IN THE

UNITED STATES COURT OF APPEALS

FOR THE FEDERAL CIRCUIT

AKAMAI TECHNOLOGIES, INC.,

Plaintiff-Appellant,

and

THE MASSACHUSETTS INSTITUTE OF TECHNOLOGY,

Plaintiff-Appellant,

v.

LIMELIGHT NETWORKS, INC.,

Defendant-Cross Appellant.

Appeals from the United States District Court for the District of Massachusetts in case nos. 06-CV-11109 and 06-CV-11585, Judge Rya W. Zobel.

PRINCIPAL BRIEF FOR PLAINTIFF-APPELLANT AKAMAI TECHNOLOGIES, INC. ON REHEARING EN BANC

Of Counsel:

ROBERT S. FRANK, JR.
CHOATE, HALL & STEWART LLP
Two International Place
Boston, MA 02110

Attorney for Plaintiff-Appellant
The Massachusetts Institute of Technology

DONALD R. DUNNER
KARA F. STOLL
FINNEGAN, HENDERSON, FARABOW, GARRETT & DUNNER, LLP
901 New York Avenue, NW
Washington, DC 20001-4413
(202) 408-4000

JENNIFER S. SWAN
FINNEGAN, HENDERSON, FARABOW, GARRETT & DUNNER, LLP
3300 Hillview Avenue
Palo Alto, CA 94304-1203
(650) 849-6676

Attorneys for Plaintiff-Appellant
Akamai Technologies, Inc.

June 20, 2011

CERTIFICATE OF INTEREST

Counsel for Akamai Technologies, Inc. certifies the following:

1. The full name of every party or amicus represented by Counsel is:

   Akamai Technologies, Inc.

2. The name of the real party in interest represented by me is:

   None

3. All parent corporations and any publicly held companies that own 10 percent or more of the stock of the party represented by me are:

   None

4. The names of all law firms and the partners or associates that appeared for the party or amicus now represented by me in the trial court or agency or are expected to appear in this Court are:

   Donald R. Dunner
   Kara F. Stoll
   Elizabeth D. Ferrill
   Finnegan, Henderson,
   Farabow, Garrett & Dunner, LLP
   901 New York Avenue, NW
   Washington, DC 20001
   Telephone: (202) 408-4000

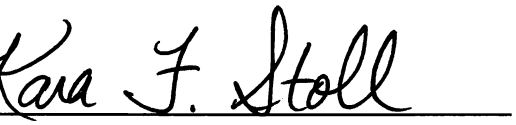
   Jennifer Swan
   Finnegan, Henderson,
   Farabow, Garrett & Dunner, LLP
   3300 Hillview Ave.
   Palo Alto, CA 94304

   Robert S. Frank, Jr.
   Carlos J. Perez-Albuerne
   Choate, Hall & Stewart LLP
   Two International Place
   Boston, MA 02110

Sarah Chapin Columbia
McDermott, Will & Emery LLP
28 State Street
Boston, MA 02109-1775

Respectfully submitted,

Date: June 20, 2011



Kara F. Stoll
FINNEGAN, HENDERSON, FARABOW,
GARRETT & DUNNER, LLP
901 New York Avenue, NW
Washington, DC 20001
(202) 408-4000

Attorney for Plaintiff-Appellant
Akamai Technologies, Inc.

TABLE OF CONTENTS

TABLE OF AUTHORITIES

Page(s)

CASES

STATUTES

OTHER AUTHORITIES

STATEMENT OF RELATED CASES

No appeal in or from these Civil Action Case Nos. 06-CV-11109 and 06-CV-11585 was previously before this or any other appellate court.

On April 20, 2011, the Court granted Akamai's request for rehearing en banc. (*See* Order of April 20, 2011, Granting En Banc Review, Case Nos. 2009-1372, -1380, -1416, -1417.) On May 26, 2011, the Court granted rehearing en banc in *McKesson Technologies, Inc. v. Epic Systems Corp.*, 2011 WL 1365548 (Fed. Cir. 2011) ("*McKesson*"). (S*ee* Order of May 26, 2011, Granting En Banc Review, Case No. 2009-1291.) Both cases involve issues of joint infringement. Accordingly, the *McKesson* case may be affected by this appeal.

I. RESPONSE TO THE EN BANC QUESTION

This Court has requested answers to the following question: If separate entities each perform separate steps of a method claim, under what circumstances would that claim be directly infringed and to what extent would each of the parties be liable?

Akamai answers as follows: A method claim is directly infringed when every step of the claim is practiced in the United States, whether by a single entity or by entities whose actions combine to perform all the steps of the claim.

As this Court has already held, principles of vicarious liability allow acts of one party in the performance of a step or steps of a method claim to be attributed to another. The most common type of vicarious liability is based on the principles of respondeat superior, or agency law. In such a situation, the agent's (servant's) acts are attributed to the principal (master) such that the parties can be seen as acting as a single entity. This is not, however, the ***only*** kind of vicarious liability. Rather, there are at least three other common law doctrines of vicarious liability that may apply in patent cases.

First, as this Court has already held in *BMC Resources, Inc. v. Paymentech, L.P.*, 498 F.3d 1373, 1381 (Fed. Cir. 2007), if one party "directs or controls" another to perform a step or steps of a method claim, those steps may be attributed to the directing or controlling party as if it performed them itself. This doctrine

prevents a potential infringer from immunizing itself from infringement by performing nearly all the steps of a method claim while directing or controlling the performance of the remaining steps by another. Under common law principles of torts, it makes no difference whether the directed or controlled party is acting as an "agent" of the other for this doctrine to apply. Although the party that was directed or controlled may not be liable, the party that exercises "direction or control" over the entire process such that every step is attributable to the directing or controlling party is liable.

Second, again applying common law principles of torts, performance of the steps of a method claim by two or more parties acting in concert should make such parties jointly and severally liable for direct infringement. As expressly defined in the Restatement (Second) of Torts, parties are acting in concert when they act in accordance with an implied or express agreement to cooperate in a particular line of conduct. This doctrine applies whenever the parties act in concert to perform the steps that constitute a method claim, whether they are partners, part of a joint enterprise, or have a contractual relationship. Each circumstance is a recognized form of vicarious liability in which all are liable for the acts of each other committed as part of their expressly or tacitly agreed-upon activity.

Finally, under common law principles of torts, even if one party is unaware that it is carrying out certain steps of a patented method, the other party possessing

either actual or constructive knowledge that all the steps that constitute the patented method are indeed being carried out should be held liable for patent infringement. In such circumstances, the "innocent" party without such knowledge but who is merely carrying out certain steps in isolation would not be held liable.

These standards, adopted from common law, are flexible enough to accommodate situations where multiple parties infringe a patented method, but narrow enough to avoid liability for a truly innocent party who, without knowledge of the overall method, performs some steps of a claim. There is no basis under precedent, the language of the Patent Act, or the policies underlying the Patent Act, for ignoring these common law principles of torts and restricting infringement of a method claim to the conduct of a single actor, and there is certainly no support for limiting liability for joint direct infringement to a narrow, rigid agency or contractual relationship.

II. PRELIMINARY STATEMENT

A method claim is directly infringed under 35 U.S.C. § 271(a) when all of the steps of a method are performed. The fundamental error of law committed by the Panel in the instant case stems from this Court's misplaced insertion of a "single entity" requirement into § 271(a). While it is well established under this Court's precedent that a method claim can only be directly infringed when all the steps of the method are performed, there is no basis in this Court's precedent, or

the policy underlying the Patent Act, to restrict infringement of a method claim to only a single actor. (*See* § IV.A.1, *infra.*) It is this misreading of § 271(a) that led to the Panel's requirement of an agency or contractual relationship—a rule that dramatically restricts liability for infringement and threatens the value of untold numbers of issued patents.

Indeed, it contravenes the goals of the Patent Act to immunize a party from infringement of a patent claim if that party performs only some of the steps and has another party perform the other step or steps. It is equally problematic to allow two or more parties to avoid liability for infringement if those parties come together and agree explicitly or implicitly to perform the steps of a patented method. Rather, principles of vicarious liability allow acts of one party in performance of a step or steps of a method claim to be attributed to another in a variety of scenarios.

Without a doubt, the most common type of vicarious liability is based on the principles of respondeat superior, or agency law. This, however, is not the *only* kind of vicarious liability. For example, as this Court has already held, if one party "directs or controls" another to perform a step or steps of a method claim, those steps may be attributed to the directing or controlling party. At the very least, a flexible fact-based "direction or control" test, as initially set forth in *BMC Resources,* as opposed to a rigid agency or contractual relationship test, should

apply. (*See* § IV.B.1, *infra*.) There is simply no basis for reading a "direction or control" test as necessarily requiring an agency relationship or a contract between the parties in order to impose liability. Further, the focus of a flexible fact-based "direction or control" test should be on both direction as well as control—the standard should not be converted into a control-only test.

Additionally, there is no basis under precedent, the Patent Act, or the policies underlying the Patent Act for restricting liability for direct infringement of a method claim to only those circumstances where one actor dominates another. Those who act in concert, partners, and joint enterprisers are all vicariously liable for the acts of each other committed as part of their expressly or tacitly agreed-upon activity. The Restatement (Second) of Torts, this Court's precedent, and numerous pre-1952 cases all acknowledge this type of liability. Accordingly, and consistent with tort law principles, parties who act in concert to carry out the steps that constitute a patented method should be jointly liable.

Finally, consistent with principles of tort law, an independent actor is liable for direct infringement if that actor knows that its actions may be combined with another's, and such conduct results in tortious injury. Applying this basic principle, numerous courts have imposed liability on a defendant in situations where the actions of independent parties have combined together to commit a tort, reasoning that the defendant knew of the combined conduct. This Court should

apply this common law tort doctrine to patent infringement, which is itself a tort. Moreover, this doctrine is particularly attractive because an "innocent" infringer, without knowledge of the steps performed by the other party, would not be liable for infringement.

Each of the above circumstances is a recognized form of vicarious liability. In all of these situations, it is uncontroversial that liability for patent infringement under § 271(a) should apply. These doctrines of vicarious liability, based on common law tort principles and supported by precedent, provide a sensible, workable standard for patent infringement under § 271(a) and are consistent with the Supreme Court's preference for flexible fact-based standards that avoid bright-line rules. The doctrines are consistent with the language of the statute and afford inventors a meaningful right to exclude. Further, the proposed test for joint infringement is broad enough to encompass the "direction or control" test set forth in *BMC Resources*, but is not so narrow as to restrict liability where it otherwise should apply.

In this case, Limelight was the mastermind behind the performance of the accused method. Limelight developed the accused process and provided detailed instructions and a contract to direct its customers to perform those few steps of the claimed process that it did not perform. Specifically, according to Limelight's process, Limelight performed all of the steps of asserted claim 34 except the

"tagging" step and all of the steps of asserted claims 19-21 except the "tagging" and "serving" steps. As the Panel recognized, Limelight's contract explicitly set forth that the customer would perform tagging and serving steps and, in exchange, Limelight would provide a service guarantee (with that service requiring the performance of the remaining steps of the claim). Given Limelight's role in developing, performing, and orchestrating the performance of the accused method, this Court should reinstate the jury verdict of joint infringement, which was properly decided under the "direction or control" standard announced in *BMC Resources*.

III. FACTS

Given that the en banc Court has asked the parties to address a specific legal question, Akamai will forego the conventional statement of facts at this point in the brief. Akamai will instead provide a detailed statement of facts below when applying the broad principles involved to the facts in this case. (*See* § IV.G., *infra*.)

IV. ARGUMENT

A. There Is No Basis to Restrict Infringement of a Method Claim to the Conduct of a Single Actor

1. This Court Erred by Adopting a Single-Actor Rule for Assessing Joint Liability for Infringement

The genesis of the rigid test applied by the *Akamai* Panel can be found in certain language from this Court's decision in *BMC Resources*, which is the first in

a recent line of cases where this Court sought to expand on the proper framework for deciding joint infringement questions. In *BMC Resources*, this Court explained that liability for joint infringement can be imposed where one party has sufficient "direction or control" over the performance of the claimed method. 498 F.3d at 1380-81. Although *BMC Resources* also discussed how principles of equity do not allow a "mastermind" to avoid infringement, *id.* at 1381, decisions subsequent to *BMC Resources* progressively narrowed and limited the applicable test, until finally, in *Akamai*, the Federal Circuit announced that to establish joint infringement under § 271(a), there must exist either a strict agency relationship or a contractual relationship between the parties. *Akamai Techs. Inc. v. Limelight Networks, Inc.*, 629 F.3d 1311, 1320 (Fed. Cir. 2010) ("This court therefore holds as a matter of Federal Circuit law that there can only be joint infringement when there is an agency relationship between the parties who perform the method steps or when one party is contractually obligated to the other to perform the steps.").

This Court's erroneous progression toward the agency or contract standard as the sole means for establishing joint infringement, however, is based primarily on the mistaken view that only a single entity can infringe a method claim. For example, in *Muniauction, Inc. v. Thomson Corp.*, 532 F.3d 1318, 1329 (Fed Cir. 2008), the Court stated that:

> In *BMC Resources*, this court clarified the proper standard for whether a method claim is directly infringed by the combined actions of

> multiple parties. The court's analysis was founded on the proposition that direct infringement requires a single party to perform every step of a claimed method. 498 F.3d at 1380 (concluding that this requirement derived directly from 35 U.S.C. § 271(a)); *see also NTP, Inc. v. Research in Motion*, 418 F.3d 1282, 1317-18 (Fed. Cir. 2005).[1]

Muniauction, 532 F.3d at 1329.

Similarly, subsequent cases, including *Akamai* and *McKesson Technologies, Inc. v. Epic Systems Corp.*, 2010-1291 (Fed. Cir. Apr. 12, 2011), have all reiterated and relied on this single entity rule. *Akamai*, 629 F.3d at 1318 ("It is well settled that direct infringement requires a single party to perform every step of a claimed method.") (citing *BMC Res.*, 498 F.3d at 1378-79; *Warner-Jenkinson Co. v. Hilton Davis Chem. Co.*, 520 U.S. 17, 40 (1997)); *McKesson*, No. 2010-1291, slip op. at 6 ("A method claim is directly infringed only if each step of the claimed method is performed by a single party.") (citing *BMC Res.*, 498 F.3d at 1378-79)).

All of these cases cite to each other and to *BMC Resources*. Yet, neither *BMC Resources* nor any of the authorities cited therein provides legal support for the assertion that a method claim can only be directly infringed by a single entity. Indeed, there is not a single decision from this Court or the Supreme Court, including *Warner-Jenkinson; Canton Bio-Medical, Inc. v. Integrated Liner Technologies, Inc.*, 216 F.3d 1367 (Fed. Cir. 2000); *General Foods Corp. v.*

---

[1] *NTP* discusses the all elements rule for infringement. It does not, however, provide any analysis or state that "direct infringement requires a single party."

Studiengesellschaft Kohle mbH, 972 F.2d 1272 (Fed. Cir. 1992); *Joy Technologies, Inc. v. Flakt, Inc.*, 6 F.3d 770 (Fed. Cir. 1993); *Fromson v. Advance Offset Plate, Inc.*, 720 F.2d 1565 (Fed. Cir. 1983); and *Cross Medical Products, Inc. v. Medtronic Sofamor Danek, Inc.*, 424 F.3d 1293 (Fed. Cir. 2005), all relied on by *BMC Resources*, that provides a sound, reasoned basis for the single entity restriction the Court has imposed on § 271(a).

In *Warner-Jenkinson*, the Supreme Court, in the context of clarifying the doctrine of equivalents, held that "[e]ach element contained in a patent claim is deemed material to defining the scope of the patented invention, and thus the doctrine of equivalents must be applied to individual elements of the claim, not to the invention as a whole." 520 U.S. at 29. Similarly, *Canton Bio-Medical*, 216 F.3d at 1370, and *General Foods*, 972 F.2d at 1274, merely note that each and every element of a method claim must be practiced to constitute infringement. None of these cases, however, describes the parties ***who*** must practice each element.[2] Likewise, *Joy Technologies*, 6 F.3d at 773, referencing § 271(a), notes that "[t]he making, using, or selling of a patented invention is the usual meaning of the expression 'direct infringement.'" Accordingly, although these cases suggest ***what*** constitutes direct infringement of a method claim—that is, the practice of

[2] In fact, in a recent case, the Supreme Court noted that "[d]irect infringement has long been understood to *require no more than* the unauthorized use of a patented invention." *Global-Tech Appliances, Inc. v. SEB S.A.*, 563 U.S. ___, slip op. at 5 n.2 (May 31, 2011) (emphasis added).

each and every step of a claimed method—not one of these cases addresses the issue of *who* must practice the steps.

BMC Resources further cites *Fromson* and *Cross Medical*. A careful inspection of these cases, however, demonstrates that the single entity rule was not the basis for the holding in either case. *Fromson* turned on the proper construction of the claim term "reaction." 720 F.2d at 1571. Because the claim term was construed improperly, the court reversed and remanded the case to the district court. *Id.* *BMC Resources* relied on a statement in the "Background" section of the *Fromson* opinion: "Because the claims include the application of a diazo coating or other light sensitive layer and because Advance's customers, not Advance, applied the diazo coating, Advance cannot be liable for direct infringement with respect to those plates but could be liable for contributory infringement." *Id.* at 1568. Indeed, the statement on its face appears contradictory. In any event, there is no reasoned analysis in *Fromson* supporting the existence of the single entity rule.

Similarly, in *Cross Medical*, Cross Medical had asserted that Medtronic infringed claims to an apparatus because the Medtronic apparatus was capable of being operated in an infringing manner by the physicians in an operating room. 424 F.3d at 1310. Cross Medical argued that Medtronic was liable for direct infringement because of its interactions with the physicians in the operating room.

Id. at 1311. Although the Court noted in rejecting Medtronic's argument that the physicians in the operating room were not "agents" of Medtronic, *id.*, there was no reasoned analysis on this point. The decision did not hold that direct infringement is limited to one person.[3]

Moreover, far from supporting the single entity rule, the district court cases cited in *BMC Resources* actually support the proposition that two actors can directly infringe a claim. For example, *BMC Resources* cites *Shields v. Halliburton Co.*, 493 F.Supp. 1376 (W.D. La. 1980). In *Shields*, the court found that the combined actions of the parties (Halliburton and Brown & Root) jointly infringed:

> When infringement results from the ***participation and combined action of several parties***, they are all ***joint infringers and jointly liable*** for patent infringement. *New Jersey Patent Co. v. Schaeffer*, 159 F. 171 (E.D. Pa.1908), aff'd 178 F. 276 (3rd Cir. 1909). Infringement of a patented process or method ***cannot*** be avoided by having another perform one step of the process or method. *Metal Film Company v. Milton* [sic *Metlon*] *Corporation*, 316 F. Supp. 96 (S.D.N.Y. 1970).

Shields, 493 F. Supp. at 1389 (emphases added).

---

[3] Similarly, caselaw such as *Dynacore Holdings Corp. v. U.S. Phillips Corp.*, 363 F.3d 1263, 1272 (Fed. Cir. 2004) (cited in *BMC Resources*), and *Aro Manufacturing Co. v. Convertible Top Replacement Co.*, 365 U.S. 336, 366-67 (1961) (cited in *McKesson*), which note that inducement or contributory infringement requires a showing of direct infringement, do not discuss a single entity requirement for direct infringement.

Further, *New Jersey Patent* and *Metal Film*, the cases cited in *Shields*, certainly allow for direct infringement by two or more parties. In *New Jersey Patent*, the district court noted that "[w]here an infringement of a patent is brought about by ***concert of action*** between a defendant and complainants' licensee, all engaged directly and intentionally become joint infringers." 159 F. at 173 (emphasis added). In *Metal Film*, contracting out a step of a method claim did not preclude liability for the "mastermind": "That defendants choose to have the vacuum metallizing, which was a conventional step (used, for example, in producing the Prindle laminated yarn), done by outside suppliers does not mitigate their infringement of the overall process." 316 F. Supp. at 110 n.12. It was apparently the discussion in *Shields*, *New Jersey Patent*, and *Metal Film* that led to the statement in *BMC Resources* that one party cannot simply contract out a step of a method claim to avoid liability for infringement. *BMC Res.*, 498 F.3d at 1381 (citing *Shields*, 493 F. Supp. at 1389). This important aspect of the *BMC Resources* analysis was stripped out by subsequent Federal Circuit cases.

Two other district court opinions cited in *BMC Resources*, including *Faroudja Laboratories, Inc. v. Dwin Electronics, Inc,* No. 97-20010 SW, 1999 WL 111788 (N.D. Cal. Feb. 24, 1999), and *Mobil Oil Corp. v. Filtrol Corp.*, 501 F.2d 282 (9th Cir. 1974), also did not rely on a single entity rule. While *Faroudja* found no direct infringement when different parties performed different steps of a method

claim, it did so because there was not a significant enough relationship between the parties. 1999 WL 111788 at *6-7. *Faroudja*, however, expressly noted that courts have found direct infringement where two or more actors worked together to infringe a patent. *Id.* at *5 ("It is true that several district courts have found a party liable for direct infringement of a process patent even where the various steps included in the patent are performed by distinct entities. However, these cases indicate that some connection between the different entities justified that finding."). *Mobil Oil* also stated that joint infringement cannot be found in situations where two actors perform steps but have *no* connection, but it does *not* say that joint infringement can never be found. 501 F.2d at 291-92. In *Mobil Oil*, the combined actions of the defendants did not complete all of the method steps (i.e., neither completed a washing step of the claim) and, accordingly, it was the failure of anyone to perform a claim step that resulted in no liability. *Id.*

As can be seen from the foregoing discussion, not one of the cases cited by *BMC Resources* provided a reasoned analysis as to why only a single actor can infringe a method claim. Some of the cases cited in the opinion were cited for the recitation of the all elements rule, others merely restated the single entity rule without supporting reasoning, and others actually supported the opposite proposition, i.e., that more than one entity can infringe a method claim.

2. The Statute Also Does Not Support a Single Entity Rule

It is not surprising that none of the above cases provides a sufficient basis for limiting infringement of a method claim to only a single actor, as the statute plainly does not require such an outcome. Section 271(a) of the 1952 Patent Act imposes liability on "whoever . . . uses . . . any patented invention." There is nothing in this language to suggest that "whoever" refers to a single entity when applied to method claims. Rather, according to common dictionary definitions, "whoever" in § 271(a) means "[w]hatever person or ***persons***." *See* American Heritage College Dictionary 1540 (3d ed. 1997) (emphasis added). "Dictionaries of the English language provide the ordinary meaning of words used in statutes." *Bayer AG v. Housey Pharms., Inc.*, 340 F.3d 1367, 1371 (Fed. Cir. 2003). Moreover, 1 U.S.C. § 1 states: "In determining the meaning of any Act of Congress, unless the context indicates otherwise—words importing the singular include and apply to several persons, parties or things . . . ." Accordingly, consistent with the plain meaning and as indicated by Congress in 1 U.S.C. § 1, "whoever" in § 271(a) means person ***or*** persons.

