ☐ MERGER☒ OTHER: CORRECTIVE DOCUMENT (original at Reel/Frame 012313 / 0205)

EXEC. DATE: 10/4/2000

EXECUTION DATE(S) ON THE DECLARATION IF FILED HERewith: (NOTE: IF DATES ON DECLARATION AND ASSIGNMENT DIFFER SEE ATTY!)

4.5 APPL. NO.(S) OR PAT NO.(S). OTHERS ON ADDITIONAL SHEET(S) attached? ☐ YES ☒ NO

A. PAT. APP. NO.(S) series code/serial no.	M#	1 st INVENTOR if not in item 1	B. PATENT NO(S)	M#	1 st INVENTOR if not in item 1
09/987,723	283002	Farber			

5. Name & Address of Party to Whom Correspondence Concerning Document Should be Mailed:

Pillsbury Winthrop LLP
Intellectual Property Group
P.O. Box 10500 McLean, VA 22102

6. NUMBER INVOLVED:

APPLNS 1 + PATS 0 = TOTAL = 1

7. AMOUNT OF FEE DUE: (Code 581)

ABOVE TOTAL x \$40 = \$40

5.5 ATTY DKT:

P 283002

8. PLEASE CHARGE TO OUR DEPOSIT ACCOUNT

NUMBER: 03-3975

UNDER ORDER NO

018404

0000003

MATTER NO.

CLIENT REF.

dup. sheet not required

CLIENT NO.

MATTER NO.

9. To the best of my knowledge and belief, the foregoing information is true and correct and any attached copy is a true copy of the original document.

10. Total number of pages including this cover sheet, attachments and document (do not file dup. Cover sheet)

3

Signature

Attorney: Brian Sirtzky

Reg. No. 37497

Date: September 13, 2002

Atty/Sec: BS/BS

TEL: (703) 905-2185

FAX: (703) 905-2500

FILE WITH PTO RETURN RECEIPT (PAT-103A)

BOX ASSIGNMENTS

11-23-2001

DO NOT USE FOR TRADEMARKS



101899326

TO THE ASSISTANT COMMISSIONER OF
SIR: PLEASE RECORD THE ATTACHED C. OF.Jc973 U.S. PTO
09/987723
11/16/01

1. NAME OF CONVEYING PARTY(IES) (ASSIGNORS(S)):

1. KINETECH, Inc.

3.
5.
7.2.
4.
6.
8.ADDITIONAL NAME(S) OF CONVEYING PARTY(IES) ATTACHED? ☐ YES ☒ NO

2. PARTY(IES) (ASSIGNEE(S)) RECEIVING INTEREST:

NAME: DIGITAL ISLAND, INC.

ADDRESS: 45 Fremont Street, 12th Floor, San Francisco, CA 94150

ADDITIONAL NAME(S) & ADDRESS(ES) ATTACHED? ☐ YES ☒ NO

3. NATURE OF CONVEYANCE (DOCUMENT):

(Submit herewith only one document for recordation—multiple copies of same Assignment signed by different inventors is one document)☒ ASSIGNMENT OF ☐ WHOLE ☒ PART INTEREST
☐ ORIGINAL ☒ FACSIMILE/PHOTOCOPY
☐ CHANGE OF NAME ☐ VERIFIED TRANSLATION
☐ SECURITY ☐ MERGER ☐ OTHER:

EXEC. DATE: September 27, 2000

EXECUTION DATE(S) ON THE DECLARATION IF FILED HEREWITH: (NOTE: IF DATES ON DECLARATION AND ASSIGNMENT DIFFER SEE ATTY!)

4.5 APPL. NO.(S) OR PAT NO.(S). OTHERS ON ADDITIONAL SHEET(S) attached? ☐ YES ☒ NO

A. PAT. APP. NO.(S)	M#	1 st INVENTOR	B. PATENT NO(S)	M#	1 st INVENTOR
series code/serial no.		if not in item 1			if not in item 1
Cont. of 09/283,160	0283002	FARBER			

5. Name & Address of Party to Whom Correspondence
Concerning Document Should be Mailed:Pillsbury Winthrop LLP
Intellectual Property Group
1600 Tysons Boulevard
McLean, VA 221026. NUMBER INVOLVED:
APPLNS 1 + PATS 0 = TOTAL = 17. AMOUNT OF FEE ENCLOSED: (Code 581)
ABOVE TOTAL x \$40 = \$40

5.5 ATTY DKT:

P 283002

8. IF ABOVE FEE IS MISSING OR INADEQUATE CHARGE
INSUFFICIENCY TO DEPOSIT ACCOUNT NUMBER: 03-3975

UNDER ORDER NO 7018

283002

MATTER NO.

CLIENT REF.

dup. sheet not required

CLIENT NO.

MATTER NO.

9. To the best of my knowledge and belief, the foregoing information is true and correct and any attached copy is a true copy of the original document.

10. Total number of pages including this
cover sheet, attachments and document
(do not file dup. Cover sheet)

2

Signature

Attorney: Brian Sirtzky

Reg. No. 37497

Atty/Sec: BS/jig

TEL: (703) 905-2185

Date: November 15, 2001

FAX: (703) 905-2500

FILE WITH PTO RETURN RECEIPT (PAT-103A)

11/19/2001 AADQF01 00000031 09987723

03 FC:581

40.00 DP

Page 282 of 582

GOOG-1002

ASSIGNMENT

WHEREAS, KINETECH, Inc. a Delaware corporation having offices at 3140 Whisperwoods Court, Northbrook, Illinois 60062 ("KINETECH"), owns, by assignment, all right, title, and interest in U.S. Patent No. 5,978,791, entitled "Data Processing System Using Substantially Unique Identifiers To Identify Data Items, Whereby Identical Data Items Have The Same Identifiers", and any invention claimed therein; and

Delaware

Digital Island, Inc., a California corporation having offices at 45 Fremont Street, 12th Floor, San Francisco CA 94150 ("DI"), desires to own fifty percent of KINETECH's entire right, title, and interest in and to the invention, in all countries throughout the world, and in and to U.S. Letters Patent No. 5,978,791;

NOW THEREFORE, be it known that, for good and valuable consideration, receipt of which is hereby acknowledged, KINETECH hereby sells, assigns, transfers, and sets over to DI, its lawful successors and assigns, FIFTY percentage of KINETECH's entire right, title, and interest in and to U.S. Patent No. 5,978,791, the invention claimed therein, any other patent applications directed to the invention, all counterpart applications, including continuation applications, divisional applications, and all Letters Patent of the United States that may be granted thereon, and all reissues, reexaminations, and extensions thereof; and all rights to claim priority on the basis of such application, and all counterpart applications for Letters Patent that may be filed for the invention in any foreign country and all Letters Patent that may be granted on the invention in any foreign country, and all extensions, renewals, and reissues thereof; and

KINETECH hereby authorizes and requests the Commissioner of Patents and Trademarks of the United States and any official of any foreign country whose duty it is to issue patents on applications as described above, to issue all Letters Patent for this invention to DI, its successors and assigns, in accordance with the terms of this Assignment;

AND, KINETECH HEREBY further covenants that KINETECH has the full right to convey the interest assigned by this Assignment, KINETECH will take all action and execute all documents necessary to perfect the interest assigned hereby, and KINETECH has not executed and will not execute any agreement in conflict with this Assignment;

AND, KINETECH HEREBY further covenants and agrees that KINETECH, through its officers and employees, will, without further consideration, communicate with DI, its successors and assigns, any facts known to KINETECH and its officers and employees respecting the invention and testify in any legal proceeding, sign all lawful papers when called upon to do so, execute and deliver all papers that may be necessary or desirable to perfect the title to the invention in said DI, its successors and assigns, execute all divisional, continuation, and reissue applications, make all rightful oaths, and generally do everything possible to aid DI, its successors and assigns, to obtain and enforce proper patent protection for the invention in the United States and any foreign country, it being understood that any expense incident to the execution of such papers shall be borne by DI, its successors and assigns.

IN TESTIMONY WHEREOF, each party has caused its authorized representative to execute this Assignment.

KINETECH, INC.

By

Name

Title

Date

EZRA GOLDMAN

VICE PRESIDENT

10/4/2000

Digital Island, Inc.

By

Name

Title

Date

HOWARD KASKY

Vice President

10/4/2000

RECORDATION FORM COVER SHEET
PATENT APPLICATIONS & PATENTS ONLY

TO THE ASSISTANT COMMISSIONER OF PATENTS AND TRADEMARKS:
 SIR: PLEASE RECORD THE ATTACHED ORIGINAL DOCUMENTS OR COPY THEREOF.

1. NAME OF CONVEYING PARTY(IES) (ASSIGNORS(S)):

1. Digital Island, Inc.

3.

5.

7.

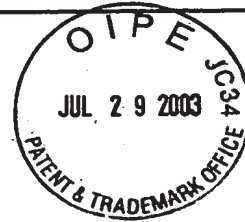
2.

4.

6.

8.

ADDITIONAL NAME(S) OF CONVEYING PARTY(IES) ATTACHED? ☐ YES ☐ NO



2. PARTY(IES) (ASSIGNEE(S)) RECEIVING INTEREST:

NAME: Cable & Wireless Internet Services, Inc.

ADDRESS: 45 Fremont Street, San Francisco, California 94150

ADDITIONAL NAME(S) & ADDRESS(ES) ATTACHED? ☐ YES ☒ NO

3. NATURE OF CONVEYANCE (DOCUMENT):

(Submit herewith only one document for recordation—multiple copies of same Assignment signed by different inventors is one document)

- ☐ ASSIGNMENT OF ☐ WHOLE ☐ PART INTEREST
☐ ORIGINAL ☐ FACSIMILE/PHOTOCOPY
☒ CHANGE OF NAME ☐ VERIFIED TRANSLATION
☐ SECURITY ☐ MERGER ☐ OTHER:

EXEC. DATE: February 12, 2002

EXECUTION DATE(S) ON THE DECLARATION IF FILED HERewith: (NOTE: IF DATES ON DECLARATION AND ASSIGNMENT DIFFER SEE ATTY!)

4.5 APPL. NO.(S) OR PAT NO.(S). OTHERS ON ADDITIONAL SHEET(S) attached? ☐ YES ☒ NO

A. PAT. APP. NO.(S) series code/serial no.	M#	1 st INVENTOR if not in item 1	B. PATENT NO.(S)	M#	1 st INVENTOR if not in item 1
09/987,723	283002	Farber			

5. Name & Address of Party to Whom Correspondence Concerning Document Should be Mailed:

Pillsbury Winthrop LLP
 Intellectual Property Group
 P.O. Box 10500 McLean, VA 22102

6. NUMBER INVOLVED:

APPLNS 1 + PATS 0 = TOTAL = 1

7. AMOUNT OF FEE DUE: (Code 581)
 ABOVE TOTAL x \$40 = \$40

5.5 ATTY DKT:

P 283002

8. PLEASE CHARGE TO OUR DEPOSIT ACCOUNT
 NUMBER: 03-3975

MATTER NO.

CLIENT REF.

UNDER ORDER NO

018404

0000003

dup. sheet not required

CLIENT NO.

MATTER NO.

9. To the best of my knowledge and belief, the foregoing information is true and correct and any attached copy is a true copy of the original document.

B. Siritzky
 Signature

10. Total number of pages including this cover sheet, attachments and document (do not file dup. Cover sheet)

4

Attorney: Brian Siritzky

Reg. No. 37497

Atty/Sec: BS/BS

TEL: (703) 905-2185

Date: September 13, 2002

FAX: (703) 905-2500

FILE WITH PTO RETURN RECEIPT (PAT-103A)

Delaware

PAGE 1

The First State

I, HARRIET SMITH WINDSOR, SECRETARY OF STATE OF THE STATE OF DELAWARE, DO HEREBY CERTIFY THE ATTACHED IS A TRUE AND CORRECT COPY OF THE CERTIFICATE OF AMENDMENT OF "DIGITAL ISLAND, INC.", CHANGING ITS NAME FROM "DIGITAL ISLAND, INC." TO "CABLE & WIRELESS INTERNET SERVICES, INC.", FILED IN THIS OFFICE ON THE TWELFTH DAY OF FEBRUARY, A.D. 2002, AT 9 O'CLOCK A.M.

Draft Only



Harriet Smith Windsor
Harriet Smith Windsor, Secretary of State

3016554 8100

AUTHENTICATION: 1936267

STATE OF DELAWARE
SECRETARY OF STATE
DIVISION OF CORPORATIONS
FILED 09:00 AM 02/18/2008
020098043 - 3016554

**CERTIFICATE OF AMENDMENT
OF THE
CERTIFICATE OF INCORPORATION
OF
DIGITAL ISLAND, INC.,
a Delaware corporation**

The undersigned William M. Austin and Howard Lasky hereby certify that:

ONE: They are the duly elected and acting Chief Executive Officer and Secretary, respectively, of said corporation.

TWO: The Amended and Restated Certificate of Incorporation of said corporation, last filed on July 2, 1999 shall be amended as set forth in this Certificate of Amendment.

THREE: Article First of the Certificate of Incorporation of said corporation is hereby amended to read in its entirety as follows:

"FIRST. The name of the corporation is Cable & Wireless Internet Services, Inc. (hereinafter sometimes called the "Corporation")."

FOUR: This amendment of the Certificate of Incorporation herein certified has been duly adopted in accordance with the provisions of Sections 228 and 242 of the General Corporation Law of the State of Delaware.

The undersigned further declare under penalty of perjury that the matters set forth in this Certificate of Amendment are true and correct of their own knowledge.

IN WITNESS WHEREOF, the undersigned have executed this Certificate of Amendment at San Francisco, California, on this 12th day of February, 2002.



William M. Austin, Chief Executive Officer



Howard E. Lasky, Secretary

Please type a plus sign (+) inside this box → ☐

Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it displays a valid OMB control number.



TRANSMITTAL FORM

(to be used for all correspondence after initial filing)

TRANSMITTAL FORM (to be used for all correspondence after initial filing)	Application Number	09/987,723	
	Filing Date	November 15, 2001	
	First Named Inventor	David A. FARBER et al.	
	Group Art Unit	2177	
	Examiner Name	Homere, J.	
Total Number of Pages in This Submission		Attorney Docket Number	39685-0036

ENCLOSURES (check all that apply)

<input type="checkbox"/> Fee Transmittal Form <input type="checkbox"/> Fee Attached <input type="checkbox"/> Amendment / Response <input type="checkbox"/> After Final <input type="checkbox"/> Affidavits/declaration(s) <input type="checkbox"/> Extension of Time Request <input type="checkbox"/> Express Abandonment Request <input checked="" type="checkbox"/> Information Disclosure Statement <input type="checkbox"/> Certified Copy of Priority Document(s) <input type="checkbox"/> Response to Missing Parts/Incomplete Application <input type="checkbox"/> Response to Missing Parts under 37 CFR 1.52 or 1.53	<input checked="" type="checkbox"/> Assignment Papers (for an Application) <input type="checkbox"/> Drawing(s) <input type="checkbox"/> Licensing-related Papers <input type="checkbox"/> Petition <input type="checkbox"/> Petition to Convert to a Provisional Application <input checked="" type="checkbox"/> Power of Attorney, Revocation Change of Correspondence Address <input type="checkbox"/> Terminal Disclaimer <input type="checkbox"/> Request for Refund <input type="checkbox"/> CD, Number of CD(s) _____	<input type="checkbox"/> After Allowance Communication to Group <input type="checkbox"/> Appeal Communication to Board of Appeals and Interferences <input type="checkbox"/> Appeal Communication to Group (Appeal Notice, Brief, Reply Brief) <input type="checkbox"/> Proprietary Information <input type="checkbox"/> Status Letter <input checked="" type="checkbox"/> Other Enclosure(s) (please identify below): Revocation of Power of Attorney and Power of Attorney (from Kinetech, Inc. and Cable & Wireless Internet Services)
RECEIVED JUL 30 2003 Technology Center 2100		

SIGNATURE OF APPLICANT, ATTORNEY, OR AGENT

Firm or Individual name	HELLER EHRMAN WHITE & MCAULIFFE, Customer No. 26633
Signature	<i>[Signature]</i> 37497
Date	7/29/2003

CERTIFICATE OF MAILING

I hereby certify that this correspondence is being deposited with the United States Postal Service as first class mail in an envelope addressed to: Assistant Commissioner for Patents, Washington, D.C. 20231 on this date: <input type="text"/>			
Typed or printed name			
Signature		Date	

Burden Hour Statement: This form is estimated to take 0.2 hours to complete. Time will vary depending upon the needs of the individual case. Any comments on the amount of time you are required to complete this form should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, Washington, DC 20231. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. SEND TO: Assistant Commissioner for Patents, Washington, DC 20231.



IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Docket No.: 39685-0036

In re patent application of:
David A. FARBER et al.

Confirmation No.: 7771

Application No.: 09/987,723

Art Unit: 2177

Filed: November 15, 2001

Examiner: Homere, J.

For: IDENTIFYING AND REQUESTING DATA IN NETWORK USING IDENTIFIERS
WHICH ARE BASED ON CONTENTS OF DATA

CHANGE OF ADDRESS

RECEIVED

JUL 30 2003

Technology Center 2100

Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:

Applicants respectfully request that the official correspondence address for the above captioned application be changed to Brian Siritzky and the attorneys of Heller Ehrman White & McAuliffe LLP, located at 1666 K Street, N.W., Suite 300, Washington, D.C. 20006. Please direct all written communications to Brian Siritzky at the above mentioned address, and direct all telephone communications to (202) 912-2000.

Respectfully submitted,

Date: July 29, 2003

HELLER EHRMAN WHITE &
MCAULIFFE
1666 K Street, N.W., Suite 300
Washington, DC 20006
Telephone: (202) 912-2000
Facsimile: (202) 912-2020

By B Siritzky

Brian Siritzky
Attorney for Applicant
Registration No. 37,497
Customer No. 26633



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER OF PATENTS AND TRADEMARKS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/987,723	11/15/2001	David A. Farber	P 283002 TN-Cont-2	7771

7590

05/14/2003

PILLSBURY WINTHROP
1600 TYSONS BOULEVARD
MCLEAN, VA 22102

EXAMINER

HOMERE, JEAN RAYMOND

ART UNIT

PAPER NUMBER

2177

DATE MAILED: 05/14/2003

Please find below and/or attached an Office communication concerning this application or proceeding.

Draft Only

Office Action Summary

Applicati n No.

09/987,723

Applicant(s)

FARBER ET AL.

Examiner

Jean R. Homere

Art Unit

2177

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 1 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133).
- Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

- 1) ☒ Responsive to communication(s) filed on 18 November 2002.
- 2a) ☐ This action is **FINAL**. 2b) ☒ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

- 4) ☐ Claim(s) 46-49 and 54-65 is/are pending in the application.
- 4a) Of the above claim(s) _____ is/are withdrawn from consideration.
- 5) ☐ Claim(s) _____ is/are allowed.
- 6) ☐ Claim(s) _____ is/are rejected.
- 7) ☐ Claim(s) _____ is/are objected to.
- 8) ☒ Claim(s) 46-49 and 54-65 are subject to restriction and/or election requirement.

