RUSS AUGUST & KABAT
Marc A. Fenster, State Bar No. 181067
Andrew D. Weiss, State Bar No. 232974
12424 Wilshire Boulevard, 12th Floor
Los Angeles, California 90025
Tel: (310) 826-7474
Fax: (310) 826-6991
Email: mfenster@raklaw.com
Email: aweiss@raklaw.com

Attorneys for Proxyconn, Inc.

UNITED STATES DISTRICT COURT

CENTRAL DISTRICT OF CALIFORNIA

SOUTHERN DIVISION

PROXYCONN, INC.,

Plaintiff

vs.

MICROSOFT CORPORATION; HEWLETT-PACKARD COMPANY; ACER AMERICA CORPORATION; and DELL INC.,

Defendants.

Case No. SA CV 11 -1681-DOC(JPRx) [Consolidated with Case Nos. SA CV11-1682 DOC (JPRx), SA CV11-1683 DOC (JPRx), SA CV11-1684 DOC (JPRx), and SA CV12-0889 DOC (JPRx)]

THIRD AMENDED CONSOLIDATED COMPLAINT FOR PATENT INFRINGEMENT

JURY TRIAL DEMANDED

Plaintiff Proxyconn, Inc. ("Proxyconn") alleges as follows:

PARTIES

1. Plaintiff Proxyconn is a California corporation with its principal place of business located at 3211 S. Shannon Street, Santa Ana, California 92704.

2. Defendant Microsoft Corporation ("Microsoft") is a Washington corporation with its principal place of business at One Microsoft Way, Redmond, Washington 98052. Microsoft has appointed Corporation Service Company, 2730 Gateway Oaks Drive, Suite 100, Sacramento, California 95833, as its agent for service of process.

3. Defendant Hewlett-Packard Company ("HP") is a Delaware corporation with its principal place of business at 3000 Hanover Street, Palo Alto, California 94304. HP has appointed CT Corporation System, 818 W. Seventh Street, Los Angeles, California 90017, as its agent for service of process.

4. Defendant Acer America Corporation ("Acer") is a California corporation with its principal place of business at 333 West San Carlos Street, Suite 1500, San Jose, California 95110. Acer has appointed C T Corporation System, 818 West 7th Street, Los Angeles, California 90017, as its agent for service of process.

5. Defendant Dell Inc. ("Dell") is a Delaware corporation with its principal place of business at 1 Dell Way, Round Rock, Texas 78682. Dell has appointed Corporation Service Company, 2711 Centerville Road, Suite 400, Wilmington, Delaware 19808, as its agent for service of process.

6. Microsoft, HP, Acer and Dell shall be referred to collectively as "Defendants."

JURISDICTION AND VENUE

7. This action arises under the patent laws of the United States, Title 35 of the United States Code. Accordingly, this Court has subject matter jurisdiction under 28 U.S.C. §§ 1331 and 1338(a).

8. Venue is proper in this District under 28 U.S.C. §§ 1391 (b)-(d) and 1400(b) because each defendant is subject to personal jurisdiction in this District, has committed acts of patent infringement in this District, or has a regular and established place of business in this District.

9. Joinder is appropriate in this case pursuant to 35 U.S.C. §299. On information and belief, Microsoft has agreed to indemnify and defend HP, Acer and Dell because of the relation of Proxyconn's claims to Microsoft's products. Defendants have agreed to the consolidation of the pending actions filed by Proxyconn against Defendants.[1]

FACTUAL BACKGROUND

10. Proxyconn was founded in 2001 in Santa Ana, California. It remains based in Santa Ana today.

11. Proxyconn was started to address the problem of demands by users of networks, such as the Internet, to instantly receive content over the network. While other solutions simply relied on increasing available bandwidth, Proxyconn sought a more intelligent solution that could be used with existing technology and bandwidth.

12. As a result, Proxyconn created a technology that used existing technological limitations while making the use of networks effectively many times faster than previously possible. Proxyconn filed a patent application on its novel technology. As a result of that patent application, Proxyconn was awarded United States Patent No. 6,757,717 ("the '717 patent").

13. Proxyconn's technology and method were used by hundreds of ISPs and hundreds of thousands of users in the United States and throughout the world, and is still being used.

[1]The actions consolidated into this one are: *Proxyconn, Inc. v. Hewlett-Packard Company*, Case No. SA CV 11-1682-DOC, *Proxyconn, Inc. v. Dell Inc.*, Case No. SA CV 11-1683-DOC, *Proxyconn, Inc. v. Acer America Corporation*, Case No. SA CV 11-1684-DOC and *Proxyconn, Inc. v. Microsoft et al.*, Case No. SA CV 12-889-DOC.

14. Proxyconn is the owner by assignment of the '717 patent. The '717 patent is entitled "System and Method for Data Access." The '717 patent issued on June 29, 2004. A true and correct copy of the '717 patent is attached hereto as Exhibit A.

COUNT I

(Infringement of U.S. Patent No. 6,757,717 Against Microsoft)

15. Microsoft has been and still is directly (literally and under the doctrine of equivalents) infringing at least claims 1, 10, 11 and 22 of the '717 patent, literally and under the doctrine of equivalents, by making, using, selling, offering to sell, or importing, without license or authority, software that creates, transmits, receives, or compares digital digests on data, including, but not limited to, its use of Remote Differential Compression ("RDC") technology in at least its Windows Server 2003 R2, Windows Server 2008, Windows Small Business Server 2003, Windows Small Business Server 2008, Windows Small Business Server 2011, Windows XP with Service Pack 3, Windows Vista, and Windows 7 operating systems and its use of BranchCache. For example, on information and belief, Microsoft uses its Distributed File System ("DFS") Replication product, which uses RDC, on its servers. *See* http://msdn.microsoft.com/en-us/library/windows/desktop/bb540025%28v=vs.85%29.aspx. Attached as Exhibit B to this complaint is an exemplary chart illustrating how Microsoft's making, using, selling, offering to sell, or importing, without license or authority, DFS Replication and RDC infringes claims 1, 10, 11 and 22 of the '717 patent.

16. Since November 3, 2011, Microsoft has been and still is indirectly infringing, by way of inducing infringement by others of the '717 patent, by, among other things, making, using, importing, offering for sale, and/or selling, without license or authority, software for use in systems that thereby fall within the scope of at least claims 1, 10, 11 and 22 of the '717 patent. Such software includes, but is not limited to, the Remote Differential Compression ("RDC") technology

used in at least its Windows Server 2003 R2, Windows Server 2008, Windows Small Business Server 2003, Windows Small Business Server 2008, Windows Small Business Server 2011, Windows XP with Service Pack 3, Windows Vista, and Windows 7 operating systems and the BranchCache technology. This software is used in infringing computer systems made, used, imported, offered for sale, and/or sold by direct infringers of the '717 patent in the United States, such as computer manufacturers (for example, HP, Dell and Acer) and end-users (for example, customers that purchase Microsoft's software and use it in their computer systems). The systems using Microsoft's software include a sender computer and a receiver computer communicating through a network, with each computer equipped with a method for creating digital digests on data and the receiving computer including a means for comparing digital digests. Microsoft induces others to directly infringe by inducing or encouraging the use of its infringing RDC and BrachCache technologies. *See, e.g.*, http://technet.microsoft.com/en-us/library/cc754372.aspx and http://msdn.microsoft.com/en-us/library/windows/desktop/aa372963%28v=VS.85%29.aspx. Since November 3, 2011, when the original complaint in *Proxyconn, Inc. v. Microsoft Corp. et al.*, Case No. 11-cv-1681-DOC was filed, Microsoft has had knowledge of the '717 patent and, by continuing the actions described above, has had the specific intent to, or were willfully blind to the fact that its actions would, induce infringement of the '717 patent. *See, e.g.*, http://technet.microsoft.com/en-us/library/cc754372.aspx and http://msdn.microsoft.com/en-us/library/windows/desktop/aa372963%28v=VS.85%29.aspx. Indeed, Microsoft has been aware of Proxyconn and its products since at least the summer of 2003, when Microsoft reviewed Proxyconn's technology. On information and belief, as a result of its awareness of Proxyconn and its technology, Microsoft has been aware of the patent since it issued on June 29, 2004. Thus, by making, using, importing, offering for sale, and/or selling such software, Microsoft has injured Proxyconn

and is thus liable to Proxyconn for infringement of the '717 patent under 35 U.S.C. § 271(b) after November 3, 2011.

17. Since November 3, 2011, Microsoft has also been and still is indirectly infringing, by way of contributing to the infringement by others of the '717 patent, by, among other things, making, using, importing, offering for sale, and/or selling, without license or authority, software for use in systems that thereby fall within the scope of at least claims 1, 10, 11 and 22 of the '717 patent. Such software includes, but is not limited to, the RDC technology used in at least its Windows Server 2003 R2, Windows Server 2008, Windows Small Business Server 2003, Windows Small Business Server 2008, Windows Small Business Server 2011, Windows XP with Service Pack 3, Windows Vista, and Windows 7 operating systems and its BranchCache technology. This software is used in infringing computer systems made, used, imported, offered for sale, and/or sold by direct infringers of the '717 patent in the United States, such as computer manufacturers (for example, HP, Dell and Acer) and end-users (for example, customers that purchase Microsoft's software and install it in this computer systems). The systems using Microsoft's software include a sender computer and a receiver computer communicating through a network, with each computer equipped with a method for creating digital digests on data and the receiving computer including a means for comparing digital digests. Microsoft contributes to others directly infringing by inducing or encouraging the use of its infringing RDC and BranchCache technologies. *See, e.g.*, http://technet.microsoft.com/en-us/library/cc754372.aspx and http://msdn.microsoft.com/en-us/library/windows/desktop/aa372963%28v=VS.85%29.aspx. Microsoft's accused software, including the RDC and BrachCache technologies, is a material part of the invention, and is especially made or especially adapted for use in the infringement of '717 patent and is not a staple article or commodity of commerce suitable for substantial noninfringing uses. Since November 3, 2011, when the original

complaint in *Proxyconn, Inc. v. Microsoft Corp. et al.*, Case No. SA CV 11-1681-DOC was filed, Microsoft has had knowledge of the '717 patent and, by continuing the actions described above, has had the specific intent to, or were willfully blind to the fact that its actions would, induce infringement of the '717 patent. *See, e.g.*, http://technet.microsoft.com/en-us/library/cc754372.aspx and http://msdn.microsoft.com/en-us/library/windows/desktop/aa372963%28v=VS.85%29.aspx. Indeed, Microsoft has been aware of Proxyconn and its products since the summer of 2003, when Microsoft reviewed Proxyconn's technology. On information and belief, as a result of its awareness of Proxyconn and its technology, Microsoft has been aware of the patent since it issued on June 29, 2004. Thus, by making, using, importing, offering for sale, and/or selling such software, Microsoft has injured Proxyconn and is thus liable to Proxyconn for infringement of the '717 patent under 35 U.S.C. § 271(c) after November 3, 2011.

18. To the extent that facts learned in discovery show that Microsoft's infringement of the '717 patent is or has been willful, Proxyconn reserves the right to request such a finding at time of trial.

19. To the extent necessary, Proxyconn has marked its products pursuant to 35 U.S.C. § 287(a).

20. As a result of Microsoft's infringement of the '717 patent, Proxyconn has suffered monetary damages in an amount adequate to compensate for Microsoft's infringement, but in no event less than a reasonable royalty for the use made of the invention by Microsoft, together with interest and costs as fixed by the Court, and Proxyconn will continue to suffer damages in the future unless Microsoft's infringing activities are enjoined by this Court.

21. Unless a permanent injunction is issued enjoining Microsoft and its agents, servants, employees, representatives, affiliates, and all others acting or in

active concert therewith from infringing the '717 patent, Proxyconn will be greatly and irreparably harmed.

COUNT II

(Infringement of U.S. Patent No. 6,757,717 Against HP)

22. HP has been and still is directly (literally and under the doctrine of equivalents) infringing at least claims 1, 10, 11 and 22 of the '717 patent, literally and under the doctrine of equivalents, by making, using, selling, offering to sell, or importing, without license or authority, computer systems that include a sender computer and a receiver computer communicating through a network, with each computer equipped with a method for creating digital digests on data and the receiving computer including a means for comparing digital digests. In particular, these computer systems contain software including, but not limited to, the Remote Differential Compression ("RDC") technology used in at least Microsoft's Windows Server 2003 R2, Windows Server 2008, Windows Small Business Server 2003, Windows Small Business Server 2008, Windows Small Business Server 2011, Windows XP with Service Pack 3, Windows Vista, and Windows 7 operating systems.

23. For example, HP directly infringes claims 1 and 10 by making, using, selling, offering to sell or importing, without license or authority, the computer systems and software described above to its customers. Indeed, HP offers a training course to its customers that includes teaching the use of Microsoft's DFS Replication product, which uses RDC. *See* http://www.hp.com/education/courses/hf847s.html?jumpid=reg_r1002_useni.

24. As another example, on information and belief, HP directly infringes claims 11 and 22 by using Microsoft's DFS Replication product on its internal servers, thereby practicing the claimed methods.

25. Attached as Exhibit B to this complaint is an exemplary chart illustrating how HP's making, using, selling, offering to sell, or importing, without

license or authority, of its computer systems and software described above infringes claims 1, 10, 11 and 22 of the '717 patent.

26. Since November 3, 2011, HP has been and still is indirectly infringing, by way of inducing infringement by others of the '717 patent, by, among other things, making, using, importing, offering for sale, and/or selling, without license or authority, personal computers and servers for use in systems that thereby fall within the scope of at least claims 1, 10, 11 and 22 of the '717 patent. Such personal computers and servers include a sender computer and a receiver computer communicating through a network, with each computer equipped with a method for creating digital digests on data and the receiving computer including a means for comparing digital digests. In particular, these computer systems contain software including, but not limited to, the Remote Differential Compression ("RDC") technology used in at least Microsoft's Windows Server 2003 R2, Windows Server 2008, Windows Small Business Server 2003, Windows Small Business Server 2008, Windows Small Business Server 2011, Windows XP with Service Pack 3, Windows Vista, and Windows 7 operating systems. HP induces its customers and end users to directly infringe by inducing or encouraging the use of the infringing RDC technology. *See* http://www.hp.com/education/courses/hf847s.html?jumpid=reg_r1002_useni. Since November 3, 2011, when the original complaint in *Proxyconn, Inc. v. Hewlett-Packard Company*, Case No. 11-cv-1682-DOC was filed, HP has had knowledge of the '717 patent and, by continuing the actions described above, has had the specific intent to, or were willfully blind to the fact that its actions would, induce infringement of the '717 patent. *See* http://www.hp.com/education/courses/hf847s.html?jumpid=reg_r1002_useni. Thus, by making, using, importing, offering for sale, and/or selling such personal computers and servers, HP has injured Proxyconn and is thus liable to Proxyconn

for infringement of the '717 patent under 35 U.S.C. § 271(b) after November 3, 2011.