Further, this interpretation of the Patent Act is supported by the authority existing ***before*** the passage of the Patent Act in 1952, and the Supreme Court has explained that the Act preserved pre-codification infringement principles: "In the context of infringement, we have already held that pre-1952 precedent survived the

passage of the 1952 Act." *Warner-Jenkinson*, 520 U.S. at 26. "Section 271(a) was merely a codification of the common law of infringement that had developed up to the time of passage of the 1952 Patent Act. It was not meant to change the law of infringement." *NTP*, 418 F.3d at 1319; *see also Microsoft Corp. v. i4i Limited Partnership*, 563 U.S. ___, slip. op. at 8-9 (June 9, 2011) (looking to pre-1952 precedent to assess common law presumption of validity at time of enactment).

In *Peerless Equipment Co. v W. H. Miner, Inc.*, 93 F.2d 98 (7th Cir. 1937), the Seventh Circuit considered whether a manufacturer could escape liability for infringing a process claim by enlisting its customer to complete the final step. The patent was for a process for making train gears, which required as one of the steps "successively compressing the mechanism to flatten down said protruding portion to increase the area of surface contact of said last-named faces." *Id.* at 102 n.2. The manufacturer did not perform this last step, but instead left it to its customers to complete. *Id.* at 105. The court upheld a finding of infringement because the manufacturer passed the nearly finished gears on to the customer "with the knowledge that the railroads will put them to use and thereby flatten the crown, thus completing the final step of the process." *Id.* Likewise, *New Jersey Patent*, discussed above, illustrates the existence of joint infringement prior to passage of the Patent Act. There, the court specifically stated that "[w]here an infringement of a patent is brought about by concert of action between a defendant and

complainants' licensee, all engaged directly and intentionally become joint infringers." 159 F. at 173.

Similarly, in *Halliburton v. Honolulu Oil Corp.*, 98 F.2d 436, 440 (9th Cir. 1938), *rev'd on validity grounds*, 306 U.S. 550 (1939), the court held in connection with a process patent that two defendants were jointly liable as infringers: "We find that the Honolulu Oil Corporation participated ***jointly*** in infringement in using the process on the wells drilled by it. We hold that there was infringement of the process by the Honolulu Oil Corporation as well as by appellee M. O. Johnston Oil Field Service Corporation." (Emphasis added.)

And, in *Jackson v. Nagle*, 47 F. 703, 704 (N.D. Cal. 1891), the court held two defendants jointly liable for infringing patents, one of which was a method patent, where one defendant performed "a portion of the [infringing] work" and the "other portions" of the infringing work were performed by the other defendant. For this reason, the court found that "the respondents must be treated and held as joint infringers." *Id.*; *see also* William C. Robinson, *The Law of Patents for Useful Inventions*, § 904 (1890) ("To use in part with intent that others shall complete the operation, . . . is likewise an infringement.").

These cases did not restrict infringement to a single entity or to parties with an agency or contractual relationship, but instead used a broad, flexible approach to determine whether the specific actions by the parties were sufficient to assess

liability for joint infringement. As mentioned above, the adoption of the 1952 Patent Act did not extinguish the viability of the pre-1952 precedents concerning § 271(a), but merely constituted "a codification of the common law of infringement" that had previously existed. *Warner-Jenkinson*, 520 U.S. at 26; *NTP*, 418 F.3d at 1319; *see also Lorillard v. Pons*, 434 U.S. 575, 580 (1978) ("Congress is presumed to be aware of an administrative or judicial interpretation of a statute and to adopt that interpretation when it re-enacts a statute without change."); *In re Air Crash Disaster*, 210 F. Supp. 2d 570, 575 (E.D. Pa. 2002)("[W]hen Congress amends an existing statute, a court must presume that any part of the statute left intact reflects Congress' intent to preserve the prevailing judicial interpretation of that portion."). Indeed, if Congress meant to abrogate the common law precedent concerning joint infringement when it enacted the 1952 Patent Act, it would have said so expressly. There is nothing in either the language of the statute or the legislative history, however, that supports such an abrogation of precedent. Accordingly, there is no basis in either the statute or the precedent for a single entity test for establishing joint infringement.

B. Common Law Principles of Tort Law Regarding Joint Liability Should Apply to Patent Infringement

Without support for limiting direct infringement to a single entity, the question becomes one of how to define the relationships between two or more parties that would be sufficient to find direct infringement. Given that patent

infringement is a tort, it is logical to examine tort law for guidance. *See Dowagiac Mfg. Co. v. Minn. Moline Plow Co.*, 235 U.S. 641, 648 (1915) (holding that patent infringement was a "tortious taking"); *Carbice Corp. of Am. v. Am. Patents Dev. Corp.*, 283 U.S. 27, 33 (1931) ("Infringement, whether direct or contributory, is essentially a tort, and implies invasion of some right of the patentee.").[4] This Court has already applied common law principles of tort law by relying on vicarious liability in developing the "direction or control" test. *See, e.g., BMC Res.*, 498 F.3d at 1379 (citing *Engle v. Dinehart*, 213 F.3d 639 (5th Cir. 2000) (unpublished decision)). This Court erred, however, in limiting the doctrines applicable to liability for joint infringement to an agency or contractual relationship.

Indeed, it has long been recognized under common law tort principles that agency liability is not the ***sole*** basis for holding joint tortfeasors liable. Over time, common law courts have developed a series of distinct but overlapping bases for joint and vicarious liability. *See* Restatement (Second) of Torts §§ 875-879 (1979). These rules work together to establish liability in a variety of "circumstances in which it is just to hold one individual accountable for the actions

[4] The Supreme Court has looked to common law principles in other cases, including *eBay Inc. v. MercExchange, L.L.C.*, 547 U.S. 388 (2006), (determining the appropriate standard for injunctions in patent cases), and in *MedImmune, Inc. v. Genentech, Inc.*, 549 U.S. 118 (2007) (examining declaratory judgment law). Similarly, in *Global-Tech*, 563 U.S. ___, slip op. at 10, the Supreme Court looked to well-established principles of criminal law when examining the issue of knowledge required under 35 U.S.C. § 271(b).

of another" in harming another party's interests. *See, e.g., Sony Corp. of Am v. Universal City Studios, Inc.*, 464 U.S. 417, 435 (1984). There are at least three such bases that should apply to patent infringement under § 271(a).

1. When One Party Directs or Controls Another Party's Performance of a Method Step, the Other Party's Performance of That Step Should Be Attributed to the Directing or Controlling Party

First, as this Court has already held in *BMC Resources*, if one party "control[s] or direct[s] [the performance of] each step of the patented process," those steps may be attributed to the directing or controlling party. *BMC Res.*, 498 F.3d at 1380-81. It should make no difference whether the directed or controlled party is acting as an "agent" of the other within the formal requirements of agency law. *See, e.g., Gershwin Publ'g Corp. v. Columbia Artists Mgmt., Inc.*, 443 F.2d 1159, 1162 (2d Cir. 1971) ("This court attached no special significance to the technical classification of the Green-Jalen relationship."). It has long been recognized that agency liability is not the basis for holding joint tortfeasors liable when one acts pursuant to the instructions of the other and performance of the very thing that was directed causes harm. Moreover, there is no basis for limiting the "direction or control" test to simply one of "control" as the Panel did here.

The concept of "direction" is expressly addressed by § 212 of the Restatement (Second) of Agency (1958).[5] This section notes that "[a] person is subject to liability for the consequences of another's conduct which results from his ***directions*** as he would be for his own personal conduct if, with knowledge of the conditions, he intends the conduct, or if he intends its consequences, unless the one directing or the one acting has a privilege or immunity not available to the other." *Id.* (emphasis added). However, as the comment to this section notes, "[t]he rule stated in this Section is ***not dependent upon the law of agency*** but results from the general rule, stated in the Restatement of Torts, that one causing and intending an act or result is as responsible as if he had personally performed the act or produced the result. *See* the Restatement (Second) of Torts, §§ 870, 876, 877. If one intends a particular result to follow from his conduct and the result follows, it is immaterial that the particular way in which it is accomplished was unintended." Restatement (Second) Agency § 212, *cmt. a* (emphasis added); *see also Kellogg v. Payne*, 21 Iowa 575 at *2 (1866) ("[I]f a person employ another, although by express and independent contract, to erect a nuisance, or do any other work directly or necessarily injurious to a third person, he will be liable to such third person for damages resulting from the nuisance, or work. But this liability

[5] *BMC Resources*, which also addressed the concept of control, cited to a different section of the Restatement (Second) of Agency, specifically, § 220, which notes that even for "control," the evidence of "control needed to establish the relation of master and servant may be ***very attenuated***." (emphasis added).

rests upon the idea that he is a co-trespasser, by reason of his directing or participating in the work done, and not on the principle of respondeat superior."); *Gershwin*, 443 F.2d at 1162 ("Although vicarious liability was initially predicated upon the agency doctrine of respondeat superior, this court recently held that even in the absence of an employer-employee relationship one may be vicariously liable if he has the right and ability to supervise the infringing activity and also has a direct financial interest in such activities."(citation omitted)).

In addition to § 212, the Restatement (Second) Torts, § 877(a) further supports Akamai's position that "direction or control" need not amount to an agency or contractual relationship. This section states that for harm resulting to a third person from the tortious conduct of another, one is subject to liability if he "orders" the conduct. The Restatement also makes clear that "one who accomplishes a particular consequence is as responsible for it when accomplished through directions to another as when accomplished by himself." *Id.* at *cmt. a.* This section, however, also makes abundantly clear that an agency relationship is not necessary to impose liability, specifically emphasizing that such imputation is "independent of" and not limited to the master-servant relationship. *Id.* Thus, at the very least, this Court should return to the more flexible "direction or control" standard set forth in *BMC Resources* and supported by the caselaw and the Restatements.

2. Joint Actors Should Be Liable When Acting in Concert

In addition, joint actors should be liable when they act in concert to perform the steps that constitute a method claim. There is no basis under precedent, the Patent Act, or the policies underlying the Patent Act, for restricting principles of vicarious liability to only those circumstances where one actor dominates another as described above. (*See* § IV.B.1, *supra.*) "Respondeat superior is ***not*** the only kind of vicarious liability . . . those who act in concert, partners, and joint enterprisers are all vicariously liable for the acts of each other committed as part of their expressly or tacitly agreed-upon activity." Dan B. Dobbs, *The Law of Torts*, (West Group 2000) (emphasis added).

As noted in the Restatement (Second) of Torts, § 876(a), a person is subject to liability when he or she "does a tortious act in concert with the other or pursuant to a common design." *See also* W. Page Keeton et al., Prosser & Keeton on Torts, § 46, at 322 (5th ed. 1984) ("The original meaning of a 'joint tort' was that of vicarious liability for concerted actions. All persons who acted in concert to commit a trespass, in pursuance of a common design, were held liable for the entire result."). As defined in the comments section to this provision:

> Parties are acting in concert when they act in accordance with an agreement to cooperate in a particular line of conduct or to accomplish a particular result. The agreement need not be expressed in words and may be implied and understood to exist from the conduct itself. Whenever two or more persons commit tortious acts in concert,

> each becomes subject to liability for the acts of the others, as well as for his own acts. The theory of the early common law was that there was a mutual agency of each to act for the others, which made all liable for the tortious acts of any one.

See Restatement (Second) of Torts, § 876(a), *cmt. a.*[6]

Persons or entities who commit torts acting in concert are jointly liable. *Id.* Moreover, the independent acts of each party in themselves need not be tortious standing alone. While comment c to § 876(a) states that "it is essential that the conduct of the actor be in itself tortious," the comment goes on to explain that an actor "who innocently, rightfully and carefully does an act that has the effect of furthering the tortious conduct or cooperating in the tortious design of another" is not liable. *Id.* Thus, this comment merely explains that innocent actors are not liable. It does not mean that each party in a concerted action must itself perform a tort. This is supported by case law.

For example, in *Aeroglide Corp. v. Zeh*, 301 F.2d 420, 422 (2d Cir. 1962), the Second Circuit imposed joint liability on the individual participants of a strict liability tort—conversion—where the tort involved the combined participation and

[6] Although § 876(a) contains the caveat that "[t]he Institute takes no position on whether the rules stated in this Section [§ 876] are applicable when the conduct of either the actor or the other is free from intent to do harm or negligence but involves strict liability for the resulting harm," the comment to this caveat makes clear that this relates to cases involving "liability for the escape of animals and for abnormally dangerous conduct for which there is strict liability." Restatement (Second) of Torts, *cmt. f*; *In re Hassan*, Bankruptcy No. 04-20332-7, 2010 WL 5348770, at *12 n.34 (Bankr. D. Kan. Dec. 21, 2010). (noting that the caveat is limited to the escape of animals and dangerous conduct, and does not implicate nondangerous strict liability torts such as conversion).

action of the defendants and where the actions of each individual defendant, viewed alone, were likely not tortious. Further, in *Taylor v. Conti*, 177 A.2d 670, 672 (Conn. 1962), the court imposed liability where the harm to plaintiff would likely not have occurred without ***both*** defendants' actions. In that case, the defendant property owner had contracted with the defendant independent contractor to perform grading work and soil removal to improve his land. *Id.* at 671. This work resulted in silt and soil erosion on plaintiff's property for which both defendants were held liable. *Id.* at 672. The relationship between defendant property owner and defendant independent contractor did not appear to be an agency relationship at least in part due to the fact that defendant was using the removed soil for "its [own] purposes" independent of the defendant property owner. *Id.*

Thus, by applying these common law principles from tort law, one can formulate a test that applies in the context of joint infringement whereby infringement may be found if two or more parties act in concert to carry out the steps of a patented method. This test provides a sensible workable standard for patent infringement, and is consistent with both the common law and the U.S. Supreme Court's preference for flexible fact-based standards that avoid bright-line rules. (*See* § IV.G., *infra.*)

There is ample support for this test in patent cases. For example, in *On Demand Machine Corp. v. Ingram Industries, Inc.*, 442 F.3d 1331, 1345 (Fed. Cir. 2006), this Court agreed with a jury instruction that direct infringement occurs when participants work together in concert to perform the steps of a patented method. The jury instruction stated:

> It is not necessary for the acts that constitute infringement to be performed by one person or entity. When infringement results from the participation and combined action(s) of more than one person or entity, they are all joint infringers and jointly liable for patent infringement. Infringement of a patented process or method cannot be avoided by having another perform one step of the process or method. Where the infringement is the result of the participation and combined action(s) of one or more persons or entities, they are joint infringers and are jointly liable for the infringement.

Id. at 1344-45. This Court explained that it could "discern no flaw in this instruction as a statement of law." *Id.* at 1345; *see also McKesson*, No. 2010-1291, slip op. at 11-12. (Newman J., dissenting).

Although *BMC Resources* dismissed this language in *On Demand* as dicta, this test was also applied in *Shields*, 493 F. Supp. at 1389, and *Metal Film,* 316 F. Supp. at 110 n.12, both discussed above. (See § IV.A.1, *supra*.) Likewise, as discussed above, in *New Jersey Patent*, 159 F. at 173, the court noted that "[w]here an infringement of a patent is brought about by ***concert of action*** between a defendant and complainants' licensee, all engaged directly and intentionally become joint infringers." Similarly, in *Faroudja*, 1999 U.S. WL 111788, at * 5-6,

also discussed above, the district court examined several divided infringement fact patterns from previous cases, including *E.I. duPont De Nemours & Co. v. Monsanto Co.*, 903 F. Supp. 680, 735 (D. Del. 1995), *aff'd without op.*, 92 F.3d 1208 (Fed Cir. 1996); *Shields*, 493 F. Supp. at 1389; *Free Standing Stuffer, Inc. v. Holly Development Co.*, 187 U.S.P.Q. 323, 333 (N.D. Ill. 1974); and *Metal Film,* 316 F. Supp. at 110, and noted that "each demonstrate that the entities found to directly infringe patented processes ***worked in concert*** with other entities to complete the process of infringement." (Emphasis added.) These fact patterns included having someone perform a step (*Monsanto*), instructing another to perform a step (*Free Standing Stuffer*), contracting out a step (*Shields*), and arranging with an outside supplier to perform a step (*Metal Film*). *BMC Resources* itself recognized that "[a] party cannot avoid infringement . . . simply by contracting out steps of a patented process to another entity." 498 F.3d at 1381 (citing *Shields*, 493 F. Supp. at 1389).

Other examples exist under the patent law where parties acting in concert have been held jointly and severably liable for the actions of the other. For example, in *Evident Corp. v. Church & Dwight Co.*, 399 F.3d 1310 (Fed. Cir. 2005), this Court faced the question of how to apportion an attorney-fee award in a case involving inequitable conduct. Applying the "common law principle of mutual agency," this Court explained that a "partnership, or every member thereof,

is liable for torts committed by one of the members acting in the scope of the firm business." *Id.* at 1316. Thus, while acknowledging that inequitable conduct is not a tort, the Court held that the "principle of joint responsibility equally applies to Peroxydent and its partners." *Id.* "Because of the close, intertwined relationship between the Peroxydent partners, the Evident shareholders, and the inventors of the '782 patent, Peroxydent cannot be said to be innocent of the underlying inequitable conduct." *Id.* Further, Fed. R. Civ. P. 65(d)(2)(C) specifically includes "persons who are in active concert or participation" as parties to be bound by an injunction.

Accordingly, and consistent with principles of tort law, Akamai asserts that joint action by parties "acting in concert" to carry out the steps that constitute a patented method should make such parties jointly liable.

3. Even Independent Actors Are Liable If They Knew of the Combined Conduct

Finally, liability should attach where two parties together perform all the steps of a patented method even if one of the parties is ***unaware*** that the other party has carried out such steps. For example, one party might knowingly (either actually or constructively) carry out four of the five steps that constitute a method patent, and then cause its customers to unknowingly carry out the remaining step. In this circumstance, the party who knowingly carried out the four steps would be liable for infringement—even if the "innocent" customer carrying out the fifth step would avoid liability.

Such a circumstance is analogous, for example, to a scenario where one actor, while carefully driving a car, causes harm to someone as a result of an unknown defect caused by the negligence of a third party, such as the car's owner. *See S.E. Greyhound Lines v. Callahan*, 13 So.2d 660, 663 (Ala. 1943). In that situation, only the owner is liable, even though its acts alone would not have injured the plaintiff. *See id.* Similarly, where a railroad company caused a gas leak, and an individual lit a match, causing grave harm to others, the court recognized that the individual's liability turned on whether he knew of the gas leak, and the railroad was liable even if the individual was not. *See Watson v. Ky. & Ind. Bridge & R.R. Co.*, 126 S.W. 146, 150 (Ky.), *modified* 129 S.W. 341 (Ky. 1910). Additionally, the Tenth Circuit held some but not all defendants liable in a securities case where a joint actor who had approached the plaintiffs, and whose actions were necessary to the tort, was actually "a victim of the scheme rather than a knowing participant in it," and thus not liable. *See Stadia Oil & Uranium Co. v. Wheelis*, 251 F.2d 269, 276 (10th Cir. 1957).

Perhaps the most applicable fact pattern to the issue before this en banc Court is that of *Peerless*, 93 F.2d at 98. There, the Seventh Circuit considered whether a manufacturer could escape liability for infringing a process claim by enlisting its customer to complete the final step. As discussed above, the patent was for a process for making train gears, in which the manufacturer did not

perform this last step, but instead left it to its customers to complete. *Id.* at 105. The court upheld a finding of infringement because the manufacturer passed the nearly finished gears on to the customer "with the knowledge that the railroads will put them to use and thereby flatten the crown, thus completing the final step of the process." *Id.* Under this fact pattern, Akamai submits that the manufacturer would be directly liable for patent infringement while the customers performing the last steps would not.

C. A Flexible Approach to Joint Liability Is Consistent with Supreme Court Policy

Each of the above three circumstances is based on common law rules for joint liability and provides a sensible, workable standard for patent infringement. Further, these common law tort principles are consistent with the U.S. Supreme Court's preference for flexible fact-based standards that avoid bright-line rules, as well as the Court's preference for applying common law doctrines applicable to other areas of the law to patent law.

The Supreme Court's preference for flexible standards is well known and is illustrated by a number of its recent decisions. In *KSR International Co. v. Teleflex Inc.*, 550 U.S. 398, 415 (2007), for example, the Supreme Court rejected a rigid approach of using solely a "TSM test" for determining obviousness, noting that it may be a test for determining obviousness, but it was not the only test. *See also Bilski v. Kappos*, 130 S. Ct. 3218, 3225-26 (2010) (rejecting bright-line machine or

transformation test for determining patentable subject matter); *eBay*, 547 U.S. at 392-93 (rejecting the Federal Circuit's bright-line grant of permanent injunctions when validity and infringement have been found); *Pfaff v. Wells Elecs., Inc.*, 525 U.S. 55, 65-66 (1998) (rejecting a bright-line rule that an invention cannot be "on sale" unless and until it is reduced to practice).

D. Applying the Common Law of Torts to Cases of Joint Infringement Will Not Subvert the Statutory Scheme of Indirect Infringement

BMC Resources and other cases suggest that establishing joint or collaborative direct infringement under § 271(a) would subvert the statutory scheme of indirect infringement under §§ 271(b) and (c), which require knowledge and/or intent for liability: "[A] patentee would rarely, if ever, need to bring a claim for indirect infringement." *BMC Res.*, 498 F.3d at 1381. This is simply not correct.

That two parties jointly directly infringe a method claim (i.e., each performs different steps of the method) does not subvert induced or contributory infringement. A manufacturer or vendor of a machine designed to carry out a patented method would still need to be sued for indirect infringement under §§ 271(b) and/or (c) if the manufacturer or vendor does not itself practice the method. But, for there even to be the possibility of indirect infringement, there must ***first*** be a direct infringement. Because indirect infringement requires a

threshold showing of direct infringement under current law, a patentee that cannot establish that a single entity directly infringes a claim cannot bring an action for indirect infringement.