Application Papers

- 9) ☐ The specification is objected to by the Examiner.
- 10) ☐ The drawing(s) filed on _____ is/are: a) ☐ accepted or b) ☐ objected to by the Examiner.
- Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).
- 11) ☐ The proposed drawing correction filed on _____ is: a) ☐ approved b) ☐ disapproved by the Examiner.
- If approved, corrected drawings are required in reply to this Office action.
- 12) ☐ The oath or declaration is objected to by the Examiner.

Priority under 35 U.S.C. §§ 119 and 120

- 13) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) ☐ All b) ☐ Some * c) ☐ None of:
- ☐ Certified copies of the priority documents have been received.
 - ☐ Certified copies of the priority documents have been received in Application No. _____.
 - ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).
- * See the attached detailed Office action for a list of the certified copies not received.
- 14) ☐ Acknowledgment is made of a claim for domestic priority under 35 U.S.C. § 119(e) (to a provisional application).
- a) ☐ The translation of the foreign language provisional application has been received.
- 15) ☐ Acknowledgment is made of a claim for domestic priority under 35 U.S.C. §§ 120 and/or 121.

Attachment(s)

- | | |
|--|---|
| 1) <input type="checkbox"/> Notice of References Cited (PTO-892) | 4) <input type="checkbox"/> Interview Summary (PTO-413) Paper No(s). _____ |
| 2) <input type="checkbox"/> Notice of Draftsperson's Patent Drawing Review (PTO-948) | 5) <input type="checkbox"/> Notice of Informal Patent Application (PTO-152) |
| 3) <input type="checkbox"/> Information Disclosure Statement(s) (PTO-1449) Paper No(s) _____ | 6) <input type="checkbox"/> Other: |

Art Unit: 2177

DETAILED ACTION

Election/Restrictions

1. Restriction to one of the following inventions is required under 35 U.S.C. 121:

- I. Claims 54-65 are drawn to method(s) for delivering contents of a file, classified in class 707, subclass 10.
- II. Claims 46-49 are drawn to method(s) for deleting, removing and synchronizing data items, classified in class 707, subclass 201.

The inventions are distinct, each from the other because of the following reasons:

2. Inventions I and II are related as sub-combinations disclosed as usable together in a single combination. The sub-combinations are distinct from each other if they are shown to be separately usable. In the instant case, invention I has separate utility such as delivering the contents of a file over a network while invention II is useable for deleting, removing or synchronizing data files on a network. See M.P.E.P. § 806.05(d).

3. These inventions are distinct for the reasons given above and have acquired a separate status in the art as shown by their different classification. Restriction for examination purposes as indicated is proper.

4. These inventions are distinct for the reasons given above and the search required for Group I is not required for Group II. Restriction for examination purposes as indicated is proper.

Art Unit: 2177

5. Because these inventions are distinct for the reasons given above and have acquired a separate status in the art because of their recognized divergent subject matter, restriction for examination purposes as indicated is proper.

6. A telephone call was made to Brian Siritzky on 05/08/03 to request an oral election to the above restriction requirement, but did not result in an election being made.

7. Applicant is advised that the reply to this requirement to be complete must include an election of the invention to be examined even though the requirement be traversed (37 CFR 1.143).

8. Applicant is reminded that upon the cancellation of claims to a non-elected invention, the inventorship must be amended in compliance with 37 CFR 1.48(b) if one or more of the currently named inventors is no longer an inventor of at least one claim remaining in the application. Any amendment of inventorship must be accompanied by a request under 37 CFR 1.48(b) and by the fee required under 37 CFR 1.17(i).

9. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Jean R. Homere whose telephone number is (703)-308-6647. The examiner can normally be reached on Monday-Friday from 09:30 a.m.-6:00 p.m.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, John Breene, can be reached on Monday-Friday from 8:00 a.m. to 3:30 p.m. at (703)-305-9790.

Art Unit: 2177

Any response to this action should be mailed to: Commissioner of Patents and Trademarks Washington, D.C. 20231, **or faxed to:** (703) 746-7239, (for formal communications intended for entry), **or faxed to:** (703) 746-7238, (for after final communications intended for entry), **Or:** (703) 746-7240 (for informal or draft communications, please label "PROPOSED" or "DRAFT"). Hand-delivered responses should be brought to Crystal Park II, 2121 Crystal Drive, Arlington, VA., Sixth Floor (Receptionist).

Any inquiry of a general nature or relating to the status of this application should be directed to the Group receptionist whose telephone number is (703) 305-3900.

Jean R. Homere
Primary Examiner
Art Unit 2177

Jrh
May 13, 2003



UNITED STATES DEPARTMENT OF COMMERCE
Patent and Trademark Office

ASSISTANT SECRETARY AND COMMISSIONER
OF PATENTS AND TRADEMARKS
Washington, D.C. 20231

#5
11/20/02
AW.

CHANGE OF ADDRESS/POWER OF ATTORNEY

FILE LOCATION 2FC1 SERIAL NUMBER 09987723 PATENT NUMBER

THE CORRESPONDENCE ADDRESS HAS BEEN CHANGED TO CUSTOMER # 909

THE PRACTITIONERS OF RECORD HAVE BEEN CHANGED TO CUSTOMER # 909

THE FEE ADDRESS HAS BEEN CHANGED TO CUSTOMER # 909

ON 04/23/02 THE ADDRESS OF RECORD FOR CUSTOMER NUMBER 909 IS:

PILLSBURY WINTHROP LLP
1600 TYSONS BOULEVARD
MCLEAN VA 22102

AND THE PRACTITIONERS OF RECORD FOR CUSTOMER NUMBER 909 ARE:

16773	17698	18221	20508	24238	25323	27248	28429	28458	28872
30793	31204	31361	31542	31678	32243	32995	34393	35861	35959
36004	36239	36787	37087	38821	38825	40862	41646	41835	

RECEIVED

NOV 18 2002

Technology Center 2100

PTO INSTRUCTIONS: PLEASE TAKE THE FOLLOWING ACTION WHEN THE CORRESPONDENCE ADDRESS HAS BEEN CHANGED TO CUSTOMER NUMBER: RECORD, ON THE NEXT AVAILABLE CONTENTS LINE OF THE FILE JACKET, 'ADDRESS CHANGE TO CUSTOMER NUMBER'. LINE THROUGH THE OLD ADDRESS ON THE FILE JACKET LABEL AND ENTER ONLY THE 'CUSTOMER NUMBER' AS THE NEW ADDRESS. FILE THIS LETTER IN THE FILE JACKET. WHEN ABOVE CHANGES ARE ONLY TO FEE ADDRESS AND/OR PRACTITIONERS OF RECORD, FILE LETTER IN THE FILE JACKET. THIS FILE IS ASSIGNED TO GAU 2171.

RECEIPT FROM PTO FOR INDICATED ITEMS

(Do NOT Use for New or Continuing Applications of Any Kind)
Use 2 postcards for all New Applns. (cont/Div/CIP, too)
Use this sheet when filing CPA

Appln. No: 09/987,723	Atty: Brian Siritzky
First Inventor: FARBER et al.	Date: September 13, 2002
	Matter No: 0000003
	Client No: 018404

ENCLOSED:

- ☐ Response/Amendment ☐ Appendix ☐ Cover Sheet ☐ Cited/Listed Documents
☐ Completion Request for R 53(f)/PCT Nat.

No. of Pages Abstract

No. of Pages Spec and Claims

No. of Numbered Claims Only# No. of Sheets of Drawings (Figs)

☐ 1 Set Formal ☐ 1 Set Informal ☐ Cover Letter

☐ Declaration☒ # of pages☒ Assignment☒ Cover Sheet☐ Small Entity Declaration☐ Extension Petition (PAT-111)# ☐ No. of Priority Documents☐ IDS Letter ☐ Cited Appln(s) ☐ Foreign Sch Rep/OA☐ PTO-1449 ☐ Cited Documents☐ Issue Fee Transmittal Form PTOL-85(b) in duplicate\$ Amount Requested be Charged to our Dep. Acct. No. 03-3975OTHER: **Current DUE DATE:**

(Submit Single Copy Only)



**IN THE UNITED STATES PATENT AND TRADEMARK
OFFICE
REQUEST FOR FILING
(RULE 53(b)(1))**

Page 1 of 4

jc973 U.S. PTO

jc973 U.S. PTO
09/987723

11/15/01

For Design or Utility Applications

(DO NOT USE FOR CIPs)

Rule 53(b)(1) PATENT APPLICATION:

☒ Continuation)
) application under 37 CFR 1.53(b)(1)
☐ Divisional)
 application under 37 CFR 1.53(b)(1)
 of pending prior application of

Group Art Unit: 2771

Examiner: Jean HOMERE

Inventor(s): Farber et al.

Parent Appln. No.:	09	283,160
	Series Code ↑	Serial No. ↑

Atty. Dkt. P 283002

TN-Cont-2

New M#

Client Ref

Parent Filed: April 1, 1999

This Appln. Filed: November 15, 2001

Title: Identifiying Data in a Data Processing System

Hon. Commissioner of Patents
Washington, DC 20231

Date: November 15, 2001

(Parent Matter No. 252465)

Sir:

To effect the above-requested filing today:

1. **Attached** is a copy (**which must be filed**) of the prior application, including:

☒ Abstract

☒ Specification and claims (94 pages) **(must be attached)**

Drawings (**must be attached if originally filed**): 31 sheet(s)/set: ☐ 1 set informal;

☒ Formal of size

☒ A4

□ 11"

1A. Always X one box, only:

(1) ☒ Copy of Signed declaration or oath as originally filed in prior application attached

(2) ☐ NO declaration or fee is enclosed; therefore, this is a filing under Rule 53(f).

2. ☐ This application is hereby filed by less than all of the inventors named in the prior application. Petition is hereby made requesting deletion as inventor(s) of the following who is/are **not** inventor(s) of the invention being claimed in this application (DELETE THE FOLLOWING INVENTOR(S)):

1. _____
3. _____
5. _____
7. _____

2. _____

4. _____

6. _____

8. _____

2.5 THE INVENTOR(S) FOR THIS NEW APPLICATION IS(ARE):

1. David A. FARBER
3. _____
5. _____
7. _____

2. Ronald D. LACHMAN
4. _____
6. _____
8. _____

3. The entire disclosure of the prior application is considered as being part of the disclosure of the accompanying application and is hereby incorporated therein by reference thereto.

4. ☐ Priority is claimed under 35 U.S.C. 119/365 based on filing in _____ of _____ (country)
- | | Application No. | Filing Date | | Application No. | Filing Date |
|-----|-----------------|-------------|-----|-----------------|-------------|
| (1) | _____ | _____ | (2) | _____ | _____ |
| (3) | _____ | _____ | (4) | _____ | _____ |
| (5) | _____ | _____ | (6) | _____ | _____ |
- a. ☐ _____ (No.) Certified copy/copies attached.
- b. ☐ Certified copy/copies previously filed on _____ in _____
U.S. Application No. _____ / _____, filed on _____
series code ↑ serial no.
- c. ☐ Certified copy/copies filed during International stage of PCT/ _____ / _____
4. (a) ☐ Domestic priority is claimed from _____ / _____, filed _____
PCT/ _____
- (b) ☐ Benefit is claimed of Provisional Application No. 60/_____, filed _____.
5. ☒ Prior application is assigned to Kinetech, Inc. and Digital Island, Inc. respectively
by assignment recorded April 1, 1995/ October 5, 2000 Reel 9873/11273 Frame 0463/0164.
(Date)
6. ☒ Attached is the following number of Assignments (including original and all later successive ones by different assignors): 2 and respective **new** Cover Sheets. (Do **NOT** file old cover sheets.)

(Assignments in parent **must be refiled** with new Cover Sheets in this continuing application if you want it/them recorded against the continuing application.)

Please return the recorded Assignment to the undersigned.
7. ☒ The power of attorney in the prior application is to Dale S. Lazar, Reg. No. 28,872
(Name and Reg. No.)
whose current address is as in item 8 below.
- a. ☒ Recognize as associate attorney Brian Siritzky, Reg. No. 37,497
(Name, Reg. No. and Address)
8. **Address all future communications to Intellectual Property Group of Pillsbury Winthrop LLP, 1600 Tysons Boulevard, McLean, VA 22102**
9. ☒ **Amend the specification** by inserting before the first line the sentence:--This is a
☒ continuation ☐ division of Application No. 09/283,160, filed April 1, 1999
series code ↑ serial no.
which is a continuation of 08/960,079, filed October 24, 1997, now abandoned, which is --
a continuation of 08/425,160, filed April 11, 1995, now abandoned.
9. (a) ☐ **Amend the specification** by inserting before the first line: --This application claims the benefit of
Provisional Application No. 60/_____, filed _____.
10. **Small Entity Status** ☒ is **Not** claimed ☐ is claimed (**pre-filing confirmation required**)

(No.) Small Entity Statement(s) (**not** essential since 9/8/00) were/are:
☐ filed in above prior application
☐ attached.
11. Petition to extend the life of the above prior application to at least the date hereof
(one box) ☐ is being concurrently filed in that prior application (Use Form PAT-111).
(must be) ☐ was previously filed in that prior application (Check length of prior extension).
(X'd) ☒ is not necessary for copendency (**Double check** before X'ing this box).

12. ☒ **INFORMATION DISCLOSURE STATEMENT:** Attached is Form PTO-1449 listing all of the documents cited by Applicant and the PTO in the parent application(s) relied upon under 35 USC 120 and referenced in item 9 above. Per Rule 98(d) copies of those documents are not required now. Please consider those documents and advise that they have been considered in this new application as by returning a copy of the enclosed Form PTO-1449 with the Examiner's initials in the left column per MPEP 609. .
13. ☐ Attached is a Rule 103(a) Petition to Suspend Action.
14. ☒ **PRELIMINARY AMENDMENT to be entered before fee calculation:** (Do not make amendments here except for correction of improper multiple dependencies or cancellation of whole claims or multiple dependencies for purpose of reducing the filing fee per MPEP §§ 506 and 607; do not cancel all claims).

Please cancel claims 1-45 and 50-53.

FILING FEE

THE FOLLOWING FILING FEE IS BASED ON

-->>>>> CLAIMS AS FILED AND CHANGED BY PRELIMINARY AMENDMENT IN ITEM 14 <<<<<<<

NOTE: If box 1A2 is X'd, do not pay fees,
but leave lines 15-22 and 27-32 blank.

PTO: PLEASE NOTE CLAIM CANCELLATIONS IF BOX 14 ABOVE IS X'D.

				Large/Small Entity		Fee Code
15. Basic Filing Fee Design Application				\$330/\$165		106/26
16. Basic Filing Fee Utility Application				\$740/\$370	+740	101/201
17. Total Effective Claims	4	minus 20 =	0	x \$18/\$9	+0	103/203
18. Independent Claims	3	minus 3 =	0	x \$84/\$42	+0	102/202
19. If <u>any</u> proper multiple dependent claim (ignore improper) is present,				\$280/\$140	+0	104/204
20.				Subtotal =	\$740	
21. If "petition" box 13 above is X'd, add petition fee. \$130					+0	122
21A. If box 6 above is X'd, add Assignment recording fee \$ 40					+80	581
22.				TOTAL FILING FEE ATTACHED =	\$820	

(carry forward to Item 31)

23. ☐ ATTACHED:
24. ☒ Preliminary Amendment attached (to be entered after assigning Appln. No.)
25. ☐ See **NONPUBLICATION REQUEST** under Rule 213(a) attached (Pat-258)

26.

**ADDITIONAL FEE CALCULATION FOR
PRELIMINARY AMENDMENT
PER BOXES 24/25**

	Claims remaining after amendment	Highest number previously paid for	Present Extra	Additional Fee	File Code
			Large/Small Entity		
27.	Total Effective Claims <u>*16</u>	minus ** <u>20</u>	= <u>0</u> x \$18/\$9	= <u>\$ 0</u>	(103/203)
28.	Independent Claims <u>*14</u>	minus *** <u>3</u>	= <u>11</u> x \$84/\$42	= + <u>924</u>	(102/202)
29.	If amendment enters proper multiple dependent claim(s) into this application for the first time, add (per application)			\$280/\$140 + <u>0</u>	(104/204)
30.	ADDITIONAL FEE			<u>\$ 0</u>	
31.	plus FEE from item 22 on page 3			+ <u>820</u>	
32.	<u>TOTAL FEE ATTACHED</u>			<u>\$ 1744</u>	

33. *If the entry in this space is less than a entry in the next space, the "Present Extra" result is "0"

34. **If the "Highest number previously paid for" (see item 17 above) is less than 20, write "20" in this space

35. If the "Highest number previously paid for" (see item 18 above) is less than 3, write "3" in this space

Our Deposit Account No. 03-3975

Our Order No. 007018

0283002

C#

M#

CHARGE STATEMENT: Upon the filing of a Declaration pursuant to Rule 60(b) or 60(d), the Commissioner is hereby authorized to charge any fee specifically authorized hereafter, or any missing or insufficient fee(s) filed, or asserted to be filed, or which should have been filed herewith or concerning any paper filed hereafter, and which may be required under Rules 16-18 (~~missing or insufficient fee only~~) now or hereafter relative to this application and the resulting Official document under Rule 20, or credit any overpayment, to our Account/Order Nos. shown above for which purpose a duplicate copy of this sheet is attached.

This CHARGE STATEMENT does not authorize charge of the issue fee until/unless an issue fee transmittal form is filed.

**Pillsbury Winthrop LLP
Intellectual Property Group**

1600 Tysons Boulevard
McLean, VA 22102
Tel: (703) 905-2000

By Atty: Brian Sritzky

Reg. No. 37497

Sig: 

Fax: (703) 905-2500
Tel: (703) 905-2185

BS/BS
Atty./Sec.

NOTE No. 1: File this Request in duplicate with 2 postcard receipts (PAT-103) & attachments

NOTE No. 2: Is extension in parent necessary for copendency? **DOUBLE CHECK** Item 11 above.
If yes, printout Pat-111 and head it in parent.

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re PATENT APPLICATION of

FARBER et al.

Group Art Unit: 2177

Examiner: Homere, Jean R.