27. Since November 3, 2011, HP has also been and still is indirectly infringing, by way of contributing to the infringement by others of the '717 patent, by, among other things, making, using, importing, offering for sale, and/or selling, without license or authority, systems that thereby fall within the scope of at least claims 1, 10, 11 and 22 of the '717 patent. Such personal computers and servers include a sender computer and a receiver computer communicating through a network, with each computer equipped with a method for creating digital digests on data and the receiving computer including a means for comparing digital digests. In particular, these computer systems contain software including, but not limited to, the Remote Differential Compression ("RDC") technology used in at least Microsoft's Windows Server 2003 R2, Windows Server 2008, Windows Small Business Server 2003, Windows Small Business Server 2008, Windows Small Business Server 2011, Windows XP with Service Pack 3, Windows Vista, and Windows 7 operating systems. HP contributes to its customers directly infringing by inducing or encouraging the use of its infringing RDC technology. *See* http://www.hp.com/education/courses/hf847s.html?jumpid=reg_r1002_useni. HP's systems and software are a material part of the invention, and are especially made or especially adapted for use in the infringement of '717 patent and are not a staple article or commodity of commerce suitable for substantial noninfringing uses. Since November 3, 2011, when the original complaint in *Proxyconn, Inc. v. Hewlett-Packard Company*, Case No. 11-cv-1682-DOC was filed, HP has had knowledge of the '717 patent and, by continuing the actions described above, has had the specific intent to, or were willfully blind to the fact that its actions would, induce infringement of the '717 patent. *See* http://www.hp.com/education/courses/hf847s.html?jumpid=reg_r1002_useni. Thus, by making, using, importing, offering for sale, and/or selling such software,

HP has injured Proxyconn and is thus liable to Proxyconn for infringement of the '717 patent under 35 U.S.C. § 271(c) after November 3, 2011.

28. To the extent that facts learned in discovery show that HP's infringement of the '717 patent is or has been willful, Proxyconn reserves the right to request such a finding at time of trial.

29. To the extent necessary, Proxyconn has marked its products pursuant to 35 U.S.C. § 287(a).

30. As a result of HP's infringement of the '717 patent, Proxyconn has suffered monetary damages in an amount adequate to compensate for HP's infringement, but in no event less than a reasonable royalty for the use made of the invention by HP, together with interest and costs as fixed by the Court, and Proxyconn will continue to suffer damages in the future unless HP's infringing activities are enjoined by this Court.

31. Unless a permanent injunction is issued enjoining HP and its agents, servants, employees, representatives, affiliates, and all others acting or in active concert therewith from infringing the '717 patent, Proxyconn will be greatly and irreparably harmed.

COUNT III

(Infringement of U.S. Patent No. 6,757,717 Against Acer)

32. Acer has been and still is directly (literally and under the doctrine of equivalents) infringing at least claims 1, 10, 11 and 22 of the '717 patent, literally and under the doctrine of equivalents, by making, using, selling, offering to sell, or importing, without license or authority, computer systems that include a sender computer and a receiver computer communicating through a network, with each computer equipped with a method for creating digital digests on data and the receiving computer including a means for comparing digital digests. In particular, these computer systems contain software including, but not limited to, the Remote Differential Compression ("RDC") technology used in at least Microsoft's

Windows Server 2003 R2, Windows Server 2008, Windows Small Business Server 2003, Windows Small Business Server 2008, Windows Small Business Server 2011, Windows XP with Service Pack 3, Windows Vista, and Windows 7 operating systems.

33. For example, Acer directly infringes claims 1 and 10 by making, using, selling, offering to sell or importing, without license or authority, the computer systems and software described above to its customers.

34. As another example, on information and belief, Acer directly infringes claims 11 and 22 by using Microsoft's DFS Replication product on its internal servers, thereby practicing the claimed methods. Indeed, Acer markets the ability to use Microsoft's DFS Replication on its servers. *See* http://static.acer.com/up/Resource/Acer/Storage/SAN/AN1600_F1/Docs/20110408/AN1600%20F1%20Longspecs%20US%2004_11_11.pdf, at 3-4.

35. Attached as Exhibit B to this complaint is an exemplary chart illustrating how Acer's making, using, selling, offering to sell, or importing, without license or authority, of its computer systems and software described above infringes claims 1, 10, 11 and 22 of the '717 patent.

36. Since November 3, 2011, Acer has been and still is indirectly infringing, by way of inducing infringement by others of the '717 patent, by, among other things, making, using, importing, offering for sale, and/or selling, without license or authority, personal computers and servers for use in systems that thereby fall within the scope of at least claims 1, 10, 11 and 22 of the '717 patent. Such personal computers and servers include a sender computer and a receiver computer communicating through a network, with each computer equipped with a method for creating digital digests on data and the receiving computer including a means for comparing digital digests. In particular, these computer systems contain software including, but not limited to, the Remote Differential Compression ("RDC") technology used in at least Microsoft's Windows Server

2003 R2, Windows Server 2008, Windows Small Business Server 2003, Windows Small Business Server 2008, Windows Small Business Server 2011, Windows XP with Service Pack 3, Windows Vista, and Windows 7 operating systems. Acer induces its customers and end users to directly infringe by inducing or encouraging the use of the infringing RDC technology. *See* http://static.acer.com/up/Resource/Acer/Storage/SAN/AN1600_F1/Docs/20110408/AN1600%20F1%20Longspecs%20US%2004_11_11.pdf, at 3-4. Since November 3, 2011, when the original complaint in *Proxyconn, Inc. v. Acer America Corporation*, Case No. 11-cv-1684-DOC was filed, Acer has had knowledge of the '717 patent and, by continuing the actions described above, has had the specific intent to, or were willfully blind to the fact that its actions would, induce infringement of the '717 patent. *See* http://static.acer.com/up/Resource/Acer/Storage/SAN/AN1600_F1/Docs/20110408/AN1600%20F1%20Longspecs%20US%2004_11_11.pdf, at 3-4. Thus, by making, using, importing, offering for sale, and/or selling such personal computers and servers, Acer has injured Proxyconn and is thus liable to Proxyconn for infringement of the '717 patent under 35 U.S.C. § 271(b) after November 3, 2011.

37. Since November 3, 2011, Acer has also been and still is indirectly infringing, by way of contributing to the infringement by others of the '717 patent, by, among other things, making, using, importing, offering for sale, and/or selling, without license or authority, systems that thereby fall within the scope of at least claims 1, 10, 11 and 22 of the '717 patent. Such personal computers and servers include a sender computer and a receiver computer communicating through a network, with each computer equipped with a method for creating digital digests on data and the receiving computer including a means for comparing digital digests. In particular, these computer systems contain software including, but not limited to, the Remote Differential Compression ("RDC") technology used in at least Microsoft's Windows Server 2003 R2, Windows Server 2008, Windows

Small Business Server 2003, Windows Small Business Server 2008, Windows Small Business Server 2011, Windows XP with Service Pack 3, Windows Vista, and Windows 7 operating systems. Acer contributes to its customers directly infringing by inducing or encouraging the use of its infringing RDC technology. *See* http://static.acer.com/up/Resource/Acer/Storage/SAN/AN1600_F1/Docs/20110408/AN1600%20F1%20Longspecs%20US%2004_11_11.pdf, at 3-4. Acer's systems and software are a material part of the invention, and are especially made or especially adapted for use in the infringement of '717 patent and are not a staple article or commodity of commerce suitable for substantial noninfringing uses. Since November 3, 2011, when the original complaint in *Proxyconn, Inc. v. Acer America Corporation*, Case No. 11-cv-1684-DOC was filed, Acer has had knowledge of the '717 patent and, by continuing the actions described above, has had the specific intent to, or were willfully blind to the fact that its actions would, induce infringement of the '717 patent. *See* http://static.acer.com/up/Resource/Acer/Storage/SAN/AN1600_F1/Docs/20110408/AN1600%20F1%20Longspecs%20US%2004_11_11.pdf, at 3-4. Thus, by making, using, importing, offering for sale, and/or selling such software, Acer has injured Proxyconn and is thus liable to Proxyconn for infringement of the '717 patent under 35 U.S.C. § 271(c) after November 3, 2011.

38. To the extent that facts learned in discovery show that Acer's infringement of the '717 patent is or has been willful, Proxyconn reserves the right to request such a finding at time of trial.

39. To the extent necessary, Proxyconn has marked its products pursuant to 35 U.S.C. § 287(a).

40. As a result of Acer's infringement of the '717 patent, Proxyconn has suffered monetary damages in an amount adequate to compensate for Acer's infringement, but in no event less than a reasonable royalty for the use made of the

invention by Acer, together with interest and costs as fixed by the Court, and Proxyconn will continue to suffer damages in the future unless Acer's infringing activities are enjoined by this Court.

41. Unless a permanent injunction is issued enjoining Acer and its agents, servants, employees, representatives, affiliates, and all others acting or in active concert therewith from infringing the '717 patent, Proxyconn will be greatly and irreparably harmed.

COUNT IV

(Infringement of U.S. Patent No. 6,757,717 Against Dell)

42. Dell has been and still is directly (literally and under the doctrine of equivalents) infringing at least claims 1, 10, 11 and 22 of the '717 patent, literally and under the doctrine of equivalents, by making, using, selling, offering to sell, or importing, without license or authority, computer systems that include a sender computer and a receiver computer communicating through a network, with each computer equipped with a method for creating digital digests on data and the receiving computer including a means for comparing digital digests. In particular, these computer systems contain software including, but not limited to, the Remote Differential Compression ("RDC") technology used in at least Microsoft's Windows Server 2003 R2, Windows Server 2008, Windows Small Business Server 2003, Windows Small Business Server 2008, Windows Small Business Server 2011, Windows XP with Service Pack 3, Windows Vista, and Windows 7 operating systems.

43. For example, Dell directly infringes claims 1 and 10 by making, using, selling, offering to sell or importing, without license or authority, the computer systems and software described above to its customers. Indeed, Dell markets the ability to use Microsoft's DFS Replication on its servers. *See* http://content.dell.com/us/en/business/d/business~solutions~power~en/Documents

~ps1q10-20100266-Sherbak.pdf.aspx?c=us&cs=OWR08&l=en&s=bsd&redirect=1.

44. As another example, on information and belief, Dell directly infringes claims 11 and 22 by using Microsoft's DFS Replication product on its internal servers, thereby practicing the claimed methods.

45. Attached as Exhibit B to this complaint is an exemplary chart illustrating how Dell's making, using, selling, offering to sell, or importing, without license or authority, of its computer systems and software described above infringes claims 1, 10, 11 and 22 of the '717 patent.

46. Since November 3, 2011, Dell has been and still is indirectly infringing, by way of inducing infringement by others of the '717 patent, by, among other things, making, using, importing, offering for sale, and/or selling, without license or authority, personal computers and servers for use in systems that thereby fall within the scope of at least claims 1, 10, 11 and 22 of the '717 patent. Such personal computers and servers include a sender computer and a receiver computer communicating through a network, with each computer equipped with a method for creating digital digests on data and the receiving computer including a means for comparing digital digests. In particular, these computer systems contain software including, but not limited to, the Remote Differential Compression ("RDC") technology used in at least Microsoft's Windows Server 2003 R2, Windows Server 2008, Windows Small Business Server 2003, Windows Small Business Server 2008, Windows Small Business Server 2011, Windows XP with Service Pack 3, Windows Vista, and Windows 7 operating systems. Dell induces its customers and end users to directly infringe by inducing or encouraging the use of the infringing RDC technology. *See* http://content.dell.com/us/en/business/d/business~solutions~power~en/Documents~ps1q10-20100266-Sherbak.pdf.aspx?c=us&cs=OWR08&l=en&s=bsd&redirect=1. Since November

3, 2011, when the original complaint in *Proxyconn, Inc. v. Dell Inc.*, Case No. 11-cv-1683-DOC was filed, Dell has had knowledge of the '717 patent and, by continuing the actions described above, has had the specific intent to, or were willfully blind to the fact that its actions would, induce infringement of the '717 patent. *See* http://content.dell.com/us/en/business/d/business~solutions~power~en/Documents~ps1q10-20100266-Sherbak.pdf.aspx?c=us&cs=OWR08&l=en&s=bsd&redirect=1. Thus, by making, using, importing, offering for sale, and/or selling such personal computers and servers, Dell has injured Proxyconn and is thus liable to Proxyconn for infringement of the '717 patent under 35 U.S.C. § 271(b) after November 3, 2011.

47. Since November 3, 2011, Dell has also been and still is indirectly infringing, by way of contributing to the infringement by others of the '717 patent, by, among other things, making, using, importing, offering for sale, and/or selling, without license or authority, systems that thereby fall within the scope of at least claims 1, 10, 11 and 22 of the '717 patent. Such personal computers and servers include a sender computer and a receiver computer communicating through a network, with each computer equipped with a method for creating digital digests on data and the receiving computer including a means for comparing digital digests. In particular, these computer systems contain software including, but not limited to, the Remote Differential Compression ("RDC") technology used in at least Microsoft's Windows Server 2003 R2, Windows Server 2008, Windows Small Business Server 2003, Windows Small Business Server 2008, Windows Small Business Server 2011, Windows XP with Service Pack 3, Windows Vista, and Windows 7 operating systems. Dell contributes to its customers directly infringing by inducing or encouraging the use of its infringing RDC technology. *See* http://content.dell.com/us/en/business/d/business~solutions~power~en/Documents

~ps1q10-20100266-Sherbak.pdf.aspx?c=us&cs=OWR08&l=en&s=bsd&redirect=1. Dell's systems and software are a material part of the invention, and are especially made or especially adapted for use in the infringement of '717 patent and are not a staple article or commodity of commerce suitable for substantial noninfringing uses. Since November 3, 2011, when the original complaint in *Proxyconn, Inc. v. Dell Inc.*, Case No. 11-cv-1683-DOC was filed, Dell has had knowledge of the '717 patent and, by continuing the actions described above, has had the specific intent to, or were willfully blind to the fact that its actions would, induce infringement of the '717 patent. *See* http://content.dell.com/us/en/business/d/business~solutions~power~en/Documents~ps1q10-20100266-Sherbak.pdf.aspx?c=us&cs=OWR08&l=en&s=bsd&redirect=1. Thus, by making, using, importing, offering for sale, and/or selling such software, Dell has injured Proxyconn and is thus liable to Proxyconn for infringement of the '717 patent under 35 U.S.C. § 271(c) after November 3, 2011.

48. To the extent that facts learned in discovery show that Dell's infringement of the '717 patent is or has been willful, Proxyconn reserves the right to request such a finding at time of trial.

49. To the extent necessary, Proxyconn has marked its products pursuant to 35 U.S.C. § 287(a).

50. As a result of Dell's infringement of the '717 patent, Proxyconn has suffered monetary damages in an amount adequate to compensate for Dell's infringement, but in no event less than a reasonable royalty for the use made of the invention by Dell, together with interest and costs as fixed by the Court, and Proxyconn will continue to suffer damages in the future unless Dell's infringing activities are enjoined by this Court.

51. Unless a permanent injunction is issued enjoining Dell and its agents, servants, employees, representatives, affiliates, and all others acting or in active concert therewith from infringing the '717 patent, Proxyconn will be greatly and irreparably harmed.