Thus, rather than subvert the statutory scheme, the Court's requirement that each step of a claim be performed by a single entity or on its behalf to find direct infringement actually precludes a patentee in some cases from ever being able to prevail on a claim under §§ 271(b) or (c). This was amply demonstrated in the *McKesson* case. There, the patent owner accused the defendant of inducing two other parties, a doctor and a patient, to jointly perform the steps of a method claim. The defendant, however, could not be held liable for inducement because there was no direct infringement under the rigid test adopted in *Akamai*. This unjust result, that a patent can never be infringed in a situation such as this, cannot be correct and demonstrates the error of a single entity requirement of § 271(a). *See McKesson*, No. 2010-1291, slip op. at 6 (Newman, J., dissenting) ("A patent that can never be infringed is not a patent in the definition of the law, for a patent that cannot be infringed does not have the 'right to exclude.' This court's elimination of infringement, by creating a new but far-reaching restriction is inappropriate.")

In sum, that two parties jointly directly infringe a method claim (i.e., each performing different steps of the method) does not subvert induced or contributory infringement.

E. Precluding Application of the Common Law Principles of Joint Liability Makes Method Claims Unfairly Vulnerable to Loopholes in the Law

Patents are often the result of a significant investment of money, resources, innovation, and research. Unduly restricting joint infringers to only those who have a master-servant agency or contractual relationship effectively defeats the value of many multi-participant patent claims and renders all method claims vulnerable to loopholes in the protection of such inventions. Indeed, recent articles directed to in-house counsel provide specific instructions on how to structure language of contracts "so that no mastermind exists" in order to avoid infringement liability. Tonya M. Gray, *Contract Clauses Offer Protection in Infringement Suits*, In-House Texas, vol.25, no.41 (Jan. 11, 2010). Likewise, companies have "formed a strategic partnership, enabled their two [products] to work together, and collaborated to sell the two [products] as a unit" and yet were not liable for infringement of a method claim even though, together, their two products performed every element of the method claim. *See Golden Hour Data Sys., Inc. v. emsCharts, Inc.*, 614 F.3d 1367, 1380-81 (Fed. Cir. 2010).

As aptly stated by the Sixth Circuit in *Thomson-Houston Electric Co. v. Ohio Brass Co.*:

> From the earliest times, all who take part in a trespass, either by actual participation therein or by aiding and abetting it, have been held to be jointly and severally liable for the injury inflicted. There must be some concert of action between him who does the injury and him who

> is charged with aiding and abetting, before the latter can be held liable. When that is present, however, the joint liability of both the principal and the accomplice has been invariably enforced. If this healthful rule is not to apply to trespass upon patent property, then, indeed, the protection which is promised by the constitution and laws of the United States to inventors is a poor sham.

80 F. 712, 721 (6th Cir. 1897). As also noted by Judge Newman, "[a] patent that cannot be enforced on any theory of infringement, is not a statutory patent right. It is a cynical, and expensive, delusion to encourage innovators to develop new interactive procedures, only to find that the courts will not recognize the patent because the participants are independent entities." *McKesson*, No. 2010-1291, slip op. at 17 (Newman J., dissenting). These considerations are particularly important in the Internet Age where many genuine process inventions will involve the combined actions of two or more computer devices controlled by two or more persons.

This is exactly the case with the invention at issue here, embodied in Akamai's '703 patent. As explained in the '703 patent specification, one novel and advantageous aspect of Akamai's invention (and the one to which several claims at issue in the '703 patent are directed) is having a first entity (the Content Provider) provide the web-page base document to an end user, while a second entity provides other objects in the web page to the end user. As claimed in the '703 patent, it is this very shared responsibility that is core to this particular aspect of the invention, a point emphasized in the patent's specification (*See, e.g.*, A267 col.2 ll.17-22

("There remains a significant need in the art to provide a decentralized hosting solution that . . . enables the Content Provider to maintain control over its content" while still providing unlimited, cost-effective global delivery of the Provider's content while addressing the other deficiencies in the prior art).)[7] As evidenced by Limelight's use of Akamai's invention, the jury's infringement verdict, and the $40 million damages award in this case, it cannot be reasonably disputed that the claimed process constitutes a useful and innovative process. Given the purpose of the patent laws, there is no reason for not protecting inventions of this type. "[T]his is a case of new technology adapted to public benefit—an advance supported by patent policy. Today's holding, and the few recent cases on which it builds, have the curious effect of removing from patent eligibility the burgeoning body of interactive computer-managed advances." *McKesson*, No. 2010-1291, slip op. at 17 (Newman J., dissenting).

Further, and contrary to the suggestion of the Panel, creative claim drafting does not provide a solution to the problem. *Akamai*, 629 F.3d at 1321-22. There is no reason why the viability of a novel and unobvious patent should depend solely on whether the claim drafter had the foresight to draft claims to cover solely one direct infringer. In fact, requiring such wordsmithing by the claim drafter is

[7] While Akamai does have other related process patents based on the same specification, this should not prevent Akamai from having proper protection for the multi-participant aspect of its invention.

inconsistent with the patentee's obligation to clearly claim the invention and exalts form over substance. It should not be adopted by this Court.

Moreover, cases like *Golden Hour* demonstrate that even if claims are drafted in the "correct" manner, such that they do not require the participation of multiple entities, the issue of joint infringement can still arise. In *Golden Hour*, the defendants created a strategic partnership to perform the claimed steps. 614 F.3d at 1380-81. Thus, joint infringement arises in many cases even though the claims might have been drafted consistent with the Court's advice to use "proper" claim drafting. *See, e.g.*, *Akamai*, 629 F.3d at 1322.

Nor is it a solution to this problem to suggest that holders of affected patents could seek reissue or reexamination. A reexamination or reissue to "correct" a multi-party claim would certainly be looked upon by the PTO as a broadening request on the grounds that it broadens the scope of a claim from one that no party infringed to one that at least one party infringed. As such, for reexaminations, single-party claims could not be sought at all and, for reissues, such claims could only be sought within two years of issuance. *See* 35 U.S.C. § 251; 35 U.S.C. § 305; MPEP §§ 1412.03(I), 2250 at 2200-76 (8th ed., rev. 8, July 2010). Thus, this remedy would not be available to most patentees. As set forth above, the most effective remedy is to apply a fact-based flexible standard supported—as has been

explained above —by the statute, by common law tort principles, and by precedent.

F. Liability Will Typically Extend to Those Carrying Out Steps of the Claimed Method, but Should Not Extend to "Innocent" Actors

Under common law tort rules, liability will typically extend to those who direct or control the performance of the method, act in concert, or knowingly combine their acts with another to carry out the steps of a claimed method. As explained below, however, these rules should not extend to "innocent" actors.

1. "Innocent" Actors

A party that exercises "direction or control" over the entire process such that every step is attributable to the directing or controlling party is liable. *BMC Res.*, 498 F.3d at 1380-81. But this does not mean that the directed or controlled party, who may have acted innocently and unknowingly in performing a single step of a claimed process, is necessarily liable.

Likewise, as described above, an independent actor is liable if that party knows that its actions may be combined with another's, and such conduct results in tortious harm. But, under this common law doctrine, courts have typically exonerated an innocent actor. For example, in *Kennecott Copper Corp. v. McDonell*, 413 P.2d 749, 753 (Ariz. 1966), the court exonerated a construction company for faulty construction where another party had "not only supplied the

plans and specifications but also actively supervised" the construction, and the court held the latter party liable for the injury caused by the acts of the construction company. Similarly, in *Watson*, 126 S.W. at 150, discussed above, the court held that the railroad may be liable even if the individual were found to escape liability.

Therefore, an innocent infringer, such as an Internet user who performs one step of a claim, not knowing that the other steps are being performed by someone else, would not be liable under a direction-or-control theory, an acting-in-concert theory or a knowingly-combine-to-perform-the steps theory. Thus, there is no risk that a truly innocent infringer would be liable for patent infringement under the tests proposed by Akamai.

2. Joint and Several Liability

"A suit for infringement may be analogized to other tort actions; all infringers are jointly and severally liable." Wright, Miller & Kane, 7 *Fed. Practice and Procedure Civil 3d* § 1614 & n.45 (citing *Thomson-Houston*, 80 F. at 721("An infringement of a patent is a tort analogous to trespass or trespass on the case. From the earliest time, all who take part in a trespass, either by actual participation therein or by aiding and abetting it, have been held to be jointly and severally liable for the injury inflicted.")). Thus, where courts normally apply joint and several liability, such liability should similarly apply in situations where two or more parties infringe a method claim, e.g., where two parties act in concert.

Accordingly, there will be situations where consumers may be technically liable for patent infringement, but only where the consumer directed or controlled the performance of the method, acted in concert with another, or knew that its actions may be combined with another's, and such conduct results in tortious harm. In such situations, however, these consumers could hardly be deemed "innocent."

Moreover, the idea of consumer infringers has always been a theoretical risk in patent law. For example, a consumer who uses a patented product has always been potentially liable as a direct infringer under § 271(a). *See, e.g.*, *Centillion Data Sys. LLC v. Qwest Commc'ns Int'l, Inc.*, 631 F.3d 1279, 1285 (Fed. Cir. 2011) (finding the customer actions controlled the system as a whole, so that the customer was a direct infringer.) But, as a practical matter, there has never been any real incentive for indiscriminately suing such consumers for patent infringement. The potential of consumer liability—already present in the law—is simply not a sufficient justification for requiring a strict agency or contract test for imposing liability for patent infringement.

Further, there are ways to protect such consumers. One way would be to analogize their conduct to that involved in cases concerning de minimis infringement. While parties committing de minimis acts of infringement are subject to liability, some courts have protected such parties in the context of calculating damages. For example, in *Micro Chemical, Inc. v. Great Plains*

Chemical Co., No. 88-Z-499, 1997 U.S. Dist LEXIS 23653, at *4 (D. Colo. Nov. 19, 1997), *aff'd in part and remanded in part*, 194 F.3d 1250 (Fed. Cir. 1999), the defendants argued that any infringement was de minimis because, at the time the patent in question issued, only fifteen allegedly infringing machines were in use and these machines were converted to a noninfringing design within four to six months. *Id.* The court, however, rejected the defendants' argument, holding that "de minimis infringement ***more properly relates to damages, and does not create an exception to liability***[;] [a]lthough defendants' . . . machines may have infringed Microchem's patent, damages determined at trial may be slight or non-existent." *Id.* (emphasis added).

Likewise, in his concurring opinion in *Embrex, Inc. v. Service Engineering Corp.*, 216 F.3d 1343, 1352-53 (Fed. Cir. 2000), Chief Judge Rader specifically stated that the Federal Circuit "has not tolerated the notion that a little infringement—de minimis infringement—is acceptable infringement or not infringement at all." According to Chief Judge Rader, § 271(a) "leaves no leeway to excuse infringement because the infringer only infringed a little." *Id.* at 1352. Instead, Chief Judge Rader explained that "the statute accommodates concerns about de minimis infringement in damages calculations." *Id.*

Moreover, a patent owner would not need to add such consumers to a lawsuit. The Supreme Court has explained that "[i]t has long been the rule that it is

not necessary for all joint tortfeasors to be named as defendants in a single lawsuit." *Temple v. Synthes Corp.*, 498 U.S. 5, 7 (1990). Similarly, the 1966 Advisory Committee Notes to Fed. R. Civ. P. 19(a), the rule governing joinder in the federal courts, state that "a tortfeasor with the usual 'joint-and-several' liability is merely a ***permissive party*** to an action against another with like liability." (Emphasis added.). Under Fed. R. Civ. P. 19(a), "permissive parties" need not be added to the case. This rule—that joint tortfeasors are permissive parties—has long been applied to patent cases. For example, in *Isogon Corp. v. Amdahl Corp.*, No. 97 Civ. 6219(SAS), 1997 WL 759435, at *2-3 (S.D.N.Y. Dec. 10, 1997), the court held that direct infringer customers were not necessary parties under Rule 19 to a patent-infringement action against an inducer. As the court explained, "the fact that a direct and indirect infringer [are] jointly liable for damages [does not] support the conclusion that a direct infringer is a necessary party to infringement claims against the indirect infringer." *Id.* at *3.

Likewise, in *Stabilisierungsfonds Fur Wein v. Kaiser Stuhl Wine Distributors Pty. Ltd.*, 647 F.2d 200 (D.C. Cir. 1981), the court explained that it has been "long held that in patent, trademark, literary property, and copyright infringement cases, any member of the distribution chain can be sued as an alleged joint tortfeasor. Since joint tortfeasors are jointly and severally liable, the victim of trademark infringement may sue as many or as few of the alleged wrongdoers as

he chooses; those left out of the lawsuit, commentary underscores, are not indispensable parties." *Id.* at 207 (citations omitted); *see also Kar Kraft Eng'g v. Shelby*, No. 06-14034, 2007 WL 1544397, at *3-4 (E.D. Mich. May 25, 2007) (rejecting argument that an alleged contributory trademark infringer was a necessary party under Rule 19); *see also* Wright, Miller & Kane, 7 *Fed. Practice and Procedure Civil* 3d § 1614 ("The question of who must be joined as defendants in patent, copyright, and trademark suits for infringement also is fairly easy to answer. A suit for infringement may be analogized to other tort actions; all infringers are jointly and severally liable. Thus, plaintiff may choose whom to sue and is not required to join all infringers in a single action." (footnote omitted)). Further, the Federal Circuit has ruled in a number of inducement cases where direct infringers were not parties. *See, e.g.*, *Lucent Techs., Inc. v. Gateway, Inc.*, 580 F.3d 1301, 1317-18 (Fed. Cir. 2009); *Moleculon Res. Corp. v. CBS, Inc.*, 793 F.2d 1261, 1272 (Fed. Cir. 1986).

G. Akamai Should Prevail Under Each of the Above Common Law Doctrines of Joint Liability

As set forth in extensive detail above, Akamai respectfully submits that the precedent, the statute, the policy underlying the Patent Act, and principles of fairness demonstrate that the Court should adopt a flexible standard based on principles of common law tort liability. There is simply no reason why a party who performs all steps of a method claim is liable, but parties who perform some

steps and (1) direct or control the performance of others; (2) act in concert with another to perform all the steps; ***or*** (3) knowingly combine their acts with those of another to perform all the steps should be able to avoid liability. Under this standard, there can be no reasonable question that Limelight should be held liable as a joint infringer, as will be shown below. It is first useful, however, to review briefly the facts at issue in this case.

1. Akamai's Inventive Method

The '703 patent is directed to an improved method of delivering web page content. *Akamai*, 629 F.3d at 1315. A web page typically includes a "base document," which serves as an outline for the web page, and "embedded objects," such as images or videos, that fill the outline. (A269, col.5 ll.23-27.) Traditionally, the entirety of this web page, including both the page itself and embedded objects, was delivered by a single entity. (A17241; A274, col.15 ll.33-45; A274 col.16 ll.37-69.) As explained in the specification, one novel and advantageous aspect of Akamai's invention—and the one to which several asserted claims are directed—involves having a first entity (the "Content Provider") deliver the base document and perhaps some of the objects, while a second entity (the "Content Delivery Network" or "CDN") delivers other objects in the web page. (A267, col.2 ll.7-22.) The CDN delivers content for many Content Providers from locations close to Internet end-users, reducing demand from the Content Providers'

servers. *Akamai*, 629 F.3d at 1315. This was a breakthrough in web content delivery, as it "provide[d] a scalable solution that could efficiently deliver large amounts of web content and handle flash crowds." *Id.*

As the '703 patent itself teaches, a novel aspect of this invention was that it relieved "Content Providers"—the first entities—from delivery of certain of their web page content, while still enabling them "to maintain control over" that content. (A267, col.2 ll.7-22). Their web page content would be delivered by the CDN—the second entity—over a "global network" that was highly available and could scale to protect against "flash crowds" that might visit the first entity's web site. Thus, the '703 patent's specification emphasized the "joint" nature of the activities (of the first and second entities) that were contemplated by the inventors. Asserted claims 19-21 claim this subject matter.

To make this system work, the inventors had to develop a method for Internet users to receive content from the CDN. (A339:40.) To this end, the claims at issue require that the embedded objects be "tagged" so that requests by end-user computers for the embedded objects are directed to the CDN. (A269, col.6 ll.41-46.) It is logical and consistent with the specification that the Content Provider (the customer) performs this step, as it is the customer who decides what content it wishes to be diverted from its website to the CDN.

At trial, Akamai asserted independent claims 19 and 34 and dependent claims 20-21 of the '703 patent. Claim 34 recites (with the tagging step in italics):

> 34. A content delivery method, comprising:
> distributing a set of page objects across a network of content servers managed by a domain other than a content provider domain, wherein the network of content servers are organized into a set of regions;
>
> for a given page normally served from the content provider domain, *tagging at least some of the embedded objects of the page* so that requests for the objects resolve to the domain instead of the content provider domain;
>
> in response to a client request for an embedded object of the page:
>
> resolving the client request as a function of a location of the client machine making the request and current Internet traffic conditions to identify a given region; and
>
> returning to the client an IP address of a given one of the content servers within the given region that is likely to host the embedded object and that is not overloaded.

(A276, col.20 ll.32-52) (emphasis added.)

Asserted claim 19 also requires "tagging" (A276, col.19 l.12) and additionally recites the step of "serving [i.e., delivering] the given page from the content provider domain" (*id.* at ll.15-16). Asserted claim 19 reads:

> 19. A content delivery service, comprising:
> replicating a set of page objects across a wide area network of content servers managed by a domain other than a content provider domain;
>
> for a given page normally served from the content provider domain, tagging the embedded objects of the page so that requests for the page objects resolve to the domain instead of the content provider domain;

> responsive to a request for the given page received at the content provider domain, serving the given page from the content provider domain; and
>
> serving at least one embedded object of the given page from a given content server in the domain instead of from the content provider domain.

(A276, col. 19 ll.6-20.) Claims 20-21 depend from claim 19. (*Id.* at ll.21-30.)

2. Limelight and Its Customers Perform All of the Steps of the Method Claimed in the '703 Patent

After Akamai had significant commercial success with the invention in the '703 patent, Limelight, Akamai's direct competitor, orchestrated a divided process that includes every step of the asserted claims of the '703 patent. According to that process, Limelight performs almost all the steps of the asserted claims while its customers (following the directions provided by Limelight) perform the remaining one or two steps, including the steps of tagging (claims 19-21 and 34) and serving the page with the tag (claims 19-21). *Akamai*, 629 F.3d at 1317.

Limelight and its customers also have a contractual relationship. When Limelight's customers (the Content Providers) choose to use Limelight's services for delivery of a particular object, they are contractually obligated to perform the tagging and/or serving steps if they want Limelight's service guarantee. (A587: 122; A17231; A17807-08.) In addition, Limelight provides Content Providers with the specific virtual hostname tag ("xyz.vo.llnwd.net") that the Content Provider must use to tag the embedded objects and explicit instructions on how to

perform the claim steps that Limelight does not perform. (A587: 122; A17220, A17231; A17237; A17790.) Limelight fully expects and desires that many customers who sign Limelight's contract and receive Limelight's detailed directions and a unique tag will, in fact, perform the missing claim steps because, otherwise, Limelight will not get paid by those customers. (A17803; A587, 122:19-22.)

As the *Akamai* Panel recognized, Limelight's contract with its customers details the specific claim steps that are to be performed by the customer and the overall result of their "divided process":

> This divided process is explicitly set forth in Limelight's standard customer contract, which states:
>
> Customer [i.e., content provider] shall be responsible for identifying via the then current [Limelight] process all [URLs] of the Customer Content to enable such Customer Content to be delivered by [Limelight]
>
> and
>
> Customer shall provide [Limelight] with all cooperation and information necessary for [Limelight] to implement the [Content Delivery Service].

Akamai, 629 F.3d at 1317. The contract further contemplates that the Content Provider must deliver (i.e., "serve") the web pages containing the tags when requested by the user. (A441-42:37-38.) Otherwise, Limelight's network will not "see" the user's request for content and will be unable to meet the contract's

service guarantee. (*Id.*) In this regard, the contract states:

> Service Interruptions caused by . . . failure of [Content Provider] origin server (equipment down, ***not serving content*** [*e.g.* pages], broken links or similar issues that would prevent the [Limelight] Service from working successfully, . . .) are ineligible for [Limelight's] availability guarantee compensation.

(A17807 (emphasis added).)

3. The Jury Verdict of Infringement Under the "Direction or Control" Test

In the district court, because some of the claim steps were performed by Limelight and some by its customers, Limelight argued it was not liable for direct infringement. After a three-week trial, the jury returned a verdict of infringement and awarded over $40 million in lost-profit damages. (A93-99.) The jury was properly instructed on the *BMC Resources* "direction or control" standard and heard evidence that Limelight: (1) creates, assigns, and provides the Content Provider a unique Limelight hostname or "tag"; (2) provides explicit step-by-step instructions to perform the tagging step; (3) offers technical assistance to help Content Providers with their performance of the claim step; and (4) contractually requires Content Providers to provide "cooperation and information" if they use Limelight's service. (A441-42: 37-38, A17807-08.)

Following the verdict, the district court (Judge Zobel) initially denied Limelight's JMOL motion, finding that "unlike in *BMC Resources*, here there was evidence that not only was there a contractual relationship between Limelight and

its customers, but that [Limelight] provided those customers with instructions explaining how to utilize its content delivery service." *Akamai,* 629 F.3d at 1318. Subsequently however, on Limelight's motion for reconsideration, the district court analogized the facts before it to those at issue in *Muniauction*, reversed its previous decision, and granted JMOL of noninfringement. *Id.*

4. The Jury Properly Found Liability Under *BMC Resources's* Flexible "Direction or Control" Test

Under a flexible fact-based *BMC Resources* direction or control test, Limelight should be liable for infringing the method claims of the '703 patent, as the jury properly found. The question of direction or control is a factual one, the jury was properly instructed based on the *BMC Resources* "direction or control" standard, *see Akamai*, 629 F. 3d at 1317-18 & n.2, and substantial evidence supports the jury verdict. (*See* A396:35-36, A587:122-23, A17220, A17231, A17789-95, A17803, A17807-08 .)

The Panel's determination that the agreement between Limelight and its customers "calls for its customers to assign a unique hostname, requires content providers to perform certain claim steps if they choose to use Limelight's service, and provides instructions and technical assistance [by Limelight] for performing those steps," *Akamai*, 629 F.3d at 1321, is more than sufficient, as found by the jury, to establish joint liability under *BMC Resources. See Akamai,* 629 F.3d at 1317-18. Indeed, as discussed above, according to Limelight's contract and

detailed instructions provided to its customers, a customer who desires to have its particular content served by Limelight according to Limelight's service guarantee, is required by Limelight's process to tag (claims 19-21 and 34) and serve the page with the tag (claims 19-21) if the customer wishes to benefit from Limelight's service.