Continuation of Appln. No. 09/283,160

Filed: November 15, 2001

For: **IDENTIFYING AND REQUESTING DATA IN NETWORK USING IDENTIFIERS
WHICH ARE BASED ON CONTENTS OF DATA (AS AMENDED)**

* * * * *

November 15, 2001

PRELIMINARY AMENDMENT

Hon. Commissioner of Patents
and Trademarks
Washington, D.C. 20231

Sir:

Please amend this application as follows:

IN THE TITLE:

Please replace the title with the following:

**--IDENTIFYING AND REQUESTING DATA IN NETWORK USING IDENTIFIERS WHICH
ARE BASED ON CONTENTS OF DATA—**

IN THE SPECIFICATION:

At the top of the first page, just under the title, insert:

This is a continuation of Application No. 09/283,160, filed April 1, 1999, which is a continuation of 08/960,079, filed October 24, 1997, now abandoned, which is a continuation of 08/425,160, filed April 11, 1995, now abandoned.

IN THE CLAIMS:

Please add the following new claims:

--54. (New) A content delivery method, in a system in which a plurality of files are distributed across a network of servers, at least some of the files being cached versions of files from a source server distinct from the network of servers, the content delivery method comprising:

for a particular file, determining a name using a given function of the data, said data being the data which comprises the contents of the particular file; and

in response to a request for the particular file, the request including at least the name of the particular file, causing the particular file to be provided from a given one of the servers of the network of servers.

55. (New) A content delivery method, in a system in which a plurality of files are distributed across a network of servers, at least some of the files being cached versions of files from a source server distinct from the servers in the network, wherein data in a file in the system may represent a digital message, a digital image, a video signal or an audio signal, the content delivery method comprising:

determining a name for a particular file, the name being determined using an MD5 function of the data, said data being the data which comprises the contents of the particular file; and

in response to a request for the particular file, the request including at least the name of the particular file, providing the particular data file from a given one of the servers of the network of servers, said providing being based on the determined name.

56. (New) A content delivery method, in a system in which a plurality of files are distributed across a network of servers, wherein some processors in the network communicate with each other using a TCP/IP communication protocol, the content delivery method comprising:

for a particular file, the contents of said file representing a digital image, determining a name for the particular file, wherein the name is determined using a given function of the data which comprises the contents of the particular file; and

in response to a request for the particular file, the request including at least the name of the particular file, providing the particular file from a given one of the servers of the network of servers.

57. (New) A content delivery method comprising:

causing a plurality of files to be distributed across a network of servers, at least some of the files being cached versions of files from a source server which is distinct from the network of servers;

for a particular file, determining a name, the name being determined using a given function of the data, said data used by said function being data which comprises the contents of the particular file; and

in response to a request for the particular file, the request including at least the name of the particular file, causing the particular file to be provided from a given one of the servers of the network of servers.

58. (New) A content delivery method, in a system in which a plurality of files are distributed across a network of servers, at least some of the files being cached versions of files from a source server which is distinct from the network of servers, the content delivery method comprising:

determining a name for a particular file, the name being determined using a given function of the data which comprises the contents of the particular file; and

in response to a request for the particular file, the request including at least the name of the particular file, providing the particular file from a given one of the servers of the network of servers,

wherein the contents of the particular file may represent a digital message, a digital image, a video signal or an audio signal.

59. (New) A method, in a network comprising a plurality of processors, some of the processors functioning as servers and some of the processors functioning as clients, wherein some processors in the network communicate with each other using a TCP/IP communication protocol, wherein a key is required to identify a file on the network, the method comprising:

storing some files on a first server in the network and storing copies of some of the files from the first server on a set of cache servers distinct from the first server;

for a particular file, determining a different cache key from an ordinarily used cache key for the file, the different cache key being determined using a message function MD5 of the data, wherein said data comprises the contents of the particular file; and

responsive to a client request for the particular file, the request including the different cache key for the file, causing the particular file to be provided to the client,

wherein the contents of the file may represent: a page in memory, a digital message, a digital image, a video signal or an audio signal.

60. (New) A content delivery method comprising:

distributing a set of files from a first server across a network of servers distinct from the first server;

applying an MD5 function to the contents of a particular file to obtain a True Name for the file;

in response to a request for the particular file, the request including at least the True Name of the particular file, causing the particular file to be provided from a given one of the servers of the network of servers, wherein the request for the particular file is resolved based on a measure of availability of at least one of the servers.

61. (New) A method as in claim 60 wherein the measure of availability for a server is based on at least one of:

- (a) a measurement of bandwidth to the server;
- (b) a measurement of a cost of a connection to the server, and
- (c) a measurement of reliability of a connection to the server.

62. (New) A content delivery method comprising:

distributing a plurality of files across a network of servers, at least some of the files being cached versions of files from a source server distinct from the servers in the network;

for a particular file, determining a True Name using a given function of the data which comprises the contents of the particular file;

obtaining a request for the particular file, the request including at least the True Name of the particular file; and

responsive to the request, causing the particular file to be provided from one of the servers of the network of servers.

63. (New) A content delivery method, comprising:

distributing files across a network of servers;

for a particular file having a contextual name specifying a location in the network at which the file may be located, determining another name for the particular file, the other name including a data identifier determined using a given function of the data, where said data used by the given function comprises the contents of the particular file;

obtaining a request for the particular file, the request including the contextual name and the other name of the particular file,

responsive to the request, providing the particular file from one of the servers of the network of servers, said providing being based on the other name of the particular item.

64. (New) A content delivery method, comprising:

distributing a set of files across a network of servers;

for a particular file representing a digital image, the file having a contextual name specifying a location in the network at which the file may be located, determining another name for the particular file, the other name including a True Name for the file which was determined using a message digest function of the data, where said data used by the given function comprises the contents of the particular file;

obtaining a request for the particular file, the request including the contextual name and the True Name of the particular file; and

responsive to the request, providing the particular file from one of the servers of the network of servers, said providing being based on the True Name of the particular item.

65. (New) A method comprising:

applying an MD5 function to the contents of an image file containing data representing a digital image to obtain a True Name for the file;

distributing copies of the image file from a first server across a network of servers distinct from the first server;

obtaining a request for the image file, the request including at least the True Name of the file; and

responsive to the request, causing a copy of the image file to be provided from one of the servers of the network of servers.

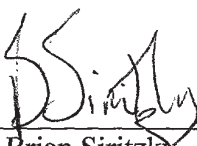
IN THE ABSTRACT OF THE DISCLOSURE

Please replace the Abstract of the Disclosure with the attached new Abstract of the Disclosure.

REMARKS

By this Amendment, new claims 54 to 65 have been added. In addition, a new Abstract has been provided and the title has been replaced. No new matter has been added by these amendments, and approval of these amendments to the specification, title and abstract is respectfully requested.

Respectfully submitted,

By 
Brian Siritzky
Reg. No. 37497
Tel. No.: (703) 905-2185
Fax No.: (703) 905-2500

1600 Tysons Boulevard,
McLean, Virginia 22102
(703) 905-2000

30238692v1

Draft Only

Abstract of the Disclosure

In a system in which a set of data items are distributed across a network of servers, at least some of the data items being cached versions of data items from a source server, a content delivery method includes determining a data identifier for a particular data item, the data identifier being determined using a given function of the data comprising the particular data item; and responsive to a request for the particular data item, the request including at least the data identifier of the particular data item, providing the particular data item from a given one of the servers of the network of servers. The request for the particular data item may be resolved based on a measure of availability of at least one of the servers, where the measure of availability may be a measurement of bandwidth to the server; a measurement of a cost of a connection to the server, and/or a measurement of a reliability of a connection to the server. The function used to determine the identifier may be a message digest function or a hash function.

APPLICATION UNDER UNITED STATES PATENT LAWS

Atty. Dkt. No. PW 283002
(M#)

Invention: IDENTIFYING DATA IN A DATA PROCESSING SYSTEM

Inventor (s): David A. FARBER
Ronald D. LACHMAN

Pillsbury Winthrop LLP
Intellectual Property Group
1600 Tysons Boulevard
McLean, VA 22102
Attorneys
Telephone: (703) 905-2000

This is a:

- ☒ Provisional Application
- ☐ Regular Utility Application
- ☒ Continuing Application
☒ The contents of the parent are incorporated by reference
- ☐ PCT National Phase Application
- ☐ Design Application
- ☐ Reissue Application
- ☐ Plant Application
- ☐ Substitute Specification
Sub. Spec Filed _____
in App. No. _____ / _____
- ☐ Marked up Specification re
Sub. Spec. filed _____
In App. No. _____ / _____

SPECIFICATION

IDENTIFYING DATA IN A DATA PROCESSING SYSTEMBACKGROUND OF THE INVENTION1. Field of the invention

5 This invention relates to data processing systems and, more particularly, to data processing systems wherein data items are identified by substantially unique identifiers which depend on all of the data in the data items and only on the data in the data items.

10 2. Background of the Invention

Data processing (DP) systems, computers, networks of computers, or the like, typically offer users and programs various ways to identify the data in the systems.

15 Users typically identify data in the data processing system by giving the data some form of name. For example, a typical operating system (OS) on a computer provides a file system in which data items are named by alphanumeric identifiers. Programs typically
20 identify data in the data processing system using a location or address. For example, a program may identify a record in a file or database by using a record number which serves to locate that record.

In all but the most primitive operating
25 systems, users and programs are able to create and use collections of named data items, these collections themselves being named by identifiers. These named collections can then, themselves, be made part of other named collections. For example, an OS may provide
30 mechanisms to group files (data items) into directories (collections). These directories can then, themselves be made part of other directories. A data item may thus be identified relative to these nested directories using a

sequence of names, or a so-called pathname, which defines a path through the directories to a particular data item (file or directory).

5 As another example, a database management system may group data records (data items) into tables and then group these tables into database files (collections). The complete address of any data record can then be specified using the database file name, the table name, and the record number of that data record.

10 Other examples of identifying data items include: identifying files in a network file system, identifying objects in an object-oriented database, identifying images in an image database, and identifying articles in a text database.

15 In general, the terms "data" and "data item" as used herein refer to sequences of bits. Thus a data item may be the contents of a file, a portion of a file, a page in memory, an object in an object-oriented program, a digital message, a digital scanned image, a part of a
20 video or audio signal, or any other entity which can be represented by a sequence of bits. The term "data processing" herein refers to the processing of data items, and is sometimes dependent on the type of data item being processed. For example, a data processor for
25 a digital image may differ from a data processor for an audio signal.

In all of the prior data processing systems the names or identifiers provided to identify data items (the data items being files, directories, records in the
30 database, objects in object-oriented programming, locations in memory or on a physical device, or the like) are always defined relative to a specific context. For instance, the file identified by a particular file name can only be determined when the directory containing the
35 file (the context) is known. The file identified by a pathname can be determined only when the file system (context) is known. Similarly, the addresses in a

process address space, the keys in a database table, or domain names on a global computer network such as the Internet are meaningful only because they are specified relative to a context.

5 In prior art systems for identifying data items there is no direct relationship between the data names and the data item. The same data name in two different contexts may refer to different data items, and two different data names in the same context may refer to the
10 same data item.

 In addition, because there is no correlation between a data name and the data it refers to, there is no *a priori* way to confirm that a given data item is in fact the one named by a data name. For instance, in a DP
15 system, if one processor requests that another processor deliver a data item with a given data name, the requesting processor cannot, in general, verify that the data delivered is the correct data (given only the name). Therefore it may require further processing, typically on
20 the part of the requestor, to verify that the data item it has obtained is, in fact, the item it requested.

 A common operation in a DP system is adding a new data item to the system. When a new data item is added to the system, a name can be assigned to it only by
25 updating the context in which names are defined. Thus such systems require a centralized mechanism for the management of names. Such a mechanism is required even in a multi-processing system when data items are created and identified at separate processors in distinct
30 locations, and in which there is no other need for communication when data items are added.

 In many data processing systems or environments, data items are transferred between different locations in the system. These locations may
35 be processors in the data processing system, storage devices, memory, or the like. For example, one processor may obtain a data item from another processor or from an

external storage device, such as a floppy disk, and may incorporate that data item into its system (using the name provided with that data item).

5 However, when a processor (or some location) obtains a data item from another location in the DP system, it is possible that this obtained data item is already present in the system (either at the location of the processor or at some other location accessible by the processor) and therefore a duplicate of the data item is
10 created. This situation is common in a network data processing environment where proprietary software products are installed from floppy disks onto several processors sharing a common file server. In these systems, it is often the case that the same product will
15 be installed on several systems, so that several copies of each file will reside on the common file server.

 In some data processing systems in which several processors are connected in a network, one system is designated as a cache server to maintain master copies
20 of data items, and other systems are designated as cache clients to copy local copies of the master data items into a local cache on an as-needed basis. Before using a cached item, a cache client must either reload the cached item, be informed of changes to the cached item, or
25 confirm that the master item corresponding to the cached item has not changed. In other words, a cache client must synchronize its data items with those on the cache server. This synchronization may involve reloading data items onto the cache client. The need to keep the cache
30 synchronized or reload it adds significant overhead to existing caching mechanisms.

 In view of the above and other problems with prior art systems, it is therefore desirable to have a mechanism which allows each processor in a multiprocessor
35 system to determine a common and substantially unique identifier for a data item, using only the data in the data item and not relying on any sort of context.

It is further desirable to have a mechanism for reducing multiple copies of data items in a data processing system and to have a mechanism which enables the identification of identical data items so as to reduce multiple copies. It is further desirable to determine whether two instances of a data item are in fact the same data item, and to perform various other systems' functions and applications on data items without relying on any context information or properties of the data item.

It is also desirable to provide such a mechanism in such a way as to make it transparent to users of the data processing system, and it is desirable that a single mechanism be used to address each of the problems described above.

SUMMARY OF THE INVENTION

This invention provides, in a data processing system, a method and apparatus for identifying a data item in the system, where the identity of the data item depends on all of the data in the data item and only on the data in the data item. Thus the identity of a data item is independent of its name, origin, location, address, or other information not derivable directly from the data, and depends only on the data itself.

This invention further provides an apparatus and a method for determining whether a particular data item is present in the system or at a location in the system, by examining only the data identities of a plurality of data items.

Using the method or apparatus of the present invention, the efficiency and integrity of a data processing system can be improved. The present invention improves the design and operation of a data storage system, file system, relational database, object-oriented database, or the like that stores a plurality of data items, by making possible or improving the design and

operation of at least some or all of the following features:

5 the system stores at most one copy of any data item at a given location, even when multiple data names in the system refer to the same contents;

 the system avoids copying data from source to destination locations when the destination locations already have the data;

10 the system provides transparent access to any data item by reference only to its identity and independent of its present location, whether it be local, remote, or offline;

15 the system caches data items from a server, so that only the most recently accessed data items need be retained;

 when the system is being used to cache data items, problems of maintaining cache consistency are avoided;

20 the system maintains a desired level of redundancy of data items in a network of servers, to protect against failure by ensuring that multiple copies of the data items are present at different locations in the system;

25 the system automatically archives data items as they are created or modified;

 the system provides the size, age, and location of groups of data items in order to decide whether they can be safely removed from a local file system;

30 the system can efficiently record and preserve any collection of data items;

 the system can efficiently make a copy of any collection of data items, to support a version control mechanism for groups of the data items;

35 the system can publish data items, allowing other, possibly anonymous, systems in a network to gain access to the data items and to rely on the availability of the data items;

the system can maintain a local inventory of all the data items located on a given removable medium, such as a diskette or CD-ROM, the inventory is independent of other properties of the data items such as their name, location, and date of creation;

the system allows closely related sets of data items, such as matching or corresponding directories on disconnected computers, to be periodically resynchronized with one another;

the system can verify that data retrieved from another location is the desired or requested data, using only the data identifier used to retrieve the data;

the system can prove possession of specific data items by content without disclosing the content of the data items, for purposes of later legal verification and to provide anonymity;

the system tracks possession of specific data items according to content by owner, independent of the name, date, or other properties of the data item, and tracks the uses of specific data items and files by content for accounting purposes.

Other objects, features, and characteristics of the present invention as well as the methods of operation and functions of the related elements of structure, and the combination of parts and economies of manufacture, will become more apparent upon consideration of the following description and the appended claims with reference to the accompanying drawings, all of which form a part of this specification.

BRIEF DESCRIPTION OF THE DRAWINGS

FIGURE 1 depicts a typical data processing system in which a preferred embodiment of the present invention operates;

FIGURE 2 depicts a hierarchy of data items stored at any location in such a data processing system;

FIGURES 3-9 depict data structures used to implement an embodiment of the present invention; and FIGURES 10(a)-28 are flow charts depicting operation of various aspects of the present invention.

5 DETAILED DESCRIPTION OF THE PRESENTLY PREFERRED
 EXEMPLARY EMBODIMENTS

 An embodiment of the present invention is now described with reference to a typical data processing system 100, which, with reference to FIGURE 1, includes one or more processors (or computers) 102 and various storage devices 104 connected in some way, for example by a bus 106.

 Each processor 102 includes a CPU 108, a memory 110 and one or more local storage devices 112. The CPU 108, memory 110, and local storage device 112 may be internally connected, for example by a bus 114. Each processor 102 may also include other devices (not shown), such as a keyboard, a display, a printer, and the like.

 In a data processing system 100, wherein more than one processor 102 is used, that is, in a multiprocessor system, the processors may be in one of various relationships. For example, two processors 102 may be in a client/server, client/client, or a server/server relationship. These inter-processor relationships may be dynamic, changing depending on particular situations and functions. Thus, a particular processor 102 may change its relationship to other processors as needed, essentially setting up a peer-to-peer relationship with other processors. In a peer-to-peer relationship, sometimes a particular processor 102 acts as a client processor, whereas at other times the same processor acts as a server processor. In other words, there is no hierarchy imposed on or required of processors 102.

 In a multiprocessor system, the processors 102 may be homogeneous or heterogeneous. Further, in a

multiprocessor data processing system 100, some or all of the processors 102 may be disconnected from the network of processors for periods of time. Such disconnection may be part of the normal operation of the system 100 or it may be because a particular processor 102 is in need of repair.

Within a data processing system 100, the data may be organized to form a hierarchy of data storage elements, wherein lower level data storage elements are combined to form higher level elements. This hierarchy can consist of, for example, processors, file systems, regions, directories, data files, segments, and the like. For example, with reference to FIGURE 2, the data items on a particular processor 102 may be organized or structured as a file system 116 which comprises regions 117, each of which comprises directories 118, each of which can contain other directories 118 or files 120. Each file 120 being made up of one or more data segments 122.

In a typical data processing system, some or all of these elements can be named by users given certain implementation specific naming conventions, the name (or pathname) of an element being relative to a context. In the context of a data processing system 100, a pathname is fully specified by a processor name, a filesystem name, a sequence of zero or more directory names identifying nested directories, and a final file name. (Usually the lowest level elements, in this case segments 122, cannot be named by users.)

In other words, a file system 116 is a collection of directories 118. A directory 118 is a collection of named files 120 -- both data files 120 and other directory files 118. A file 120 is a named data item which is either a data file (which may be simple or compound) or a directory file 118. A simple file 120 consists of a single data segment 122. A compound file 120 consists of a sequence of data segments 122. A data

0993723 11501
TEST 2222222

segment 122 is a fixed sequence of bytes. An important property of any data segment is its size, the number of bytes in the sequence.