PRAYER FOR RELIEF

Proxyconn prays for the following relief:

1. A judgment that each of the Defendants has directly infringed (either literally or under the doctrine of equivalents) one or more claims of the '717 patent;

2. A judgment that each of the Defendants has indirectly infringed (either literally or under the doctrine of equivalents) one or more claims of the '717 patent since November 3, 2011;

3. A permanent injunction enjoining each of the Defendants and its officers, directors, agents, servants, affiliates, employees, divisions, branches, subsidiaries, parents, and all others acting in active concert or participation with it, from directly or indirectly infringing the '717 patent;

4. An award of damages resulting from each Defendant's acts of direct infringement (either literal or under the doctrine of equivalents) in accordance with 35 U.S.C. § 284;

5. An award of damages resulting from each Defendant's acts of indirect infringement (either literal or under the doctrine of equivalents) in accordance with 35 U.S.C. § 284, beginning at least from the date of the filing of the original complaint against each Defendant;

6. A judgment and order requiring each of the Defendants to provide an accounting and to pay supplemental damages to Proxyconn, including, without limitation, prejudgment and post-judgment interest; and

7. Any and all other relief to which Proxyconn may show itself to be entitled.

JURY TRIAL DEMANDED

Proxyconn hereby demands a trial by jury of all issues so triable.

Dated: November 1, 2012

Respectfully submitted,

RUSS AUGUST & KABAT

By: Marc A. Fenster

Marc A. Fenster, Cal. Bar No. 181067
Email: mfenster@raklaw.com
Andrew D. Weiss, Cal. Bar No. 232974
Email: aweiss@raklaw.com
12424 Wilshire Boulevard, 12th Floor
Los Angeles, California 90025
Telephone: (310) 826-7474
Facsimile: (310) 826-6991

Attorneys for Plaintiff
PROXYCONN, INC.

EXHIBIT A

US006757717B1

(12) **United States Patent**
Goldstein

(10) **Patent No.: US 6,757,717 B1**
(45) **Date of Patent: Jun. 29, 2004**

(54) **SYSTEM AND METHOD FOR DATA ACCESS**

(75) Inventor: **Leonid Goldstein**, Herzlia (IL)

(73) Assignee: **ProxyConn, Inc.**, Irvine, CA (US)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 0 days.

(21) Appl. No.: **09/398,007**

(22) Filed: **Sep. 16, 1999**

(51) **Int. Cl.**[7] **G06F 15/16**
(52) **U.S. Cl.** 709/217; 707/1; 707/10; 709/203; 709/225; 709/229
(58) **Field of Search** 709/217, 203, 709/229, 225; 707/10, 1

(56) **References Cited**

U.S. PATENT DOCUMENTS

4,897,781	A		1/1990	Chang et al.
5,384,565	A		1/1995	Gregory
5,682,514	A	*	10/1997	Yohe et al. 707/1
5,732,265	A		3/1998	Dewitt et al.
5,835,943	A		11/1998	Yohe et al.
5,852,717	A		12/1998	Bhide et al.
5,864,837	A		1/1999	Maimone
5,878,213	A	*	3/1999	Bittinger et al. 707/10
5,909,569	A	*	6/1999	Housel et al. 707/10
5,919,247	A	*	7/1999	Van Hoff et al. 709/217
5,924,116	A		7/1999	Aggarwal et al.
5,931,947	A	*	8/1999	Burns et al. 713/201
5,978,791	A		11/1999	Farber et al.
6,003,087	A	*	12/1999	Housel et al. 707/203
6,006,034	A		12/1999	Heath et al.
6,012,085	A		1/2000	Yohe et al.
6,021,491	A	*	2/2000	Renaud 713/179
6,073,173	A	*	6/2000	Bittinger et al. 707/513
6,085,249	A	*	7/2000	Wang et al. 709/229
6,101,543	A	*	8/2000	Alden et al. 709/229
6,122,637	A	*	9/2000	Yohe et al. 709/204
6,134,583	A	*	10/2000	Herriot 709/217
6,148,340	A	*	11/2000	Bittinger et al. 709/203
6,256,664	B1	*	7/2001	Donoho et al. 709/204
6,279,041	B1	*	8/2001	Baber et al. 709/232

OTHER PUBLICATIONS

RFC2068: Hypertext Transfer Protocol—HTTP/1.1, Jan. 1997, pp. 9–12, 29–35, 70–94, 101–138.
RFC791: Internet Protocol Darpa Internet Program Protocol Specification, Sep. 1981, pp. 3, 11–14, 26.
RFC761: DOD Standard Transmission Control Protocol, Jan. 1980, pp.4, 15–17.
RFC1826: IP Authentication Header, Aug. 1995, pp. 1–10.
RFC1864: The Content-MD5 Header Field, Oct. 1995, pp.1–2.
RFC1321: The MD5 Message-Digest Algorithm, Apr. 1992, pp. 1–21.
Dave Fogle, Ethernet Frame Types, article from Feb. 1996 issue of LAN Magazine/Network Magazine, pp. 1–4.
W. Richard Stevens, TCP/IP Illustrated, vol. 1, 1994, pp. 33–37, 143–147, 223–228.

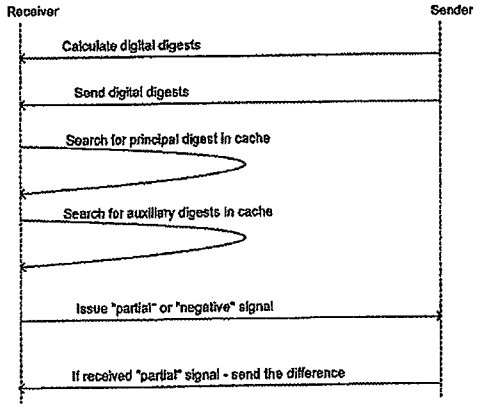
* cited by examiner

Primary Examiner—Jack B. Harvey
Assistant Examiner—Hai V. Nguyen
(74) *Attorney, Agent, or Firm*—J. D. Harriman, II, Esq.; Coudert Brothers LLP

(57) **ABSTRACT**

The invention provides a system for data access in a packet-switched network, including a sender/computer including an operating unit, a first memory, a permanent storage memory and a processor and a remote receiver/computer including an operating unit, a first memory, a permanent storage memory and a processor, the sender/computer and the receiver/computer communicating through the network; the sender/computer further including device for calculating digital digests on data; the receiver/computer further including a network cache memory and device for calculating digital digests on data in the network cache memory; and the receiver/computer and/or the sender/computer including device for comparison between digital digests. The invention also provides a method and apparatus for increased data access in a packet-switched network.

34 Claims, 9 Drawing Sheets



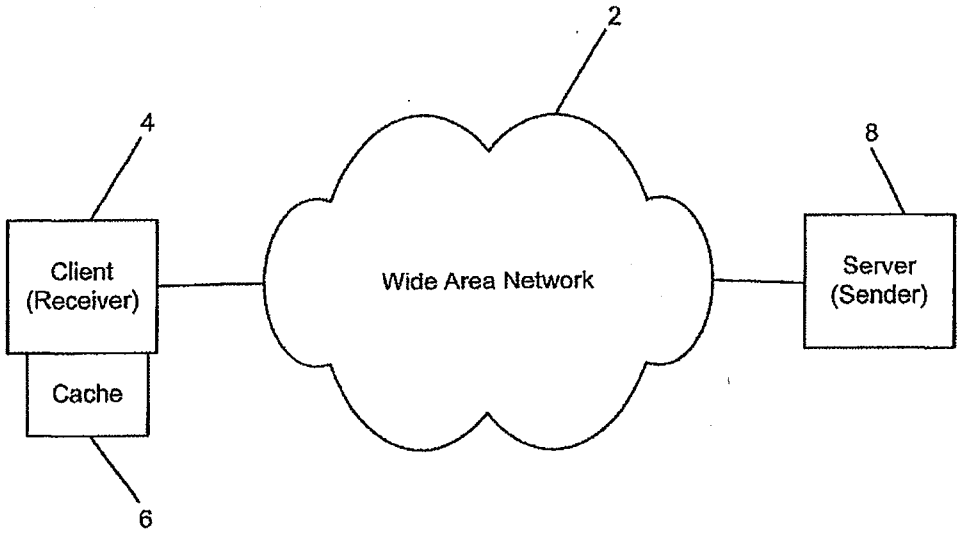


FIG. 1 (PRIOR ART)

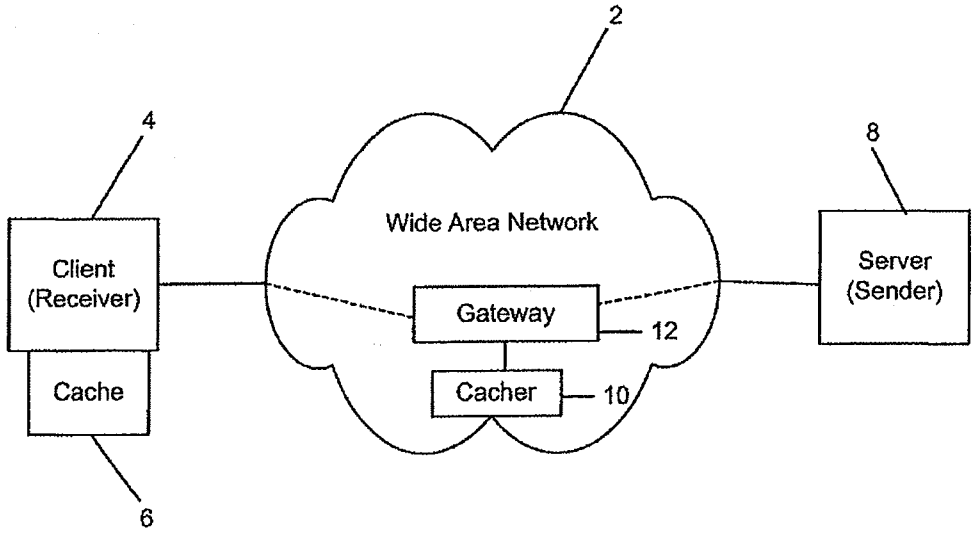


FIG. 2 (PRIOR ART)

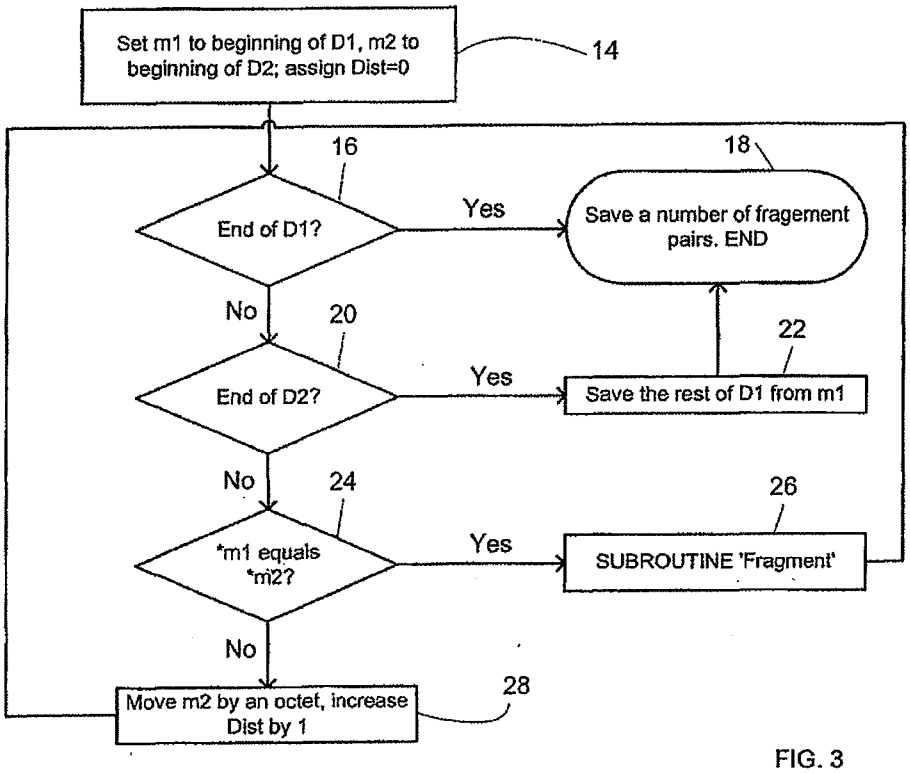
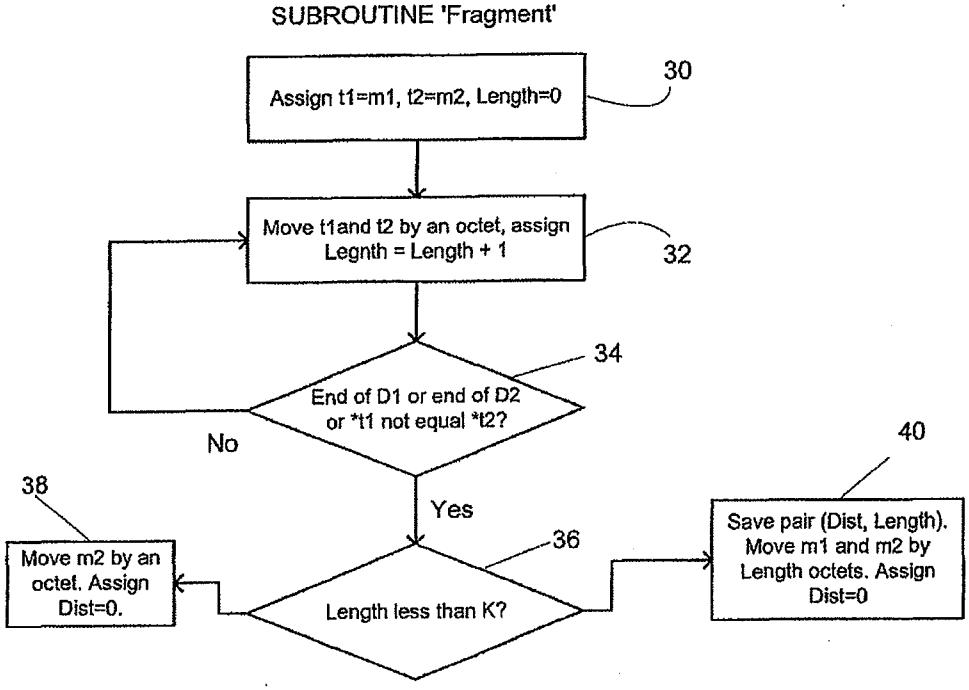

Set m1 to beginning of D1, m2 to beginning of D2; assign Dist=0
14
16
End of D1?
Yes
18
Save a number of fragement pairs. END
No
20
End of D2?
Yes
22
Save the rest of D1 from m1
No
24
*m1 equals *m2?
Yes
26
SUBROUTINE 'Fragment'
No
Move m2 by an octet, increase Dist by 1
28

FIG. 3


SUBROUTINE 'Fragment'
Assign t1=m1, t2=m2, Length=0
30
Move t1and t2 by an octet, assign Legnth = Length + 1
32
End of D1 or end of D2 or *t1 not equal *t2?
34
No
Yes
40
38
Move m2 by an octet. Assign Dist=0.
36
Length less than K?
Save pair (Dist, Length). Move m1 and m2 by Length octets. Assign Dist=0