While the *Akamai* Panel emphasized that the contract does not "obligate" customers to perform the claim steps of tagging and serving in the sense of creating an agency relationship or a claim for breach of contract if those steps are not performed, the panel recognized that the contract "calls for [Limelight's] customers to assign a unique hostname, requires content providers to perform certain claim steps if they choose to use Limelight's service, and provides instructions and technical assistance [by Limelight] for performing those steps." *Akamai*, 629 F.3d at 1321. Moreover, it is only when a customer "choose[s] to use Limelight's service" that the steps of the accused method are performed and, thus, this is where the focus of the "direction or control" test should be, not on whether Limelight directs or controls its customers in some other abstract sense. In other words, the Court should focus on whether Limelight directs or controls the performance of the accused process. It is irrelevant that Limelight does not direct or control its customers in other senses. As this Court stated in both *BMC Resources* and *Muniaction*, a party is liable for joint infringement when it

"exercises 'control or direction' ***over the entire process*** such that every step is attributable to the controlling party." *Muniauction*, 532 F.3d at 1329 (emphasis added); *BMC Resources*, 498 F.3d at 1380 ("control or direct each step of the patented process"). The focus of the direction or control test should be on the claimed process, not whether there is a formal agency relationship between the parties.

Accordingly, the jury verdict of joint infringement should be upheld in this case even under the "direction or control" standard, and in fact, under any reasonable test that does not require a formal agency relationship.

5. Limelight's Activities in Concert with Its Customers Subject Limelight to Liability for Joint Infringement

In addition, under the "acting in concert" test, Limelight and its customers acted in accordance with an express agreement to perform the steps that constitute the asserted method claims of the '703 patent. As noted by the *Akamai* Panel, Limelight's contract with its customers details the specific claim steps that are to be performed and the overall result of their "divided process." *Akamai*, 629 F.3d at 1317.

Thus, there exists an agreement between the parties in which each party is aware of the other and both are deliberately engaged in a common plan or design, demonstrating that they are acting in concert. Further, their concerted actions result in the performance of the steps of the claimed method. Under these facts,

Limelight and its customers are acting in concert, and, accordingly, this Court should impose liability for joint infringement. The joint activity performed by Limelight and its Content Provider customers is precisely what is described in the '703 patent and recited in several of the asserted claims. Enforcing the jury's verdict—one that was supported by substantial evidence in any event—would preserve the value of Akamai's pioneering invention.

6. The Contractual Relationship Between Limelight and Content Providers Makes Limelight Liable for Direct Infringement

While "acting in concert" liability does not require a strict contractual obligation to perform the steps that constitute a method claim, such an obligation exists in this case and is strong evidence that Limelight and its customers were acting in concert. The contractual obligation is also strong evidence (in addition to the separate arguments presented above) that Limelight exercised "direction or control."

As noted above, Akamai recognizes that the Panel discounted the presence of Limelight's contract because, according to the Panel, the contract did not "obligate" Limelight's customers (the Content Providers) to perform the tagging and/or serving steps. *Akamai*, 629 F.3d at 1321. With due deference to the Panel, Akamai submits that the Panel misapprehended the significance of the contract involved in this case. More specifically, it is true that for customers who do not

choose to use the patented process, there is no contractual obligation. But for those customers who do choose to use Limelight's service (Akamai's patented process)—which are the only entities relevant to this appeal—the contract includes a binding obligation requiring the Content Providers to tag if Limelight's service guarantees are to be enforced. Indeed, the Panel ignored the fact that by entering into a contractual agreement, the Content Providers almost certainly intended to use Limelight's service.

From the outset, Limelight specified that, in order to perform the claimed method, the Content Provider, and not Limelight, would "be responsible for identifying . . . all [URLs] of the Customer Content." *Akamai*, 629 F.3d at 1317; (*cf.* A276 col.19 ll.12-14 ("tagging the embedded objects of the page so that requests for the page objects resolve to the domain instead of the Content Provider domain")). Moreover, Limelight allocates to the Content Provider the responsibility for serving the page. The Content Provider's consent to these responsibilities is evidenced by its execution of the contract and performance under its terms.

Respectfully, Akamai submits that the Panel's focus on the Content Providers' "independent discretion," *Akamai*, 629 F.3d at 1321, is misdirected because, under the terms of Limelight's contract, the Content Providers who use Limelight's service do not have discretion whether or not to tag (as was the case

for the bidders in *Muniauction*); they have discretion only with respect to which content they will tag. In this manner, Limelight does not simply direct a particular result which may or may not satisfy the relevant claim limitations, but instead contractually imposes on the Content Providers who use Limelight's service the obligation to perform particular steps of the claimed method (tagging and/or serving) in such a way that infringement will result.

Further, that the Content Provider has some discretion with respect to which content it wants to tag is irrelevant because Akamai's patent claims are not limited by which types of content are tagged and/or served. Rather, the claim elements are satisfied by the acts of tagging and/or serving themselves. And Limelight, through the terms of its contract, has explicitly directed those who will perform the steps (the Content Provider who uses Limelight's service) and what these steps will be. *Id.* This obligation is not only highly relevant to the acting in concert issue, but also fits squarely within the admonition in *BMC Resources* that "[a] party cannot avoid infringement, however, by contracting out steps of a patented process to another entity." *BMC Resources*, 498 F.3d at 1381.

7. Limelight Knew of the Customer's Conduct and Is Thus Liable

Finally, even if the customers do have "independent discretion" to tag only the content they wish to divert to Limelight's servers, this should not absolve Limelight of any liability. It cannot be disputed that Limelight knows that its

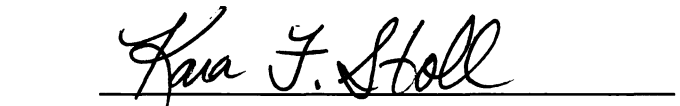
conduct will be combined with that of its customer to perform the method that constitutes the claimed invention. Limelight knows the steps that it takes and clearly knows that its customer is performing the tagging and serving steps because (1) the contract says that the customer will perform these steps; and (2) more importantly, it is only in this context that Limelight's process is used. Indeed, in many cases, Limelight only gets paid for content that is tagged by the customer—i.e., when Limelight's infringing process is actually used. Under this common law doctrine, it is irrelevant whether the customer knows the steps performed by Limelight, and in such a scenario, the customer is not liable. In this manner, this case is similar to *Peerless*, 93 F.2d at 98, and this Court should accordingly reinstate the jury verdict of joint infringement.

V. CONCLUSION

For all the reasons noted above, this Court should apply common law principles of torts and reinstate the jury verdict of infringement in this case. Should this Court adopt a new standard for the determination of joint infringement under either 35 U.S.C. §§ 271(a) or (b), Akamai respectfully submits that, at a minimum, this Court should remand for a new trial based on that standard.

Respectfully submitted,

Date: June 20, 2011



Donald R. Dunner
Kara F. Stoll
Finnegan, Henderson,
Farabow, Garrett & Dunner, LLP
901 New York Avenue, NW
Washington, DC 20001
Telephone: (202) 408-4000

Jennifer S. Swan
Finnegan, Henderson,
Farabow, Garrett & Dunner, LLP
3300 Hillview Ave.
Palo Alto, CA 94304

Attorneys for Plaintiff-Appellant
Akamai Technologies, Inc.

ADDENDUM

629 F.3d 1311, 97 U.S.P.Q.2d 1321
(Cite as: 629 F.3d 1311)

United States Court of Appeals,
Federal Circuit.
AKAMAI TECHNOLOGIES, INC., Plaintiff–Appellant,
and
The Massachusetts Institute of Technology, Plaintiff–Appellant,
v.
LIMELIGHT NETWORKS, INC., Defendant–Cross–Appellant.

Nos. 2009–1372, 2009–1380, 2009–1416, 2009–1417.
Dec. 20, 2010.
Rehearing Denied April 20, 2011.

Background: Patent holders brought action against competitor alleging infringement of patents for Internet server architecture and related software. The United States District Court for the District of Massachusetts, Rya W. Zobel, J., 494 F.Supp.2d 34, construed the claims and granted judgment as matter of law for competitor after jury verdict for plaintiffs, 614 F.Supp.2d 90. Parties appealed.

Holdings: The Court of Appeals, Linn, Circuit Judge, held that:
(1) customers of patentee's competitor did not have agency relationship with competitor;
(2) phrase, "given object of a participating content provider is associated with an alphanumeric string," limited tagged alphanumeric strings to those strings including object's original Uniform Resource Locator (URL); and
(3) phrase, "given name server that receives the DNS query being close to the client local name server as determined by given location information," and phrase, "selecting a given one of the name servers in the content delivery network," required selection of name server by alternative domain name system.

Affirmed.

West Headnotes

[1] Patents 291 259(1)

291 Patents
291XII Infringement
291XII(A) What Constitutes Infringement
291k259 Contributory Infringement; Inducement
291k259(1) k. In general. Most Cited Cases

An accused infringer's control over its customers' access to an online system, coupled with instructions on how to use that system, is not enough to establish direct patent infringement.

[2] Patents 291 259(1)

291 Patents
291XII Infringement
291XII(A) What Constitutes Infringement
291k259 Contributory Infringement; Inducement
291k259(1) k. In general. Most Cited Cases

Customers of patentee's competitor did not have agency relationship with competitor, as required for infringement of patent on Internet server architecture and related software under joint patent infringement theory, on basis that competitor had provided instructions on use of its service and required customers to perform some steps of claimed method to take advantage of that service; customers decided what content, if any, they would like delivered by competitor's content delivery network and then performed step of "tagging" that content and they also performed step of "serving" their own web pages, but agency relationship did not arise when one party simply provided direction, no matter how explicit, to another party. Restatement (Third) of Agency § 1.01.

[3] Patents 291 228.1

291 Patents
291XII Infringement
291XII(A) What Constitutes Infringement
291k228 Patents for Processes

291k228.1 k. In general. Most Cited Cases

Direct patent infringement requires a single party to perform every step of a claimed method.

[4] Patents 291 259(1)

291 Patents
291XII Infringement
291XII(A) What Constitutes Infringement
291k259 Contributory Infringement; Inducement
291k259(1) k. In general. Most Cited Cases

The performance of a method step may be attributed to an accused patent infringer when the relationship between the accused infringer and another party performing a method step is that of principal and agent, applying generally accepted principles of the law of agency as explicated by the Supreme Court and the Restatement of Agency; while control or direction is a consideration, as is the extent to which instructions, if any, may be provided, what is essential is not merely the exercise of control or the providing of instructions, but whether the relationship between the parties is such that acts of one may be attributed to the other. Restatement (Third) of Agency § 1.01.

[5] Patents 291 259(1)

291 Patents
291XII Infringement
291XII(A) What Constitutes Infringement
291k259 Contributory Infringement; Inducement
291k259(1) k. In general. Most Cited Cases

Joint patent infringement through an agency relationship occurs when a party is contractually obligated to the accused infringer to perform a method step; both parties must consent that the agent is acting on the principal's behalf and subject to the principal's control. Restatement (Third) of Agency § 1.01.

[6] Patents 291 259(1)

291 Patents
291XII Infringement
291XII(A) What Constitutes Infringement
291k259 Contributory Infringement; Inducement
291k259(1) k. In general. Most Cited Cases

When assessing patent infringement based on the actions of joint parties, it is not enough to determine for whose benefit the actions serve, for in any relationship there may be benefits that inure in some respects to both parties; there can only be joint infringement when there is an agency relationship between the parties who perform the method steps or when one party is contractually obligated to the other to perform the steps. Restatement (Third) of Agency § 1.01.

[7] Patents 291 259(1)

291 Patents
291XII Infringement
291XII(A) What Constitutes Infringement
291k259 Contributory Infringement; Inducement
291k259(1) k. In general. Most Cited Cases

A party that engages another to perform a step of a claimed method as its agent cannot escape liability for patent infringement simply by designating its agent an independent contractor if all the elements that otherwise reflect an agency relationship are present. Restatement (Third) of Agency § 1.01.

[8] Patents 291 101(3)

291 Patents
291IV Applications and Proceedings Thereon
291k101 Claims
291k101(3) k. Limitations in general. Most Cited Cases

Phrase, "given object of a participating content provider is associated with an alphanumeric string," in patent for Internet server architecture and related software, limited tagged alphanumeric strings to those strings including object's original Uniform Resource Locator (URL); alphanumeric strings including object's original URL were not merely discussed as

preferred embodiment, but, instead, written description specifically referred to strings including object's original URL as "the invention."

[9] Patents 291 324.5

291 Patents
291XII Infringement
291XII(B) Actions
291k324 Appeal
291k324.5 k. Scope and extent of review in general. Most Cited Cases

Patent claim construction is reviewed de novo.

[10] Patents 291 165(3)

291 Patents
291IX Construction and Operation of Letters Patent
291IX(B) Limitation of Claims
291k165 Operation and Effect of Claims in General
291k165(3) k. Construction of language of claims in general. Most Cited Cases

Patents 291 167(1)

291 Patents
291IX Construction and Operation of Letters Patent
291IX(B) Limitation of Claims
291k167 Specifications, Drawings, and Models
291k167(1) k. In general. Most Cited Cases

Patent claim construction analysis begins by considering the language of the claims themselves; however, the written description can provide guidance as to the meaning of the claims, thereby dictating the manner in which the claims are to be construed, even if the guidance is not provided in explicit definitional format.

[11] Patents 291 101(2)

291 Patents
291IV Applications and Proceedings Thereon
291k101 Claims
291k101(2) k. Construction in general. Most Cited Cases

Phrase, "given name server that receives the [domain name system (DNS)] query being close to the client local name server as determined by given location information," and phrase, "selecting a given one of the name servers in the content delivery network," in patents describing framework including set of "hosting" or "ghost" servers used to store and deliver Internet website's embedded objects, required selection of name server by alternative DNS.

[12] Patents 291 101(11)

291 Patents
291IV Applications and Proceedings Thereon
291k101 Claims
291k101(11) k. Process or method claims. Most Cited Cases

Structural element of alternative domain name system (DNS) framework explicitly and properly had been included in claims, in patent describing framework including set of "hosting" or "ghost" servers used to store and deliver Internet website's embedded objects, where all of asserted claims had explicitly referred to alternative DNS as detail associated with claimed method.

Patents 291 328(2)

291 Patents
291XIII Decisions on the Validity, Construction, and Infringement of Particular Patents
291k328 Patents Enumerated
291k328(2) k. Original utility. Most Cited Cases

6,108,703. Not Infringed.

Patents 291 328(2)

291 Patents
291XIII Decisions on the Validity, Construction, and Infringement of Particular Patents
291k328 Patents Enumerated
291k328(2) k. Original utility. Most Cited Cases

6,553,413, 7,103,645. Construed.

***1313** Donald R. Dunner, Finnegan, Henderson, Farabow, Garrett & Dunner, ***1314** LLP, of Washington, DC, argued for all plaintiffs-appellants. With him on the brief were Kara F. Stoll and Elizabeth D. Ferrill. Of counsel on the brief was Robert S. Frank, Jr., Choate, Hall & Stewart LLP, of Boston, MA, for The Massachusetts Institute of Technology. Of counsel were G. Mark Edgarton and Carlos Perez–Albuerne.

Alexander F. Mackinnon, Kirkland & Ellis LLP, of Los Angeles, CA, argued for defendant-cross appellant. With him on the brief were Robert G. Krupka, and Nick G. Saros. Of counsel on the brief was Dion Messer, Limelight Networks, Inc., of Tempe, AZ.

Before RADER, Chief Judge, LINN and PROST, Circuit Judges.

LINN, Circuit Judge.

Akamai Technologies, Inc. and the Massachusetts Institute of Technology (collectively, "Akamai") appeal the district court's judgment as a matter of law ("JMOL") overturning a jury verdict of infringement by Limelight Networks, Inc. ("Limelight") of claims 19–21 and 34 of U.S. Patent No. 6,108,703 (the "'703 patent"). *See Akamai Techs., Inc. v. Limelight Networks, Inc.*, 614 F.Supp.2d 90 (D.Mass.2009) (" *JMOL Opinion* "). Akamai also appeals the district court's construction of claim 1 of U.S. Patent No. 7,103,645 (the "'645 patent") and claims 8, 18, and 20 of U.S. Patent No. 6,553,413 (the "'413 patent"). Limelight cross appeals the district court's denial of JMOL relating to the jury's award of lost profits. *See Akamai Techs., Inc. v. Limelight Networks, Inc.*, Nos.2009–1372, –1380, –1416, –1417, 2010 WL 331770 (Fed.Cir. Jan.27, 2010) (finding Limelight's cross appeal in this case proper as to the lost profits determination).

Because Limelight did not perform all of the steps of the asserted method claims, and the record contains no basis on which to attribute to Limelight the actions of its customers who carried out the other steps, this court affirms the finding of noninfringement and does not reach Limelight's cross-appeal regarding damages. This court also affirms the district court's judgment of noninfringement of the '645 and '413 patents based on its rulings on claim construction.

BACKGROUND

I. The Technology and the Nature of the Dispute

Information is typically delivered over the Internet from websites. Websites are collections of documents written using a standard page description language known as Hypertext Markup Language ("HTML"). Each web page is a separate HTML file with an identifying string of characters known as a Uniform Resource Locator ("URL"). Typically, a full URL (e.g., "http:// www. cafc. uscourts. gov/ forms") consists of several elements: a protocol (e.g., "http://"); a domain name (also referred to herein as a "hostname") (e.g., "www. cafc. uscourts. gov"); and sometimes a path (e.g., "/forms"). A typical web page consists of a base HTML document that includes text interspersed with various types of content such as images, video, and sound—referred to as objects. Most of these objects are not incorporated into the web page in their entirety, but instead are simply included as links, in the form of separate URLs, which reference the actual object stored elsewhere on the same computer or another computer in the same domain (a group of networked computers that share a common domain name). These objects are referred to in the patents***1315** as "embedded objects." An embedded object's URL is typically the same as that of the web page containing the embedded object, with the object's name appended thereto (e.g., "http:// www. cafc. uscourts. gov/ forms/ pic. jpg").

The Internet maintains a Domain Name System ("DNS"), which uses computers, known as domain name servers ("DNS servers"), to convert the hostname of a URL into a numeric Internet Protocol ("IP") address, which identifies one or more computers that store content ("content servers"). This conversion process is referred to as "resolving." A user requesting a web page using a web browser (e.g., Netscape Navigator® or Microsoft Internet Explorer®) will receive an IP address from a local DNS server that corresponds to the content server for the requested web page. In response, the user's computer sends a request for the web page directly to that content server using the IP address. The content server sends the requested web page—the base HTML document and any embedded objects' URLs—to the user's computer. The user's web browser then requests each embedded object from the content provider's server using that object's URL in the same manner that it requested the web page until all of the objects have been retrieved and the web page is fully displayed on the user's computer.

This process of retrieving web content can be slow and unreliable. For example, Internet congestion problems may occur when a single content server receives many simultaneous requests for the same web page—sometimes referred to as "flash crowds." In addition, users may experience poor content delivery performance when the user's computer is located far away from the content server it is accessing. One known solution to these content delivery problems is called mirroring, in which an entire website is replicated on multiple servers in different locations. Mirroring, however, has scalability problems, including costs required by the multiple hosting facilities, additional overhead associated with keeping mirror sites synchronized, and a ceiling on the number of website copies that may be maintained concurrently. '703 patent col.1 ll.34–61.[FN1] In response to these known problems with delivering content, Akamai sought to provide a scalable solution that could efficiently deliver large amounts of web content and handle flash crowds. Akamai obtained the three patents at issue, which all share the same specification and disclose a system for allowing a content provider to outsource the storage and delivery of discrete portions of its website content.

> FN1. Because the specifications of all three patents are substantially identical, we refer throughout to the specification as it appears in the '703 patent.

All three patents include method claims directed to a content delivery service that delivers the base document of a web site from a content provider's computer while individual embedded objects of the website are stored on an object-by-object basis on a Content Delivery Network ("CDN"). CDNs are systems of computers strategically placed at various geographical locations to maximize the efficient delivery of information over the Internet to users accessing the network. The embedded objects are stored on and served from the CDN's "hosting" or "ghost" servers. Instead of maintaining identical copies of the entire web site content at a single location or at multiple locations by mirroring as taught by the prior art, only embedded objects are replicated on and served from ***1316** a CDN. To allow users accessing a content provider's web page to receive embedded objects from a CDN, the URL of the embedded object must point to a CDN hosting or ghost server instead of to a computer within the content provider's domain. To this end, the specification of the patents describes modifying the embedded object's URL, "to condition the URL to be served by the global hosting servers." '703 patent col.6 ll.41–46. This process of modifying an embedded object's URL to link to an object on the CDN is referred to as "tagging."

Akamai and Limelight operate and compete in the market for CDN services. Limelight's accused service delivers content providers' embedded objects from its CDN. According to Limelight's contracts with its content provider customers, to use Limelight's CDN service, the content provider must perform several steps. First, the content provider must choose which embedded objects, if any, it would like to be served from Limelight's CDN. The content provider must then tag the URL of each chosen object as instructed by Limelight. Limelight then replicates the properly tagged objects on some or all of its servers and directs a user's request for one of these objects to an appropriate Limelight server.

II. Proceedings Before the District Court

On June 23, 2006, Akamai sued Limelight in the United States District Court for the District of Massachusetts asserting infringement of the '645, ' 703, and '413 patents. After a trial on infringement of independent claims 19 and 34 and dependent claims 20–21 of the '703 patent, a jury returned a verdict of infringement and awarded $40.1 million in lost profits and $1.4 million in reasonable royalty damages. The two independent claims asserted at trial cover methods that require tagging at least some embedded objects in a content provider's web page so that requests for those objects resolve to a domain name other than the content provider's domain name. Claim 19 also requires serving the requested web page from the content provider's domain. Claims 19 and 34 read as follows, with steps at the heart of this dispute emphasized:

19. A content delivery service, comprising:

replicating a set of page objects across a wide area network of content servers managed by a domain other than a content provider domain;

for a given page normally served from the content provider domain, tagging the embedded objects of the page so that requests for the page objects re-

solve to the domain instead of the content provider domain;

responsive to a request for the given page received at the content provider domain, serving the given page from the content provider domain; and

serving at least one embedded object of the given page from a given content server in the domain instead of from the content provider domain.

'703 patent col.19 ll.6–20.

34. A content delivery method, comprising:

distributing a set of page objects across a network of content servers managed by a domain other than a content provider domain, wherein the network of content servers are organized into a set of regions;

*for a given page normally served from the content provider domain, tagging***1317** at least some of the embedded objects of the page so that requests for the objects resolve to the domain instead of the content provider domain;*

in response to a client request for an embedded object of the page:

resolving the client request as a function of a location of the client machine making the request and current Internet traffic conditions to identify a given region; and

returning to the client an IP address of a given one of the content servers within the given region that is likely to host the embedded object and that is not overloaded.