5 A single processor 102 may access one or more file systems 116, and a single storage device 104 may contain one or more file systems 116, or portions of a file system 116. For instance, a file system 116 may span several storage devices 104.

10 In order to implement controls in a file system, file system 116 may be divided into distinct regions, where each region is a unit of management and control. A region consists of a given directory 118 and is identified by the pathname (user defined) of the directory.

15 In the following, the term "location", with respect to a data processing system 100, refers to any of a particular processor 102 in the system, a memory of a particular processor, a storage device, a removable storage medium (such as a floppy disk or compact disk),
20 or any other physical location in the system. The term "local" with respect to a particular processor 102 refers to the memory and storage devices of that particular processor.

25 In the following, the terms "True Name", "data identity" and "data identifier" refer to the substantially unique data identifier for a particular data item. The term "True File" refers to the actual file, segment, or data item identified by a True Name.

30 A file system for a data processing system 100 is now described which is intended to work with an existing operating system by augmenting some of the operating system's file management system codes. The embodiment provided relies on the standard file management primitives for actually storing to and
35 retrieving data items from disk, but uses the mechanisms of the present invention to reference and access those data items.

109037723 - 11001

The processes and mechanisms (services) provided in this embodiment are grouped into the following categories: primitive mechanisms, operating system mechanisms, remote mechanisms, background mechanisms, and extended mechanisms.

Primitive mechanisms provide fundamental capabilities used to support other mechanisms. The following primitive mechanisms are described:

1. Calculate True Name;
2. Assimilate Data Item;
3. New True File;
4. Get True Name from Path;
5. Link path to True Name;
6. Realize True File from Location;
7. Locate Remote File;
8. Make True File Local;
9. Create Scratch File;
10. Freeze Directory;
11. Expand Frozen Directory;
12. Delete True File;
13. Process Audit File Entry;
14. Begin Grooming;
15. Select For Removal; and
16. End Grooming.

Operating system mechanisms provide typical familiar file system mechanisms, while maintaining the data structures required to offer the mechanisms of the present invention. Operating system mechanisms are designed to augment existing operating systems, and in this way to make the present invention compatible with, and generally transparent to, existing applications. The following operating system mechanisms are described:

1. Open File;
2. Close File;
3. Read File;
4. Write File;
5. Delete File or Directory;

6. Copy File or Directory;
7. Move File or Directory;
8. Get File Status; and
9. Get Files in Directory.

5 Remote mechanisms are used by the operating system in responding to requests from other processors. These mechanisms enable the capabilities of the present invention in a peer-to-peer network mode of operation. The following remote mechanisms are described:

- 10 1. Locate True File;
2. Reserve True File;
3. Request True File;
4. Retire True File;
5. Cancel Reservation;
- 15 6. Acquire True File;
7. Lock Cache;
8. Update Cache; and
9. Check Expiration Date.

Background mechanisms are intended to run occasionally and at a low priority. These provide automated management capabilities with respect to the present invention. The following background mechanisms are described:

- 25 1. Mirror True File;
2. Groom Region;
3. Check for Expired Links; and
4. Verify Region; and
5. Groom Source List.

Extended mechanisms run within application programs over the operating system. These mechanisms provide solutions to specific problems and applications. The following extended mechanisms are described:

- 30 1. Inventory Existing Directory;
2. Inventory Removable, Read-only Files;
- 35 3. Synchronize directories;
4. Publish Region;
5. Retire Directory;

6. Realize Directory at location;
7. Verify True File;
8. Track for accounting purposes; and
9. Track for licensing purposes.

5 The file system herein described maintains
sufficient information to provide a variety of mechanisms
not ordinarily offered by an operating system, some of
which are listed and described here. Various processing
performed by this embodiment of the present invention
10 will now be described in greater detail.

In some embodiments, some files 120 in a data
processing system 100 do not have True Names because they
have been recently received or created or modified, and
thus their True Names have not yet been computed. A file
15 that does not yet have a True Name is called a scratch
file. The process of assigning a True Name to a file is
referred to as assimilation, and is described later.
Note that a scratch file may have a user provided name.

Some of the processing performed by the present
20 invention can take place in a background mode or on a
delayed or as-needed basis. This background processing
is used to determine information that is not immediately
required by the system or which may never be required.
As an example, in some cases a scratch file is being
25 changed at a rate greater than the rate at which it is
useful to determine its True Name. In these cases,
determining the True Name of the file can be postponed or
performed in the background.

Data Structures

30 The following data structures, stored in memory
110 of one of more processors 102 are used to implement
the mechanisms described herein. The data structures can
be local to each processor 102 of the system 100, or they
can reside on only some of the processors 102.

The data structures described are assumed to reside on individual peer processors 102 in the data processing system 100. However, they can also be shared by placing them on a remote, shared file server (for instance, in a local area network of machines). In order to accommodate sharing data structures, it is necessary that the processors accessing the shared database use the appropriate locking techniques to ensure that changes to the shared database do not interfere with one another but are appropriately serialized. These locking techniques are well understood by ordinarily skilled programmers of distributed applications.

It is sometimes desirable to allow some regions to be local to a particular processor 102 and other regions to be shared among processors 102. (Recall that a region is a unit of file system management and control consisting of a given directory identified by the pathname of the directory.) In the case of local and shared regions, there would be both local and shared versions of each data structure. Simple changes to the processes described below must be made to ensure that appropriate data structures are selected for a given operation.

The local directory extensions (LDE) table 124 is a data structure which provides information about files 120 and directories 118 in the data processing system 100. The local directory extensions table 124 is indexed by a pathname or contextual name (that is, a user provided name) of a file and includes the True Name for most files. The information in local directory extension table 124 is in addition to that provided by the native file system of the operating system.

The True File registry (TFR) 126 is a data store for listing actual data items which have True Names, both files 120 and segments 122. When such data items occur in the True File registry 126 they are known as True Files. True Files are identified in True File

registry 126 by their True Names or identities. The table True File registry 126 also stores location, dependency, and migration information about True Files.

5 The region table (RT) 128 defines areas in the network storage which are to be managed separately. Region table 128 defines the rules for access to and migration of files 120 among various regions with the local file system 116 and remote peer file systems.

10 The source table (ST) 130 is a list of the sources of True Files other than the current True File registry 126. The source table 130 includes removable volumes and remote processors.

15 The audit file (AF) 132 is a list of records indicating changes to be made in local or remote files, these changes to be processed in background.

20 The accounting log (AL) 134 is a log of file transactions used to create accounting information in a manner which preserves the identity of files being tracked independent of their name or location.

The license table (LT) 136 is a table identifying files, which may only be used by licensed users, in a manner independent of their name or location, and the users licensed to use them.

Detailed Descriptions of the Data Structures

25 The following table summarizes the fields of an local directory extensions table entry, as illustrated by record 138 in FIGURE 3.

Field	Description
Region ID	identifies the region in which this file is contained.
Pathname	the user provided name or contextual name of the file or directory, relative to the region in which it occurs.
True Name	the computed True Name or identity of the file or directory. This True Name is not always up to date, and it is set to a special value when a file is modified and is later recomputed in the background.

5

10

Field	Description
Type	indicates whether the file is a data file or a directory.
Scratch File ID	the physical location of the file in the file system, when no True Name has been calculated for the file. As noted above, such a file is called a scratch file.
Time of last access	the last access time to this file. If this file is a directory, this is the last access time to any file in the directory.
Time of last modification	the time of last change of this file. If this file is a directory, this is the last modification time of any file in the directory.
Safe flag	indicates that this file (and, if this file is a directory, all of its subordinate files) have been backed up on some other system, and it is therefore safe to remove them.
Lock flag	indicates whether a file is locked, that is, it is being modified by the local processor or a remote processor. Only one processor may modify a file at a time.
Size	the full size of this directory (including all subordinate files), if all files in it were fully expanded and duplicated. For a file that is not a directory this is the size of the actual True File.
Owner	the identity of the user who owns this file, for accounting and license tracking purposes.

15

Each record of the True File registry 126 has the fields shown in the True File registry record 140 in FIGURE 4. The True File registry 126 consists of the database described in the table below as well as the actual True Files identified by the True File IDs below.

20

Field	Description
True Name	computed True Name or identity of the file.

Field	Description
Compressed File ID	compressed version of the True File may be stored instead of, or in addition to, an uncompressed version. This field provides the identity of the actual representation of the compressed version of the file.
Grooming delete count	tentative count of how many references have been selected for deletion during a grooming operation.
Time of last access	most recent date and time the content of this file was accessed.
Expiration	date and time after which this file may be deleted by this server.
Dependent processors	processor IDs of other processors which contain references to this True File.
Source IDs	source ID(s) of zero or more sources from which this file or data item may be retrieved.
True File ID	identity or disk location of the actual physical representation of the file or file segment. It is sufficient to use a filename in the registration directory of the underlying operating system. The True File ID is absent if the actual file is not currently present at the current location.
Use count	number of other records on this processor which identify this True File.

A region table 128, specified by a directory pathname, records storage policies which allow files in the file system to be stored, accessed and migrated in different ways. Storage policies are programmed in a configurable way using a set of rules described below.

Each region table record 142 of region table 128 includes the fields described in the following table (with reference to FIGURE 5):

Field	Description
Region ID	internally used identifier for this region.
Region file system	file system on the local processor of which this region is a part.
Region pathname	a pathname relative to the region file system which defines the location of this region. The region consists of all files and directories subordinate to this pathname, except those in a region subordinate to this region.
Mirror processor(s)	zero or more identifiers of processors which are to keep mirror or archival copies of all files in the current region. Multiple mirror processors can be defined to form a mirror group.
Mirror duplication count	number of copies of each file in this region that should be retained in a mirror group.
Region status	specifies whether this region is local to a single processor 102, shared by several processors 102 (if, for instance, it resides on a shared file server), or managed by a remote processor.
Policy	<p>the migration policy to apply to this region. A single region might participate in several policies. The policies are as follows (parameters in brackets are specified as part of the policy):</p> <ul style="list-style-type: none"> region is a cached version from [processor ID]; region is a member of a mirror set defined by [processor ID]. region is to be archived on [processor ID]. region is to be backed up locally, by placing new copies in [region ID]. region is read only and may not be changed. region is published and expires on [date]. Files in this region should be compressed.

10

A source table 130 identifies a source location for True Files. The source table 130 is also used to

identify client processors making reservations on the current processor. Each source record 144 of the source table 130 includes the fields summarized in the following table, with reference to FIGURE 6:

5	Field	Description
	source ID	internal identifier used to identify a particular source.
	source type	type of source location: Removable Storage Volume Local Region Cache Server Mirror Group Server Cooperative Server Publishing Server Client
10	source rights	includes information about the rights of this processor, such as whether it can ask the local processor to store data items for it.
	source availability	measurement of the bandwidth, cost, and reliability of the connection to this source of True Files. The availability is used to select from among several possible sources.
15	source location	information on how the local processor is to access the source. This may be, for example, the name of a removable storage volume, or the processor ID and region path of a region on a remote processor.

The audit file 132 is a table of events ordered by timestamp, each record 146 in audit file 132 including the fields summarized in the following table (with reference to FIGURE 7):

20	Field	Description
	Original Name	path of the file in question.
	Operation	whether the file was created, read, written, copied or deleted.
	Type	specifies whether the source is a file or a directory.

Field	Description
Processor ID	ID of the remote processor generating this event (if not local).
Timestamp	time and date file was closed (required only for accessed/modified files).
Pathname	Name of the file (required only for rename).
True Name	computed True Name of the file. This is used by remote systems to mirror changes to the directory and is filled in during background processing.

Each record 148 of the accounting log 134 records an event which may later be used to provide information for billing mechanisms. Each accounting log entry record 148 includes at least the information summarized in the following table, with reference to FIGURE 8:

Field	Description
date of entry	date and time of this log entry.
type of entry	Entry types include create file, delete file, and transmit file.
True Name	True Name of data item in question.
owner	identity of the user responsible for this action.

Each record 150 of the license table 136 records a relationship between a licensable data item and the user licensed to have access to it. Each license table record 150 includes the information summarized in the following table, with reference to FIGURE 9:

Field	Description
True Name	True Name of a data item subject to license validation.

Field	Description
licensee	identity of a user authorized to have access to this object.

Various other data structures are employed on some or all of the processors 102 in the data processing system 100. Each processor 102 has a global freeze lock (GFL) 152 (FIGURE 1), which is used to prevent synchronization errors when a directory is frozen or copied. Any processor 102 may include a special archive directory (SAD) 154 into which directories may be copied for the purposes of archival. Any processor 102 may include a special media directory (SMD) 156, into which the directories of removable volumes are stored to form a media inventory. Each processor has a grooming lock 158, which is set during a grooming operation. During this period the grooming delete count of True File registry entries 140 is active, and no True Files should be deleted until grooming is complete. While grooming is in effect, grooming information includes a table of pathnames selected for deletion, and keeps track of the amount of space that would be freed if all of the files were deleted.

Primitive Mechanisms

The first of the mechanisms provided by the present invention, primitive mechanisms, are now described. The mechanisms described here depend on underlying data management mechanisms to create, copy, read, and delete data items in the True File registry 126, as identified by a True File ID. This support may be provided by an underlying operating system or disk storage manager.

The following primitive mechanisms are described:

1. Calculate True Name;
2. Assimilate Data Item;

3. New True File;
4. Get True Name from Path;
5. Link Path to True Name;
6. Realize True File from Location;
7. Locate Remote File;
8. Make True File Local;
9. Create Scratch File;
10. Freeze Directory;
11. Expand Frozen Directory;
12. Delete True File;
13. Process Audit File Entry;
14. Begin Grooming;
15. Select For Removal; and
16. End Grooming.

15 1. Calculate True Name

A True Name is computed using a function, MD, which reduces a data block B of arbitrary length to a relatively small, fixed size identifier, the True Name of the data block, such that the True Name of the data block is virtually guaranteed to represent the data block B and only data block B.

The function MD must have the following properties:

1. The domain of the function MD is the set of all data items. The range of the function MD is the set of True Names.
2. The function MD must take a data item of arbitrary length and reduce it to an integer value in the range 0 to N-1, where N is the cardinality of the set of True Names. That is, for an arbitrary length data block B, $0 \leq MD(B) < N$.
3. The results of MD(B) must be evenly and randomly distributed over the range of N, in such a way that simple or regular

changes to B are virtually guaranteed to produce a different value of MD(B).

4. It must be computationally difficult to find a different value B' such that MD(B)=MD(B').

5. The function MD(B) must be efficiently computed.

A family of functions with the above properties are the so-called message digest functions, which are used in digital security systems as techniques for authentication of data. These functions (or algorithms) include MD4, MD5, and SHA.

In the presently preferred embodiments, either MD5 or SHA is employed as the basis for the computation of True Names. Whichever of these two message digest functions is employed, that same function must be employed on a system-wide basis.

It is impossible to define a function having a unique output for each possible input when the number of elements in the range of the function is smaller than the number of elements in its domain. However, a crucial observation is that the actual data items that will be encountered in the operation of any system embodying this invention form a very sparse subset of all the possible inputs.

A colliding set of data items is defined as a set wherein, for one or more pairs x and y in the set, MD(x) = MD(y). Since a function conforming to the requirements for MD must evenly and randomly distribute its outputs, it is possible, by making the range of the function large enough, to make the probability arbitrarily small that actual inputs encountered in the operation of an embodiment of this invention will form a colliding set.

To roughly quantify the probability of a collision, assume that there are no more than 2^{30} storage devices in the world, and that each storage device has an

average of at most 2^{20} different data items. Then there are at most 2^{50} data items in the world. If the outputs of MD range between 0 and 2^{128} , it can be demonstrated that the probability of a collision is approximately 1 in 2^{29} . Details on the derivation of these probability values are found, for example, in P. Flajolet and A.M. Odlyzko, "Random Mapping Statistics," *Lecture Notes in Computer Science 434: Advances in Cryptology -- Eurocrypt '89 Proceedings*, Springer-Verlag, pp. 329-354.

Note that for some less preferred embodiments of the present invention, lower probabilities of uniqueness may be acceptable, depending on the types of applications and mechanisms used. In some embodiments it may also be useful to have more than one level of True Names, with some of the True Names having different degrees of uniqueness. If such a scheme is implemented, it is necessary to ensure that less unique True Names are not propagated in the system.

While the invention is described herein using only the True Name of a data item as the identifier for the data item, other preferred embodiments use tagged, typed, categorized or classified data items and use a combination of both the True Name and the tag, type, category or class of the data item as an identifier. Examples of such categorizations are files, directories, and segments; executable files and data files, and the like. Examples of classes are classes of objects in an object-oriented system. In such a system, a lower degree of True Name uniqueness is acceptable over the entire universe of data items, as long as sufficient uniqueness is provided per category of data items. This is because the tags provide an additional level of uniqueness.

A mechanism for calculating a True Name given a data item is now described, with reference to FIGURES 10(a) and 10(b).

A simple data item is a data item whose size is less than a particular given size (which must be defined

in each particular implementation of the invention). To determine the True Name of a simple data item, with reference to FIGURE 10(a), first compute the MD function (described above) on the given simple data item (Step S212). Then append to the resulting 128 bits, the byte length modulo 32 of the data item (Step S214). The resulting 160-bit value is the True Name of the simple data item.

A compound data item is one whose size is greater than the particular given size of a simple data item. To determine the True Name of an arbitrary (simple or compound) data item, with reference to FIGURE 10(b), first determine if the data item is a simple or a compound data item (Step S216). If the data item is a simple data item, then compute its True Name in step S218 (using steps S212 and S214 described above), otherwise partition the data item into segments (Step S220) and assimilate each segment (Step S222) (the primitive mechanism, Assimilate a Data Item, is described below), computing the True Name of the segment. Then create an indirect block consisting of the computed segment True Names (Step S224). An indirect block is a data item which consists of the sequence of True Names of the segments. Then, in step S226, assimilate the indirect block and compute its True Name. Finally, replace the final thirty-two (32) bits of the resulting True Name (that is, the length of the indirect block) by the length modulo 32 of the compound data item (Step S228). The result is the True Name of the compound data item.

Note that the compound data item may be so large that the indirect block of segment True Names is itself a compound data item. In this case the mechanism is invoked recursively until only simple data items are being processed.

Both the use of segments and the attachment of a length to the True Name are not strictly required in a system using the present invention, but are currently

considered desirable features in the preferred embodiment.

2. Assimilate Data Item

5 A mechanism for assimilating a data item (scratch file or segment) into a file system, given the scratch file ID of the data item, is now described with reference to FIGURE 11. The purpose of this mechanism is to add a given data item to the True File registry 126. If the data item already exists in the True File registry
10 126, this will be discovered and used during this process, and the duplicate will be eliminated.