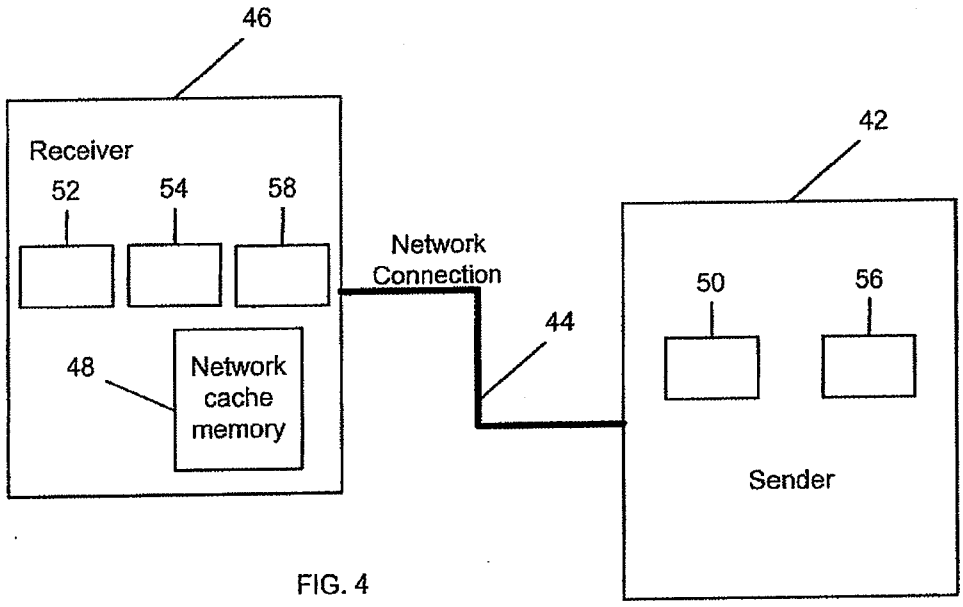

46
Receiver
52
54
58
48
Network cache memory
Network Connection
44
42
50
56
Sender

FIG. 4

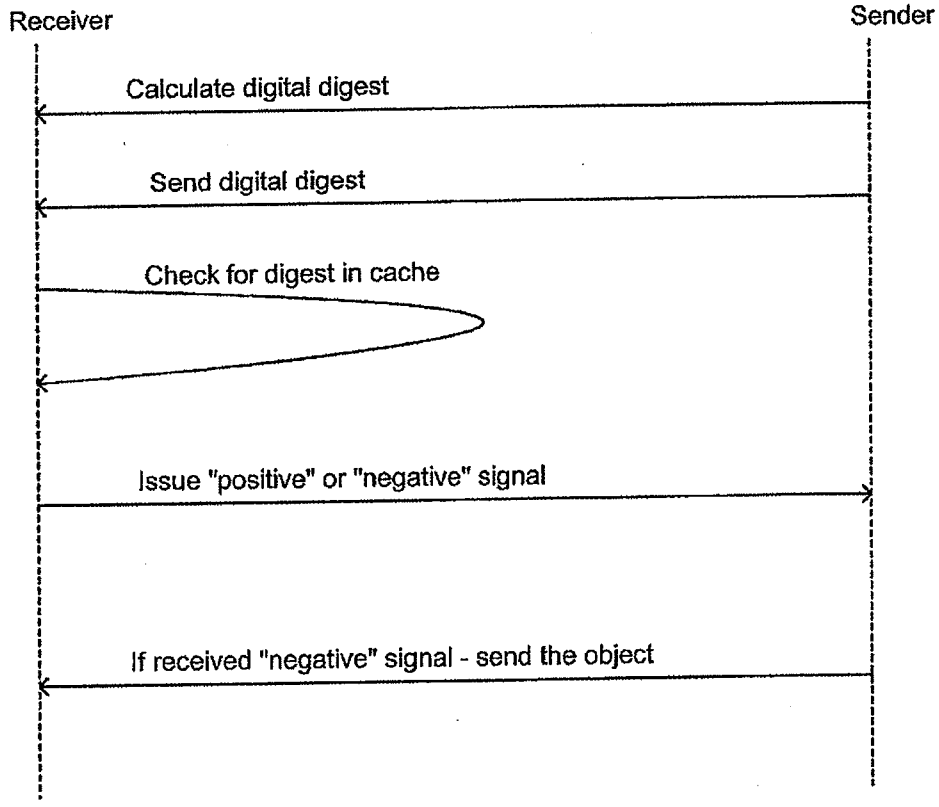

Receiver
Sender
Calculate digital digest
Send digital digest
Check for digest in cache
Issue "positive" or "negative" signal
If received "negative" signal - send the object

FIG. 5

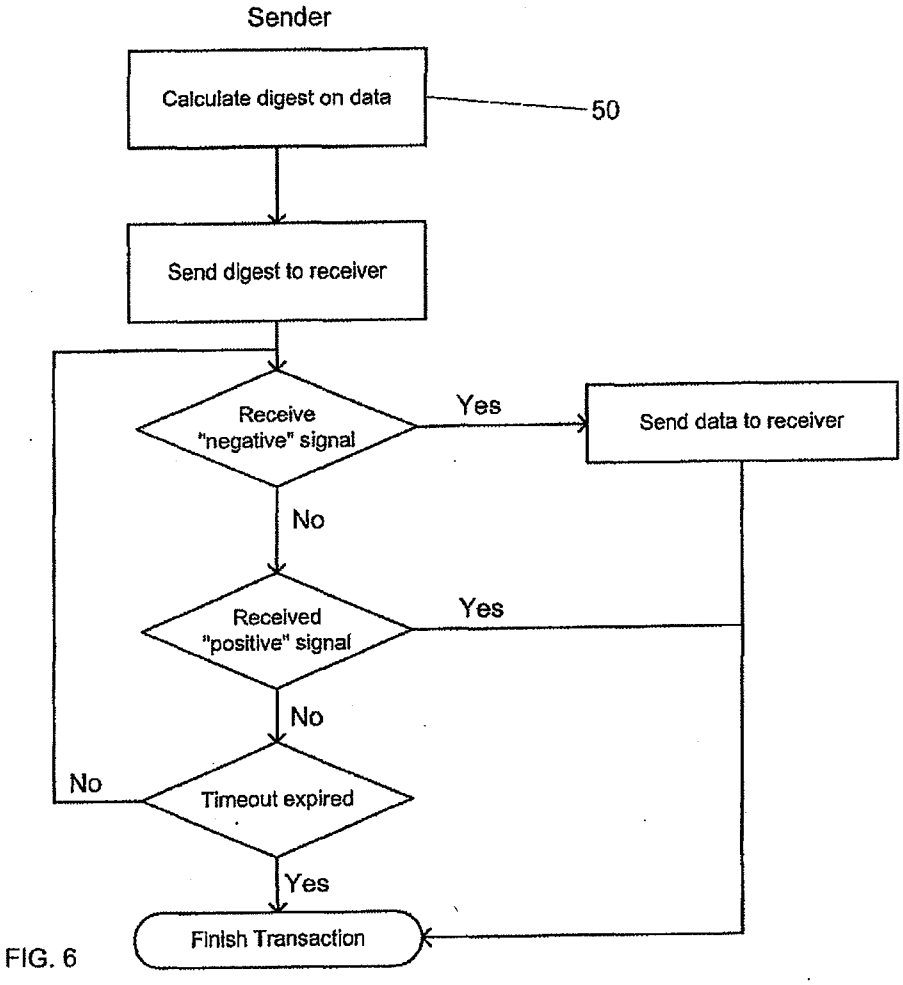

Sender
Calculate digest on data
50
Send digest to receiver
Receive "negative" signal
Yes
Send data to receiver
No
Received "positive" signal
Yes
No
No
Timeout expired
Yes
Finish Transaction

FIG. 6

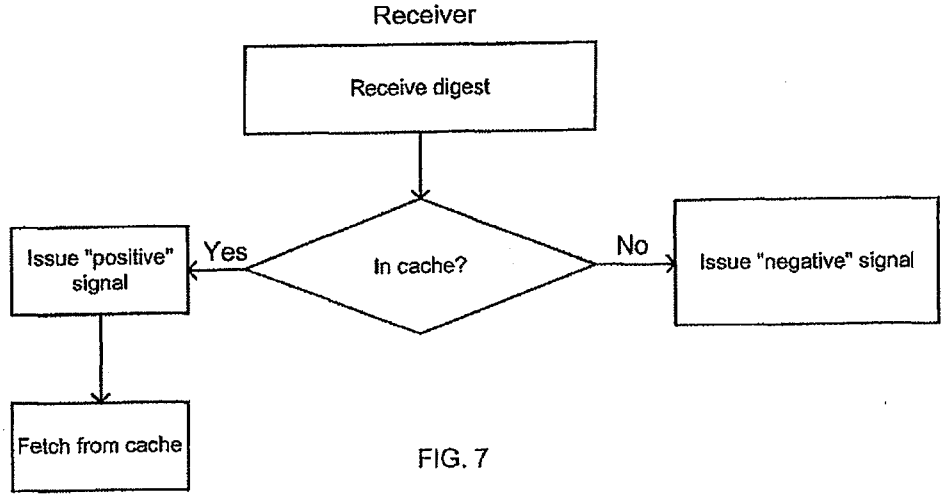

Receiver
Receive digest
Issue "positive" signal
Yes
In cache?
No
Issue "negative" signal
Fetch from cache

FIG. 7

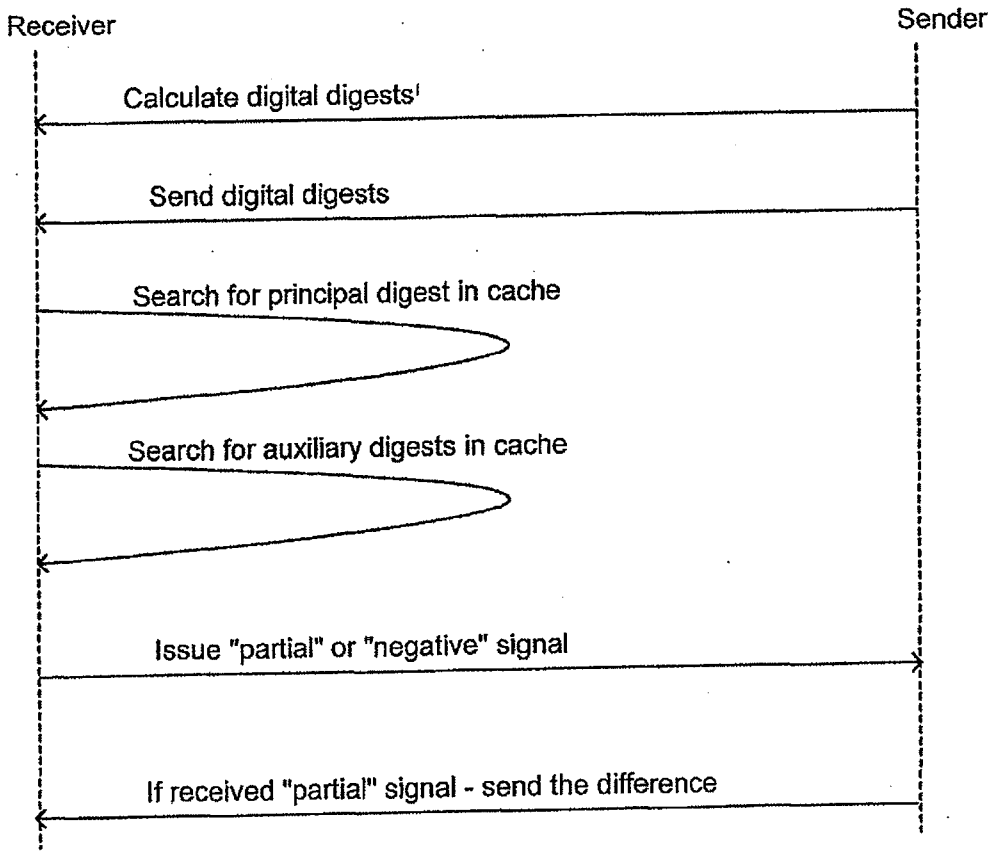

Receiver
Sender
Calculate digital digests'
Send digital digests
Search for principal digest in cache
Search for auxiliary digests in cache
Issue "partial" or "negative" signal
If received "partial" signal - send the difference

FIG. 8

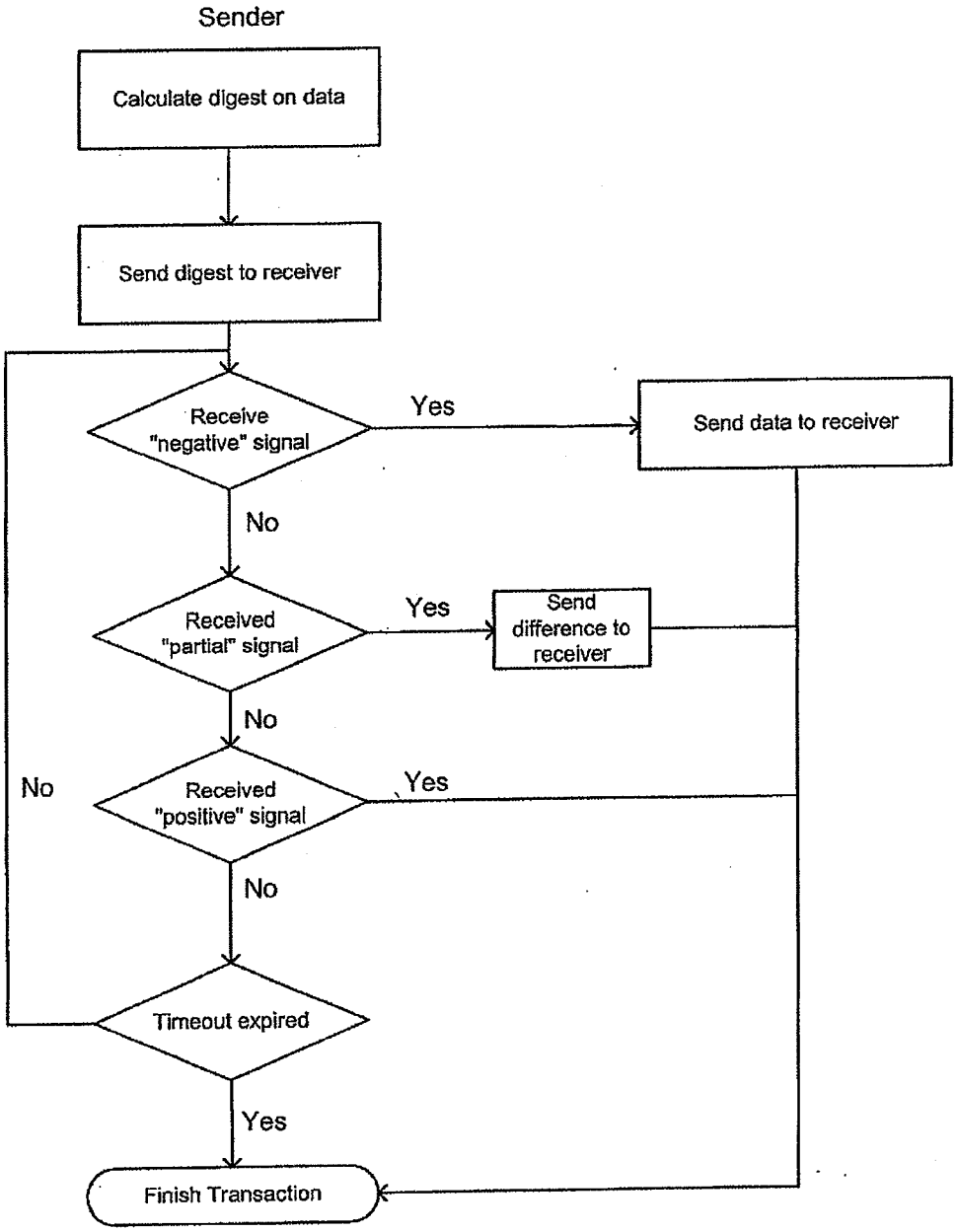

Sender
Calculate digest on data
Send digest to receiver
Receive "negative" signal
Yes
Send data to receiver
No
Received "partial" signal
Yes
Send difference to receiver
No
No
Received "positive" signal
Yes
No
Timeout expired
Yes
Finish Transaction