'703 patent col.20 ll.32–52.

It is undisputed that Limelight does not itself perform every step of the asserted claims. *JMOL Opinion* at 116. Limelight provides the information necessary for its customers, the content providers, to modify their web pages or Internet address routing information to use the Limelight service. However, the content providers perform the actual tagging step (emphasized above) themselves. There are two tagging methods used by Limelight's customers. As described by the district court:

> In the first method, the customer changes the hostname address of one or more page objects in the initial web page to point to Limelight's servers (the "prepend method"). In the second method, the customer adds or changes alias information in its DNS record so that the hostname addresses of the page objects resolve to Limelight's servers without requiring any change to the customer's initial web page (the "CNAME method").

JMOL Opinion at 117 n. 23. The content provider also serves the web page from its own domain. Limelight performs the rest of the steps of the asserted claims. This divided process is explicitly set forth in Limelight's standard customer contract, which states:

> Customer [i.e., content provider] shall be responsible for identifying via the then current [Limelight] process all [URLs] of the Customer Content to enable such Customer Content to be delivered by [Limelight]
>
> and
>
> Customer shall provide [Limelight] with all cooperation and information necessary for [Limelight] to implement the [Content Delivery Service].

J.A. 17807.

Because Limelight itself does not perform all the steps of the asserted claims, Akamai presented a theory of joint liability at trial. Akamai relied on the reasoning expressed by this court in *BMC Resources* that while "[i]nfringement requires, as it always has, a showing that a defendant has practiced each and every element of the claimed invention," joint liability may be found when one party "control[s] or direct[s]" the activities of another party. *BMC Res., Inc. v. Paymentech, L.P.*, 498 F.3d 1373, 1380 (Fed.Cir.2007). The district court, following *BMC Resources*, instructed the jury that Limelight could only be found to infringe if "the content provider, when [tagging objects], acts under the direction and control [FN2] of Limelight such that Limelight can properly be deemed to ***1318** be the one to do it." *JMOL Opinion* at 118. The district court added that the jury "should review the evidence, decide how the Limelight systems work, how does the interaction with the content provider work, and, specifically, does Limelight direct and control the modifications or does the content provider

carry out these tasks entirely independently." *Id.*

> FN2. The district court initially instructed the jury that Limelight must both direct *and* control the actions of the Content Provider, but then issued a correcting instruction that "[i]t is either direct or control, control or direct; it doesn't have to be both." *JMOL Opinion* at 118 n. 26.

[1] Following the verdict finding infringement, Limelight moved for JMOL of noninfringement on the ground that substantial evidence did not support the verdict that Limelight directs or controls all the steps in the asserted claims. Initially, the district court denied the motion "because, unlike in *BMC Resources*, here there was evidence that not only was there a contractual relationship between Limelight and its customers, but that it provided those customers with instructions explaining how to utilize its content delivery service." *JMOL Opinion* at 119. Subsequently, this court issued its decision in *Muniauction, Inc. v. Thomson Corp.*, 532 F.3d 1318 (Fed.Cir.2008), and Limelight moved for reconsideration. *Muniauction*, applying *BMC Resources*, held that an accused infringer's control over its customers' access to an online system, coupled with instructions on how to use that system, was not enough to establish direct infringement. *Id.* at 1328–30. On reconsideration, the district court granted JMOL of noninfringement to Limelight holding that there was "no material difference between Limelight's interaction with its customers and that of Thomson in *Muniauction*." *JMOL Opinion* at 122.

Akamai appeals and this court has jurisdiction under 28 U.S.C. § 1295(a)(1).

DISCUSSION

I. Joint Infringement of the '703 Patent [FN3]

> FN3. Because Akamai waived any assertion of indirect infringement before trial, the question before us is one of direct infringement only. Feb. 26, 2008 Trial Tr. at 46:4–22.

[2] On appeal, Akamai asserts that we should reverse the district court's JMOL of noninfringement of the '703 patent because substantial evidence supports the jury's determination that Limelight exercises control or direction over the entire claimed process. Akamai attempts to distinguish *Muniauction*, arguing that Limelight: (1) creates and assigns a unique hostname for the content provider; (2) provides explicit step-by-step instructions to perform the tagging and serving claim steps; (3) offers technical assistance to help content providers with their performance of the claim steps; and (4) contractually requires content providers to perform the tagging and serving claim steps if they utilize the Limelight service. Limelight responds that Akamai's evidence is indistinguishable from that found legally insufficient in *Muniauction* and therefore we should affirm.

[3] It is well settled that direct infringement requires a single party to perform every step of a claimed method. *BMC Resources*, 498 F.3d at 1378–79 (citing *Warner–Jenkinson Co., Inc. v. Hilton Davis Corp.*, 520 U.S. 17, 40, 117 S.Ct. 1040, 137 L.Ed.2d 146 (1997)). In both *BMC Resources* and *Muniauction* this court confronted the situation in which more than one party is required to perform the steps of a claimed method. The court concluded that there can be no infringement unless "one party exercises ***1319** 'control or direction' over the entire process such that every step is attributable to the controlling party." *Muniauction*, 532 F.3d at 1329 (citing *BMC Resources*, 498 F.3d at 1380–81). In assessing whether "control or direction" is present, the court in *BMC Resources* made reference to the legal principle that imposed "vicarious liability on a party for the acts of another in circumstances showing that the liable party controlled the conduct of the acting party." *BMC Resources*, 498 F.3d at 1379 (citing *Engle v. Dinehart*, 213 F.3d 639 (5th Cir.2000) (unpublished decision); Restatement (Second) of Agency § 220 cmt. d). The court concluded that "[it] would be unfair indeed for the mastermind in such situations to escape liability." *Id.* at 1381. Moreover, the court in *BMC Resources* also explained that "[a] party cannot avoid infringement ... simply by contracting out steps of a patented process to another entity." *Id.*

While the "control or direction" test of *BMC Resources* established a foundational basis on which to determine liability for direct infringement of method claims by joint parties, it left several questions unanswered, including the question of whether the furnishing of instructions is sufficient to attribute the actions of the instructed party to the accused. *Muniauction* addressed the question about instructions

and, in concluding that the instructions in that case were not enough, reiterated the notion of vicarious liability mentioned in *BMC Resources.* The court in *Muniauction* held that the requisite level of control or direction over the acts committed by a third party is met in circumstances in which "the law would traditionally hold the accused direct infringer vicariously liable for the acts committed by another party." 532 F.3d at 1330. Thus, both *BMC Resources* and *Muniauction* set forth relevant factors in assessing liability for joint infringement.

[4][5] While control or direction is a consideration, as is the extent to which instructions, if any, may be provided, what is essential is not merely the exercise of control or the providing of instructions, but whether the relationship between the parties is such that acts of one may be attributed to the other. Implicit in this court's holdings in *BMC Resources* and *Muniauction* is that the performance of a method step may be attributed to an accused infringer when the relationship between the accused infringer and another party performing a method step is that of principal and agent, applying generally accepted principles of the law of agency as explicated by the Supreme Court and the Restatement of Agency. The Restatement defines agency as "the fiduciary relationship that arises when one person (a 'principal') manifests assent to another person (an 'agent') that the agent shall act on the principal's behalf and subject to the principal's control, and the agent manifests assent or otherwise consents so to act." Restatement (Third) of Agency § 1.01. For an agency relationship to exist, and thus, for infringement to be found, both parties must consent that the agent is acting on the principal's behalf and subject to the principal's control. *See Dixson v. United States,* 465 U.S. 482, 505, 104 S.Ct. 1172, 79 L.Ed.2d 458 (1984) (citing the Restatement (Second) of Agency § 1 for the rule that an "agency relationship [is] created when one person agrees with another 'that the other shall act on his behalf and subject to his control' "). Similarly, also implicit in the court's holdings in *BMC Resources* and *Muniauction,* is that joint infringement occurs when a party is contractually obligated to the accused infringer to perform a method step.

***1320** [6] In assessing infringement based on the actions of joint parties, it is not enough to determine for whose benefit the actions serve, for in any relationship there may be benefits that inure in some respects to both parties. This court therefore holds as a matter of Federal Circuit law that there can only be joint infringement when there is an agency relationship between the parties who perform the method steps or when one party is contractually obligated to the other to perform the steps. Neither is present here.

[7] The court notes that the common law of agency encompasses not only the fiduciary relationship noted above, but also some other relationships, which may include those of independent contractors. *United States v. Hudson,* 491 F.3d 590, 595 (Fed.Cir.2007) ("As a matter of legal custom and tradition, ... nothing about the title independent contractor invariably precludes someone from being an agent under appropriate circumstances."); Restatement (Third) of Agency § 1.01 cmt. c ("The common law of agency ... additionally encompasses the employment relation.... [T]he common term 'independent contractor' is equivocal in meaning and confusing in usage because some termed independent contractors are agents while others are nonagent service providers.... This Restatement does not use the term 'independent' contractor."); Restatement (Second) of Agency § 2(3) ("An independent contractor ... may or may not be an agent."). This same principle applies to the question of joint infringement. A party that engages another to perform a step of a claimed method as its agent cannot escape liability simply by designating its agent an independent contractor if all the elements that otherwise reflect an agency relationship are present.

In this case, there is nothing to indicate that Limelight's customers are performing any of the claimed method steps as agents for Limelight. To the contrary, Limelight's CDN is a service similar to Thomson's on-line auction system in *Muniauction,* and Limelight's relationship with its customers is similar to Thomson's relationship with the bidders. In both cases, customers are provided instructions on use of the service and are required to perform some steps of the claimed method to take advantage of that service. In *Muniauction,* the customers performed the step of bidding. Here, the customers decide what content, if any, they would like delivered by Limelight's CDN and then perform the step of "tagging" that content. Limelight's customers also perform the step of "serving" their own web pages.

Akamai argues that in *Muniauction,* the direction or control provided by Thomson was "only tangen-

tially related to the claimed process" because it related to controlling access to the auction system, not directing users on what bid information to input. Akamai's Principal Br. at 44. According to Akamai, here the control or direction is directly related to the claimed step because Limelight tells providers not only how to tag, but also what hostname to use as a tag. Further, Akamai points out that by including the word "direct" in the "control or direct" test, this court in *BMC Resources* must have meant the word "direct" to mean something other than "control," and this case "presents the ultimate in direction" because of the detailed instructions and technical assistance provided to customers by Limelight. Akamai's Principal Br. at 42. However, the words in the *BMC Resources* test must be read in the context of traditional agency law. "An essential element of agency is the principal's right to control the agent's actions.***1321** Control is a concept that embraces a wide spectrum of meanings, but within any relationship of agency the principal initially states what the agent shall and shall not do, in specific or general terms." Restatement (Third) of Agency § 1.01 cmt. f. Like *BMC Resources*, the Restatement and the Supreme Court refer to the words "control" and "direction" when assessing whether an agency relationship exists, but there is no indication that an agency relationship arises when one party simply provides direction, no matter how explicit, to another party. All the elements of an agency relationship must be present. *See Meyer v. Holley*, 537 U.S. 280, 286, 123 S.Ct. 824, 154 L.Ed.2d 753 (2003) ("The Restatement [ ] specifies that the relevant principal/agency relationship demands not only control (or the right to direct or control) but also 'the manifestation of consent by one person to another that the other shall act *on his behalf*, and consent by the other so to act.' ").

Akamai also argues that the relationship between Limelight and its customers compels a finding of joint infringement because Limelight "contracts out to content providers the claim steps that it alone does not perform." This conclusion stems from Limelight's standard form contract that, according to Akamai, "obligates content providers to perform the claim steps of tagging the embedded objects and serving the tagged page so that requests for the embedded objects resolve to Limelight's network instead of the content provider's." Akamai's Principal Br. at 40. For this argument, Akamai relies on the statement in *BMC Resources* that "[a] party cannot avoid infringement ... simply by contracting out steps of a patented process to another entity." *BMC Resources*, 498 F.3d at 1381. Akamai's reliance on this statement is misplaced.

As discussed above, Limelight's customers decide what content, if any, they choose to have delivered by Limelight's CDN and only then perform the "tagging" and "serving" steps. The form contract does not *obligate* Limelight's customers to perform any of the method steps. It merely explains that the customer will have to perform the steps *if* it decides to take advantage of Limelight's service. *See Muniauction*, 532 F.3d at 1329 ("[M]ere 'arms-length cooperation' will not give rise to direct infringement by any party."). What is critical here is whether the evidence shows that the relationship between Limelight and its customers is such that the steps in question are performed by the customers as agents of Limelight or under a contractual obligation and are, thus, properly attributable to Limelight. It is true that Limelight's agreement calls for its customers to assign a unique hostname, requires content providers to perform certain claim steps if they choose to use Limelight's service, and provides instructions and offers technical assistance for performing those steps. However, none of those points establishes either Limelight's control over its customers or its customers' consent to Limelight's control. To the contrary, the agreement merely provides the customers with the tools to allow them to exercise their independent discretion and control over how and in what respect they implement the system. Limelight's customers did not perform the actions of tagging and serving as Limelight's agents and were not contractually obligated to perform those actions. Instead, the evidence leaves no question that Limelight's customers acted principally for their own benefit and under their own control.

While acknowledging the difficulty of proving infringement of claims that must be infringed by multiple parties, this court ***1322** has noted that such concerns "can usually be offset by proper claim drafting. A patentee can usually structure a claim to capture infringement by a single party." *BMC Resources*, 498 F.3d at 1381. Akamai recognizes and, indeed, asserts that the other two patents at issue in this case (the '645 and '413 patents), which share the same specification, do not implicate this joint infringement issue because of the way the asserted claims were drafted. Oral Arg. 10:35–11:10, *available at* http:// oralarguments. cafc. uscourts. gov. This court also observes that in addition to initially struc-

turing a claim to capture infringement by a single party, patentees may be able to correct a claim that can only be infringed by multiple parties by seeking a reissue patent. *See* Mark A. Lemley et al., *Divided Infringement Claims*, 33 AIPLA Q.J. 255, 278–79 (2005).

Here, the asserted claims were drafted so as to require the activities of both Limelight and its customers for a finding of infringement. Thus, Akamai put itself in a position of having to show that the allegedly infringing activities of Limelight's customers were attributable to Limelight. Akamai did not meet this burden because it did not show that Limelight's customers were acting as agents of or were contractually obligated to Limelight when performing the tagging and serving steps. Thus, the district court properly granted JMOL of noninfringement to Limelight.

* * *

Limelight argues as an alternative ground for affirmance that Akamai presented no substantial evidence that Limelight or its customers actually performed the tagging limitation as properly construed. Because we find that the district court properly granted JMOL of noninfringement on the ground stated, we need not and do not address this argument. Likewise, we do not reach Limelight's conditional cross-appeal of the damages award alleging that Akamai failed to present economic proof of a causal link between Limelight's infringement and any Akamai lost sales.

II. Claim Construction of the '645 and '413 Patents

[8] After the district court's claim construction order, *Akamai Technologies, Inc. v. Limelight Networks, Inc.*, 494 F.Supp.2d 34, 39 (D.Mass.2007) (" *Claim Construction Order* "), Akamai stipulated that it could not prove infringement of the '645 patent under the district court's construction. The district court thus entered judgment of noninfringement. The district court subsequently entered summary judgment of noninfringement of claims 8, 18, and 20 of the '413 patent. Akamai appeals the district court's construction of several terms in the '645 and '413 patents. While Limelight does not concede that the '645 and '413 patents do not implicate a joint infringement issue similar to that found in the '703 patent above, both parties agree that even if such an issue does exist, it is not properly before the court in this appeal. Oral Arg. 10:35–11:10; 30:40–31:40 (Limelight's counsel stating that the joint infringement issues for the '645 and '413 patents were not developed at the trial court). Thus, we decide Akamai's appeal of the district court's construction of several terms in the asserted claims of the '645 and '413 patents independent of any potential joint infringement issues.

[9][10] We review claim construction de novo. *Cybor Corp. v. FAS Techs., Inc.*, 138 F.3d 1448, 1451 (Fed.Cir.1998) (en banc). "We begin a claim construction ***1323** analysis by considering the language of the claims themselves." *Edward Lifesciences LLC v. Cook Inc.*, 582 F.3d 1322, 1327 (Fed.Cir.2009). However, "the written description can provide guidance as to the meaning of the claims, thereby dictating the manner in which the claims are to be construed, even if the guidance is not provided in explicit definitional format." *SciMed Life Sys., Inc. v. Advanced Cardiovascular Sys., Inc.*, 242 F.3d 1337, 1344 (Fed.Cir.2001).

A. The Technical Setting

As part of a system for efficient content delivery, the '645 and '413 patents describe a framework including a set of "hosting" or "ghost" servers used to store and deliver a website's embedded objects. '703 patent col.3 ll.4–7. To determine the location of a hosting computer on which a particular object is stored, the framework includes a second set of servers that are configured with functionality that is similar to, but not exactly the same as, a typical Internet DNS server, such that the servers resolve URLs specifically for the CDN. The specification refers to this second set of servers as "top-level" DNS servers. *Id.* col.3 ll.17–21, 31. The specification also describes a third set of servers that provide "low-level DNS" functionality. *Id.* col.3 ll.22–24. Together, the top-level and low-level servers form an "alternative domain name system." According to the patents' preferred embodiment, when a user's machine requests a web page from a content provider, the web page base document is delivered to the user's computer from the content server in the traditional manner described above. *Id.* col.3 ll.24–27. Any embedded objects in that web page that are stored on the CDN's hosting servers, however, are located using the invention's framework. First, the top-level DNS server determines the user's location in the network and uses that information to identify a list of low-level DNS servers. *Id.* col.3

ll.29–33, 60–61. The top-level DNS server then redirects the request for the embedded object to one of the identified low-level DNS servers that, in turn, resolves the request into an IP address for the appropriate hosting server that delivers the object to the user's computer. *Id.* col.3 ll.33–37. The specification does not limit the framework to two levels of DNS servers, but describes "a hierarchy of DNS servers that consisting [sic] of several levels." *Id.* col.3 ll.37–41. In addition, the top-level and low-level DNS functionality may be combined into a single DNS level. *Id.* col.5 ll.54–57.

The specification also describes load balancing across the set of hosting servers. *Id.* col.3 ll.66–67. Load balancing is the process of equalizing the workload on multiple computers. The specification describes a load balancing technique based on distributing the embedded object requests. This technique can be included in the tagging process by modifying the embedded object URL using the hostname of a "virtual server." *Id.* col.4 ll.1–5. A virtual server is simply a reference to a hosting server whose physical location is not determined until a user attempts to access a specific object. This allows users to retrieve the objects stored on hosting servers efficiently based on a number of continually changing factors (e.g., network traffic, user location). Thus, upon retrieval of a modified web page by a user, the hosting framework maintained by the CDN will resolve the virtual server hostname in the modified URL into the IP address of the appropriate hosting server from which to retrieve the object.

Claim 1 of the '645 patent provides:

***1324** In a wide area network in which an Internet domain name system (DNS) is useable to resolve DNS queries directed to participating content provider content that is available from participating content provider sites, a method of content delivery wherein participating content providers identify content to be delivered by a service provider from a set of content servers that are distinct from the participating content provider sites and associated with the service provider, wherein *a given object of a participating content provider is associated with an alphanumeric string*, the method comprising:

having the service provider establish an alternative domain name system (DNS), distinct from the Internet domain name system and any client local name server, and having authority to *resolve the alphanumeric strings associated with the objects* identified by the participating content providers so that the objects identified by the participating content providers are available to be served from the service provider's content servers, the service provider's alternative domain name system having one or more DNS levels, wherein at least one DNS level comprises a set of one or more name servers;

for each of one or more participating content providers, delivering a given object on behalf of the participating content provider, wherein the given object is delivered by the following steps;

responsive to a DNS query to the given object's associated alphanumeric string, the DNS query originating from a client local name server, receiving the DNS query at a given name server of a lowest level of the one or more DNS levels in the service provider's alternative domain name system, *the given name server that receives the DNS query being close to the client local name server as determined by given location information;*

having the given name server that receives the DNS query resolve the alphanumeric string into an IP address that the given name server then returns to the client local name server, wherein *the alphanumeric string is resolved without reference to a filename for the given object*, wherein the IP address returned as a result of the resolution is associated with a content server within a given subset of the set of content servers, the subset of the set of content being associated with the given name server, the content server associated with the IP address returned by the given name server being selected according to a load sharing algorithm enforced across the subset of the set of content servers associated with the given name server;

at the content server associated with the IP address, receiving a request for the given object, the request having the filename associated therewith;

if the given object is available for delivery from the content server associated with the IP address, serving the given object from the content server.

'645 patent col.17 ll.39–col.18 ll.29 (emphases

added).

Claim 8 is representative of the asserted claims of the '413 patent. It provides:

> A method of content delivery wherein participating content providers identify content to be delivered by a content delivery network service provider from a set of content servers associated with the content delivery network service ***1325** provider, wherein a given object of a participating content provider is associated with a [URL] that includes, *in addition to a filename, an alphanumeric string,* comprising:
>
> having the content delivery network service provider establish a domain name system (DNS) having authority to resolve the alphanumeric strings in the URLs of the objects identified by the participating content providers, the content delivery network server provider's *domain name system having one or more DNS levels,* wherein at least one DNS level comprises a set of one or more name servers;
>
> for each of one or more participating content providers, delivering a given object on behalf of the participating content provider, wherein the given object is delivered by the following steps;
>
> *responsive to a DNS query, selecting a given one of the name servers in the content delivery network service provider's domain name system;*
>
> at the given one of the name servers, resolving the alphanumeric string to an IP address, wherein the alphanumeric string is resolved without reference to the filename for the given object;
>
> at a server associated with the IP address, the server being one of the set of content servers, receiving a request for the given object, the request having the filename associated therewith;
>
> from the server, serving the given object; and
>
> caching the given object at the server so that the given object is available for delivery from the server for a given time period in the event that a new DNS query to resolve the alphanumeric string is received at the domain name system and is resolved to the IP address of the server.
>
> '413 patent col.18 ll.14–51 (emphases added).