Thereby the system stores at most one copy of any data item or file by content, even when multiple names refer to the same content.

15 First, determine the True Name of the data item corresponding to the given scratch File ID using the Calculate True Name primitive mechanism (Step S230). Next, look for an entry for the True Name in the True File registry 126 (Step S232) and determine whether a
20 True Name entry, record 140, exists in the True File registry 126. If the entry record includes a corresponding True File ID or compressed File ID (Step S237), delete the file with the scratch File ID (Step S238). Otherwise store the given True File ID in the
25 entry record (step S239).

If it is determined (in step S232) that no True Name entry exists in the True File registry 126, then, in Step S236, create a new entry in the True File registry 126 for this True Name. Set the True Name of the entry
30 to the calculated True Name, set the use count for the new entry to one, store the given True File ID in the entry and set the other fields of the entry as appropriate.

Because this procedure may take some time to compute, it is intended to run in background after a file has ceased to change. In the meantime, the file is considered an unassimilated scratch file.

5 3. New True File

 The New True File process is invoked when processing the audit file 132, some time after a True File has been assimilated (using the Assimilate Data Item primitive mechanism). Given a local directory extensions
10 table entry record 138 in the local directory extensions table 124, the New True File process can provide the following steps (with reference to FIGURE 12), depending on how the local processor is configured:

 First, in step S238, examine the local
15 directory extensions table entry record 138 to determine whether the file is locked by a cache server. If the file is locked, then add the ID of the cache server to the dependent processor list of the True File registry table 126, and then send a message to the cache server to
20 update the cache of the current processor using the Update Cache remote mechanism (Step 242).

 If desired, compress the True File (Step S246), and, if desired, mirror the True File using the Mirror True File background mechanism (Step S248).

25 4. Get True Name from Path

 The True Name of a file can be used to identify a file by contents, to confirm that a file matches its original contents, or to compare two files. The mechanism to get a True Name given the pathname of a file
30 is now described with reference to FIGURE 13.

 First, search the local directory extensions table 124 for the entry record 138 with the given pathname (Step S250). If the pathname is not found, this process fails and no True Name corresponding to the given
35 pathname exists. Next, determine whether the local

0987723 "11501

directory extensions table entry record 138 includes a True Name (Step S252), and if so, the mechanism's task is complete. Otherwise, determine whether the local directory extensions table entry record 138 identifies a directory (Step S254), and if so, freeze the directory (Step S256) (the primitive mechanism Freeze Directory is described below).

Otherwise, in step S258, assimilate the file (using the Assimilate Data Item primitive mechanism) defined by the File ID field to generate its True Name and store its True Name in the local directory extensions entry record. Then return the True Name identified by the local directory extensions table 124.

5. Link Path to True Name

The mechanism to link a path to a True Name provides a way of creating a new directory entry record identifying an existing, assimilated file. This basic process may be used to copy, move, and rename files without a need to copy their contents. The mechanism to link a path to a True Name is now described with reference to FIGURE 14.

First, if desired, confirm that the True Name exists locally by searching for it in the True Name registry or local directory extensions table 135 (Step S260). Most uses of this mechanism will require this form of validation. Next, search for the path in the local directory extensions table 135 (Step S262). Confirm that the directory containing the file named in the path already exists (Step S264). If the named file itself exists, delete the File using the Delete True File operating system mechanism (see below) (Step S268).

Then, create an entry record in the local directory extensions with the specified path (Step S270) and update the entry record and other data structures as follows: fill in the True Name field of the entry with the specified True Name; increment the use count for the

True File registry entry record 140 of the corresponding
True Name; note whether the entry is a directory by
reading the True File to see if it contains a tag (magic
number) indicating that it represents a frozen directory
5 (see also the description of the Freeze Directory
primitive mechanism regarding the tag); and compute and
set the other fields of the local directory extensions
appropriately. For instance, search the region table 128
to identify the region of the path, and set the time of
10 last access and time of last modification to the current
time.

6. Realize True File from Location

This mechanism is used to try to make a local
copy of a True File, given its True Name and the name of
15 a source location (processor or media) that may contain
the True File. This mechanism is now described with
reference to FIGURE 15.

First, in step S272, determine whether the
location specified is a processor. If it is determined
20 that the location specified is a processor, then send a
Request True File message (using the Request True File
remote mechanism) to the remote processor and wait for a
response (Step S274). If a negative response is received
or no response is received after a timeout period, this
25 mechanism fails. If a positive response is received,
enter the True File returned in the True File registry
126 (Step S276). (If the file received was compressed,
enter the True File ID in the compressed File ID field.)

If, on the other hand, it is determined in step
30 S272 that the location specified is not a processor,
then, if necessary, request the user or operator to mount
the indicated volume (Step S278). Then (Step S280) find
the indicated file on the given volume and assimilate the
file using the Assimilate Data Item primitive mechanism.
35 If the volume does not contain a True File registry 126,
search the media inventory to find the path of the file

on the volume. If no such file can be found, this mechanism fails.

At this point, whether or not the location is determined (in step S272) to be a processor, if desired, verify the True File (in step S282).

7. Locate Remote File

This mechanism allows a processor to locate a file or data item from a remote source of True Files, when a specific source is unknown or unavailable. A client processor system may ask one of several or many sources whether it can supply a data object with a given True Name. The steps to perform this mechanism are as follows (with reference to FIGURE 16).

The client processor 102 uses the source table 145 to select one or more source processors (Step S284). If no source processor can be found, the mechanism fails. Next, the client processor 102 broadcasts to the selected sources a request to locate the file with the given True Name using the Locate True File remote mechanism (Step S286). The request to locate may be augmented by asking to propagate this request to distant servers. The client processor then waits for one or more servers to respond positively (Step S288). After all servers respond negatively, or after a timeout period with no positive response, the mechanism repeats selection (Step S284) to attempt to identify alternative sources. If any selected source processor responds, its processor ID is the result of this mechanism. Store the processor ID in the source field of the True File registry entry record 140 of the given True Name (Step S290).

If the source location of the True Name is a different processor or medium than the destination (Step S290a), perform the following steps:

(i) Look up the True File registry entry record 140 for the corresponding True Name, and add the

source location ID to the list of sources for the True Name (Step S290b); and

5 (ii) If the source is a publishing system, determine the expiration date on the publishing system for the True Name and add that to the list of sources. If the source is not a publishing system, send a message to reserve the True File on the source processor (Step S290c).

10 Source selection in step S284 may be based on optimizations involving general availability of the source, access time, bandwidth, and transmission cost, and ignoring previously selected processors which did not respond in step S288.

8. Make True File Local

15 This mechanism is used when a True Name is known and a locally accessible copy of the corresponding file or data item is required. This mechanism makes it possible to actually read the data in a True File. The mechanism takes a True Name and returns when there is a
20 local, accessible copy of the True File in the True File registry 126. This mechanism is described here with reference to the flow chart of FIGURE 17.

First, look in the True File registry 126 for a True File entry record 140 for the corresponding True
25 Name (Step S292). If no such entry is found this mechanism fails. If there is already a True File ID for the entry (Step S294), this mechanism's task is complete. If there is a compressed file ID for the entry (Step S296), decompress the file corresponding to the file ID
30 (Step S298) and store the decompressed file ID in the entry (Step S300). This mechanism is then complete.

If there is no True File ID for the entry (Step S294) and there is no compressed file ID for the entry (Step S296), then continue searching for the requested
35 file. At this time it may be necessary to notify the user that the system is searching for the requested file.

00987723 44501

If there are one or more source IDs, then select an order in which to attempt to realize the source ID (Step S304). The order may be based on optimizations involving general availability of the source, access time, bandwidth, and transmission cost. For each source in the order chosen, realize the True File from the source location (using the Realize True File from Location primitive mechanism), until the True File is realized (Step S306). If it is realized, continue with step S294. If no known source can realize the True File, use the Locate Remote File primitive mechanism to attempt to find the True File (Step S308). If this succeeds, realize the True File from the identified source location and continue with step S296.

15 9. Create Scratch File

A scratch copy of a file is required when a file is being created or is about to be modified. The scratch copy is stored in the file system of the underlying operating system. The scratch copy is eventually assimilated when the audit file record entry 146 is processed by the Process Audit File Entry primitive mechanism. This Create Scratch File mechanism requires a local directory extensions table entry record 138. When it succeeds, the local directory extensions table entry record 138 contains the scratch file ID of a scratch file that is not contained in the True File registry 126 and that may be modified. This mechanism is now described with reference to FIGURE 18.

First determine whether the scratch file should be a copy of the existing True File (Step S310). If so, continue with step S312. Otherwise, determine whether the local directory extensions table entry record 138 identifies an existing True File (Step S316), and if so, delete the True File using the Delete True File primitive mechanism (Step S318). Then create a new, empty scratch file and store its scratch file ID in the local directory

extensions table entry record 138 (Step S320). This mechanism is then complete.

5 If the local directory extensions table entry record 138 identifies a scratch file ID (Step S312), then the entry already has a scratch file, so this mechanism succeeds.

10 If the local directory extensions table entry record 138 identifies a True File (S316), and there is no True File ID for the True File (S312), then make the True File local using the Make True File Local primitive mechanism (Step S322). If there is still no True File ID, this mechanism fails.

15 There is now a local True File for this file. If the use count in the corresponding True File registry entry record 140 is one (Step S326), save the True File ID in the scratch file ID of the local directory extensions table entry record 138, and remove the True File registry entry record 140 (Step S328). (This step makes the True File into a scratch file.) This
20 mechanism's task is complete.

25 Otherwise, if the use count in the corresponding True File registry entry record 140 is not one (in step S326), copy the file with the given True File ID to a new scratch file, using the Read File OS mechanism and store its file ID in the local directory extensions table entry record 138 (Step S330), and reduce the use count for the True File by one. If there is insufficient space to make a copy, this mechanism fails.

10. Freeze Directory

30 This mechanism freezes a directory in order to calculate its True Name. Since the True Name of a directory is a function of the files within the directory, they must not change during the computation of the True Name of the directory. This mechanism requires
35 the pathname of a directory to freeze. This mechanism is described with reference to FIGURE 19.

09987433:11504

In step S332, add one to the global freeze lock. Then search the local directory extensions table 124 to find each subordinate data file and directory of the given directory, and freeze each subordinate directory found using the Freeze Directory primitive mechanism (Step S334). Assimilate each unassimilated data file in the directory using the Assimilate Data Item primitive mechanism (Step S336). Then create a data item which begins with a tag or marker (a "magic number") being a unique data item indicating that this data item is a frozen directory (Step S337). Then list the file name and True Name for each file in the current directory (Step S338). Record any additional information required, such as the type, time of last access and modification, and size (Step S340). Next, in step S342, using the Assimilate Data Item primitive mechanism, assimilate the data item created in step S338. The resulting True Name is the True Name of the frozen directory. Finally, subtract one from the global freeze lock (Step S344).

20 11. Expand Frozen Directory

This mechanism expands a frozen directory in a given location. It requires a given pathname into which to expand the directory, and the True Name of the directory and is described with reference to FIGURE 20.

25 First, in step S346, make the True File with the given True Name local using the Make True File Local primitive mechanism. Then read each directory entry in the local file created in step S346 (Step S348). For each such directory entry, do the following:

30 Create a full pathname using the given pathname and the file name of the entry (Step S350); and

link the created path to the True Name (Step S352) using the Link Path to True Name primitive mechanism.

12. Delete True File

This mechanism deletes a reference to a True Name. The underlying True File is not removed from the True File registry 126 unless there are no additional references to the file. With reference to FIGURE 21, this mechanism is performed as follows:

If the global freeze lock is on, wait until the global freeze lock is turned off (Step S354). This prevents deleting a True File while a directory which might refer to it is being frozen. Next, find the True File registry entry record 140 given the True Name (Step S356). If the reference count field of the True File registry 126 is greater than zero, subtract one from the reference count field (Step S358). If it is determined (in step S360) that the reference count field of the True File registry entry record 140 is zero, and if there are no dependent systems listed in the True File registry entry record 140, then perform the following steps:

- (i) If the True File is a simple data item, then delete the True File, otherwise,
- (ii) (the True File is a compound data item) for each True Name in the data item, recursively delete the True File corresponding to the True Name (Step S362).
- (iii) Remove the file indicated by the True File ID and compressed file ID from the True File registry 126, and remove the True File registry entry record 140 (Step S364).

13. Process Audit File Entry

This mechanism performs tasks which are required to maintain information in the local directory extensions table 124 and True File registry 126, but which can be delayed while the processor is busy doing more time-critical tasks. Entries 142 in the audit file 132 should be processed at a background priority as long as there are entries to be processed. With reference to

FIGURE 22, the steps for processing an entry are as follows:

Determine the operation in the entry 142 currently being processed (Step S365). If the operation indicates that a file was created or written (Step S366), then assimilate the file using the Assimilate Data Item primitive mechanism (Step S368), use the New True File primitive mechanism to do additional desired processing (such as cache update, compression, and mirroring) (Step S369), and record the newly computed True Name for the file in the audit file record entry (Step S370).

Otherwise, if the entry being processed indicates that a compound data item or directory was copied (or deleted) (Step S376), then for each component True Name in the compound data item or directory, add (or subtract) one to the use count of the True File registry entry record 140 corresponding to the component True Name (Step S378).

In all cases, for each parent directory of the given file, update the size, time of last access, and time of last modification, according to the operation in the audit record (Step S379).

Note that the audit record is not removed after processing, but is retained for some reasonable period so that it may be used by the Synchronize Directory extended mechanism to allow a disconnected remote processor to update its representation of the local system.

14. Begin Grooming

This mechanism makes it possible to select a set of files for removal and determine the overall amount of space to be recovered. With reference to FIGURE 23, first verify that the global grooming lock is currently unlocked (Step S382). Then set the global grooming lock, set the total amount of space freed during grooming to zero and empty the list of files selected for deletion

(Step S384). For each True File in the True File registry 126, set the delete count to zero (Step S386).

15. Select For Removal

5 This grooming mechanism tentatively selects a
pathname to allow its corresponding True File to be
removed. With reference to FIGURE 24, first find the
local directory extensions table entry record 138
corresponding to the given pathname (Step S388). Then
find the True File registry entry record 140
10 corresponding to the True File name in the local
directory extensions table entry record 138 (Step S390).
Add one to the grooming delete count in the True File
registry entry record 140 and add the pathname to a list
of files selected for deletion (Step S392). If the
15 grooming delete count of the True File registry entry
record 140 is equal to the use count of the True File
registry entry record 140, and if there are no
entries in the dependency list of the True File registry
entry record 140, then add the size of the file indicated
20 by the True File ID and or compressed file ID to the
total amount of space freed during grooming (Step S394).

16. End Grooming

This grooming mechanism ends the grooming phase
and removes all files selected for removal. With
25 reference to FIGURE 25, for each file in the list of
files selected for deletion, delete the file (Step S396)
and then unlock the global grooming lock (Step S398).

Operating System Mechanisms

30 The next of the mechanisms provided by the
present invention, operating system mechanisms, are now
described.

The following operating system mechanisms are
described:

1. Open File;

2. Close File;
3. Read File;
4. Write File;
5. Delete File or Directory;
6. Copy File or Directory;
7. Move File or Directory;
8. Get File Status; and
9. Get Files in Directory.

5

1. Open File

10

15

20

25

30

35

A mechanism to open a file is described with reference to FIGURE 26. This mechanism is given as input a pathname and the type of access required for the file (for example, read, write, read/write, create, etc.) and produces either the File ID of the file to be opened or an indication that no file should be opened. The local directory extensions table record 138 and region table record 142 associated with the opened file are associated with the open file for later use in other processing functions which refer to the file, such as read, write, and close.

First, determine whether or not the named file exists locally by examining the local directory extensions table 124 to determine whether there is an entry corresponding to the given pathname (Step S400). If it is determined that the file name does not exist locally, then, using the access type, determine whether or not the file is being created by this opening process (Step S402). If the file is not being created, prohibit the open (Step S404). If the file is being created, create a zero-length scratch file using an entry in local directory extensions table 124 and produce the scratch file ID of this scratch file as the result (Step S406).

If, on the other hand, it is determined in step S400 that the file name does exist locally, then determine the region in which the file is located by searching the region table 128 to find the record 142

with the longest region path which is a prefix of the file pathname (Step S408). This record identifies the region of the specified file.

5 Next, determine using the access type, whether the file is being opened for writing or whether it is being opened only for reading (Step S410). If the file is being opened for reading only, then, if the file is a scratch file (Step S419), return the scratch File ID of the file (Step S424). Otherwise get the True Name from
10 the local directory extensions table 124 and make a local version of the True File associated with the True Name using the Make True File Local primitive mechanism, and then return the True File ID associated with the True Name (Step S420).

15 If the file is not being opened for reading only (Step S410), then, if it is determined by inspecting the region table entry record 142 that the file is in a read-only directory (Step S416), then prohibit the opening (Step S422).

20 If it is determined by inspecting the region table 128 that the file is in a cached region (Step S423), then send a Lock Cache message to the corresponding cache server, and wait for a return message (Step S418). If the return message says the file is
25 already locked, prohibit the opening.

 If the access type indicates that the file being modified is being rewritten completely (Step S419), so that the original data will not be required, then Delete the File using the Delete File OS mechanism (Step
30 S421) and perform step S406. Otherwise, make a scratch copy of the file (Step S417) and produce the scratch file ID of the scratch file as the result (Step S424).

2. Close File

 This mechanism takes as input the local
35 directory extensions table entry record 138 of an open file and the data maintained for the open file. To close

5 a file, add an entry to the audit file indicating the time and operation (create, read or write). The audit file processing (using the Process Audit File Entry primitive mechanism) will take care of assimilating the file and thereby updating the other records.

3. Read File

To read a file, a program must provide the offset and length of the data to be read, and the location of a buffer into which to copy the data read.

10 The file to be read from is identified by an open file descriptor which includes a File ID as computed by the Open File operating system mechanism defined above. The File ID may identify either a scratch file or a True File (or True File segment). If the File ID
15 identifies a True File, it may be either a simple or a compound True File. Reading a file is accomplished by the following steps:

20 In the case where the File ID identifies a scratch file or a simple True File, use the read capabilities of the underlying operating system.

In the case where the File ID identifies a compound file, break the read operation into one or more read operations on component segments as follows:

25 A. Identify the segment(s) to be read by dividing the specified file offset and length each by the fixed size of a segment (a system dependent parameter), to determine the segment number and number of segments that must be read.

30 B. For each segment number computed above, do the following:

i. Read the compound True File index block to determine the True Name of the segment to be read.

35 ii. Use the Realize True File from Location primitive mechanism to make the True File

segment available locally. (If that mechanism fails, the Read File mechanism fails).