FIG. 9

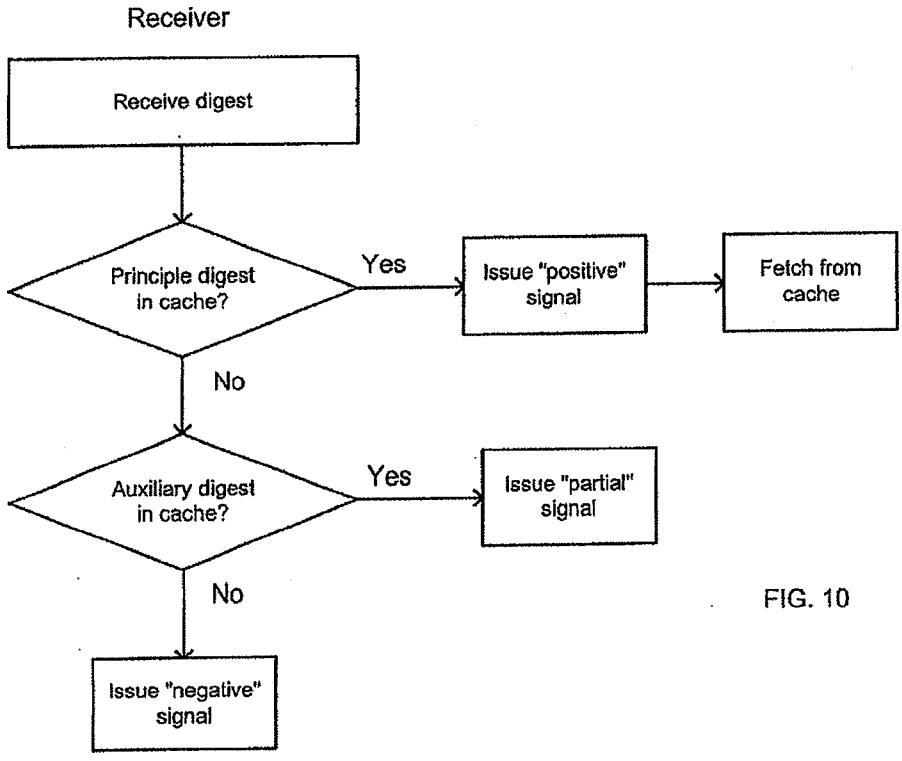

Receiver
Receive digest
Principle digest in cache?
Yes
Issue "positive" signal
Fetch from cache
No
Auxiliary digest in cache?
Yes
Issue "partial" signal
No
Issue "negative" signal

FIG. 10

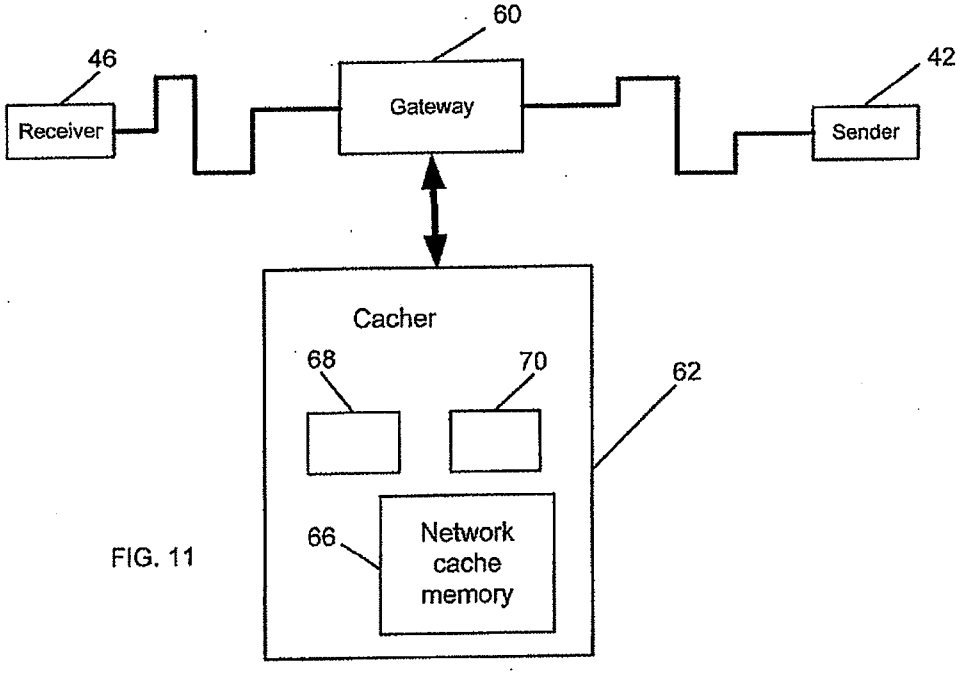

60
46
42
Receiver
Gateway
Sender
Cacher
68
70
62
66
Network cache memory

FIG. 11

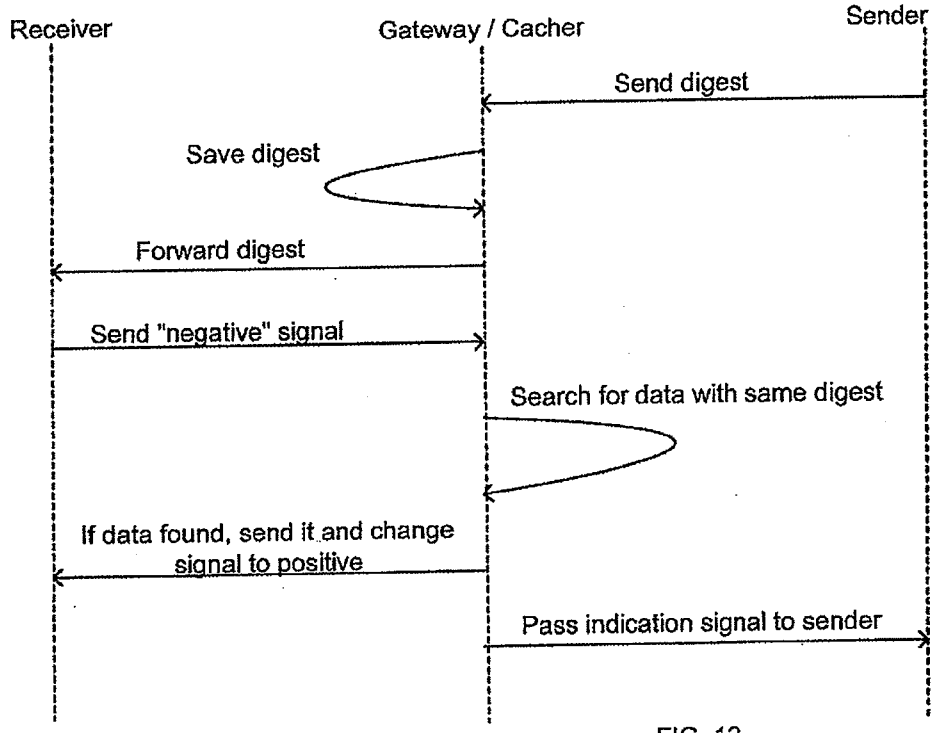

Receiver
Gateway / Cacher
Sender
Send digest
Save digest
Forward digest
Send "negative" signal
Search for data with same digest
If data found, send it and change signal to positive
Pass indication signal to sender

FIG. 12

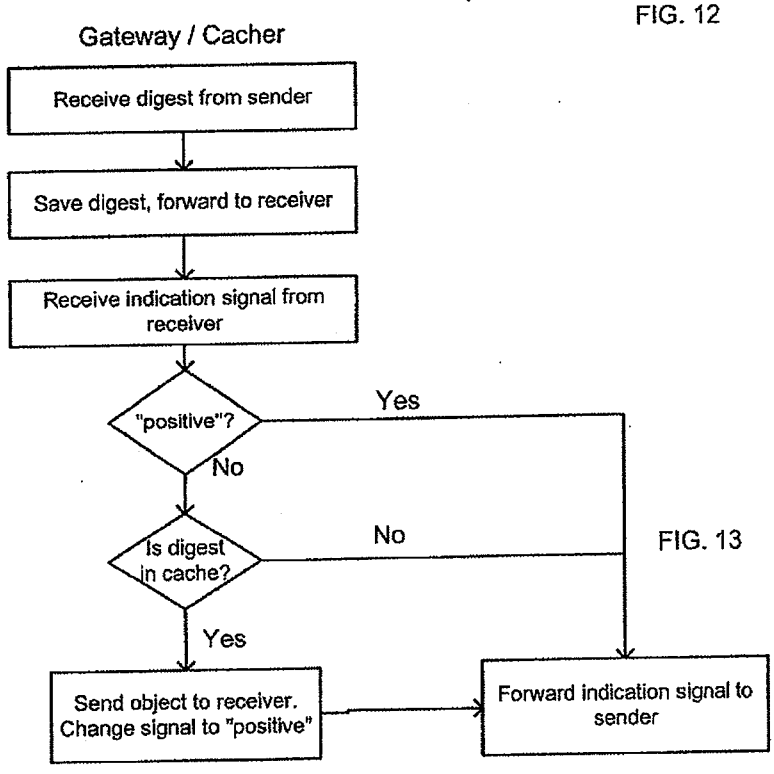

Gateway / Cacher
Receive digest from sender
Save digest, forward to receiver
Receive indication signal from receiver
"positive"?
Yes
No
Is digest in cache?
No
Yes
Send object to receiver. Change signal to "positive"
Forward indication signal to sender

FIG. 13

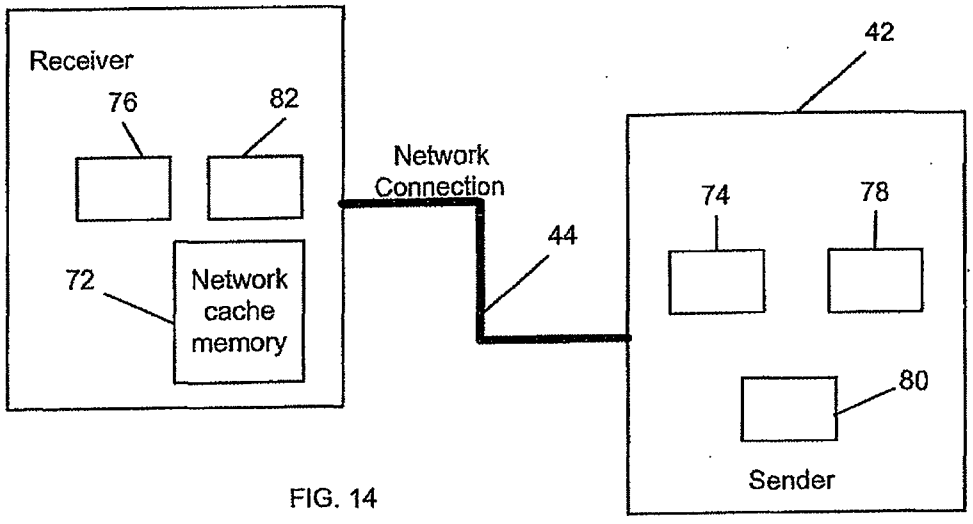

Receiver
76
82
72
Network cache memory
Network Connection
44
42
74
78
80
Sender

FIG. 14

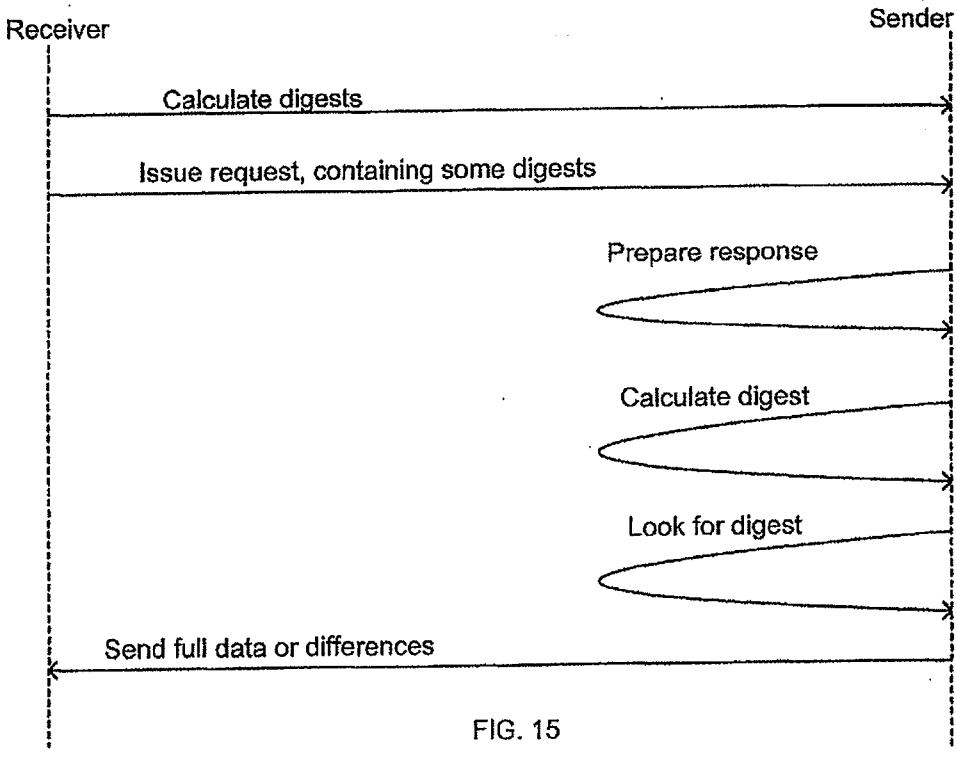

Receiver
Sender
Calculate digests
Issue request, containing some digests
Prepare response
Calculate digest
Look for digest
Send full data or differences

FIG. 15

SYSTEM AND METHOD FOR DATA ACCESS

RELATED APPLICATION

This application claims priority and is entitled to the filing date of Israeli application Ser. No. 126292 filed Sep. 18, 1998, and entitled "System And Method For Data Access," and which describes the same invention as defined herein.

FIELD OF THE INVENTION

The present invention relates to data access in networks. Specifically, the invention is concerned with a method, system and apparatus for increasing the speed of data accessing in communication networks.