B. Associated with an Alphanumeric String

Akamai appeals the construction of the term "a given object of a participating content provider is associated with an alphanumeric string" in the preamble of claim 1 of the '645 patent.[FN4] The district court construed the limitation to require that the alphanumeric string include the embedded object's original URL (the URL including the hostname of the computer on which the actual object resided within the content provider's domain). *Claim Construction Order* at 39. The court reasoned that the written description portion of the '645 patent "describes *the invention* as associating a particular object of a content provider with an alphanumeric string consisting of a virtual server hostname prepended onto the URL for the object." *Id.* at 40. The court found that "[t]he specification discloses no other way that an object is associated with an alphanumeric string, nor is there any suggestion or teaching that an association which did not include the URL for the embedded object could be used in an embodiment of the invention." *Id.* The district court declined as overly broad Akamai's proposed construction of the term "associated" according to its dictionary definition of "brought into some kind of relationship with."

> FN4. Neither party contends that the term in question is not a limitation because it is part of the preamble.

***1326** Akamai contends that the court imported a limitation from the specification into the claims and thereby improperly limited the scope of the claims to the specification's preferred embodiment. According to Akamai, nothing in the claim language supports requiring that the alphanumeric string include the original URL. Akamai relies on the parties' stipulation that "alphanumeric string" is "a character string up to 24 characters drawn from the alphabet (a-z), digits (0–9), minus signs (-), and periods(.)." Stipulated Order Establishing the Constructions for Certain Claim Terms as Agreed Upon by the Parties at 3, *Akamai Techs., Inc. v. Limelight Networks, Inc.*, No. 06–CV–11109 (D.Mass. Apr. 24, 2007). Akamai asserts that the specification and prosecution history do not define "associated" as having any meaning other than its ordinary meaning. Thus, Akamai argues that the ordinary meaning of the words in the claim

compel a broad interpretation without the limitation introduced by the district court. Akamai also argues that the specification very clearly indicates that including the object's original URL in the alphanumeric string is merely the preferred method. Akamai contends that one of ordinary skill in the art would understand that other tagging methods may be used to associate an alphanumeric string with the object.

In addition, Akamai points to the prosecution history, other claim limitations in the '645 patent, and the use of "alphanumeric string" in claim 18 of the '413 patent as evidence contradicting the district court's construction. Akamai notes that (1) none of the examples of alphanumeric strings cited by Akamai during prosecution included the original URL; (2) other claim limitations of the '645 patent use the term "alphanumeric string" as a virtual server hostname, not a URL; and (3) the preamble of claim 18 of the '413 patent requires a URL to include an alphanumeric string, not the other way around.

Limelight responds that the district court correctly limited the claim term to include the object's original URL because it reflects the '645 patent's explicit description of the invention. Reiterating the points made by the district court, Limelight asserts that the patents consistently describe "the invention" as associating an alphanumeric string with an object by prepending a virtual server hostname to the original URL that identifies the object in the absence of the CDN. Limelight adds that including the original URL in the alphanumeric string is not merely a preferred embodiment in the patents because all the examples in the patents contain the object's original URL.

This court agrees with Limelight and the district court that the claim term "a given object of a participating content provider is associated with an alphanumeric string" limits tagged alphanumeric strings to those strings including the object's original URL. Here, as in *Honeywell International, Inc. v. ITT Industries, Inc.*, 452 F.3d 1312, 1318 (Fed.Cir.2006), alphanumeric strings including the object's original URL were not merely discussed as a preferred embodiment. Instead, the written description specifically refers to strings including the object's original URL as "the invention":

> According to the present invention, load balancing across the set of hosting servers is achieved in part through a novel technique for distributing the embedded object requests. In particular, each embedded object URL is preferably modified by prepending a virtual server hostname into the URL. More generally, the virtual server hostname is inserted into the URL.
>
> '645 patent col. 4 ll.13–19.
>
> According to the invention, the embedded object URL is first modified, preferably***1327** in an off-line process, to condition the URL to be served by the global hosting servers.
>
> *Id.* col.6 ll.54–57.
>
> Thus, according to the present invention, a virtual server hostname is prepended into the URL for a given embedded object....
>
> *Id.* col.7 ll.36–38.
>
> With the above as background, the inventive global hosting framework is now described in the context of a specific example.... Instead of returning the usual page, according to the invention, the Web site returns a page with embedded object URLs that are modified according to the method illustrated in the flowchart of FIG. 4.
>
> *Id.* col.7 l.49–col.8 l.2
>
> If, however, no copy of the data on the ghost exists, a copy is retrieved from the original server or another ghost server. Note that the ghost knows who the original server was because the name was encoded into the URL that was passed to the ghost from the browser.
>
> *Id.* col.12 ll.54–60.

The specification does include language indicating that the patentee intended certain aspects of the description to represent preferred, rather than required, elements of the invention. *See, e.g.*, '645 patent col.4 ll.15–17 ("[E]ach embedded object URL is preferably modified by prepending a virtual server hostname into the URL."); *id.* col.6 ll.57–58 ("A flowchart illustrating the preferred method for modifying the object URL is illustrated in FIG. 4."). This court also acknowledges that much of the language describing a string including a URL as "the invention" occurs within the section entitled "Detailed Description of the Preferred Embodiment" or in the description of Figure 4, which is referred to as a "preferred method for

modifying the object URL." However, the specification as a whole makes clear that including the object's original URL is the only method to achieve the claimed association between an alphanumeric string and the embedded object. Indeed, it is the only method described. *Netword, LLC v. Centraal Corp.*, 242 F.3d 1347, 1352 (Fed.Cir.2001) ("Although the specification need not present every embodiment or permutation of the invention and the claims are not limited to the preferred embodiment of the invention ... neither do the claims enlarge what is patented beyond what the inventor has described as the invention.") (internal citations omitted). *See also Bell Atl. Network Servs., Inc. v. Covad Commc'ns Grp., Inc.*, 262 F.3d 1258, 1271 (Fed.Cir.2001) ("[W]hen a patentee uses a claim term throughout the entire patent specification, in a manner consistent with only a single meaning, he has defined that term 'by implication.' ") (quoting *Vitronics Corp. v. Conceptronic, Inc.*, 90 F.3d 1576, 1582 (Fed.Cir.1996)). Moreover, the specification specifically limits the object's modified URL to either prepending or inserting a virtual server hostname into the URL. '645 patent col.4 ll.15–19 ("In particular, each embedded object URL is preferably modified by prepending a virtual server hostname into the URL. More generally, the virtual server hostname is inserted into the URL."). Both of these methods include the original URL of the object in the modified string. Finally, the specification describes the proper functioning of the invention as motivation for including the object's original URL in the modified string, "the ghost knows who the original server was because the name was encoded into the URL that was passed to the ghost from the browser." *Id.* col.12 ll.56–58.

***1328** This court is not persuaded by Akamai's argument that the patentee established a broader scope during prosecution or that other uses of the term "alphanumeric string" compel a broader interpretation. Akamai argues that during prosecution the patentee made it clear that an alphanumeric string can be comprised of just a hostname as opposed to requiring the inclusion of an entire URL. Akamai refers to the patentee's description of an examiner interview in a preliminary amendment. The remarks describe the interpretation of the phrase "alphanumeric string" and cite "numerous examples of such strings, such as ... 'a1234.g.akamaitech.net,' " in the written description. *Id.* col.7 ll.14–15. However, in the specification, the reference to the indicated hostname is in the context of determining a virtual server hostname for ultimate inclusion in the tagged string. The specification does not indicate that this virtual hostname can eventually be the entire string. Instead, the specification clearly describes that the hostname will be "prepended into the URL for the given embedded object" once the hostname is determined. *See, e.g., id.* col.6 ll.63–64. In fact, all the examples in the specification indicate that the ultimate tagged string contains the object's original URL. '645 patent col.8 ll.24–25; *id.* col.8 ll.56–57; *id.* col.9 ll.25–26. Even if we agreed with Akamai that the patentee indicated in the prosecution history that the alphanumeric string associated with an object could include only a hostname, this is not enough to overcome the clear description of the invention in the specification. *See Honeywell*, 452 F.3d at 1319 ("Where, as here, the written description clearly identifies what his invention is, an expression by a patentee during prosecution that he intends his claims to cover more than what his specification discloses is entitled to little weight."); *Biogen, Inc. v. Berlex Labs.*, 318 F.3d 1132, 1140 (Fed.Cir.2003) (stating that "[r]epresentations during prosecution cannot enlarge the content of the specification."). Akamai's arguments that other uses of "alphanumeric string" in the '645 and '413 patents require a broad interpretation such that the string may include only a hostname are likewise not persuasive. None of the uses of "alphanumeric string" in either patent clearly limits the contents to just a hostname. In fact, Akamai does not explain how a string made up of just a virtual server hostname would be "associated" with the original object even under the broadest definition of that term.

Akamai argues that the district court's requirement that the alphanumeric string include an entire URL is nonsensical because DNS servers resolve hostnames, not URLs. Akamai also asserts that the district court's statement that "[t]he URL of the object is necessary to the inventive global framework in order to retrieve the object from the content provider's server if no copy exists on a ghost [i.e., content] server" in its claim construction order, *Claim Construction Order* at 40, demonstrates a "fundamental misunderstanding of the requirements of the invention." Akamai's Principal Br. at 57. According to Akamai, this statement ignores that the specification describes retrieving any missing content from either the content provider's original server or another content server in the CDN. None of these arguments are persuasive. At no place does the specification indicate that the entire string must be used by the DNS server. Even if only the hostname is used by the DNS during the resolving step, this does not mean that an alpha-

numeric string cannot contain other information not used by the DNS during this step. Indeed, this must ***1329** be the case since the specification explicitly notes that "the ghost knows who the original server was because the name was encoded into the URL that was passed to the ghost from the browser." '645 patent col.12 ll.56–58.

C. Selection by the Alternative Domain Name System

[11] Akamai also appeals the district court's construction of "the given name server that receives the DNS query being close to the client local name server as determined by given location information" in claim 1 of the ' 645 patent and "selecting a given one of the name servers in the content delivery network" in claims 8, 18, and 20 of the '413 patent. The district court interpreted both limitations to require that the name server be selected by the alternative domain name system.[FN5] *Claim Construction Order*, at 42, 45. The court found that the specification compelled this interpretation because "[r]ead in light of the specification, the invention claims an alternate DNS system that selects a DNS server in response to a user request based on the location of the user." *Id.* at 43. Akamai, citing *DSW, Inc. v. Shoe Pavilion, Inc.*, 537 F.3d 1342, 1347 (Fed.Cir.2008), argues that the district court improperly incorporated a structural limitation—the alternative domain name system—into method claims. Moreover, Akamai asserts that claim 1 of the '645 patent does not use the term selecting at all. Akamai points out that claim 1 only requires that the CDN's DNS server receiving a DNS query be close to the client's local name server. In addition, Akamai argues that nothing in the '413 patent claim language, specification, or prosecution history supports the court's requirement of selection by the alternative domain name system. Limelight responds that the district court did not import a new structural limitation because claim 1 expressly requires an alternative domain name system.

> FN5. The claim limitations and their associated construction differ slightly for the '645 and '413 patents. For the '645 patent, the limitation "the given name server that receives the DNS query being close to the client local name server as determined by given location information" was construed by the district court to be "the particular name server that receives the DNS query is selected by the alternative domain name system and is close in Internet terms to the client local name server." *Claim Construction Order* at 42. Claims 8 and 18 of the ' 413 patent include the limitation "responsive to a DNS query, selecting a given one of the name servers in the content delivery network," which is construed as "in response to a DNS query, the [CDN's] [DNS] selects a particular name server." *Id.* at 45. Claim 20 of the '413 patent includes the limitation "responsive to a DNS query received from a client local name server, selecting a given one of the name servers in the [CDN]," which is construed as "in response to a DNS query received from a client local name server, the [CDN's] [DNS] selects a particular name server." *Id.* at 45. These distinctions are not germane to the issue presented in this appeal.

[12] This court is not persuaded by Akamai's argument. *DSW* is inapposite. In *DSW* this court reversed the district court's claim construction importing a limitation from a preferred embodiment because the claim language was unambiguously broader than the preferred embodiment, not because it imported structural limitations into a method claim. *Id.* at 1347. Method claims often include structural details. *See e.g.*, *Microprocessor Enhancement Corp. v. Tex. Instruments, Inc.*, 520 F.3d 1367, 1374 (Fed.Cir.2008) ("Method claim preambles often recite the physical structures of a system in which the claimed method is practiced, and ***1330** claim 1 is no different."); *Eaton Corp. v. Rockwell Int'l Corp.*, 323 F.3d 1332, 1342 (Fed.Cir.2003) (construing a method claim as including "steps that require the operation or manipulation of the particular structure identified and described by the preamble"). All of the asserted claims in both the '645 patent and the '413 patent explicitly refer to the alternative domain name system as a detail associated with the claimed method. '645 patent col.17 ll.50–51 ("having the service provider establish an alternative domain name system (DNS)"); '413 patent col.18 ll.22–23, col.19 ll.44–45, col.20 ll.25–26 ("having the content delivery network service provider establish an alternative domain name system (DNS)"). Therefore, the structural element of the alternative DNS framework was explicitly and properly included in the claims.

Akamai also asserts that the district court's interpretation improperly limits the inventive framework to

a multi-level DNS system. Akamai points out that because the patents explicitly allow for a framework with a one-level DNS framework, a multi-level restriction is unduly limiting. '703 patent col.5 ll.56–57 ("Alternatively, there may be a single DNS level that combines the functionality of the top-level and low-level servers.").

The district court responded to this exact argument in its claim construction order. Specifically, the court explained that because the specification states that "the functionality of the top and low-level servers" may be combined in "a single DNS level," the specification requires that a single-level DNS system accomplish the same steps as the two-level system described in the preferred embodiment. *Claim Construction Order* at 45. Thus, the district court's construction *does* support a single-level DNS system, and is not limited to a multilevel DNS system. As the district court recognized, the steps described in the preferred embodiment—(1) a top-level DNS server of the CDN selects a close-by low-level DNS server and redirects the user to that server and (2) the user's local DNS server requests the object's IP address from the low-level server—can be accomplished by one DNS server. *Id.* at 46 (citing '413 patent col.9 ll.44–50). Specifically, the district court explained:

> In a single-level DNS embodiment, as suggested by the specification, the user's local name server would still contact a content delivery provider's top-level name server to resolve the IP address of a server to serve an object. This name server, however, would then *directly* communicate with a particular local name server, based on the user's location, to resolve the server's IP address and return it to the user, rather than require the user to conduct a second lookup. Thus, the user would obtain the IP address of the appropriate ghost server with only a *single* DNS request, however the selection of a particular name server would still be the result of a DNS lookup by the service provider's DNS system. Such an embodiment would satisfy the claimed "one" level of DNS, yet not be in conflict with [the district court's adopted] claim construction.
>
> *Id.* at 45–46.

This explanation is entirely consistent with the specification's description of the invention and effectively counters Akamai's argument that the court's construction improperly limits the invention to a multi-level DNS system. Akamai also asserts, however, that one of these "other techniques" could be substituted for the top-level DNS servers in order to implement a one-level DNS framework. Thus, according to Akamai, the patent, but not the district court's construction, allows for a one-level DNS framework in which "other techniques," such as "Anycasting," ***1331** would be used to select the ultimate CDN DNS server—instead of a top-level DNS server—because "the specification encompassed techniques known in the prior art." Akamai's Br. at 61 (citing *BJ Servs. Co. v. Halliburton Energy Servs., Inc.*, 338 F.3d 1368, 1372 (Fed.Cir.2003)). This court does not agree that the patent's description allows for such a broad reading of the claims. The patent disclosure supports only one method for choosing the ultimate CDN DNS server—the alternative DNS system. There is no support in the specification for any method of choosing a particular name server other than by a DNS lookup and no disclosure that would have suggested to a person of ordinary skill in the art that anything other than a DNS lookup should be contemplated. There is no evidence that, given the lack of detailed disclosure in the patent's language, a person of skill in the art would have looked to other known techniques to implement this portion of the claimed invention.

In fact, the patent repeatedly defines using DNS lookup for choosing the ultimate CDN DNS server as the "invention." As noted by the district court, the specification describes "the present invention" as "manipulat[ing] the DNS system so the name is resolved to one of the ghosts that is near the client." '703 patent col.9 ll.26–28. In addition, under the heading entitled "Brief Summary of the Invention," the specification states that "[t]o locate the appropriate hosting servers to use, the top-level DNS server determines the user's location in the network to identify a given low-level DNS server to respond to the request for the embedded object." '703 patent col.3 ll.29–33.

Akamai conceded that under the district court's construction, Limelight does not infringe the '645 patent. Akamai also does not argue that Limelight would infringe the '413 patent under this construction. Therefore, this court is left to conclude that the district court properly entered judgment in favor of Limelight on the issue of infringement.

CONCLUSION

For the foregoing reasons, this court affirms the district court's grant of Limelight's motion for JMOL of noninfringement of the '703 patent. This court also affirms the district court's entry of judgment of non-infringement of the '645 and '413 patents.

AFFIRMED

C.A.Fed. (Mass.),2010.
Akamai Technologies, Inc. v. Limelight Networks, Inc.
629 F.3d 1311, 97 U.S.P.Q.2d 1321

END OF DOCUMENT

US006108703A

United States Patent [19]

Leighton et al.

[11] Patent Number: 6,108,703

[45] Date of Patent: Aug. 22, 2000

[54] GLOBAL HOSTING SYSTEM

[75] Inventors: **F. Thomson Leighton**, Newtonville; **Daniel M. Lewin**, Cambridge, both of Mass.

[73] Assignee: **Massachusetts Institute of Technology**, Cambridge, Mass.

[21] Appl. No.: **09/314,863**

[22] Filed: **May 19, 1999**

Related U.S. Application Data

[60] Provisional application No. 60/092,710, Jul. 14, 1998.

[51] **Int. Cl.**[7] **G06F 13/00**

[52] **U.S. Cl.** **709/226**; 709/105; 709/219; 709/223; 709/224; 709/235

[58] **Field of Search** 707/10, 2, 104, 707/203, 500, 501, 511, 512, 513, 515; 709/200, 201, 203, 218, 219, 230, 235, 238, 245, 246, 226, 224, 105, 220; 711/118, 119, 120, 122, 126, 130, 200, 202, 216

[56] **References Cited**

U.S. PATENT DOCUMENTS

4,922,417	5/1990	Churm et al.	707/1
5,287,499	2/1994	Nemes	707/2
5,341,477	8/1994	Pitkin et al.	709/226
5,542,087	7/1996	Neimat et al.	707/10
5,638,443	6/1997	Stefik et al.	705/54
5,646,676	7/1997	Dewkett et al.	348/7
5,715,453	2/1998	Stewart	707/104
5,740,423	4/1998	Logan et al.	707/10
5,751,961	5/1998	Smyk	709/217
5,761,507	12/1999	Govett	395/684
5,774,660	6/1998	Brendel et al.	709/201
5,777,989	7/1998	McGarvey	370/254
5,802,291	9/1998	Balick et al.	709/202
5,832,506	11/1998	Kuzma	707/200
5,856,974	1/1999	Gervais et al.	370/392
5,870,559	2/1999	Leshem et al.	709/224
5,878,212	3/1999	Civanlar et al.	709/203
5,884,038	3/1999	Kapoor	709/226
5,903,723	5/1999	Beck et al.	709/200
5,919,247	12/1999	Van Hoff et al.	709/217
5,933,832	8/1999	Suzuoka et al.	707/101
5,945,989	8/1999	Freishtat et al.	345/329
5,956,716	9/1999	Kenner et al.	707/10
5,961,596	10/1999	Takubo et al.	709/224
5,991,809	11/1999	Kriegsman	709/226
6,003,030	12/1999	Kenner et al.	707/10
6,006,264	12/1999	Colby et al.	709/226

FOREIGN PATENT DOCUMENTS

2202572	10/1998	Canada .
865180A2	9/1998	European Pat. Off. .
9804985	2/1998	WIPO .

OTHER PUBLICATIONS

Shaw, David M. "A Low Latency, High Throughput Web Service Using Internet-wide Replication." Department of Computer Science, Johns Hopkins University, Aug. 1998, 33 pgs.

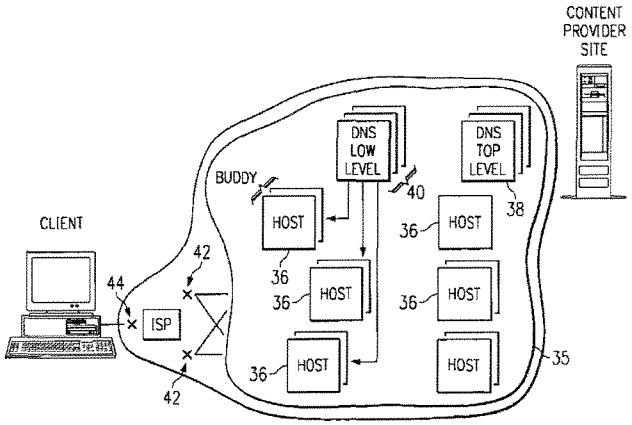
(List continued on next page.)

Primary Examiner—Dung C. Dinh
Assistant Examiner—Abdullahi E. Salad
Attorney, Agent, or Firm—David H. Judson

[57] **ABSTRACT**

The present invention is a network architecture or framework that supports hosting and content distribution on a truly global scale. The inventive framework allows a Content Provider to replicate and serve its most popular content at an unlimited number of points throughout the world. The inventive framework comprises a set of servers operating in a distributed manner. The actual content to be served is preferably supported on a set of hosting servers (sometimes referred to as ghost servers). This content comprises HTML page objects that, conventionally, are served from a Content Provider site. In accordance with the invention, however, a base HTML document portion of a Web page is served from the Content Provider's site while one or more embedded objects for the page are served from the hosting servers, preferably, those hosting servers near the client machine. By serving the base HTML document from the Content Provider's site, the Content Provider maintains control over the content.

34 Claims, 2 Drawing Sheets



OTHER PUBLICATIONS

Amir, Yair, et al. "Seamlessly Selecting the Best Copy from Internet–Wide Replicated Web Servers." Department of Computer Science, Johns Hopkins University, Jun. 1998, 14 pgs.

Bestavros, Azer. "Speculative Data Dissemination and Service to Reduce Server Load, Network Traffic and Service Time in Distributed Information Systems." In *Proceedings of ICDE '96: The 1996 International Conference on Data Engineering*, Mar. 1996, 4 pgs.

Carter, J. Lawrence, et al. "Universal Classes of Hash Function." *Journal of Computer and System Sciences*, vol. 18, No. 2, Apr. 1979, pp. 143–154.

Chankhunthod, Anawat, et al. "A Hierarchical Internet Object Cache." In *Usenix Proceedings*, Jan. 1996, pgs. 153–163.