5 iii. Determine the File ID of the True File specified by the True Name corresponding to this segment.

 iv. Use the Read File mechanism (recursively) to read from this segment into the corresponding location in the specified buffer.

10 4. Write File
 File writing uses the file ID and data management capabilities of the underlying operating system. File access (Make File Local described above) can be deferred until the first read or write.

15 5. Delete File or Directory
 The process of deleting a file, for a given pathname, is described here with reference to FIGURE 27.
 First, determine the local directory extensions table entry record 138 and region table entry record 142 for the file (Step S422). If the file has no local
20 directory extensions table entry record 138 or is locked or is in a read-only region, prohibit the deletion.

 Identify the corresponding True File given the True Name of the file being deleted using the True File registry 126 (Step S424). If the file has no True Name,
25 (Step S426) then delete the scratch copy of the file based on its scratch file ID in the local directory extensions table 124 (Step S427), and continue with step S428.

30 If the file has a True Name and the True File's use count is one (Step S429), then delete the True File (Step S430), and continue with step S428.

 If the file has a True Name and the True File's use count is greater than one, reduce its use count by one (Step S431). Then proceed with step S428.

In Step S428, delete the local directory extensions table entry record, and add an entry to the audit file 132 indicating the time and the operation performed (delete).

5 6. Copy File or Directory

A mechanism is provided to copy a file or directory given a source and destination processor and pathname. The Copy File mechanism does not actually copy the data in the file, only the True Name of the file.

10 This mechanism is performed as follows:

(A) Given the source path, get the True Name from the path. If this step fails, the mechanism fails.

(B) Given the True Name and the destination path, link the destination path to the True Name.

15 (C) If the source and destination processors have different True File registries, find (or, if necessary, create) an entry for the True Name in the True File registry table 126 of the destination processor. Enter into the source ID field of this new entry the
20 source processor identity.

(D) Add an entry to the audit file 132 indicating the time and operation performed (copy).

This mechanism addresses capability of the system to avoid copying data from a source location to a destination location when the destination already has the data. In addition, because of the ability to freeze a directory, this mechanism also addresses capability of the system immediately to make a copy of any collection of files, thereby to support an efficient version control mechanisms for groups of files.
30

7. Move File or Directory

A mechanism is described which moves (or renames) a file from a source path to a destination path. The move operation, like the copy operation, requires no
35 actual transfer of data, and is performed as follows:

(A) Copy the file from the source path to the destination path.

(B) If the source path is different from the destination path, delete the source path.

5 8. Get File Status

 This mechanism takes a file pathname and provides information about the pathname. First the local directory extensions table entry record 138 corresponding to the pathname given is found. If no such entry exists, then this mechanism fails, otherwise, gather information about the file and its corresponding True File from the local directory extensions table 124. The information can include any information shown in the data structures, including the size, type, owner, True Name, sources, time of last access, time of last modification, state (local or not, assimilated or not, compressed or not), use count, expiration date, and reservations.

 9. Get Files in Directory

 This mechanism enumerates the files in a directory. It is used (implicitly) whenever it is necessary to determine whether a file exists (is present) in a directory. For instance, it is implicitly used in the Open File, Delete File, Copy File or Directory, and Move File operating system mechanisms, because the files operated on are referred to by pathnames containing directory names. The mechanism works as follows:

 The local directory extensions table 124 is searched for an entry 138 with the given directory pathname. If no such entry is found, or if the entry found is not a directory, then this mechanism fails.

 If there is a corresponding True File field in the local directory extensions table record, then it is assumed that the True File represents a frozen directory. The Expand Frozen Directory primitive mechanism is used

to expand the existing True File into directory entries in the local directory extensions table.

Finally, the local directory extensions table 124 is again searched, this time to find each directory subordinate to the given directory. The names found are provided as the result.

Remote Mechanisms

The remote mechanisms provided by the present invention are now described. Recall that remote mechanisms are used by the operating system in responding to requests from other processors. These mechanisms enable the capabilities of the present invention in a peer-to-peer network mode of operation.

In a presently preferred embodiment, processors communicate with each other using a remote procedure call (RPC) style interface, running over one of any number of communication protocols such as IPX/SPX or TCP/IP. Each peer processor which provides access to its True File registry 126 or file regions, or which depends on another peer processor, provides a number of mechanisms which can be used by its peers.

The following remote mechanisms are described:

1. Locate True File;
2. Reserve True File;
3. Request True File;
4. Retire True File;
5. Cancel Reservation;
6. Acquire True File;
7. Lock Cache;
8. Update Cache; and
9. Check Expiration Date.

1. Locate True File

This mechanism allows a remote processor to determine whether the local processor contains a copy of

a specific True File. The mechanism begins with a True Name and a flag indicating whether to forward requests for this file to other servers. This mechanism is now described with reference to FIGURE 28.

5 First determine if the True File is available locally or if there is some indication of where the True File is located (for example, in the Source IDs field). Look up the requested True Name in the True File registry 126 (Step S432).

10 If a True File registry entry record 140 is not found for this True Name (Step S434), and the flag indicates that the request is not to be forwarded (Step S436), respond negatively (Step S438). That is, respond to the effect that the True File is not available.

15 One the other hand, if a True File registry entry record 140 is not found (Step S434), and the flag indicates that the request for this True File is to be forwarded (Step S436), then forward a request for this True File to some other processors in the system (Step S442). If the source table for the current processor identifies one or more publishing servers which should have a copy of this True File, then forward the request to each of those publishing servers (Step S436).

20 If a True File registry entry record 140 is found for the required True File (Step S434), and if the entry includes a True File ID or Compressed File ID (Step S440), respond positively (Step S444). If the entry includes a True File ID then this provides the identity or disk location of the actual physical representation of the file or file segment required. If the entry include a Compressed File ID, then a compressed version of the True File may be stored instead of, or in addition to, an uncompressed version. This field provides the identity of the actual representation of the compressed version of the file.

35 If the True File registry entry record 140 is found (Step S434) but does not include a True File ID

5 (the File ID is absent if the actual file is not
currently present at the current location) (Step S440),
and if the True File registry entry record 140 includes
one or more source processors, and if the request can be
forwarded, then forward the request for this True File to
one or more of the source processors (Step S444).

2. Reserve True File

10 This mechanism allows a remote processor to
indicate that it depends on the local processor for
access to a specific True File. It takes a True Name as
input. This mechanism is described here.

(A) Find the True File registry entry record
140 associated with the given True File. If no entry
exists, reply negatively.

15 (B) If the True File registry entry record 140
does not include a True File ID or compressed File ID,
and if the True File registry entry record 140 includes
no source IDs for removable storage volumes, then this
processor does not have access to a copy of the given
20 file. Reply negatively.

(C) Add the ID of the sending processor to the
list of dependent processors for the True File registry
entry record 140. Reply positively, with an indication
of whether the reserved True File is on line or off line.

25 3. Request True File

This mechanism allows a remote processor to
request a copy of a True File from the local processor.
It requires a True Name and responds positively by
sending a True File back to the requesting processor.

30 The mechanism operates as follows:

(A) Find the True File registry entry record
140 associated with the given True Name. If there is no
such True File registry entry record 140, reply
negatively.

(B) Make the True File local using the Make True File Local primitive mechanism. If this mechanism fails, the Request True File mechanism also fails.

5 (C) Send the local True File in either it is uncompressed or compressed form to the requesting remote processor. Note that if the True File is a compound file, the components are not sent.

10 (D) If the remote file is listed in the dependent process list of the True File registry entry record 140, remove it.

4. Retire True File

0908723 11501
This mechanism allows a remote processor to indicate that it no longer plans to maintain a copy of a given True File. An alternate source of the True File
15 can be specified, if, for instance, the True File is being moved from one server to another. It begins with a True Name, a requesting processor ID, and an optional alternate source. This mechanism operates as follows:

20 (A) Find a True Name entry in the True File registry 126. If there is no entry for this True Name, this mechanism's task is complete.

(B) Find the requesting processor on the source list and, if it is there, remove it.

25 (C) If an alternate source is provided, add it to the source list for the True File registry entry record 140.

30 (D) If the source list of the True File registry entry record 140 has no items in it, use the Locate Remote File primitive mechanism to search for another copy of the file. If it fails, raise a serious error.

5. Cancel Reservation

35 This mechanism allows a remote processor to indicate that it no longer requires access to a True File stored on the local processor. It begins with a True

Name and a requesting processor ID and proceeds as follows:

(A) Find the True Name entry in the True File registry 126. If there is no entry for this True Name, this mechanism's task is complete.

(B) Remove the identity of the requesting processor from the list of dependent processors, if it appears.

(C) If the list of dependent processors becomes zero and the use count is also zero, delete the True File.

6. Acquire True File

This mechanism allows a remote processor to insist that a local processor make a copy of a specified True File. It is used, for example, when a cache client wants to write through a new version of a file. The Acquire True File mechanism begins with a data item and an optional True Name for the data item and proceeds as follows:

(A) Confirm that the requesting processor has the right to require the local processor to acquire data items. If not, send a negative reply.

(B) Make a local copy of the data item transmitted by the remote processor.

(C) Assimilate the data item into the True File registry of the local processor.

(D) If a True Name was provided with the file, the True Name calculation can be avoided, or the mechanism can verify that the file received matches the True Name sent.

(E) Add an entry in the dependent processor list of the true file registry record indicating that the requesting processor depends on this copy of the given True File.

(F) Send a positive reply.

7. Lock Cache

This mechanism allows a remote cache client to lock a local file so that local users or other cache clients cannot change it while the remote processor is using it. The mechanism begins with a pathname and proceeds as follows:

(A) Find the local directory extensions table entry record 138 of the specified pathname. If no such entry exists, reply negatively.

(B) If an local directory extensions table entry record 138 exists and is already locked, reply negatively that the file is already locked.

(C) If an local directory extensions table entry record 138 exists and is not locked, lock the entry. Reply positively.

8. Update Cache

This mechanism allows a remote cache client to unlock a local file and update it with new contents. It begins with a pathname and a True Name. The file corresponding to the True Name must be accessible from the remote processor. This mechanism operates as follows:

Find the local directory extensions table entry record 138 corresponding to the given pathname. Reply negatively if no such entry exists or if the entry is not locked.

Link the given pathname to the given True Name using the Link Path to True Name primitive mechanism.

Unlock the local directory extensions table entry record 138 and return positively.

9. Check Expiration Date

Return current or new expiration date and possible alternative source to caller.

Background Processes and Mechanisms

The background processes and mechanisms provided by the present invention are now described. Recall that background mechanisms are intended to run occasionally and at a low priority to provide automated management capabilities with respect to the present invention.

The following background mechanisms are described:

1. Mirror True File;
2. Groom Region;
3. Check for Expired Links;
4. Verify Region; and
5. Groom Source List.

1. Mirror True File

This mechanism is used to ensure that files are available in alternate locations in mirror groups or archived on archival servers. The mechanism depends on application-specific migration/archival criteria (size, time since last access, number of copies required, number of existing alternative sources) which determine under what conditions a file should be moved. The Mirror True File mechanism operates as follows, using the True File specified, perform the following steps:

(A) Count the number of available locations of the True File by inspecting the source list of the True File registry entry record 140 for the True File. This step determines how many copies of the True File are available in the system.

(B) If the True File meets the specified migration criteria, select a mirror group server to which a copy of the file should be sent. Use the Acquire True File remote mechanism to copy the True File to the selected mirror group server. Add the identity of the selected system to the source list for the True File.

2. Groom Region

This mechanism is used to automatically free up space in a processor by deleting data items that may be available elsewhere. The mechanism depends on application-specific grooming criteria (for instance, a file may be removed if there is an alternate online source for it, it has not been accessed in a given number of days, and it is larger than a given size). This mechanism operates as follows:

Repeat the following steps (i) to (iii) with more aggressive grooming criteria until sufficient space is freed or until all grooming criteria have been exercised. Use grooming information to determine how much space has been freed. Recall that, while grooming is in effect, grooming information includes a table of pathnames selected for deletion, and keeps track of the amount of space that would be freed if all of the files were deleted.

(i) Begin Grooming (using the primitive mechanism).

(ii) For each pathname in the specified region, for the True File corresponding to the pathname, if the True File is present, has at least one alternative source, and meets application specific grooming criteria for the region, select the file for removal (using the primitive mechanism).

(iii) End Grooming (using the primitive mechanism).

If the region is used as a cache, no other processors are dependent on True Files to which it refers, and all such True Files are mirrored elsewhere. In this case, True Files can be removed with impunity. For a cache region, the grooming criteria would ordinarily eliminate the least recently accessed True Files first. This is best done by sorting the True Files in the region by the most recent access time before performing step (ii) above. The application specific

criteria would thus be to select for removal every True File encountered (beginning with the least recently used) until the required amount of free space is reached.

3. Check for Expired Links

5 This mechanism is used to determine whether dependencies on published files should be refreshed. The following steps describe the operation of this mechanism:

For each pathname in the specified region, for each True File corresponding to the pathname, perform the following step:

10 If the True File registry entry record 140 corresponding to the True File contains at least one source which is a publishing server, and if the expiration date on the dependency is past or close, then perform the following steps:

15 (A) Determine whether the True File registry entry record contains other sources which have not expired.

20 (B) Check the True Name expiration of the server. If the expiration date has been extended, or an alternate source is suggested, add the source to the True File registry entry record 140.

25 (C) If no acceptable alternate source was found in steps (A) or (B) above, make a local copy of the True File.

(D) Remove the expired source.

4. Verify Region

30 This mechanism can be used to ensure that the data items in the True File registry 126 have not been damaged accidentally or maliciously. The operation of this mechanism is described by the following steps:

(A) Search the local directory extensions table 124 for each pathname in the specified region and then perform the following steps:

(i) Get the True File name corresponding to the pathname;

(ii) If the True File registry entry 140 for the True File does not have a True File ID or compressed file ID, ignore it.

(iii) Use the Verify True File mechanism (see extended mechanisms below) to confirm that the True File specified is correct.

5. Groom Source List

10 The source list in a True File entry should be groomed sometimes to ensure there are not too many mirror or archive copies. When a file is deleted or when a region definition or its mirror criteria are changed, it may be necessary to inspect the affected True Files to
15 determine whether there are too many mirror copies. This can be done with the following steps:

For each affected True File,

(A) Search the local directory extensions table to find each region that refers to the True File.

20 (B) Create a set of "required sources", initially empty.

(C) For each region found,

(a) determine the mirroring criteria for that region,

25 (b) determine which sources for the True File satisfy the mirroring criteria, and

(c) add these sources to the set of required sources.

(D) For each source in the True File registry
30 entry, if the source identifies a remote processor (as opposed to removable media), and if the source is not a publisher, and if the source is not in the set of required sources, then eliminate the source, and use the Cancel Reservation remote mechanism to eliminate the
35 given processor from the list of dependent processors

recorded at the remote processor identified by the source.

Extended Mechanisms

5 The extended mechanisms provided by the present invention are now described. Recall that extended mechanisms run within application programs over the operating system to provide solutions to specific problems and applications.

10 The following extended mechanisms are described:

1. Inventory Existing Directory;
2. Inventory Removable, Read-only Files;
3. Synchronize Directories;
4. Publish Region;
- 15 5. Retire Directory;
6. Realize Directory at Location;
7. Verify True File;
8. Track for Accounting Purposes; and
9. Track for Licensing Purposes.

20 1. Inventory Existing Directory

This mechanism determines the True Names of files in an existing on-line directory in the underlying operating system. One purpose of this mechanism is to install True Name mechanisms in an existing file system.

25 An effect of such an installation is to eliminate immediately all duplicate files from the file system being traversed. If several file systems are inventoried in a single True File registry, duplicates across the volumes are also eliminated.

30 (A) Traverse the underlying file system in the operating system. For each file encountered, excluding directories, perform the following:

(i) Assimilate the file encountered (using the Assimilate File primitive mechanism). This

process computes its True Name and moves its data into the True File registry 126.

- (ii) Create a pathname consisting of the path to the volume directory and the relative path of the file on the media. Link this path to the computed True Name using the Link Path to True Name primitive mechanism.

2. Inventory Removable, Read-only Files

A system with access to removable, read-only media volumes (such as WORM disks and CD-ROMs) can create a usable inventory of the files on these disks without having to make online copies. These objects can then be used for archival purposes, directory overlays, or other needs. An operator must request that an inventory be created for such a volume.

This mechanism allows for maintaining inventories of the contents of files and data items on removable media, such as diskettes and CD-ROMs, independent of other properties of the files such as name, location, and date of creation.

The mechanism creates an online inventory of the files on one or more removable volumes, such as a floppy disk or CD-ROM, when the data on the volume is represented as a directory. The inventory service uses a True Name to identify each file, providing a way to locate the data independent of its name, date of creation, or location.

The inventory can be used for archival of data (making it possible to avoid archiving data when that data is already on a separate volume), for grooming (making it possible to delete infrequently accessed files if they can be retrieved from removable volumes), for version control (making it possible to generate a new version of a CD-ROM without having to copy the old version), and for other purposes.

The inventory is made by creating a volume directory in the media inventory in which each file named identifies the data item on the volume being inventoried. Data items are not copied from the removable volume during the inventory process.

10 An operator must request that an inventory be created for a specific volume. Once created, the volume directory can be frozen or copied like any other directory. Data items from either the physical volume or the volume directory can be accessed using the Open File operating system mechanism which will cause them to be read from the physical volume using the Realize True File from Location primitive mechanism.

To create an inventory the following steps are
15 taken:

(A) A volume directory in the media inventory is created to correspond to the volume being inventoried. Its contextual name identifies the specific volume.

(B) A source table entry 144 for the volume is created in the source table 130. This entry 144 identifies the physical source volume and the volume directory created in step (A).

(C) The filesystem on the volume is traversed. For each file encountered, excluding directories, the following steps are taken:

(i) The True Name of the file is computed. An entry is created in the True Name registry 124, including the True Name of the file using the primitive mechanism. The source field of the True Name registry entry 140 identifies the source table entry 144.

(ii) A pathname is created consisting of the path to the volume directory and the relative path of the file on the media. This path is linked to the computed True Name using Link Path to True Name primitive mechanism.

(D) After all files have been inventoried, the volume directory is frozen. The volume directory serves

as a table of contents for the volume. It can be copied using the Copy File or Directory primitive mechanism to create an "overlay" directory which can then be modified, making it possible to edit a virtual copy of a read-only medium.

3. Synchronize Directories

Given two versions of a directory derived from the same starting point, this mechanism creates a new, synchronized version which includes the changes from each. Where a file is changed in both versions, this mechanism provides a user exit for handling the discrepancy. By using True Names, comparisons are instantaneous, and no copies of files are necessary.