BACKGROUND OF THE INVENTION

Many known applications and protocols provide means for caching and verifying of data transmitted via a network 2 (FIG. 1, prior art). Thus, a client (receiver) 4 caches data received from network 2 in cache 6. Then, when data from a remote server (sender) 8 is requested, it first searches its local cache. If the requested data is available in the cache and is verified to be valid, the client uses it, and transmission over the network is not required. Gateway or proxy caches 10 (FIG. 2, prior art) are able to operate in a similar manner.

The most well-known techniques are as follows:

1) In response to a request from a receiver, a sender attaches to the sent data an expiration time in absolute or relative form. The receiver, and possibly proxies, cache the data together with its request until the expiration time. Then the data is retrieved from the cache. In some cases, the receiver guesses the expiration time.

The problem associated with this technique is that the data entity can be changed before the expiration time, and the receiver would use an obsolete version of the data without even knowing it. Also, when the time has expired, the data will be resent, even if it is up to date.

2) In response to a request from a receiver, the sender attaches a validator to the sent data. The validator changes at least every time the data changes; in many cases, system time is used as the validator. The receiver, and possibly proxies, cache the data together with its request. When making the next request for the same data to the same sender, the receiver includes the validator. The sender keeps track of the data and resends it only if it were changed.

The problems associated with this technique are:

a) Data is cached according to requests and senders. If the same request is directed to different servers, cached data cannot be reused.

b) Requests without concrete data cannot be cached.

c) The sender must track the cached data, which is not always possible.

None of the prior art techniques discussed above provides means for transmitting minor differences in data. Additionally, if data is retrieved through a caching proxy, there is a danger that an unauthorized user will have access to the data.

It is therefore a broad object of the present invention to provide a method, system and apparatus for increasing the speed of data access in a packet-switched network.

Another object of the present invention is to decrease data traffic throughout the network.

Still another object of the present invention is to decrease the required cache size.

A yet further object of the present invention is to maintain accessed data integrity and to improve security.

SUMMARY OF THE INVENTION

The terms "data" or "data object" as used herein refer to a file or range of octets in a file, a range of frames in a video stream or RAM-based range of octets, a transport level network packet, or the like.

The term "digital digest" as used herein refers to a fixed-size binary value calculated from arbitrary-size binary data in such a way that it depends only on the contents of the data and the low probability that two different data or objects have the same digital digest.

The term "gateway" as used herein also includes network proxies and routers.

If a sender/computer in a network is required to send data to another receiver/computer, and the receiver/computer has data with the same digital digest as that of the data to be sent, it can be assumed with sufficient probability for most practical applications that the receiver/computer has data which is exactly the same as the data to be sent. Then, the receiver/computer can use the data immediately without its actual transfer through the network. In the present invention, this idea is used in a variety of ways.

In one embodiment of the invention, a sender/computer required to send data to a receiver/computer computer initially sends a digital digest of the data. If the receiver/computer already has data with the same digital digest, it uses this data as if it were actually transmitted from the sender/computer. Additionally, digital digests for other data objects can be sent together with the principal digest. If the receiver/computer cannot find data having the principal digest, it searches for data with one of these auxiliary digests. If such data is found, the sender/computer is required to send only the difference between the requested data object and the data object corresponding to the digest.

The expression "difference between a first data or data object and a second data or data object" as used herein means any bit sequence that enables the restoration of the first data, given the second data, the bit sequence and the method employed in calculating the difference.

The invention may be implemented in a gateway system. Such a system comprises a gateway computer connected to a packet-switched network in such a way that network packets sent between at least two other computers pass through it; a caching computer connected to the gateway computer, the caching computer having a network cache memory in its permanent storage memory, means for calculating a digital digest on the data it stores and means for comparison between a digital digest calculated on data in its network cache memory and a digital digest received from the packet-switched network by the gateway computer. When this system intercepts an indication signal other than a positive indication signal for a certain digital digest from a receiver/computer computer, if it has data with the same digest, it sends this data to the receiver/computer.

In another embodiment of the present invention, a client computer sends to a server computer a request including digital digests. A sender/computer forming a response then searches for data with the same digital digests as those received. If the digest of the data in the response equals one of the received digests, the server only sends confirmation. If the digest of another data is identical to one of the received digests, only the difference(s) between these data is sent.

In accordance with the present invention, there is therefore provided a system for data access in a packet-switched

network, comprising a sender/computer including an operating unit, a first memory, a permanent storage memory and a processor and a remote receiver/computer including an operating unit, a first memory, a permanent storage memory and a processor, said sender/computer and said receiver/computer communicating through said network; said sender/computer further including means for calculating digital digests on data; said receiver/computer further including a network cache memory and means for calculating digital digests on data in said network cache memory; and said receiver/computer and/or said sender/computer including means for comparison between digital digests.

The invention also provides a system for data access in a packet-switched network, comprising a gateway computer including an operating unit, a memory and a processor connected to said packet-switched network in such a way that network packets sent between at least two other computers pass through it; a caching computer including an operating unit, a first memory, a permanent storage memory and a processor connected to said gateway computer through a fast local network; said caching computer further including a network cache memory in its permanent storage memory, means for calculating a digital digest on data stored therein and means for comparison between a digital digest calculated on data in its network cache memory and a digital digest received from said packet-switched network through said gateway computer.

In addition, the invention provides a system for data access in a packet-switched network, comprising a sender/computer including an operating unit, a first memory, a permanent storage memory and a processor and a remote receiver/computer including an operating unit, a first memory, a permanent storage memory and a processor, said sender/computer and said receiver/computer communicating through a network; said sender/computer further including means for calculating digital digests on data, and said receiver/computer further including a network cache memory, means for storing a digital digest received from said network in its permanent storage memory and means for comparison between digital digests.

The invention further provides a method performed by a sender/computer in a packet-switched network for increasing data access, said sender/computer including an operating unit, a first memory, a permanent storage memory and a processor and said sender/computer being operative to transmit data to a receiver/computer, the method comprising the steps of transmitting a digital digest of said data from said sender/computer to said receiver/computer; receiving a response signal from said receiver/computer at said sender/computer, said response signal containing a positive, partial or negative indication signal for said digital digest, and if a negative indication signal is received, transmitting said data from said sender/computer to said receiver/computer.

The invention still further provides a method for increasing data access performed by a sender/computer in a packet-switched network, said sender/computer including an operating unit, a first memory, a permanent storage memory and a processor and said sender/computer being operative to transmit principal data to a receiver/computer, said method comprising the steps of transmitting digital digests of said principal data and of one or more auxiliary data from said sender/computer to said receiver/computer; receiving a response signal at said sender/computer from said receiver/computer, said response signal containing a positive, negative or partial indication signal, and if a partial indication signal is received, said sender/computer transmitting a signal constituting the difference between said principal data and corresponding auxiliary data.

The invention yet further provides a method for increased data access performed by a receiver/computer in a packet-switched network, said receiver/computer including an operating unit, a first memory, a permanent storage memory, a processor and a network cache memory, said method comprising the steps of receiving a message containing a digital digest from said network; searching for data with the same digital digest in said network cache memory, and if data having the same digital digest as the digital digest received is not uncovered, forming a negative indication signal and transmitting it back through said network.

Still further, the invention provides a method for increased data access performed by a receiver/computer in a packet-switched network, said receiver/computer including an operating unit, a first memory, a permanent storage memory, a processor and a network cache memory, said method comprising the steps of receiving a message containing a digital digest from said network; searching for data with the same digital digest in said network cache memory, and if data having the same digital digest as the digital digest received is uncovered, forming a positive indication signal and transmitting it back through said network.

In addition, the invention provides a method for increased data access performed by a receiver/computer in a packet-switched network, said receiver/computer including an operating unit, a first memory, a permanent storage memory, a processor and a network cache memory, said method comprising the steps of receiving a message containing a principal digital digest and one or more auxiliary digital digests from said network; searching in predetermined locations in said permanent storage memory for data with a digital digest substantially identical to said principal digital digest; searching in predetermined locations in said permanent storage memory for data with a digital digest substantially identical to one of said auxiliary digital digests; and if data having the same digital digest as one of said auxiliary digital digests received is uncovered, forming a partial indication signal and transmitting it back through said network.

Yet further, the invention provides a method for increased data access performed by a computer system in a packet-switched network, said computer system including a network cache memory and being operationally interposed between a sender/computer and a receiver/computer so that data packets sent between said sender/computer and said receiver/computer are delivered through said computer system; said method comprising the steps of intercepting a message containing a digital digest transmitted from said sender/computer to said receiver/computer, and transmitting data with a digital digest substantially identical to the digital digest received from said sender/computer to said receiver/computer.

In addition, the invention provides a method for increased data access performed by a computer system in a packet-switched network, said computer system including a network cache memory and being operationally interposed between a sender/computer and a receiver/computer so that data packets sent between said sender/computer and said receiver/computer are delivered through said computer system; said method comprising the steps of intercepting a message containing a digital digest transmitted from said sender/computer to said receiver/computer; intercepting a message containing an indication signal other than a positive indication signal transmitted from said receiver/computer to said sender/computer in response to said message containing a digital digest, and transmitting data with a digital digest substantially identical to the digital digest received from said sender/computer to said receiver/computer.

Additionally, the invention provides a method for increased data access performed by a client computer in a packet-switched network, said client computer including an operating unit, a first memory and a processor, said method comprising the steps of sending a request for data from said client computer to a server, said request containing digital digests for different data; said server preparing a response to said request, searching for data with a digital digest substantially identical to one of the digital digests received in said request, and producing the difference between said response and the uncovered data.

Finally, the invention provides apparatus for increased data access in a packet-switched network, comprising a computer connected to said packet-switched network, including an operating unit, a first memory, a permanent storage memory, a processor and a network cache memory; means for calculating digital digests of data in said network cache memory; means for comparison between digital digests, and means for sending the results of comparison between a digital digest received from another computer in said network and a digital digest calculated on data in said network cache memory back to said other computer.

The invention will now be described in connection with certain preferred embodiments with reference to the following illustrative figures so that it may be more fully understood.

With specific reference now to the figures in detail, it is stressed that the particulars shown are by way of example and for purposes of illustrative discussion of the preferred embodiments of the present invention only, and are presented in the cause of providing what is believed to be the most useful and readily understood description of the principles and conceptual aspects of the invention. In this regard, no attempt is made to show structural details of the invention in more detail than is necessary for a fundamental understanding of the invention, the description taken with the drawings making apparent to those skilled in the art how the several forms of the invention may be embodied in practice.

BRIEF DESCRIPTION OF THE DRAWINGS

FIG. 1 illustrates a prior art wide-area network;

FIG. 2 illustrates a prior art wide-area network with a caching gateway;

FIG. 3 is a flow diagram of the method of calculating the difference between two data digests according to the present invention;

FIG. 4 is a block diagram of a first embodiment of a sender/computer-receiver/computer system according to the present invention;

FIG. 5 is a schematic representation illustrating the interaction between a sender/computer and a receiver/computer according to the system of FIG. 4;

FIG. 6 is a flow diagram illustrating the method of operating the sender/computer according to the present invention;

FIG. 7 is a flow diagram illustrating the method of operating the receiver/computer according to the present invention;

FIG. 8 is a schematic representation illustrating the interaction between a sender/computer and a receiver/computer according to another embodiment of the present invention;

FIG. 9 is a flow diagram illustrating the method of operating the sender/computer according to a further embodiment of the present invention;

FIG. 10 is a flow diagram illustrating the method of operating the receiver/computer according to the embodiment of FIG. 9;

FIG. 11 is a block diagram of the configuration of the gateway system according to the present invention;

FIG. 12 is a schematic representation of the interaction between a sender/computer, a receiver/computer, and the gateway configuration according to the present invention;

FIG. 13 is a flow diagram of the operation of the gateway;

FIG. 14 is a block diagram of a further configuration of a sender/computer-receiver/computer system according to the present invention; and

FIG. 15 is a schematic representation of the interaction between the sender/computer-receiver/computer system of FIG. 14.

DETAILED DESCRIPTION OF PREFERRED EMBODIMENTS

The performance gains realized by the present invention are derived from the fact that computers in common wide-area networks tend to repetitively transmit the same data over the network.

The operations described herein may take the form of electrical or optical signals. The packet-switched network may be Internet.

The term "digital digest" as used herein refers to the per se known MD5 algorithm, described in RFC 1321 by R. Rivest, which is a preferred calculation method. Other algorithms may, however, just as well be used. For example, a digital digest may be calculated according to the CRC algorithm, or by applying the CRC algorithm to different subsets or different reorderings of data, or by consecutively applying CRC and MD5. In addition, any other algorithm may be used, provided that it produces a fixed-size binary value calculated from arbitrarily-sized binary data in such a way that it depends only on the contents of said data and that the probability of two different data having the same digital digest, is low.

Whenever means for calculating the difference between two data are mentioned herein, the method as shown in FIG. 3 may be employed. The data are referred to as D1 and D2. The difference between them consists of three parts: the number of fragment pairs, the array of fragment pairs, and the remainder of D1. A fragment pair is a pair representing the distance from the beginning of this fragment to the end of the previous one, and the number of octets in the fragment (Dist,Length). A marker m1 is set at the beginning of the data D1 and a marker m2 at the beginning of D2.

An octet m1 is designated as *m1 and an octet m2 as *m2. An integer K>1, which represents a minimal length of a fragment encoded, e.g., K=3, is chosen.

As stated above, m1 is set at the beginning of D1, m2 at the beginning of D2, and Dist=0 is assigned at 14. A loop is then entered: if m1 is at the end of D1 (16), a number of fragment pairs is saved at 18, and the algorithm is completed. If m2 is at the end of D2 (20), the rest of D1 from m1 is saved at 22, a number of fragment pairs is saved, and the algorithm is completed. If *m1 equals *m2 (24), a subroutine "Fragment" is entered at 26; otherwise, m2 is moved by one octet toward the end of D2 and Dist is increased by 1 at 28.

The subroutine "Fragment" proceeds as follows: New markers t1=m1 and t2=m2 are set and Length=0 is assigned at 30. t1 and t2 are moved by an octet toward the ends of D1 and D2 and Length is increased at 32. If t1 is at the end of D1, or t2 is at the end of D2, or *t1 does not equal *t2 at the end of the fragment (34), then the Length is a length of the fragment and Dist is the distance between the beginning of

this fragment and the end of the previous one. If the Length <K as determined by 36, the fragment is dropped at 38, m2 is moved by one octet and the subroutine is terminated. Otherwise, the pair (Dist,Length) is saved, the number of pairs is increased by one, m1 and m2 are moved by Length octets toward the ends of D1 and D2, and Dist is reset to 0 at 40. The subroutine is ended.

The sequence of fragment pairs may be further reduced in size by using the per se known Huffman encoding or by using an arithmetic coding, e.g., as disclosed in U.S. Pat. No. 4,122,440.