Cormen, Thomas H., et al. *Introduction to Algorithms*, The MIT Press, Cambrdige, Massachusetts, 1994, pgs. 219–243, 991–993.

Deering, Stephen, et al. "Multicast Routing in Datagram Internetworks and Extended LANs." *ACM Transactions on Computer Systems*, vol. 8, No. 2, May 1990, pgs. 85–110.

Devine, Robert. "Design and Implementation of DDH: A Distributed Dynamic Hashing Algorithm." In *Proceedings of 4th International Conference on Foundations of Data Organizations and Algorithms,* 1993, pgs. 101–114.

Grigni, Michelangelo, et al. "Tight Bounds on Minimum Broadcasts Networks." *SIAM Journal of Discrete Mathematics,* vol. 4, No. 2, May 1991, pgs. 207–222.

Gwertzman, James, et al. "The Case for Geographical Push–Caching." *Technical Report HU TR 34–94*(excerpt), Harvard University, DAS, Cambridge, MA 02138, 1994, 2 pgs.

Gwertzman, James, et al. "World–Wide Web Cache Consistency." In *Proceedings of the 1996 USENIX Technical Conference*, Jan. 1996, 8 pgs.

Feeley, Michael, et al. "Implementing Global Memory Management in a Workstation Cluster." In *Proceedings of the 15th ACM Symposium on Operating Systems Principles,* 1995, pgs. 201–212.

Floyd, Sally, et al. "A Reliable Multicast Framework for Light–Weight Sessions and Application Level Framing." In *Proceeding of ACM SIGCOMM'95,* pgs. 342–356.

Fredman, Michael, et al. "Storing a Sparse Table with *0*(1) Worst Case Access Time." *Journal of the Association for Computing Machinery,* vol. 31., No. 3, Jul. 1984, pgs. 538–544.

Karger, David, et al. "Consistent Hashing and Random Trees: Distributed Caching Protocols for Relieving Hot Spots on the World Wide Web." In *Proceedings of the Twenty–Ninth Annual ACM Symposium on Theory of Computing* ,May 1997, pgs. 654–663.

Litwin, Withold, et al. "LH—A Scaleable, Distributed Data Structure." *ACM Transactions on Database Systems,* vol. 21, No. 4, Dec. 1996, pgs. 480–525.

Malpani, Radhika, et al. "Making World Wide Web Caching Servers Cooperate." In *Proceedings of World Wide Web Conference,* 1996, 6 pgs.

Naor, Moni, et al. "The Load, Capacity and Availability of Quorum Systems." In *Proceedings of the 35th IEEE Symposium on Foundations of Computer Science,* Nov. 1994, pgs. 214–225.

Nisan, Noam. "Psuedorandom Generators for Space–Bounded Computation." In *Proceedings of the Twenty–Second Annual ACM Symposium on Theory of Computing,* May 1990, pgs. 204–212.

Palmer, Mark, et al. "Fido: A Cache that Learns to Fetch." In *Proceedings of the 17th International Conference on Very Large Data Bases,* Sep. 1991, pgs. 255–264.

Panigraphy, Rina. *Relieving Hot Spots on the World Wide Web.* Massachusetts Institute of Technology, Jun. 1997, pgs. 1–66.

Peleg, David, et al. "The Availability of Quorum Systems." *Information and Computation* 123, 1995, 210–223.

Plaxton, C. Greg, et al. "Fast Fault–Tolerant Concurrent Access to Shared Objects." In *Proceedings of 37th IEEE Symposium on Foundations of Computer Science*, 1996, pgs. 570–579.

Rabin, Michael. "Efficient Dispersal of Information for Security, Load Balancing, and Fault Tolerance." *Journal of the ACM*, vol. 36, No. 2, Apr. 1989, pgs. 335–348.

Ravi, R., "Rapid Rumor Ramification: Approximating the Miniumum Broadcast Time." In *Proceedings of the 35th IEEE Symposium on Foundations of Computer Science*, Nov. 1994, pgs. 202–213.

Schmidt, Jeanette, et al. "Chernoff–Hoeffding Bounds for Applications with Limited Independence." In *Proceedings of the 4th ACS–SIAM Symposium on Discrete Algorithms*, 1993, pgs. 331–340.

Tarjan, Robert Endre, et al. "Storing a Sparse Table." *Communications of the ACM*, vol. 22, No. 11, Nov. 1979, pgs. 606–611.

Vitter, Jeffrey Scott, et al. "Optimal Prefetching via Data Compression." In *Proceedings of the 32nd IEEE Symposium on Foundations of Computer Science*, Nov. 1991, pgs. 121–130.

Wegman, Mark, et al. "New Hash Functions and Their Use in Authentication and Set Equality." *Journal of Computer and System Sciences* vol. 22, Jun. 1981, pgs. 265–279.

Yao, Andrew Chi–Chih. "Should Tables be Sorted?" *Journal of the Association for Computing Machinery*, vol. 28, No. 3, Jul. 1981, pgs. 615–628.

Beavan, Colin "Web Life They're Watching You." *Esquire*, Aug. 1997, pgs. 104–105.

Beavan, Colin "Web Life They're Watching You." Esquire, Aug. 1997, pp. 104–105.

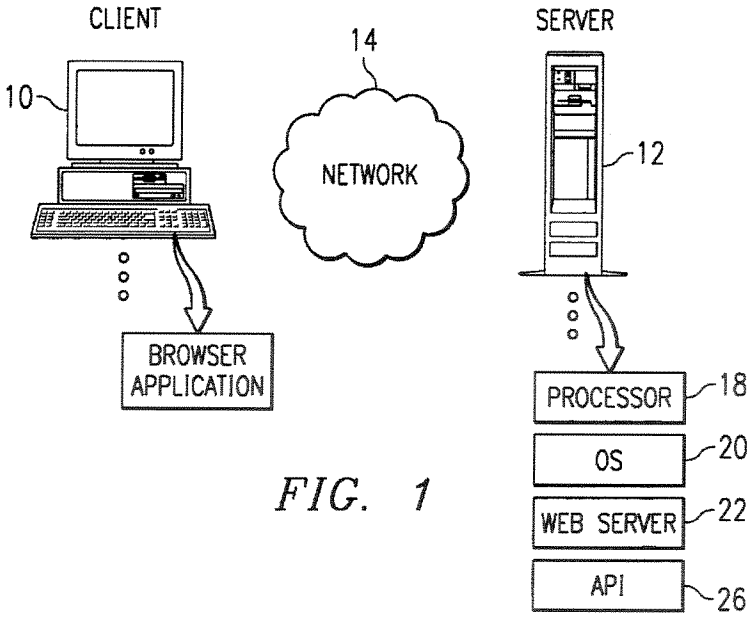


FIG. 1

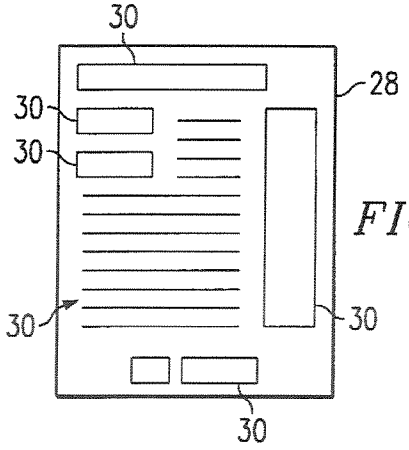


FIG. 2

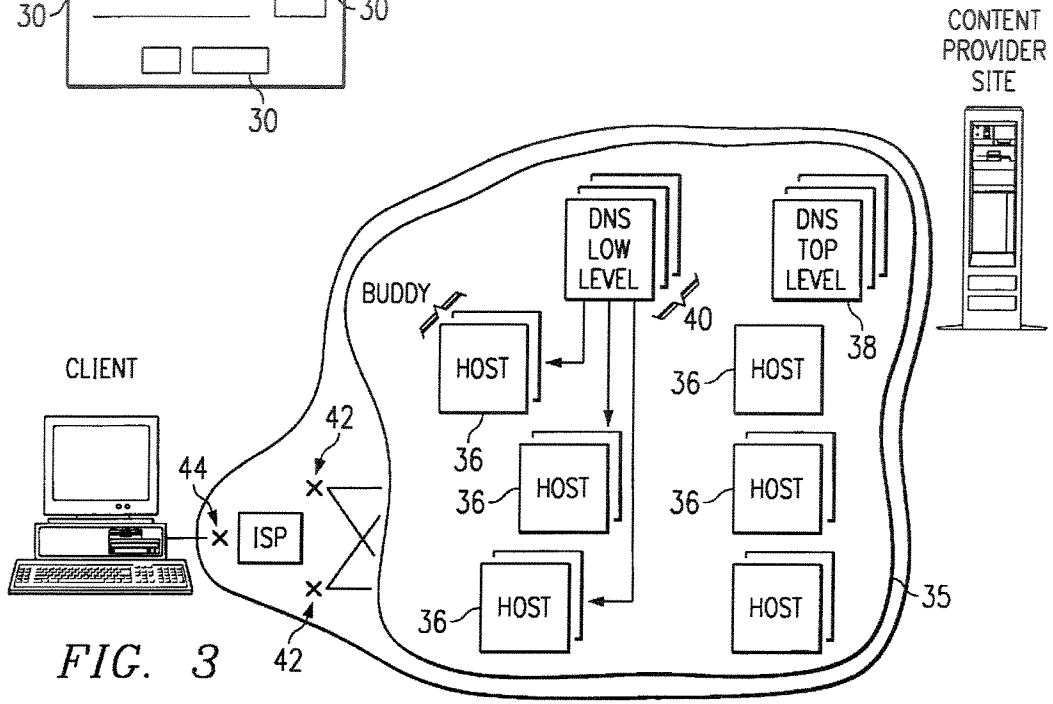


FIG. 3

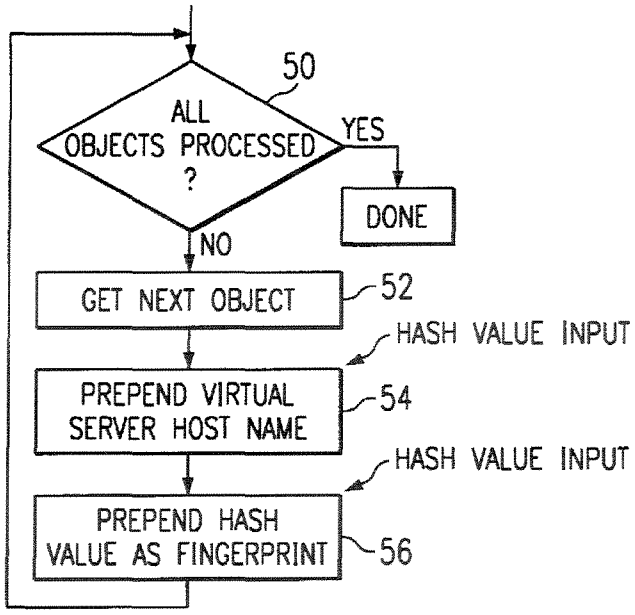

50
ALL OBJECTS PROCESSED ?
YES
DONE
NO
GET NEXT OBJECT
52
HASH VALUE INPUT
PREPEND VIRTUAL SERVER HOST NAME
54
HASH VALUE INPUT
PREPEND HASH VALUE AS FINGERPRINT
56

FIG. 4

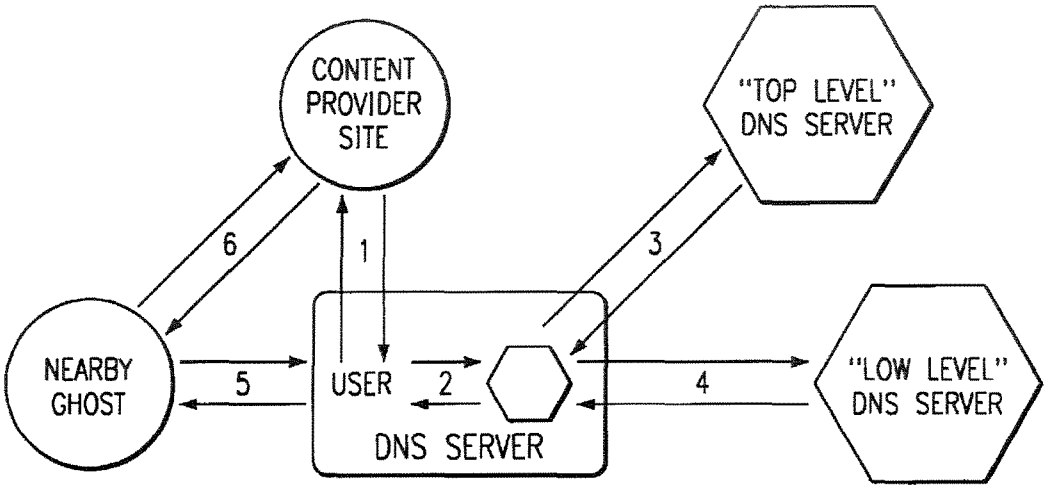

CONTENT PROVIDER SITE
"TOP LEVEL" DNS SERVER
6
1
3
NEARBY GHOST
5
USER
2
DNS SERVER
4
"LOW LEVEL" DNS SERVER

FIG. 5

GLOBAL HOSTING SYSTEM

This application is based on Provisional Application No. 60/092,710, filed Jul. 14, 1998. This application includes subject matter protected by copyright.

BACKGROUND OF THE INVENTION

1. Technical Field

This invention relates generally to information retrieval in a computer network. More particularly, the invention relates to a novel method of hosting and distributing content on the Internet that addresses the problems of Internet Service Providers (ISPs) and Internet Content Providers.

2. Description of the Related Art

The World Wide Web is the Internet's multimedia information retrieval system. In the Web environment, client machines effect transactions to Web servers using the Hypertext Transfer Protocol (HTTP), which is a known application protocol providing users access to files (e.g., text, graphics, images, sound, video, etc.) using a standard page description language known as Hypertext Markup Language (HTML). HTML provides basic document formatting and allows the developer to specify "links" to other servers and files. In the Internet paradigm, a network path to a server is identified by a so-called Uniform Resource Locator (URL) having a special syntax for defining a network connection. Use of an HTML-compatible browser (e.g., Netscape Navigator or Microsoft Internet Explorer) at a client machine involves specification of a link via the URL. In response, the client makes a request to the server identified in the link and, in return, receives a document or other object formatted according to HTML. A collection of documents supported on a Web server is sometimes referred to as a Web site.

It is well known in the prior art for a Web site to mirror its content at another server. Indeed, at present, the only method for a Content Provider to place its content closer to its readers is to build copies of its Web site on machines that are located at Web hosting farms in different locations domestically and internationally. These copies of Web sites are known as mirror sites. Unfortunately, mirror sites place unnecessary economic and operational burdens on Content Providers, and they do not offer economies of scale. Economically, the overall cost to a Content Provider with one primary site and one mirror site is more than twice the cost of a single primary site. This additional cost is the result of two factors: (1) the Content Provider must contract with a separate hosting facility for each mirror site, and (2) the Content Provider must incur additional overhead expenses associated with keeping the mirror sites synchronized.

In an effort to address problems associated with mirroring, companies such as Cisco, Resonate, Bright Tiger, F5 Labs and Alteon, are developing software and hardware that will help keep mirror sites synchronized and load balanced. Although these mechanisms are helpful to the Content Provider, they fail to address the underlying problem of scalability. Even if a Content Provider is willing to incur the costs associated with mirroring, the technology itself will not scale beyond a few (i.e., less than 10) Web sites.

In addition to these economic and scalability issues, mirroring also entails operational difficulties. A Content Provider that uses a mirror site must not only lease and manage physical space in distant locations, but it must also buy and maintain the software or hardware that synchronizes and load balances the sites. Current solutions require Content Providers to supply personnel, technology and other items necessary to maintain multiple Web sites. In summary, mirroring requires Content Providers to waste economic and other resources on functions that are not relevant to their core business of creating content.

Moreover, Content Providers also desire to retain control of their content. Today, some ISPs are installing caching hardware that interrupts the link between the Content Provider and the end-user. The effect of such caching can produce devastating results to the Content Provider, including (1) preventing the Content Provider from obtaining accurate hit counts on its Web pages (thereby decreasing revenue from advertisers), (2) preventing the Content Provider from tailoring content and advertising to specific audiences (which severely limits the effectiveness of the Content Provider's Web page), and (3) providing outdated information to its customers (which can lead to a frustrated and angry end user).

There remains a significant need in the art to provide a decentralized hosting solution that enables users to obtain Internet content on a more efficient basis (i.e., without burdening network resources unnecessarily) and that likewise enables the Content Provider to maintain control over its content.

The present invention solves these and other problems associated with the prior art.

BRIEF SUMMARY OF THE INVENTION

It is a general object of the present invention to provide a computer network comprising a large number of widely deployed Internet servers that form an organic, massively fault-tolerant infrastructure designed to serve Web content efficiently, effectively, and reliably to end users.

Another more general object of the present invention is to provide a fundamentally new and better method to distribute Web-based content. The inventive architecture provides a method for intelligently routing and replicating content over a large network of distributed servers, preferably with no centralized control.

Another object of the present invention is to provide a network architecture that moves content close to the user. The inventive architecture allows Web sites to develop large audiences without worrying about building a massive infrastructure to handle the associated traffic.

Still another object of the present invention is to provide a fault-tolerant network for distributing Web content. The network architecture is used to speed-up the delivery of richer Web pages, and it allows Content Providers with large audiences to serve them reliably and economically, preferably from servers located close to end users.

A further feature of the present invention is the ability to distribute and manage content over a large network without disrupting the Content Provider's direct relationship with the end user.

Yet another feature of the present invention is to provide a distributed scalable infrastructure for the Internet that shifts the burden of Web content distribution from the Content Provider to a network of preferably hundreds of hosting servers deployed, for example, on a global basis.

In general, the present invention is a network architecture that supports hosting on a truly global scale. The inventive framework allows a Content Provider to replicate its most popular content at an unlimited number of points throughout the world. As an additional feature, the actual content that is replicated at any one geographic location is specifically tailored to viewers in that location. Moreover, content is automatically sent to the location where it is requested, without any effort or overhead on the part of a Content Provider.

It is thus a more general object of this invention to provide a global hosting framework to enable Content Providers to retain control of their content.

The hosting framework of the present invention comprises a set of servers operating in a distributed manner. The actual content to be served is preferably supported on a set of hosting servers (sometimes referred to as ghost servers). This content comprises HTML page objects that, conventionally, are served from a Content Provider site. In accordance with the invention, however, a base HTML document portion of a Web page is served from the Content Provider's site while one or more embedded objects for the page are served from the hosting servers, preferably, those hosting servers nearest the client machine. By serving the base HTML document from the Content Provider's site, the Content Provider maintains control over the content.

The determination of which hosting server to use to serve a given embedded object is effected by other resources in the hosting framework. In particular, the framework includes a second set of servers (or server resources) that are configured to provide top level Domain Name Service (DNS). In addition, the framework also includes a third set of servers (or server resources) that are configured to provide low level DNS functionality. When a client machine issues an HTTP request to the Web site for a given Web page, the base HMTL document is served from the Web site as previously noted. Embedded objects for the page preferably are served from particular hosting servers identified by the top- and low-level DNS servers. To locate the appropriate hosting servers to use, the top-level DNS server determines the user's location in the network to identify a given low-level DNS server to respond to the request for the embedded object. The top-level DNS server then redirects the request to the identified low-level DNS server that, in turn, resolves the request into an IP address for the given hosting server that serves the object back to the client.

More generally, it is possible (and, in some cases, desirable) to have a hierarchy of DNS servers that consisting of several levels. The lower one moves in the hierarchy, the closer one gets to the best region.

A further aspect of the invention is a means by which content can be distributed and replicated through a collection of servers so that the use of memory is optimized subject to the constraints that there are a sufficient number of copies of any object to satisfy the demand, the copies of objects are spread so that no server becomes overloaded, copies tend to be located on the same servers as time moves forward, and copies are located in regions close to the clients that are requesting them. Thus, servers operating within the framework do not keep copies of all of the content database. Rather, given servers keep copies of a minimal amount of data so that the entire system provides the required level of service. This aspect of the invention allows the hosting scheme to be far more efficient than schemes that cache everything everywhere, or that cache objects only in prespecified locations.

The global hosting framework is fault tolerant at each level of operation. In particular, the top level DNS server returns a list of low-level DNS servers that may be used by the client to service the request for the embedded object. Likewise, each hosting server preferably includes a buddy server that is used to assume the hosting responsibilities of its associated hosting server in the event of a failure condition.

According to the present invention, load balancing across the set of hosting servers is achieved in part through a novel technique for distributing the embedded object requests. In particular, each embedded object URL is preferably modified by prepending a virtual server hostname into the URL. More generally, the virtual server hostname is inserted into the URL. Preferably, the virtual server hostname includes a value (sometimes referred to as a serial number) generated by applying a given hash function to the URL or by encoding given information about the object into the value. This function serves to randomly distribute the embedded objects over a given set of virtual server hostnames. In addition, a given fingerprint value for the embedded object is generated by applying a given hash function to the embedded object itself. This given value serves as a fingerprint that identifies whether the embedded object has been modified. Preferably, the functions used to generate the values (i.e., for the virtual server hostname and the fingerprint) are applied to a given Web page in an off-line process. Thus, when an HTTP request for the page is received, the base HTML document is served by the Web site and some portion of the page's embedded objects are served from the hosting servers near (although not necessarily the closest) to the client machine that initiated the request.

The foregoing has outlined some of the more pertinent objects and features of the present invention. These objects should be construed to be merely illustrative of some of the more prominent features and applications of the invention. Many other beneficial results can be attained by applying the disclosed invention in a different manner or modifying the invention as will be described. Accordingly, other objects and a fuller understanding of the invention may be had by referring to the following Detailed Description of the Preferred Embodiment.

BRIEF DESCRIPTION OF THE DRAWINGS

For a more complete understanding of the present invention and the advantages thereof, reference should be made to the following Detailed Description taken in connection with the accompanying drawings in which:

FIG. **1** is a representative system in which the present invention is implemented;

FIG. **2** is a simplified representation of a markup language document illustrating the base document and a set of embedded objects;

FIG. **3** is a high level diagram of a global hosting system according to the present invention;

FIG. **4** is a simplified flowchart illustrating a method of processing a Web page to modified embedded object URLs that is used in the present invention;

FIG. **5** is a simplified state diagram illustrating how the present invention responds to a HTTP request for a Web page.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENT

A known Internet client-server system is implemented as illustrated in FIG. **1**. A client machine **10** is connected to a Web server **12** via a network **14**. For illustrative purposes, network **14** is the Internet, an intranet, an extranet or any other known network. Web server **12** is one of a plurality of servers which are accessible by clients, one of which is illustrated by machine **10**. A representative client machine includes a browser **16**, which is a known software tool used to access the servers of the network. The Web server supports files (collectively referred to as a "Web" site) in the form of hypertext documents and objects. In the Internet

paradigm, a network path to a server is identified by a so-called Uniform Resource Locator (URL).