This mechanism lets a local processor synchronize a directory to account for changes made at a remote processor. Its purpose is to bring a local copy of a directory up to date after a period of no communication between the local and remote processor. Such a period might occur if the local processor were a mobile processor detached from its server, or if two distant processors were run independently and updated nightly.

An advantage of the described synchronization process is that it does not depend on synchronizing the clocks of the local and remote processors. However, it does require that the local processor track its position in the remote processor's audit file.

This mechanism does not resolve changes made simultaneously to the same file at several sites. If that occurs, an external resolution mechanism such as, for example, operator intervention, is required.

The mechanism takes as input a start time, a local directory pathname, a remote processor name, and a remote directory pathname name, and it operates by the following steps:

(A) Request a copy of the audit file 132 from the remote processor using the Request True File remote mechanism.

5 (B) For each entry 146 in the audit file 132 after the start time, if the entry indicates a change to a file in the remote directory, perform the following steps:

(i) Compute the pathname of the corresponding file in the local directory. Determine the
10 True Name of the corresponding file.

(ii) If the True Name of the local file is the same as the old True Name in the audit file, or if there is no local file and the audit entry indicates a new file is being created, link the new True Name in the
15 audit file to the local pathname using the Link Path to True Name primitive mechanism.

(iii) Otherwise, note that there is a problem with the synchronization by sending a message to the operator or to a problem resolution program,
20 indicating the local pathname, remote pathname, remote processor, and time of change.

(C) After synchronization is complete, record the time of the final change. This time is to be used as the new start time the next time this directory is
25 synchronized with the same remote processor.

4. Publish Region

The publish region mechanism allows a processor to offer the files in a region to any client processors for a limited period of time.

30 The purpose of the service is to eliminate any need for client processors to make reservations with the publishing processor. This in turn makes it possible for the publishing processor to service a much larger number of clients.

35 When a region is published, an expiration date is defined for all files in the region, and is propagated

into the publishing system's True File registry entry record 140 for each file.

When a remote file is copied, for instance using the Copy File operating system mechanism, the expiration date is copied into the source field of the client's True File registry entry record 140. When the source is a publishing system, no dependency need be created.

The client processor must occasionally and in background, check for expired links, to make sure it still has access to these files. This is described in the background mechanism Check for Expired Links.

5. Retire Directory

This mechanism makes it possible to eliminate safely the True Files in a directory, or at least dependencies on them, after ensuring that any client processors depending on those files remove their dependencies. The files in the directory are not actually deleted by this process. The directory can be deleted with the Delete File operating system mechanism.

The mechanism takes the pathname of a given directory, and optionally, the identification of a preferred alternate source processor for clients to use. The mechanism performs the following steps:

(A) Traverse the directory. For each file in the directory, perform the following steps:

(i) Get the True Name of the file from its path and find the True File registry entry 140 associated with the True Name.

(ii) Determine an alternate source for the True File. If the source IDs field of the TFR entry includes the preferred alternate source, that is the alternate source. If it does not, but includes some other source, that is the alternate source. If it contains no alternate sources, there is no alternate source.

(iii) For each dependent processor in the True File registry entry 140, ask that processor to retire the True File, specifying an alternate source if one was determined, using the remote mechanism.

5 6. Realize Directory at Location

This mechanism allows the user or operating system to force copies of files from some source location to the True File registry 126 at a given location. The purpose of the mechanism is to ensure that files are accessible in the event the source location becomes inaccessible. This can happen for instance if the source or given location are on mobile computers, or are on removable media, or if the network connection to the source is expected to become unavailable, or if the source is being retired.

This mechanism is provided in the following steps for each file in the given directory, with the exception of subdirectories:

(A) Get the local directory extensions table entry record 138 given the pathname of the file. Get the True Name of the local directory extensions table entry record 138. This service assimilates the file if it has not already been assimilated.

(B) Realize the corresponding True File at the given location. This service causes it to be copied to the given location from a remote system or removable media.

7. Verify True File

This mechanism is used to verify that the data item in a True File registry 126 is indeed the correct data item given its True Name. Its purpose is to guard against device errors, malicious changes, or other problems.

If an error is found, the system has the ability to "heal" itself by finding another source for

the True File with the given name. It may also be desirable to verify that the error has not propagated to other systems, and to log the problem or indicate it to the computer operator. These details are not described here.

To verify a data item that is not in a True File registry 126, use the Calculate True Name primitive mechanism described above.

The basic mechanism begins with a True Name, and operates in the following steps:

(A) Find the True File registry entry record 140 corresponding to the given True Name.

(B) If there is a True File ID for the True File registry entry record 140 then use it. Otherwise, indicate that no file exists to verify.

(C) Calculate the True Name of the data item given the file ID of the data item.

(D) Confirm that the calculated True Name is equal to the given True Name.

(E) If the True Names are not equal, there is an error in the True File registry 126. Remove the True File ID from the True File registry entry record 140 and place it somewhere else. Indicate that the True File registry entry record 140 contained an error.

8. Track for Accounting Purposes

This mechanism provides a way to know reliably which files have been stored on a system or transmitted from one system to another. The mechanism can be used as a basis for a value-based accounting system in which charges are based on the identity of the data stored or transmitted, rather than simply on the number of bits.

This mechanism allows the system to track possession of specific data items according to content by owner, independent of the name, date, or other properties of the data item, and tracks the uses of specific data items and files by content for accounting purposes. True

names make it possible to identify each file briefly yet uniquely for this purpose.

Tracking the identities of files requires maintaining an accounting log 134 and processing it for accounting or billing purposes. The mechanism operates in the following steps:

(A) Note every time a file is created or deleted, for instance by monitoring audit entries in the Process Audit File Entry primitive mechanism. When such an event is encountered, create an entry 148 in the accounting log 134 that shows the responsible party and the identity of the file created or deleted.

(B) Every time a file is transmitted, for instance when a file is copied with a Request True File remote mechanism or an Acquire True File remote mechanism, create an entry in the accounting log 134 that shows the responsible party, the identity of the file, and the source and destination processors.

(C) Occasionally run an accounting program to process the accounting log 134, distributing the events to the account records of each responsible party. The account records can eventually be summarized for billing purposes.

9. Track for Licensing Purposes

This mechanism ensures that licensed files are not used by unauthorized parties. The True Name provides a safe way to identify licensed material. This service allows proof of possession of specific files according to their contents without disclosing their contents.

Enforcing use of valid licenses can be active (for example, by refusing to provide access to a file without authorization) or passive (for example, by creating a report of users who do not have proper authorization).

One possible way to perform license validation is to perform occasional audits of employee systems. The

service described herein relies on True Names to support such an audit, as in the following steps:

5 (A) For each licensed product, record in the license table 136 the True Name of key files in the product (that is, files which are required in order to use the product, and which do not occur in other products) Typically, for a software product, this would include the main executable image and perhaps other major files such as clip-art, scripts, or online help. Also
10 record the identity of each system which is authorized to have a copy of the file.

(B) Occasionally, compare the contents of each user processor against the license table 136. For each True Name in the license table do the following:

15 (i) Unless the user processor is authorized to have a copy of the file, confirm that the user processor does not have a copy of the file using the Locate True File mechanism.

20 (ii) If the user processor is found to have a file that it is not authorized to have, record the user processor and True Name in a license violation table.

The System in Operation

25 Given the mechanisms described above, the operation of a typical DP system employing these mechanisms is now described in order to demonstrate how the present invention meets its requirements and capabilities.

30 In operation, data items (for example, files, database records, messages, data segments, data blocks, directories, instances of object classes, and the like) in a DP system employing the present invention are identified by substantially unique identifiers (True Names), the identifiers depending on all of the data in
35 the data items and only on the data in the data items.

09987723 111501

The primitive mechanisms Calculate True Name and Assimilate Data Item support this property. For any given data item, using the Calculate True Name primitive mechanism, a substantially unique identifier or True Name for that data item can be determined.

Further, in operation of a DP system incorporating the present invention, multiple copies of data items are avoided (unless they are required for some reason such as backups or mirror copies in a fault-tolerant system). Multiple copies of data items are avoided even when multiple names refer to the same data item. The primitive mechanisms Assimilate Data Items and New True File support this property. Using the Assimilate Data Item primitive mechanism, if a data item already exists in the system, as indicated by an entry in the True File registry 126, this existence will be discovered by this mechanism, and the duplicate data item (the new data item) will be eliminated (or not added). Thus, for example, if a data file is being copied onto a system from a floppy disk, if, based on the True Name of the data file, it is determined that the data file already exists in the system (by the same or some other name), then the duplicate copy will not be installed. If the data item was being installed on the system by some name other than its current name, then, using the Link Path to True Name primitive mechanism, the other (or new) name can be linked to the already existing data item.

In general, the mechanisms of the present invention operate in such a way as to avoid recreating an actual data item at a location when a copy of that data item is already present at that location. In the case of a copy from a floppy disk, the data item (file) may have to be copied (into a scratch file) before it can be determined that it is a duplicate. This is because only one processor is involved. On the other hand, in a multiprocessor environment or DP system, each processor has a record of the True Names of the data items on that

processor. When a data item is to be copied to another location (another processor) in the DP system, all that is necessary is to examine the True Name of the data item prior to the copying. If a data item with the same True Name already exists at the destination location (processor), then there is no need to copy the data item. Note that if a data item which already exists locally at a destination location is still copied to the destination location (for example, because the remote system did not have a True Name for the data item or because it arrives as a stream of un-named data), the Assimilate Data Item primitive mechanism will prevent multiple copies of the data item from being created.

Since the True Name of a large data item (a compound data item) is derived from and based on the True Names of components of the data item, copying of an entire data item can be avoided. Since some (or all) of the components of a large data item may already be present at a destination location, only those components which are not present there need be copied. This property derives from the manner in which True Names are determined.

When a file is copied by the Copy File or Directory operating system mechanism, only the True Name of the file is actually replicated.

When a file is opened (using the Open File operating system mechanism), it uses the Make True File Local primitive mechanism (either directly or indirectly through the Create Scratch File primitive mechanism) to create a local copy of the file. The Open File operating system mechanism uses the Make True File Local primitive mechanism, which uses the Realize True File from Location primitive mechanism, which, in turn uses the Request True File remote mechanism.

The Request True File remote mechanism copies only a single data item from one processor to another. If the data item is a compound file, its component

segments are not copied, only the indirect block is copied. The segments are copied only when they are read (or otherwise needed).

5 The Read File operating system mechanism actually reads data. The Read File mechanism is aware of compound files and indirect blocks, and it uses the Realize True File from Location primitive mechanism to make sure that component segments are locally available, and then uses the operating system file mechanisms to
10 read data from the local file.

Thus, when a compound file is copied from a remote system, only its True Name is copied. When it is opened, only its indirect block is copied. When the corresponding file is read, the required component
15 segments are realized and therefore copied.

In operation data items can be accessed by reference to their identities (True Names) independent of their present location. The actual data item or True File corresponding to a given data identifier or True
20 Name may reside anywhere in the system (that is, locally, remotely, offline, etc). If a required True File is present locally, then the data in the file can be accessed. If the data item is not present locally, there are a number of ways in which it can be obtained from
25 wherever it is present. Using the source IDs field of the True File registry table, the location(s) of copies of the True File corresponding to a given True Name can be determined. The Realize True File from Location primitive mechanism tries to make a local copy of a True
30 File, given its True Name and the name of a source location (processor or media) that may contain the True File. If, on the other hand, for some reason it is not known where there is a copy of the True File, or if the processors identified in the source IDs field do not
35 respond with the required True File, the processor requiring the data item can make a general request for the data item using the Request True File remote

mechanism from all processors in the system that it can contact.

As a result, the system provides transparent access to any data item by reference to its data
5 identity, and independent of its present location.

In operation, data items in the system can be verified and have their integrity checked. This is from the manner in which True Names are determined. This can be used for security purposes, for instance, to check for
10 viruses and to verify that data retrieved from another location is the desired and requested data. For example, the system might store the True Names of all executable applications on the system and then periodically redetermine the True Names of each of these applications
15 to ensure that they match the stored True Names. Any change in a True Name potentially signals corruption in the system and can be further investigated. The Verify Region background mechanism and the Verify True File extended mechanisms provide direct support for this mode
20 of operation. The Verify Region mechanism is used to ensure that the data items in the True File registry have not been damaged accidentally or maliciously. The Verify True File mechanism verifies that a data item in a True File registry is indeed the correct data item given its
25 True Name.

Once a processor has determined where (that is, at which other processor or location) a copy of a data item is in the DP system, that processor might need that other processor or location to keep a copy of that data
30 item. For example, a processor might want to delete local copies of data items to make space available locally while knowing that it can rely on retrieving the data from somewhere else when needed. To this end the system allows a processor to Reserve (and cancel the
35 reservation of) True Files at remote locations (using the remote mechanism). In this way the remote locations are

put on notice that another location is relying on the presence of the True File at their location.

A DP system employing the present invention can be made into a fault-tolerant system by providing a certain amount of redundancy of data items at multiple locations in the system. Using the Acquire True File and Reserve True File remote mechanisms, a particular processor can implement its own form of fault-tolerance by copying data items to other processors and then reserving them there. However, the system also provides the Mirror True File background mechanism to mirror (make copies) of the True File available elsewhere in the system. Any degree of redundancy (limited by the number of processors or locations in the system) can be implemented. As a result, this invention maintains a desired degree or level of redundancy in a network of processors, to protect against failure of any particular processor by ensuring that multiple copies of data items exist at different locations.

The data structures used to implement various features and mechanisms of this invention store a variety of useful information which can be used, in conjunction with the various mechanisms, to implement storage schemes and policies in a DP system employing the invention. For example, the size, age and location of a data item (or of groups of data items) is provided. This information can be used to decide how the data items should be treated. For example, a processor may implement a policy of deleting local copies of all data items over a certain age if other copies of those data items are present elsewhere in the system. The age (or variations on the age) can be determined using the time of last access or modification in the local directory extensions table, and the presence of other copies of the data item can be determined either from the Safe Flag or the source IDs, or by checking which other processors in the system have

copies of the data item and then reserving at least one of those copies.

In operation, the system can keep track of data items regardless of how those items are named by users (or regardless of whether the data items even have names). The system can also track data items that have different names (in different or the same location) as well as different data items that have the same name. Since a data item is identified by the data in the item, without regard for the context of the data, the problems of inconsistent naming in a DP system are overcome.

In operation, the system can publish data items, allowing other, possibly anonymous, systems in a network to gain access to the data items and to rely on the availability of these data items. True Names are globally unique identifiers which can be published simply by copying them. For example, a user might create a textual representation of a file on system A with True Name N (for instance as a hexadecimal string), and post it on a computer bulletin board. Another user on system B could create a directory entry F for this True Name N by using the Link Path to True Name primitive mechanism. (Alternatively, an application could be developed which hides the True Name from the users, but provides the same public transfer service.)

When a program on system B attempts to open pathname F linked to True Name N, the Locate Remote File primitive mechanism would be used, and would use the Locate True File remote mechanism to search for True Name N on one or more remote processors, such as system A. If system B has access to system A, it would be able to realize the True File (using the Realize True File from Location primitive mechanism) and use it locally. Alternatively, system B could find True Name N by accessing any publicly available True Name server, if the server could eventually forward the request to system A.

Clients of a local server can indicate that they depend on a given True File (using the Reserve True File remote mechanism) so that the True File is not deleted from the server registry as long as some client requires access to it. (The Retire True File remote mechanism is used to indicate that a client no longer needs a given True File.)

A publishing server, on the other hand, may want to provide access to many clients, and possibly anonymous ones, without incurring the overhead of tracking dependencies for each client. Therefore, a public server can provide expiration dates for True Files in its registry. This allows client systems to safely maintain references to a True File on the public server. The Check For Expired Links background mechanism allows the client of a publishing server to occasionally confirm that its dependencies on the publishing server are safe.

In a variation of this aspect of the invention, a processor that is newly connected (or reconnected after some absence) to the system can obtain a current version of all (or of needed) data in the system by requesting it from a server processor. Any such processor can send a request to update or resynchronize all of its directories (starting at a root directory), simply by using the Synchronize Directories extended mechanism on the needed directories.

Using the accounting log or some other user provided mechanism, a user can prove the existence of certain data items at certain times. By publishing (in a public place) a list of all True Names in the system on a given day (or at some given time), a user can later refer back to that list to show that a particular data item was present in the system at the time that list was published. Such a mechanism is useful in tracking, for example, laboratory notebooks or the like to prove dates of conception of inventions. Such a mechanism also

permits proof of possession of a data item at a particular date and time.

5 The accounting log file can also track the use of specific data items and files by content for accounting purposes. For instance, an information utility company can determine the data identities of data items that are stored and transmitted through its computer systems, and use these identities to provide bills to its customers based on the identities of the data items being transmitted (as defined by the substantially unique identifier). The assignment of prices for storing and transmitting specific True Files would be made by the information utility and/or its data suppliers; this information would be joined periodically with the information in the accounting log file to produce customer statements.

20 Backing up data items in a DP system employing the present invention can be done based on the True Names of the data items. By tracking backups using True Names, duplication in the backups is prevented. In operation, the system maintains a backup record of data identifiers of data items already backed up, and invokes the Copy File or Directory operating system mechanism to copy only those data items whose data identifiers are not recorded in the backup record. Once a data item has been backed up, it can be restored by retrieving it from its backup location, based on the identifier of the data item. Using the backup record produced by the backup to identify the data item, the data item can be obtained using, for example, the Make True File Local primitive mechanism.

35 In operation, the system can be used to cache data items from a server, so that only the most recently accessed data items need be retained. To operate in this way, a cache client is configured to have a local registry (its cache) with a remote Local Directory Extensions table (from the cache server). Whenever a

file is opened (or read), the Local Directory Extensions table is used to identify the True Name, and the Make True File Local primitive mechanism inspects the local registry. When the local registry already has a copy, the file is already cached. Otherwise, the Locate True File remote mechanism is used to get a copy of the file. This mechanism consults the cache server and uses the Request True File remote mechanism to make a local copy, effectively loading the cache.

The Groom Cache background mechanism flushes the cache, removing the least-recently-used files from the cache client's True File registry. While a file is being modified on a cache client, the Lock Cache and Update Cache remote mechanisms prevent other clients from trying to modify the same file.

In operation, when the system is being used to cache data items, the problems of maintaining cache consistency are avoided.

To access a cache and to fill it from its server, a key is required to identify the data item desired. Ordinarily, the key is a name or address (in this case, it would be the pathname of a file). If the data associated with such a key is changed, the client's cache becomes inconsistent; when the cache client refers to that name, it will retrieve the wrong data. In order to maintain cache consistency it is necessary to notify every client immediately whenever a change occurs on the server.