Restoration of the data is simple. Marker m2 is set at the beginning of the known D2. Then for each fragment pair (Dist, Length) from the known difference, m2 is moved by Dist octets, Length octets are copied from m2 to D1 and m2 is moved by Length. Then the rest of D1 is copied from the remainder

An embodiment of a sender/computer-receiver/computer system according to the present invention is schematically illustrated in FIG. 4. A preferred embodiment is a network computer system having at least two computers. A sender/computer **42** (also referred to herein as "sender/computer") having an operating unit, a first memory, a permanent storage memory and a processor, is connected to the network by any network connection **44**. A remote receiver/computer **46** (also referred to herein as "receiver/computer") having an operating unit, a first memory, a permanent storage memory and a processor, is also connected to the network. The receiver/computer **46** uses a part of its permanent storage memory or its first memory, or both, as network cache memory **48**. The sender/computer has calculation means **50** for calculating a digital digest on data in its first memory or in its permanent storage memory. Similarly, the receiver/computer has calculating means **52** for calculating a digital digest on data stored in its network cache memory **48**. The receiver/computer also has comparison means **54** for comparing between such a calculated digital digest and a digital digest received from the network.

An example of a first memory could be a RAM; an example of a permanent storage memory may be a disk drive, a flash RAM or a bubble memory.

It is possible to modify this system in different ways. The receiver/computer **46** and sender/computer **42** may each include means for storing the calculated digital digest in its first memory or permanent storage memory. Additionally, the receiver/computer **46** may have means for calculating a digital digest on data in its permanent storage memory outside of its cache memory. Furthermore, the system may be modified in such a way that the sender/computer **42** has means **56** for calculating the difference between two data objects.

Interaction between the receiver/computer and the sender/computer is depicted in FIGS. 5 to 7. The data sender/computer **42** calculates a digital digest on the data in means **50** and then transmits the calculated digest to receiver/computer **46**. The receiver/computer receives the digital digest from sender/computer **42** and then searches its network cache memory **48** for data with the same digest. If it finds such data, it uses it as if it were received from the sender/computer **42** and issues a positive indication signal to the sender/computer. Otherwise, it sends a negative indication signal to the sender/computer. Upon receiving a negative indication signal, the sender/computer transmits the data. Upon receiving a positive indication signal, or upon expiration of a predefined period of time, the sender/computer completes the transaction. This transaction begins with a receiver/computer sending a request to the sender/computer.

The above-described method may be modified in different ways. For example, absence of a signal from the receiver/computer for a predetermined period of time may be considered by the sender/computer to be a negative indication signal. Alternatively, the digital digests for some data may be stored in the permanent storage memory of the sender/computer and obtained from there, or a plurality of data may be processed in one transaction, a digital digest being calculated for each data object and a separate indication signal issued on each digital digest.

Another method of interaction between the receiver/computer **46** and the sender/computer **42** is illustrated in FIGS. **8–10**. The data sender/computer calculates a digital digest on the data to be sent (hereinafter, "principal digest") and for one or more other data objects (hereinafter, "auxiliary digests"). Without limiting the scope of the invention, the following data objects may be recommended: (a) a previous version of the data requested; (b) a file similar to the data requested. Then the sender/computer sends the principal and auxiliary digests to the receiver/computer. Upon receiving a message with these digital digests from the sender/computer, the receiver/computer searches its network cache memory **48** for data having the principal digest. If such data is found, it uses it as if it were received from sender/computer **42** and issues a positive indication signal to the sender/computer. Otherwise, receiver/computer **46** searches its network cache memory **48** for data with the auxiliary digests. If it finds data with a digital digest substantially equal to one of the auxiliary digests, it issues a partial indication signal to the sender/computer, with a reference to the digest. Otherwise, it issues a negative indication signal to the sender/computer. Upon receiving a negative indication signal, the sender/computer sends the data. Upon receiving a partial indication signal, the sender/computer transmits the difference between the digital digest of the data required to be sent and that of the data whose digital digest was found by the receiver/computer. This transaction may also begin with the receiver/computer sending a request to the sender/computer.

A modification of the above method is possible. For example, absence of the indication signal from the receiver/computer for a predefined period of time may be considered by the sender/computer as a negative indication signal, or the digital digests for some data may be stored in the permanent storage memory of the sender/computer and obtained from there instead of being calculated immediately before the transaction. Alternatively, a plurality of data may be processed in one transaction; a digital digest is calculated for each data object and a separate indication signal issued on every digital digest. Still alternatively, receiver/computer **46** may search not only in its network cache memory **48**, but also in predefined locations in its permanent storage memory. Sender/computer **42** may add to a digest it sends to the receiver/computer information about the possible location of the data with that digital digest in the receiver/computer's permanent storage memory.

Another embodiment of the present invention is schematically illustrated in FIG. **11**. Shown is a system comprising a gateway computer or gateway **60** including an operating unit, a first memory and a processor, and a caching computer **62** including an operating unit, a first memory, a permanent storage memory and a processor, connected to the gateway **60** through any fast network connection **64**, e.g., Ethernet. Gateway **60** is connected to a wide-area packet-switched network in such a way that network packets sent between at least two other computers **42** and **46** pass through the gateway **60**. The caching computer **62** uses a part of its

permanent storage memory for network cache memory **66**. Caching computer **62** has means **68** for calculating the digital digest of data in its network cache memory **66**, and means **70** for comparison between such a calculated digital digest and a digital digest received by gateway computer **60** from the wide-area network. It should be noted that gateway computer **60** may be integrally formed with the caching computer. The caching computer may have means for storing a calculated digital digest in its first memory or permanent storage memory.

By way of example, operations which may be performed in such a system will now be described with reference to FIGS. **12** and **13**. The gateway **60**, operationally interposed between a sender/computer **42** and a receiver/computer **46**, intercepts a digital digest sent from the sender/computer to the receiver/computer, saves it in its memory, and passes it unchanged to the receiver/computer **46**. Then the gateway **60** intercepts an indication signal other than a positive indication signal issued by the receiver/computer. If there was a digest for this indication signal, the caching computer **62** searches for data with the same digital digest in its network cache memory **66**. If that digest is found, then the gateway sends the data to the receiver/computer, changes the indication signal to positive, and then passes the indication signal to sender/computer **42**.

Further, the caching computer **62** may verify a digital digest for a data object stored in its network cache memory **66** by calculating the digital digest for that data and comparing it to the digest stored in the network cache memory. The calculated digital digest may be stored in the network cache memory **66** and the data object-digital digest pair may be marked as not requiring further verification.

Another further embodiment of the present invention is schematically illustrated in FIG. **14**. It consists of a network computer system comprising at least two computers: a sender/computer **42** including an operating unit, a first memory, a permanent storage memory and a processor which is connected to a network **44**. A remote receiver/computer **46** having an operating unit, a first memory, a permanent storage memory and a processor is also connected to the network. The receiver/computer uses a part of its permanent storage memory or its first memory, or both, for network cache memory **72**. The sender/computer **42** has means **74** for calculating a digital digest for data in its memory or in its permanent storage. The receiver/computer **46** has means **76** for calculating a digital digest for data stored in its network cache memory. The sender/computer **42** also has means **78** for comparison between such a calculated digital digest and a digital digest received from the network. The sender/computer further includes means **80** for calculating the difference between two data objects, and receiver/computer **46** includes means **82** for restoring a data object from another data object and the difference between said data object being restored and said another data object.

An interaction between the sender/computer and receiver/computer according to this system is illustrated in FIG. **15**. When receiver/computer **46** is required to request data from the server or sender/computer **42**, it calculates one or more digital digests for different data objects stored in its network cache memory **72** or in its permanent storage memory. Without limiting the scope of the invention, the following data objects may be recommended: (a) a previous version of the data requested; (b) a file similar to the data requested; (c) a data set similar to the data requested, which may be generated in a first memory; (d) a large data file or database including fragments of octets, similar to the data requested.

The receiver/computer then transmits a request for data, containing one or more of the above-mentioned digital digests. The sender/computer prepares a response to the request, and then calculates a digital digest on the data in the response. If the calculated digest is equal to one of the digital digests in the request, the sender/computer sends a confirmation. Otherwise, the sender/computer may continue searching for the data objects with the same digital digests in the predefined subset of its permanent storage memory. If it finds such data, it calculates the difference between this data and the data in the response, and sends only the difference. Otherwise, the sender/computer sends the response as prepared.

Variations of the above method are envisioned. For example, a number of requests for data may be sent simultaneously. The digital digests on the receiver/computer may be calculated earlier and stored in the permanent memory of the receiver/computer. The digital digests on the sender/computer may also be calculated earlier and stored in the permanent memory of the sender/computer.

It will be evident to those skilled in the art that the invention is not limited to the details of the foregoing illustrated embodiments and that the present invention may be embodied in other specific forms without departing from the spirit or essential attributes thereof. The present embodiments are therefore to be considered in all respects as illustrative and not restrictive, the scope of the invention being indicated by the appended claims rather than by the foregoing description, and all changes which come within the meaning and range of equivalency of the claims are therefore intended to be embraced therein.

What is claimed is:

1. A system for data access in a packet-switched network, comprising:

a sender/computer including an operating unit, a first memory, a permanent storage memory and a processor and a remote receiver/computer including an operating unit, a first memory, a permanent storage memory and a processor, said sender/computer and said receiver/computer communicating through said network;

said sender/computer further including means for creating digital digests on data;

said receiver/computer further including a network cache memory and means for creating digital digests on data in said network cache memory; and

said receiver/computer including means for comparison between digital digests.

2. The system as claimed in claim **1**, wherein said receiver/computer further includes means for a digital digest for data stored in said permanent storage memory.

3. The system as claimed in claim **1**, wherein said receiver/computer further includes means for storing said created digital digest in its first or permanent memory.

4. The system as claimed in claim **1**, wherein said sender/computer further includes means for the difference between two data objects and said receiver/computer further includes means for restoring a data object from another data object and the difference between said data object being restored and said another data object.

5. The system as claimed in claim **1**, wherein said sender/computer further includes means for the difference between two data objects and said receiver/computer further includes means for restoring a data object from another data object and the difference between said data object being restored and said another data object.

6. A system for data access in a packet-switched network, comprising:

a gateway including an operating unit, a memory and a processor connected to said packet-switched network

in such a way that network packets sent between at least two other computers pass through it;

a caching computer connected to said gateway through a fast local network, wherein said caching computer includes an operating unit, a first memory, a permanent storage memory and a processor;

said caching computer further including a network cache memory in its permanent storage memory, means for a digital digest and means for comparison between a digital digest on data in its network cache memory and a digital digest received from said packet-switched network through said gateway.

7. The system as claimed in claim 6, wherein said caching computer further includes means for a digital digest for data in its network cache memory.

8. The system as claimed in claim 6, wherein said caching computer is integrally formed with said gateway.

9. The system as claimed in claim 6, wherein said caching computer further includes means for storing said digital digest in said permanent storage memory.

10. A system for data access in a packet-switched network, comprising:

a sender/computer including an operating unit, a first memory, a permanent storage memory and a processor and a remote receiver/computer including an operating unit, a first memory, a permanent storage memory and a processor, said sender/computer and said receiver/computer communicating through a network;

said sender/computer further including means for creating digital digests on data, and

said receiver/computer further including a network cache memory, means for storing a digital digest received from said network in its permanent storage memory and means for comparison between digital digests.

11. A method performed by a sender/computer in a packet-switched network for increasing data access, said sender/computer including an operating unit, a first memory, a permanent storage memory and a processor and said sender/computer being operative to transmit data to a receiver/computer, the method comprising the steps of:

creating and transmitting a digital digest of said data from said sender/computer to said receiver/computer;

receiving a response signal from said receiver/computer at said sender/computer, said response signal containing a positive, partial or negative indication signal for said digital digest, and

if a negative indication signal is received, transmitting said data from said sender/computer to said receiver/computer.

12. The method as claimed in claim 11, wherein said sender/computer creates said digital digest for the data before transmitting it to said receiver/computer.

13. The method as claimed in claim 12, wherein said sender/computer transmits the data to said receiver/computer after a preset period of time has passed since transmitting said digital digest to said receiver/computer and a response signal has not been received.

14. The method as claimed in claim 12, wherein, when a plurality of data objects is to be sent, a digital digest is sent for each of said data objects and a response signal is sent containing a separate indication signal for each of said data objects.

15. The method as claimed in claim 12, wherein said digital digest creation comprises the step of Cyclic Redundancy Check against the contents of the data.

16. The method as claimed in claim 12, wherein said digital digest creation comprises the step of MD5 against the contents of the data.

17. A method for increasing data access performed by a sender/computer in a packet-switched network, said sender/computer including an operating unit, a first memory, a permanent storage memory and a processor and said sender/computer being operative to transmit principal data to a receiver/computer, said method comprising the steps of:

creating and transmitting digital digests of said principal data and of one or more auxiliary data from said sender/computer to said receiver/computer;

receiving a response signal at said sender/computer from said receiver/computer, said response signal containing a positive, negative or partial indication signal, and

if a partial indication signal is received, said sender/computer transmitting a signal constituting the difference between said principal data and corresponding auxiliary data.

18. The method as claimed in claim 17, wherein additional information about the probable location of auxiliary data in said permanent storage memory of the receiver/computer is encoded and transmitted together with the corresponding digital digest.

19. The method as claimed in claim 17, wherein said sender/computer creates said digital digest for data before transmitting said digital digest to said receiver/computer.

20. The method as claimed in claim 17, wherein said digital digest is obtained from the permanent storage memory of said sender/computer.

21. The method as claimed in claim 17, wherein said digital digest creation comprises the step of Cyclic Redundancy Check against the contents of the data.

22. A method for increased data access performed by a receiver/computer in a packet-switched network, said receiver/computer including an operating unit, a first memory, a permanent storage memory, a processor and a network cache memory, said method comprising the steps of:

receiving a message containing a digital digest from said network;

searching for data with the same digital digest in said network cache memory,

if data having the same digital digest as the digital digest received is not uncovered, forming a negative indication signal and transmitting it back through said network; and

creating a digital digest for data received from said network cache memory.

23. The method as claimed in claim 22, further comprising searching in predetermined locations in said permanent storage memory for data with a digital digest substantially identical to the digital digest received from said network.

24. The method as claimed in claim 22, wherein a plurality of digital digests for different data objects is received in the same message and an indication signal is generated separately for each of said data objects.

25. A method for increased data access performed by a receiver/computer in a packet-switched network, said receiver/computer including an operating unit, a first memory, a permanent storage memory, a processor and a network cache memory, said method comprising the steps of:

receiving a message containing a principal digital digest and one or more auxiliary digital digests from said network, wherein said auxiliary digital digests are correlated to data objects similar to the data object represented by said principal digest;

searching in predetermined locations in said permanent storage memory for data with a digital digest substantially identical to said principal digital digest;

searching in predetermined locations in said permanent storage memory for data with a digital digest substantially identical to one of said auxiliary digital digests; and

if data having the same digital digest as one of said auxiliary digital digests received is uncovered, forming a partial indication signal and transmitting it back through said network.

26. The method as claimed in claim **25**, further comprising the step of searching the network cache memory for data with said principal digital digest.

27. The method as claimed in claim **26**, further comprising the step of searching in said network cache memory for data with one of said auxiliary digital digests.

28. A method for increased data access performed by a computer system in a packet-switched network, said computer system including a network cache memory and being operationally interposed between a sender/computer and receiver/computer so that data packets sent between said sender/computer and said receiver/computer are delivered through said computer system; said method comprising the steps of:

intercepting a message containing a digital digest transmitted from said sender/computer to said receiver/computer, and

transmitting data with a digital digest substantially identical to the digital digest received from said sender/computer to said receiver/computer in response to said message, whereby said sender/computer is relieved of the burden of transmitting said data.

29. The method of claim **28** further comprising the step of receiving said data into the network cache memory prior to intercepting the message.

30. A method for increased data access performed by a computer system in a packet-switched network, said computer system including a network cache memory and being operationally interposed between a sender/computer and a receiver/computer so that data packets sent between said sender/computer and said receiver/computer are delivered through said computer system; said method comprising the steps of:

intercepting a message containing a digital digest transmitted from said sender/computer to said receiver/computer;

intercepting a message containing an indication signal other than a positive indication signal transmitted from said receiver/computer to said sender/computer in response to said message containing a digital digest, and

transmitting data with a digital digest substantially identical to the digital digest received from said sender/computer to said receiver/computer, whereby said sender/computer is relieved of the burden of transmitting said data.

31. The method of claim **30** further comprising the step of receiving said data into the network cache memory prior to intercepting the message.

32. A method for increased data access performed by a client computer in a packet-switched network, said client computer including an operating unit, a first memory and a processor, said method comprising the steps of:

sending a request for a single first data object from said client computer to a server, said request containing multiple digital digests for different data objects similar to said first data object;

said server preparing a response to said request, searching for a second data object with a digital digest substantially identical to one of the digital digests received in said request, and producing the difference between said first data object and the uncovered second data object.

33. The method as claimed in claim **32**, further comprising the step of transmitting said difference to said client computer.

34. The method as claimed in claim **33**, further comprising the step of using said difference for restoring the data from said response in said client computer.

* * * * *

EXHIBIT B

Claim Language	How Practiced by Defendants' Systems Using Microsoft's Distributed File System ("DFS") Replication and Remote Differential Compression ("RDC")
a sender/computer including an operating unit, a first memory, a permanent storage memory and a processor and a remote receiver/computer including an operating unit, a first memory, a permanent storage memory and a processor,	RDC is used with both a sender/computer (server) and a receiver/computer (client): "In a typical RDC scenario, a server and a client have different versions of a file. (The terms client and server refer only to the computers' roles in this scenario, not their operating systems.)"[4] The sender/computer and receiver/computer each include an operating unit, a RAM (a first memory), a hard disk drive (permanent storage memory), and a processor.
said sender/computer and said receiver/computer communicating through said network;	The sender/computer and receiver/computer communicate through the network: "Remote Differential Compression (RDC) allows data to be synchronized with a remote source using compression techniques to minimize the amount of data sent across the network."[5] "RDC is suitable for applications that move data across a wide area network (WAN) …."[6]
said sender/computer further including means for creating digital digests on data;	The sender/computer uses the FilterMax signature generator to create signatures (digital digests) of files: "The RDC client and server each use the RDC library's FilterMax signature generator to divide their copy of the file into chunks and compute a strong hash, called a signature, for each chunk of file data."[7]

[4] "About Remote Differential Compression" ¶12.
[5] "About Remote Differential Compression" ¶1.
[6] "About Remote Differential Compression" ¶6.
[7] "About Remote Differential Compression" ¶13.

Claim Language	How Practiced by Defendants' Systems Using Microsoft's Distributed File System ("DFS") Replication and Remote Differential Compression ("RDC")
said receiver/computer further including a network cache memory and	The receiver/computer has a staging folder that operates as a network cache memory: "DFS Replication uses staging folders to act as caches for new and changed files to be replicated from sending members to receiving members.... The receiving member downloads the data and builds the file in its staging folder."[8]
means for creating digital digests on data in said network cache memory;	The receiver/computer uses the FilterMax signature generator to create signatures (digital digests) of files: "The RDC client and server each use the RDC library's FilterMax signature generator to divide their copy of the file into chunks and compute a strong hash, called a signature, for each chunk of file data."[9]
and said receiver/computer including means for comparison between digital digests.	The receiver/computer compares file signatures: "The client initiates the RDC protocol by requesting the source signature list from the server. Then the client compares each source signature against the signatures in its own seed signature list."[10]

[8]"Staging Folders and Conflict and Deleted folders" (http://technet.microsoft.com/en-us/library/cc782648(WS.10,printer).aspx) ¶1.
[9]"About Remote Differential Compression" ¶13.
[10]"About Remote Differential Compression" ¶14.

Claim Language	How Practiced by Defendants' Systems Using Microsoft's Distributed File System ("DFS") Replication and Remote Differential Compression ("RDC")
10. A system for data access in a packet-switched network, comprising:	On information and belief, Defendants practice claim 10 by, for example, using DFS Replication on systems in their internal networks. Additionally, HP, Dell and Acer practice claim 10 by making, using, selling, offering for sale and/or licensing systems that use DFS Replication. Proxyconn believes that DFS Replication uses RDC.[11] In the DFS Replication context, the sender/computer is a server and the receiver/computer is also a server. Sender/computer and receiver/computer are connected in a packet-switched network: "RDC is suitable for applications that move data across a wide area network (WAN) …."[12]
a sender/computer including an operating unit, a first memory, a permanent storage memory and a processor and a remote receiver/computer including an operating unit, a first memory, a permanent storage memory and a processor	RDC is used with both a sender/computer and a receiver/computer: "In a typical RDC scenario, a server and a client have different versions of a file. (The terms client and server refer only to the computers' roles in this scenario, not their operating systems.)"[13] The sender/computer and receiver/computer each include an operating unit, a RAM (a first memory), a hard disk drive (permanent storage memory), and a processor.

[11]DFS Replication is used for exemplary purposes only. Defendants' infringement will be similar for other products that use RDC.
[12]"About Remote Differential Compression" ¶6.
[13]"About Remote Differential Compression" ¶12.

Claim Language	How Practiced by Defendants' Systems Using Microsoft's Distributed File System ("DFS") Replication and Remote Differential Compression ("RDC")
said sender/computer and said receiver/computer communicating through a network;	The sender/computer and receiver/computer communicate through a network: "Remote Differential Compression (RDC) allows data to be synchronized with a remote source using compression techniques to minimize the amount of data sent across the network."[14] "RDC is suitable for applications that move data across a wide area network (WAN) …."[15]
said sender/computer further including means for creating digital digests on data, and	The sender/computer uses the FilterMax signature generator to create signatures (digital digests) of files: "The RDC client and server each use the RDC library's FilterMax signature generator to divide their copy of the file into chunks and compute a strong hash, called a signature, for each chunk of file data."[16]
said receiver/computer further including a network cache memory,	The receiver/computer has a staging folder that is a network cache memory: "DFS Replication uses staging folders to act as caches for new and changed files to be replicated from sending members to receiving members…. After being constructed, the staged file is sent to the receiving member…. The receiving member downloads the data and builds the file in its staging folder."[17]

[14] "About Remote Differential Compression" ¶1.
[15] "About Remote Differential Compression" ¶6.
[16] "About Remote Differential Compression" ¶13.
[17] "Staging Folders and Conflict and Deleted folders" ¶1.

Claim Language	How Practiced by Defendants' Systems Using Microsoft's Distributed File System ("DFS") Replication and Remote Differential Compression ("RDC")
means for storing a digital digest received from said network in its permanent storage memory and	The receiver/computer stores file signatures received from the server in the client's permanent storage memory: "DFS Replication uses staging folders to act as caches for new and changed files to be replicated from sending members to receiving members.... The receiving member downloads the data and builds the file in its staging folder."[18] "Because the size of the signature file grows linearly with the size of the original file, comparing very large files can be expensive. This cost is reduced dramatically by applying the RDC algorithm recursively to the signature files."[19]
means for comparison between digital digests.	The receiver/computer compares file signatures: "The client initiates the RDC protocol by requesting the source signature list from the server. Then the client compares each source signature against the signatures in its own seed signature list."[20]

[18]"Staging Folders and Conflict and Deleted folders" ¶1.
[19]"About Remote Differential Compression" ¶10.
[20]"About Remote Differential Compression" ¶14.

Claim Language	How Practiced by Defendants' Systems Using Microsoft's Distributed File System ("DFS") Replication and Remote Differential Compression ("RDC")
11. A method performed by a sender/computer in a packet-switched network for increasing data access, said sender/computer including an operating unit, a first memory, a permanent storage memory and a processor and said sender/computer being operative to transmit data to a receiver/computer, the method comprising the steps of:	On information and belief, Defendants practice claim 11 by, for example, using DFS Replication on systems in their internal networks. Proxyconn believes that DFS Replication uses RDC.[21] In the DFS Replication context, the sender/computer is a server and the receiver/computer is also a server: "In a typical RDC scenario, a server and a client have different versions of a file. (The terms client and server refer only to the computers' roles in this scenario, not their operating systems.)"[22] The sender/computer is located in a packet-switched network and can communicate with the receiver/computer: "Remote Differential Compression (RDC) allows data to be synchronized with a remote source using compression techniques to minimize the amount of data sent across the network."[23] "RDC is suitable for applications that move data across a wide area network (WAN) ...."[24] The sender/computer includes an operating unit, a RAM (a first memory), a hard disk drive (permanent storage memory), and a processor.

[21] DFS Replication is used for exemplary purposes only. Defendants' infringement will be similar for other products that use RDC.
[22] "About Remote Differential Compression" ¶12.
[23] "About Remote Differential Compression" ¶1.
[24] "About Remote Differential Compression" ¶6.

Claim Language	How Practiced by Defendants' Systems Using Microsoft's Distributed File System ("DFS") Replication and Remote Differential Compression ("RDC")
creating and transmitting a digital digest of said data from said sender/computer to said receiver/computer;	The sender/computer uses the FilterMax signature generator to create signatures (digital digests) of files: "The RDC client and server each use the RDC library's FilterMax signature generator to divide their copy of the file into chunks and compute a strong hash, called a signature, for each chunk of file data."[25] The sender/computer sends signatures to the receiver/computer: "The client initiates the RDC protocol by requesting the source signature list from the server. Then the client compares each source signature against the signatures in its own seed signature list."[26]

[25] "About Remote Differential Compression" ¶13.
[26] "About Remote Differential Compression" ¶14.

Claim Language	How Practiced by Defendants' Systems Using Microsoft's Distributed File System ("DFS") Replication and Remote Differential Compression ("RDC")
receiving a response signal from said receiver/computer at said sender/computer, said response signal containing a positive, partial or negative indication signal for said digital digest, and	The sender/computer receives a response signal from the receiver/computer in form of a request for a missing chunk of a file (a negative indication signal): "If a source signature matches a seed signature, the client already has the file data for that signature. If a source signature does not appear in the client's list of seed signatures, the client must request the specified chunk (of file data) from the server. "The result of comparing the two signature lists is a needs list, which describes which chunks of file data, from where (seed or source file), are needed to construct the target file on the client computer. Each entry in the needs list is called a *needs block*."[27] The sender/computer does not receive a request for a missing chunk of a file if such a request is unnecessary (a positive indication signal).
if a negative indication signal is received, transmitting said data from said sender/computer to said receiver/computer.	If the receiver/computer requests a missing chunk of a file, the sender/computer sends the chunk of the file to receiver/computer: "The client iterates through each needs block and copies the specified chunk of the source or seed file data to the target file. Seed file data is copied locally. Source file data is downloaded from the server. The more similar the seed and source files are, the less network bandwidth is required to create the target file."[28]

[27] "About Remote Differential Compression" ¶14-15.
[28] "About Remote Differential Compression" ¶16.

Claim Language	How Practiced by Defendants' Systems Using Microsoft's Distributed File System ("DFS") Replication and Remote Differential Compression ("RDC")
22. A method for increased data access performed by a receiver/computer in a packet-switched network, said receiver/computer including an operating unit, a first memory, a permanent storage memory, a processor and a network cache memory, said method comprising the steps of:	On information and belief, Defendants practice claim 22 by, for example, using DFS Replication on systems in their internal networks. Proxyconn believes that DFS Replication uses RDC.[29] In the DFS Replication context, the receiver/computer is a server in a packet-switched network: "RDC is suitable for applications that move data across a wide area network (WAN) …."[30] The receiver/computer includes an operating unit, a RAM (a first memory), a hard disk drive (permanent storage memory), and a processor. The receiver/computer also has a staging folder that is in a network cache memory: "DFS Replication uses staging folders to act as caches for new and changed files to be replicated from sending members to receiving members…. The receiving member downloads the data and builds the file in its staging folder."[31]

[29]DFS Replication is used for exemplary purposes only. Defendants' infringement will be similar for other products that use RDC.
[30]"About Remote Differential Compression" ¶6.
[31]"Staging Folders and Conflict and Deleted folders" ¶1.

Claim Language	How Practiced by Defendants' Systems Using Microsoft's Distributed File System ("DFS") Replication and Remote Differential Compression ("RDC")
receiving a message containing a digital digest from said network;	The receiver/computer receives a message containing signatures (digital digests) over a network: "The client initiates the RDC protocol by requesting the source signature list from the server. Then the client compares each source signature against the signatures in its own seed signature list."[32] "The RDC client and server each use the RDC library's FilterMax signature generator to divide their copy of the file into chunks and compute a strong hash, called a signature, for each chunk of file data."[33]
searching for data with the same digital digest in said network cache memory,	The receiver/computer searches for data with the same signature by consulting its own signature list: "The client initiates the RDC protocol by requesting the source signature list from the server. Then the client compares each source signature against the signatures in its own seed signature list."[34]

[32] "About Remote Differential Compression" ¶14.
[33] "About Remote Differential Compression" ¶13.
[34] "About Remote Differential Compression" ¶14.

Claim Language	How Practiced by Defendants' Systems Using Microsoft's Distributed File System ("DFS") Replication and Remote Differential Compression ("RDC")
if data having the same digital digest as the digital digest received is not uncovered, forming a negative indication signal and transmitting it back through said network; and	If the receiver/computer does not locate data having the same signature, it requests that the file be sent to it (a negative indication signal): "The client iterates through each needs block and copies the specified chunk of the source or seed file data to the target file. Seed file data is copied locally. Source file data is downloaded from the server. The more similar the seed and source files are, the less network bandwidth is required to create the target file."[35] "The client iterates through each needs block and copies the specified chunk of the source or seed file data to the target file. Seed file data is copied locally. Source file data is downloaded from the server. The more similar the seed and source files are, the less network bandwidth is required to create the target file."[36]
creating a digital digest for data received from said network cache memory.	The receiver/computer uses the FilterMax signature generator to create a signature (digital digest) of the file from its staging folders (network cache memory): "The RDC client and server each use the RDC library's FilterMax signature generator to divide their copy of the file into chunks and compute a strong hash, called a signature, for each chunk of file data."[37] "DFS Replication uses staging folders to act as caches for new and changed files to be replicated from sending members to receiving members.... The receiving member downloads the data and builds the file in its staging folder."[38]

[35]"About Remote Differential Compression" ¶16.
[36]"About Remote Differential Compression" ¶16.
[37]"About Remote Differential Compression" ¶13.

[38]"Staging Folders and Conflict and Deleted folders" ¶1.