By using an embodiment of the present invention, the cache key uniquely identifies the data it represents. When the data associated with a name changes, the key itself changes. Thus, when a cache client wishes to access the modified data associated with a given file name, it will use a new key (the True Name of the new file) rather than the key to the old file contents in its cache. The client will always request the correct data, and the old data in its cache will be

eventually aged and flushed by the Groom Cache background mechanism.

Because it is not necessary to immediately notify clients when changes on the cache server occur, the present invention makes it possible for a single server to support a much larger number of clients than is otherwise possible.

In operation, the system automatically archives data items as they are created or modified. After a file is created or modified, the Close File operating system mechanism creates an audit file record, which is eventually processed by the Process Audit File Entry primitive mechanism. This mechanism uses the New True File primitive mechanism for any file which is newly created, which in turn uses the Mirror True File background mechanism if the True File is in a mirrored or archived region. This mechanism causes one or more copies of the new file to be made on remote processors.

In operation, the system can efficiently record and preserve any collection of data items. The Freeze Directory primitive mechanism creates a True File which identifies all of the files in the directory and its subordinates. Because this True File includes the True Names of its constituents, it represents the exact contents of the directory tree at the time it was frozen. The frozen directory can be copied with its components preserved.

The Acquire True File remote mechanism (used in mirroring and archiving) preserves the directory tree structure by ensuring that all of the component segments and True Files in a compound data item are actually copied to a remote system. Of course, no transfer is necessary for data items already in the registry of the remote system.

In operation, the system can efficiently make a copy of any collection of data items, to support a version control mechanism for groups of the data items.

09987723 111501

The Freeze Directory primitive mechanism is used to create a collection of data items. The constituent files and segments referred to by the frozen directory are maintained in the registry, without any need to make copies of the constituents each time the directory is frozen.

Whenever a pathname is traversed, the Get Files in Directory operating system mechanism is used, and when it encounters a frozen directory, it uses the Expand Frozen Directory primitive mechanism.

A frozen directory can be copied from one pathname to another efficiently, merely by copying its True Name. The Copy File operating system mechanism is used to copy a frozen directory.

Thus it is possible to efficiently create copies of different versions of a directory, thereby creating a record of its history (hence a version control system).

In operation, the system can maintain a local inventory of all the data items located on a given removable medium, such as a diskette or CD-ROM. The inventory is independent of other properties of the data items such as their name, location, and date of creation.

The Inventory Existing Directory extended mechanism provides a way to create True File Registry entries for all of the files in a directory. One use of this inventory is as a way to pre-load a True File registry with backup record information. Those files in the registry (such as previously installed software) which are on the volumes inventoried need not be backed up onto other volumes.

The Inventory Removable, Read-only Files extended mechanism not only determines the True Names for the files on the medium, but also records directory entries for each file in a frozen directory structure. By copying and modifying this directory, it is possible to create an on line patch, or small modification of an

existing read-only file. For example, it is possible to create an online representation of a modified CD-ROM, such that the unmodified files are actually on the CD-ROM, and only the modified files are online.

5 In operation, the system tracks possession of specific data items according to content by owner, independent of the name, date, or other properties of the data item, and tracks the uses of specific data items and files by content for accounting purposes. Using the
10 Track for Accounting Purposes extended mechanism provides a way to know reliably which files have been stored on a system or transmitted from one system to another.

True Names in Relational and Object-Oriented Databases

15 Although the preferred embodiment of this invention has been presented in the context of a file system, the invention of True Names would be equally valuable in a relational or object-oriented database. A relational or object-oriented database system using True Names would have similar benefits to those of the file
20 system employing the invention. For instance, such a database would permit efficient elimination of duplicate records, support a cache for records, simplify the process of maintaining cache consistency, provide location-independent access to records, maintain archives
25 and histories of records, and synchronize with distant or disconnected systems or databases.

 The mechanisms described above can be easily modified to serve in such a database environment. The True Name registry would be used as a repository of
30 database records. All references to records would be via the True Name of the record. (The Local Directory Extensions table is an example of a primary index that uses the True Name as the unique identifier of the desired records.)

In such a database, the operations of inserting, updating, and deleting records would be implemented by first assimilating records into the registry, and then updating a primary key index to map the key of the record to its contents by using the True Name as a pointer to the contents.

The mechanisms described in the preferred embodiment, or similar mechanisms, would be employed in such a system. These mechanisms could include, for example, the mechanisms for calculating true names, assimilating, locating, realizing, deleting, copying, and moving True Files, for mirroring True Files, for maintaining a cache of True Files, for grooming True Files, and other mechanisms based on the use of substantially unique identifiers.

While the invention has been described in connection with what is presently considered to be the most practical and preferred embodiments, it is to be understood that the invention is not to be limited to the disclosed embodiment, but on the contrary, is intended to cover various modifications and equivalent arrangements included within the spirit and scope of the appended claims.

WHAT IS CLAIMED IS:

1. In a data processing system, an apparatus comprising:

5 identity means for determining, for any of a plurality of data items in the system, a substantially unique identifier, said identifier depending on all of the data in the data item and only on the data in the data item; and

10 existence means for determining whether a particular data item is present in the system, by examining the identifiers of the plurality of data items.

2. An apparatus as in claim 1, further comprising:

15 local existence means for determining whether an instance of a particular data item is present at a particular location in the system, based on the identifier of the data item.

20 3. An apparatus as in claim 2, wherein each location contains a distinct plurality of data items, and wherein said local existence means determines whether a particular data item is present at a particular location in the system by examining the identifiers of the plurality of data items at said particular location in the system.

25 4. An apparatus as in claim 2, further comprising:

30 data associating means for making and maintaining, for a data item in the system, an association between the data item and the identifier of the data item; and

access means for accessing a particular data item using the identifier of the data item.

5. An apparatus as in claim 2, further comprising:

duplication means for copying a data item from a source to a destination in the data processing system, by providing said destination with the data item only if it is determined using the data identifier that the data item is not present at the destination.

6. An apparatus as in claim 4, further comprising:

assimilation means for assimilating a new data item into the system, said assimilation means invoking said identity means to determine the identifier of the new data item and invoking said data associating means to associate the new data item with its identifier.

7. An apparatus as in claim 4, further comprising:

duplication means for duplicating a data item from a source location to a destination location in the data processing system, based on the identifier of the data item, said duplication means invoking said local existence means to determine whether an instance of the data item is present at the destination location, and invoking said access means to provide said destination with the data item only if said local existence means determines that no instance of the data item is present at the destination.

8. An apparatus as in claim 7, further comprising:

backup means for making copies of data items in the system, said backup means maintaining a backup record of identifiers of data items backed up, and invoking duplication means to copy only those data items whose data identifiers are not recorded in the backup record.

9. An apparatus as in claim 8, further comprising:

recovery means for retrieving a data item previously backed up by said backup means, based on the identifier of the data item, said recovery means using the backup record to identify the data item, and invoking access means to retrieve the data item.

10. An apparatus as in claim 2, wherein a location is a computer among a network of computers, the apparatus further comprising:

remote existence means for determining whether a data item is present at a remote location in the system from a current location in the system, based on the identifier of the data item, said remote location using local existence means at the remote location to determine whether the data item is present at the remote location, and providing the current location with an indication of the presence of the data item at the remote location.

11. An apparatus as in claim 4, wherein a location is a computer among a network of computers, the apparatus further comprising:

requesting means for requesting a data item at a current location in the system from a remote location in the system, based on the identifier of the data item, said remote location using access means at the remote location to obtain the data item and to send it to the current location if it is present.

12. An apparatus as in claim 1, further comprising:

context means for making and maintaining a context association between at least one contextual name of a data item in the system and the identifier of the data item; and

referencing means for obtaining the identifier of a data item in the system given a contextual name for the data item, using said context association.

5 13. An apparatus as in claim 12, further comprising:

assignment means for assigning a data item to a contextual name, invoking said identity means to determine the identifier of the data item, and invoking said context means to make or modify the context association between the contextual name of the data item and the identifier of the data item.

14. An apparatus as in claim 12, further comprising:

15 data associating means for making and maintaining, for a data item in the system, an association between the data item and the identifier of the data item;

20 access means for accessing a particular data item using the identifier of the particular data item; and

contextual name access means for accessing a data item in the system for a given context name of the data item, determining the data identifier associated with the given context name, and invoking said access means to access the data item using the data identifier.

15. An apparatus as in claim 11, further comprising:

30 transparent access means for accessing a data item from one of several locations, using the identifier of the data item, said transparent access means invoking said local existence means to determine if the particular data item is present at the current location, and, in the case when the particular data item is not present at the

current location, invoking said requesting means to obtain the data item from a remote location.

16. An apparatus as in claim 15, further comprising:

5 identifier copy means for copying an identifier of a data item from a source location to a destination location.

17. An apparatus as in claim 15, further comprising:

10 context means for making and maintaining a context association between a contextual name of a data item in the system and the identifier of the data item;
context copy means for copying a data item from a source location to a destination location, given the
15 contextual name of the data item, by copying only the context association between the contextual identifier and the data identifier from the source location to the destination location; and

20 transparent referencing means for obtaining a data item from one of several locations the system given a contextual name for the data item, said transparent referencing means invoking said context association to determine the data identifier of a data item given a contextual name, and invoking said transparent access
25 means to access the data item from one of several locations given the identifier of the data item.

18. An apparatus as in claim 1, wherein at least some of said data items are compound data items, each compound data item including at least some component
30 data items in a fixed sequence, and wherein the identity means determines the identifier of a compound data item based on each component data item of the compound data item.

19. An apparatus as in claim 18, wherein said compound data items are files and said component data items are segments, and wherein the identity means determines the identifier of a file based on the identifier of each data segment of the file.

20. An apparatus as in claim 18, wherein said compound data items are directories and said component data items are files or subordinate directories, and wherein the identity means determines the identifier of a given directory based on each file and subordinate directory within the given directory.

21. An apparatus as in claim 11, further comprising:
means for advertising a data item from a location in the system to at least one other location in the system, said means for advertising providing each of said at least one other location with the data identifier of the data item, and providing the data item to only those locations of said other locations that request said data item in response to said providing.

22. An apparatus as in claim 18, further comprising:
local existence means for determining whether a particular data item is present at a particular location in the system, based on the identifier of the data item; and

compound copy means for copying a data item from a source to a destination in the data processing system, said compound copy means invoking said local existence means to determine whether the data item is present at the destination, and to determine, when the data item is a compound data item, whether the component data items of the compound data item are present at the destination, and providing said destination with the data

item only if said local existence means determines that
the data item is not present at the destination, and
providing said destination with each component data item
only if said local existence means determines that the
5 component data item is not present at the destination.

23. An apparatus as in claim 11, further
comprising:

means for verifying the integrity a data item
obtained from said requesting means in response to
10 providing said requesting with a particular data
identifier, to confirm that the data item obtained from
the requesting means is the same data item as the data
item requested, said verifying means invoking said
identity means to determine the data identifier of the
15 obtained data item, and comparing said determined data
identifier with said particular data identifier to verify
said obtained data item.

24. An apparatus as in claim 2, wherein a
location is at least one of a storage location and a
20 processing location, and wherein a storage location is at
least one of a data storage device and a data storage
volume, and wherein a processing location is at least one
of a data processor and a computer.

25. An apparatus as in claim 3, wherein at
25 least some of said data items are compound data items,
each compound data item including at least some component
data items in a fixed sequence, and wherein the identity
means determines the identifier of a compound data item
based on the identifier of each component data item of
30 the compound data item.

26. An apparatus as in claim 3, further
comprising:

context associating means for making and
maintaining a context association, for any data item in
the system, between the identifier of the data item and
at least one contextual name of the data item at a
5 particular location in the system;

means for obtaining the identifier of a data
item in the system given a contextual name for the data
item at a particular location in the system; and

logical copy means for associating the data
10 identifier corresponding to a contextual name at a source
location with a contextual name at a destination location
in the data processing system.

27. An apparatus as in claim 25, wherein said
compound data items are files and said component data
15 items are segments, and wherein the identity means
determines the identifier of a file based on the
identifier of each data segment of the file.

28. An apparatus as in claim 25, further
comprising:
20 compound copy means for copying a data item
from a source location to a destination location in the
data processing system, said compound copy means invoking
said local existence means to determine whether the data
item is present at the destination, and to determine,
25 when the data item is a compound data item, whether the
component data items of the compound data item are
present at the destination, and providing said
destination with the data item only if said local
existence means determines that the data item is not
30 present at the destination, and providing said
destination with each component data item only if said
local existence means determines that the component data
item is not present at the destination.

29. An apparatus as in any of claims 1-28, wherein a data item is at least one of a file, a database record, a message, a data segment, a data block, a directory, and an instance an object class.

5 30. A method of identifying a data item in a data processing system for subsequent access to the data item, the method comprising the steps of:

 determining a substantially unique identifier for the data item, said identifier depending on all of
10 the data in the data item and on the data in the data item; and

 accessing a data item in the system using the identifier of the data item.

15 31. A method as in claim 30, further comprising the step of:

 making and maintaining, for a plurality of data items in the system, an association between each of the data items and the identifier of each of the data items, wherein said accessing step accesses a data item via the
20 association.

 32. A method as in claim 31, further comprising the step of:

 assimilating a new data item into the system, by determining the identifier of the new data item and
25 associating the new data item with its identifier.

 33. A method for duplicating a given data item from a source location to a destination location in a data processing system, the method comprising the steps of:

30 determining a substantially unique identifier for the given data item, said identifier depending on all of the data in the data item and only on the data in the data item;

determining, using said data identifier,
whether said data item is present at said destination
location; and

5 based on said determining, providing said
destination location with said data item only if said
data item is not present at said destination.

34. A method as in claim 33, wherein said
given data item is a compound data item having a
plurality of component data items, the method further
10 comprising the steps of:

for each data item of said component data
items,

obtaining the component data
15 identifier of the data item by determining a
substantially unique identifier for the data
item, said identifier depending on all of the
data in the data item and only on the data in
the data item;

determining, using said obtained
20 component data identifier, whether said data
item is present at said destination; and

based on said determining, providing
said destination with said data item only if
said data item is not present at said
25 destination.

35. A method for determining whether a
particular data item is present in a data processing
system, the method comprising the steps of:

(A) for each data item of a plurality of data
30 items in the system,

(i) determining a substantially unique
identifier for the data item, said identifier
depending on all of the data in the data item
and only on the data in the data item; and

(ii) making and maintaining a set of identifiers of said plurality of data items; and

(B) for the particular data item,

(i) determining a particular substantially unique identifier for the data item, said identifier depending on all of the data in the data item and only on the data in the data item; and

(ii) determining whether said particular identifier is in said set of data items.

36. A method of backing up, of a plurality of data items, data items modified since a previous backup time in a data processing system, the method comprising the steps of:

(A) maintaining a backup record of identifiers of data items backed up at the previous backup time; and

(B) for each of said plurality of data items, (i) determining a substantially unique identifier for the data item, said identifier depending on all of the data in the data item and only on the data in the data item;

(ii) determining those data items of the plurality of data items whose identifiers are not in the backup record; and

(iii) based on said determining, copying only those data items whose data identities are not recorded in the backup record.

37. A method as in claim 36, further comprising the step of:

recording in the backup record the identifiers of those data items copied in said step of copying.

38. A method of locating a particular data item at a location in a data processing system, the
5 method comprising the steps of:

- (A) determining a substantially unique identifier for the data item, said identifier depending on all of the data in the data item and only on the data in the data item;
- 10 (B) requesting the particular data item by sending the data identifier of the data item from the requestor location to at least one location of a plurality of provider locations in the system; and
- 15 (C) on at least some of said provider locations,
 - (a) for each data item of a plurality of data items at said provider locations,
 - 20 (i) determining a substantially unique identifier for the data item, said identifier depending on all of the data in the data item and only on the data in the data item; and
 - (ii) making and maintaining a set of
25 identifiers of data items,
 - (b) determining, based on said set of identifiers, whether the data item corresponding to the requested data identifier is present at said provider
30 location; and
 - (c) based on said determining, when said provider location determines that the particular data item is present at the provider location, notifying said
35 requestor that the provider has a copy of the given data item.

39. The method of claim 38, further comprising the steps of:

- 5 (a) for each data item of a plurality of data items at said provider locations,
making and maintaining an association between the data item and the identifier of the data item,
10 (b) in response to said notifying, said client location copying said data item from one of said responding remote locations, using said association to access the data item given the data identifier.

40. A method of locating a particular data item among a plurality of locations, each of said locations having a plurality of data items, the method comprising the steps of:

- 15 determining, for the particular data item and for each data item of the plurality of data items, a substantially unique identifier for the data item, said identifier depending on all of the data in the data item and only on the data in the data item; and
20 determining the presence of the particular data item in each of said plurality of locations by determining whether the identifier of the particular data item is present at each of said locations.
25

41. The method of claim 30, wherein said step of accessing further comprises the steps of, for a given data identifier and for a given current location and a remote location in the system:

- 30 determining whether the data item corresponding to the given data identifier is present at the current location, and
based on said determining, if said data item is not present at the current location, fetching the data

item from a remote location in the system to the current location.

42. The method of claim 41, further comprising the steps of:

5 for each contextual name at a location,
 making and maintaining a context
 association between the context name of a data item and
 the identifier of said data item, and when some context
 association changes at said current location, and
10 notifying said remote location of a
 modification to the context association.

43. The method of claim 42, further comprising the step of:

15 at said remote location, updating the
 association between the contextual identifier of the data
 item and the identifier of the data item.

44. The method of claim 43, further comprising the step of:

20 from said remote location, notifying all other
 locations that said data item has been modified, by
 providing the contextual identifier and data identifier
 of said data item to said other locations.

45. The method of claim 44, further comprising the step of, at each location notified that the data item
25 has been modified:

 modifying an association between the contextual
 identifier of the data item and the data identifier of
 the data item, to record that the data item has been
 modified.

30 46. A method of eliminating a data item at a
 given location in a data processing system when said data

item can be obtained from another location in the system,
the method comprising the steps of:

5 determining a substantially unique identifier
for the data, said identifier depending on all of the
data in the data item and only on the data in the data
item;

 making and maintaining a source association
between the data identifier and at least one location at
which said data item is known to be present; and
10 based on said source association, if said data
item is present at said other location, removing the data
item from the given location.

47. A method of deleting a data item from a
location in a data processing system, the method
15 comprising the steps of:

 for each of a plurality of data items in the
system:

 determining a substantially unique identifier
for the data, said identifier depending on all of the
20 data in the data item and only on the data in the data
item; and

 making and maintaining, an association between
each of the data items and the unique identifier of the
data items; and

25 for a given data item:

 determining a substantially unique identifier
for the data, said identifier depending on all of the
data in the data item and only on the data in the data
item; and

30 determining whether a contextual identifier or
a compound data item or a remote processor in the system
refers to the unique identifier of the data item, and
based on said determining, deleting said data item and
its association if no other contextual identifier or
35 compound data item or remote processor refers to said
data item.