A representative Web server **12** is a computer comprising a processor **18**, an operating system **20**, and a Web server program **22**, such as Netscape Enterprise Server. The server **12** also includes a display supporting a graphical user interface (GUI) for management and administration, and an Application Programming Interface (API) that provides extensions to enable application developers to extend and/or customize the core functionality thereof through software programs including Common Gateway Interface (CGI) programs, plug-ins, servlets, active server pages, server side include (SSI) functions or the like.

A representative Web client is a personal computer that is x86-, PowerPC®-or RISC-based, that includes an operating system such as IBM® OS/2® or Microsoft Windows '95, and that includes a Web browser, such as Netscape Navigator 4.0 (or higher), having a Java Virtual Machine (JVM) and support for application plug-ins or helper applications. A client may also be a notebook computer, a handheld computing device (e.g., a PDA), an Internet appliance, or any other such device connectable to the computer network.

As seen in FIG. **2**, a typical Web page comprises a markup language (e.g. HTML) master or base document **28**, and many embedded objects (e.g., images, audio, video, or the like) **30**. Thus, in a typical page, twenty or more embedded images or objects are quite common. Each of these images is an independent object in the Web, retrieved (or validated for change) separately. The common behavior of a Web client, therefore, is to fetch the base HTML document, and then immediately fetch the embedded objects, which are typically (but not always) located on the same server. According to the present invention, preferably the markup language base document **28** is served from the Web server (i.e., the Content Provider site) whereas a given number (or perhaps all) of the embedded objects are served from other servers. As will be seen, preferably a given embedded object is served from a server (other than the Web server itself) that is close to the client machine, that is not overloaded, and that is most likely to already have a current version of the required file.

Referring now to FIG. **3**, this operation is achieved by the hosting system of the present invention. As will be seen, the hosting system **35** comprises a set of widely-deployed servers (or server resources) that form a large, fault-tolerant infrastructure designed to serve Web content efficiently, effectively, and reliably to end users. The servers may be deployed globally, or across any desired geographic regions. As will be seen, the hosting system provides a distributed architecture for intelligently routing and replicating such content. To this end, the global hosting system **35** comprises three (3) basic types of servers (or server resources): hosting servers (sometimes called ghosts) **36**, top-level DNS servers **38**, and low-level DNS servers **40**. Although not illustrated, there may be additional levels in the DNS hierarchy. Alternatively, there may be a single DNS level that combines the functionality of the top level and low-level servers. In this illustrative embodiment, the inventive framework **35** is deployed by an Internet Service Provider (ISP), although this is not a limitation of the present invention. The ISP or ISPs that deploy the inventive global hosting framework **35** preferably have a large number of machines that run both the ghost server component **36** and the low-level DNS component **40** on their networks. These machines are distributed throughout the network; preferably, they are concentrated around network exchange points **42** and network access points **44**, although this is not a requirement. In addition, the ISP preferably has a small number of machines running the top-level DNS **38** that may also be distributed throughout the network.

Although not meant to be limiting, preferably a given server used in the framework **35** includes a processor, an operating system (e.g., Linux, UNIX, Windows NT, or the like), a Web server application, and a set of application routines used by the invention. These routines are conveniently implemented in software as a set of instructions executed by the processor to perform various process or method steps as will be described in more detail below. The servers are preferably located at the edges of the network (e.g., in points of presence, or POPs).

Several factors may determine where the hosting servers are placed in the network. Thus, for example, the server locations are preferably determined by a demand driven network map that allows the provider (e.g., the ISP) to monitor traffic requests. By studying traffic patterns, the ISP may optimize the server locations for the given traffic profiles.

According to the present invention, a given Web page (comprising a base HTML document and a set of embedded objects) is served in a distributed manner. Thus, preferably, the base HTML document is served from the Content Provider that normally hosts the page. The embedded objects, or some subset thereof, are preferentially served from the hosting servers **36** and, specifically, given hosting servers **36** that are near the client machine that in the first instance initiated the request for the Web page. In addition, preferably loads across the hosting servers are balanced to ensure that a given embedded object may be efficiently served from a given hosting server near the client when such client requires that object to complete the page.

To serve the page contents in this manner, the URL associated with an embedded object is modified. As is well-known, each embedded object that may be served in a page has its own URL. Typically, the URL has a hostname identifying the Content Provider's site from where the object is conventionally served, i.e., without reference to the present invention. According to the invention, the embedded object URL is first modified, preferably in an off-line process, to condition the URL to be served by the global hosting servers. A flowchart illustrating the preferred method for modifying the object URL is illustrated in FIG. **4**.

The routine begins at step **50** by determining whether all of the embedded objects in a given page have been processed. If so, the routine ends. If not, however, the routine gets the next embedded object at step **52**. At step **54**, a virtual server hostname is prepended into the URL for the given embedded object. The virtual server hostname includes a value (e.g., a number) that is generated, for example, by applying a given hash function to the URL. As is well-known, a hash function takes arbitrary length bit strings as inputs and produces fixed length bit strings (hash values) as outputs. Such functions satisfy two conditions: (1) it is infeasible to find two different inputs that produce the same hash value, and (2) given an input and its hash value, it is infeasible to find a different input with the same hash value. In step **54**, the URL for the embedded object is hashed into a value xx,xxx that is then included in the virtual server hostname. This step randomly distributes the object to a given virtual server hostname.

The present invention is not limited to generating the virtual server hostname by applying a hash function as described above. As an alternative and preferred

embodiment, a virtual server hostname is generated as follows. Consider the representative hostname a1234.g.akamaitech.net. The 1234 value, sometimes referred to as a serial number, preferably includes information about the object such as its size (big or small), its anticipated popularity, the date on which the object was created, the identity of the Web site, the type of object (e.g., movie or static picture), and perhaps some random bits generated by a given random function. Of course, it is not required that any given serial number encode all of such information or even a significant number of such components. Indeed, in the simplest case, the serial number may be a simple integer. In any event, the information is encoded into a serial number in any convenient manner. Thus, for example, a first bit is used to denote size, a second bit is used to denote popularity, a set of additional bits is used to denote the date, and so forth. As noted above in the hashing example, the serial number is also used for load balancing and for directing certain types of traffic to certain types of servers. Typically, most URLs on the same page have the same serial number to minimize the number of distinguished name (DN) accesses needed per page. This requirement is less important for larger objects.

Thus, according to the present invention, a virtual server hostname is prepended into the URL for a given embedded object, and this hostname includes a value (or serial number) that is generated by applying a given function to the URL or object. That function may be a hash function, an encoding function, or the like.

Turning now back to the flowchart, the routine then continues at step **56** to include a given value in the object's URL. Preferably, the given value is generated by applying a given hash function to the embedded object. This step creates a unique fingerprint of the object that is useful for determining whether the object has been modified. Thereafter, the routine returns to step **50** and cycles.

With the above as background, the inventive global hosting framework is now described in the context of a specific example. In particular, it is assumed that a user of a client machine in Boston requests a Content Provider Web page normally hosted in Atlanta. For illustrative purposes, It is assumed that the Content Provider is using the global hosting architecture within a network, which may be global, international, national, regional, local or private. FIG. **5** shows the various components of the system and how the request from the client is processed. This operation is not to be taken by way of limitation, as will be explained.

Step **1**: The browser sends a request to the Provider's Web site (Item 1). The Content Provider site in Atlanta receives the request in the same way that it does as if the global hosting framework were not being implemented. The difference is in what is returned by the Provider site. Instead of returning the usual page, according to the invention, the Web site returns a page with embedded object URLs that are modified according to the method illustrated in the flowchart of FIG. **4**. As previously described, the URLs preferably are changed as follows:

Assume that there are 100,000 virtual ghost servers, even though there may only be a relatively small number (e.g., 100) physically present on the network. These virtual ghost servers or virtual ghosts are identified by the hostname: ghostxxxxx.ghosting.com, where xxxxx is replaced by a number between 0 and 99,999. After the Content Provider Web site is updated with new information, a script executing on the Content Provider site is run that rewrites the embedded URLs. Preferably, the embedded URLs names are hashed into numbers between 0 and 99,999, although this range is not a limitation of the present invention. An embedded URL is then switched to reference the virtual ghost with that number. For example, the following is an embedded URL from the Provider's site:

<IMG SRC=http://www.provider.com/TECH/images/space.story.gif>

If the serial number for the object referred to by this URL is the number 1467, then preferably the URL is rewritten to read:

<IMG SRC=http: //ghost467.ghosting.akamai.com/www.provider.com/TECH/images/sp ace.story.gif>.

The use of serial numbers in this manner distributes the embedded URLs roughly evenly over the 100,000 virtual ghost server names. Note that the Provider site can still personalize the page by rearranging the various objects on the screen according to individual preferences. Moreover, the Provider can also insert advertisements dynamically and count how many people view each ad.

According to the preferred embodiment, an additional modification to the embedded URLs is made to ensure that the global hosting system does not serve stale information. As previously described, preferably a hash of the data contained in the embedded URL is also inserted into the embedded URL itself. That is, each embedded URL may contain a fingerprint of the data to which it points. When the underlying information changes, so does the fingerprint, and this prevents users from referencing old data.

The second hash takes as input a stream of bits and outputs what is sometimes referred to as a fingerprint of the stream. The important property of the fingerprint is that two different streams almost surely produce two different fingerprints. Examples of such hashes are the MD2 and MD5 hash functions, however, other more transparent methods such as a simple checksum may be used. For concreteness, assume that the output of the hash is a 128 bit signature. This signature can be interpreted as a number and then inserted into the embedded URL. For example, if the hash of the data in the picture space.story.gif from the Provider web site is the number 28765, then the modified embedded URL would actually look as follows:

<IMGSRC=http://ghost1467.ghosting.akamai.com/28765/www.provider.com /TECH/images/space.story.gif'>.

Whenever a page is changed, preferably the hash for each embedded URL is recomputed and the URL is rewritten if necessary. If any of the URL's data changes, for example, a new and different picture is inserted with the name space.story.gif, then the hash of the data is different and therefore the URL itself will be different. This scheme prevents the system from serving data that is stale as a result of updates to the original page.

For example, assume that the picture space.story.gif is replaced with a more up-to-date version on the Content Provider server. Because the data of the pictures changes, the hash of the URL changes as well. Thus, the new embedded URL looks the same except that a new number is inserted for the fingerprint. Any user that requests the page after the update receives a page that points to the new picture. The old picture is never referenced and cannot be mistakenly returned in place of the more up-to-date information.

In summary, preferably there are two hashing operations that are done to modify the pages of the Content Provider. First, hashing can be a component of the process by which a serial number is selected to transform the domain name into a virtual ghost name. As will be seen, this first transformation serves to redirect clients to the global hosting

system to retrieve the embedded URLs. Next, a hash of the data pointed to by the embedded URLs is computed and inserted into the URL. This second transformation serves to protect against serving stale and out-of-date content from the ghost servers. Preferably, these two transformations are performed off-line and therefore do not pose potential performance bottlenecks.

Generalizing, the preferred URL schema is as follows. The illustrative domain www.domainname.com/frontpage.jpg is transformed into:

xxxx.yy.zzzz.net/aaaa/www.domainname.com/frontpage.jpg,

where:

xxxx=serial number field

yy=lower level DNS field

zzzz=top level DNS field

aaaa=other information (e.g., fingerprint) field.

If additional levels of the DNS hierarchy are used, then there may be additional lower level DNS fields, e.g., xxxx.y_1y_1.y_2y_2 zzz.net/aaaa/. . . .

Step **2**: After receiving the initial page from the Content Provider site, the browser needs to load the embedded URLs to display the page. The first step in doing this is to contact the DNS server on the user's machine (or at the user's ISP) to resolve the altered hostname, in this case: ghost1467.ghosting.akamai.com. As will be seen, the global hosting architecture of the present invention manipulates the DNS system so that the name is resolved to one of the ghosts that is near the client and is likely to have the page already. To appreciate how this is done, the following describes the progress of the DNS query that was initiated by the client.

Step **3**: As previously described, preferably there are two types of DNS servers in the inventive system: top-level and low-level. The top level DNS servers **38** for ghosting.com have a special function that is different from regular DNS servers like those of the .com domain. The top level DNS servers **38** include appropriate control routines that are used to determine where in the network a user is located, and then to direct the user to a akamai.com (i.e., a low level DNS) server **40** that is close-by. Like the .com domain, akamai-.com preferably has a number of top-level DNS servers **38** spread throughout the network for fault tolerance. Thus, a given top level DNS server **38** directs the user to a region in the Internet (having a collection of hosting servers **36** that may be used to satisfy the request for a given embedded object) whereas the low level DNS server **40** (within the identified region) identifies a particular hosting server within that collection from which the object is actually served.

More generally, as noted above, the DNS process can contain several levels of processing, each of which serves to better direct the client to a ghost server. The ghost server name can also have more fields. For example, "a123.g.g.akamaitech.net" may be used instead of "a123.ghost.akamai.com." If only one DNS level is used, a representative URL could be "a123.akamai.com."

Although other techniques may be used, the user's location in the network preferably is deduced by looking at the IP address of the client machine making the request. In the present example, the DNS server is running on the machine of the user, although this is not a requirement. If the user is using an ISP DNS server, for example, the routines make the assumption that the user is located near (in the Internet sense) this server. Alternatively, the user's location or IP address could be directly encoded into the request sent to the top level DNS. To determine the physical location of an IP address in the network, preferably, the top level DNS server builds a network map that is then used to identify the relevant location.

Thus, for example, when a request comes in to a top level DNS for a resolution for a1234.g.akamaitech.net, the top level DNS looks at the return address of the requester and then formulates the response based on that address according to a network map. In this example, the a1234 is a serial number, the g is a field that refers to the lower level DNS, and akamaitech refers to the top level DNS. The network map preferably contains a list of all Internet Protocol (IP) blocks and, for each IP block, the map determines where to direct the request. The map preferably is updated continually based on network conditions and traffic.

After determ ining where in the network the request originated, the top level DNS server redirects the DNS request to a low level DNS server close to the user in the netw ork. The ability to redirect requests is a standard feature in the DNS system. In addition, this redirection can be done in such a way that if the local low level DNS server is down, there is a backup server that is contacted.

Preferably, the TTL (time to live) stamp on these top level DNS redirections for the ghosting.com domain is set to be long. This allows DNS caching at the user's DNS s ervers and/or the ISP's DNS servers to prevent the top level DNS servers from being overloaded. If the TTL for ghosting.akamai.com in the DNS server at the user's machine or ISP has expired, then a top level server is contacted, and a new redirection to a local low level ghosting.akamai.com DNS server is returned with a new TTL stamp. It should be noted the system does not cause a substantially larger number of top level DNS lookups than what is done in the current centralized hosting solutions. This is because the TTL of the top level redirections are set to be high and, thus, the vast majority of users are directed by their local DNS straight to a nearby low level ghosting.akamai.com DNS server.

Moreover, fault tolerance for the top level DNS servers is provided automatically by DNS similarly to what is done for the popular .com domain. Fault tolerance for the low level DNS servers preferably is provided by returning a list of possible low level DNS servers instead of just a single server. If one of the low level DNS servers is down, the user will still be able to contact one on the list that is up and running.

Fault tolerance can also be handled via an "overflow control" mechanism wherein the client is redirected to a low-level DNS in a region that is known to have sufficient capacity to serve the object. This alternate approach is very useful in scenarios where there is a large amount of demand from a specific region or when there is reduced capacity in a region. In general, the clients are directed to regions in a way that minimizes the overall latency experienced by clients subject to the constraint that no region becomes overloaded. Minimizing overall latency subject to the regional capacity constraints preferably is achieved using a min-cost multicommodity flow algorithm.

Step **4**: At this point, the user has the address of a close-by ghosting.com DNS server **38**. The user's local DNS server contacts the close-by low level DNS server **40** and requests a translation for the name ghost1467.ghosting.akamai.com. The local DNS server is responsible for returning the IP address of one of the ghost servers **36** on the network that is close to the user, not overloaded, and most likely to already have the required data.

The basic mechanism for mapping the virtual ghost names to real ghosts is hashing. One preferred technique is so-called consistent hashing, as described in U.S. Ser. No. 09/042,228, filed Mar. 13, 1998, and in U.S. Ser. No. 09/088,825, filed Jun. 2, 1998, each titled Method And Apparatus For Distributing Requests Among A Plurality Of

Resources, and owned by the assachusetts Institute of Technology, which applications are ncorporated herein by reference. Consistent hash functions ake the system robust under machine failures and crashes. It also allows the system to grow gracefully, without changing where most items are located and without perfect information about the system.

According to the invention, the virtual ghost names may be hashed into real ghost addresses using a table lookup, where the table is continually updated based on network conditions and traffic in such a way to insure load balancing and fault tolerance. Preferably, a table of resolutions is created for each serial number. For example, serial number 1 resolves to ghost 2 and 5, serial number 2 resolves to ghost 3, serial number 3 resolves to ghosts 2,3,4, and so forth. The goal is to define the resolutions so that no ghost exceeds its capacity and that the total number of all ghosts in all resolutions is minimized. This is done to assure that the system can take maximal advantage of the available memory at each region. This is a major advantage over existing load balancing schemes that tend to cache everything everywhere or that only cache certain objects in certain locations no matter what the loads are. In general, it is desirable to make assignments so that resolutions tend to stay consistent over time provided that the loads do not change too much in a short period of time. This mechanism preferably also takes into account how close the ghost is to the user, and how heavily loaded the ghost is at the moment.

Note that the same virtual ghost preferably is translated to different real ghost addresses according to where the user is located in the network. For example, assume that ghost server 18.98.0.17 is located in the United States and that ghost server 132.68.1.28 is located in Israel. A DNS request for ghost1487.ghosting.akamai.com originating in Boston will resolve to 18.98.0.17, while a request originating in Tel-Aviv will resolve to 132.68.1.28.

The low-level DNS servers monitor the various ghost servers to take into account their loads while translating virtual ghost names into real addresses. This is handled by a software routine that runs on the ghosts and on the low level DNS servers. In one embodiment, the load information is circulated among the servers in a region so that they can compute resolutions for each serial number. One algorithm for computing resolutions works as follows. The server first computes the projected load (based on number of user requests) for each serial number. The serial numbers are then processed in increasing order of load. For each serial number, a random priority list of desired servers is assigned using a consistent hashing method. Each serial number is then resolved to the smallest initial segment of servers from the priority list so that no server becomes overloaded. For example, if the priority list for a serial number is 2,5,3,1,6, then an attempt is made first to try to map the load for the serial number to ghost 2. If this overloads ghost 2, then the load is assigned to both ghosts 2 and 5. If this produced too much load on either of those servers, then the load is assigned to ghosts 2,3, and 5, and so forth. The projected load on a server can be computed by looking at all resolutions that contain that server and by adding the amount of load that is likely to be sent to that server from that serial number. This method of producing resolutions is most effective when used in an iterative fashion, wherein the assignments starts in a default state, where every serial number is mapped to every ghost. By refining the resolution table according to the previous procedure, the load is balanced using the minimum amount of replication (thereby maximally conserving the available memory in a region).

The TTL for these low level DNS translations is set to be short to allow a quick response when heavy load is detected on one of the ghosts. The TTL is a parameter that can be manipulated by the system to insure a balance between timely response to high load on ghosts and the load induced on the low level DNS servers. Note, however, that even if the TTL for the low level DNS translation is set to 1–2 minutes, only a few of the users actually have to do a low level DNS lookup. Most users will see a DNS translation that is cached on their machine or at their ISP. Thus, most users go directly from their local DNS server to the close-by ghost that has the data they want. Those users that actually do a low level DNS lookup have a very small added latency, however this latency is small compared to the advantage of retrieving most of the data from close by.

As noted above, fault tolerance for the low level DNS servers is provided by having the top level DNS return a list of possible low level DNS servers instead of a single server address. The user's DNS system caches this list (part of the standard DNS system), and contacts one of the other servers on the list if the first one is down for some reason. The low level DNS servers make use of a standard feature of DNS to provide an extra level of fault tolerance for the ghost servers. When a name is translated, instead of returning a single name, a list of names is returned. If for some reason the primary fault tolerance method for the ghosts (known as the Buddy system, which is described below) fails, the client browser will contact one of the other ghosts on the list.

Step **5**: The browser then makes a request for an object named a123.ghosting.akamai.com/.../www.provider.com/TECH/images/space.story.gif from the close-by ghost. Note that the name of the original server (www.provider.com) preferably is included as part of the URL. The software running on the ghost parses the page name into the original host name and the real page name. If a copy of the file is already stored on the ghost, then the data is returned immediately. If, however, no copy of the data on the ghost exists, a copy is retrieved from the original server or another ghost server. Note that the ghost knows who the original server was because the name was encoded into the URL that was passed to the ghost from the browser. Once a copy has been retrieved it is returned to the user, and preferably it is also stored on the ghost for answering future requests.

As an additional safeguard, it may be preferable to check that the user is indeed close to the server. This can be done by examining the IP address of the client before responding to the request for the file. This is useful in the rare case when the client's DNS server is far away from the client. In such a case, the ghost server can redirect the user to a closer server (or to another virtual address that is likely to be resolved to a server that is closer to the client). If the redirect is to a virtual server, then it must be tagged to prevent further redirections from taking place. In the preferred embodiment, redirection would only be done for large objects; thus, a check may be made before applying a redirection to be sure that the object being requested exceeds a certain overall size.

Performance for long downloads can also be improved by dynamically changing the server to which a client is connected based on changing network conditions. This is especially helpful for audio and video downloads (where the connections can be long and where quality is especially important). In such cases, the user can be directed to an alternate server in mid-stream. The control structure for redirecting the client can be similar to that described above, but it can also include software that is placed in the client's browser or media player. The software monitors the performance of the client's connection and perhaps the status of the network as well. If it is deemed that the client's connection can be improved by changing the server, then the system directs the client to a new server for the rest of the connection.

Fault tolerance for the ghosts is provided by a buddy system, where each ghost has a designated buddy ghost. If a ghost goes down, its buddy takes over its work (and IP address) so that service is not interrupted. Another feature of the system is that the buddy ghost does not have to sit idle waiting for a failure. Instead, all of the machines are always active, and when a failure happens, the load is taken over by the buddy and then balanced by the low level DNS system to the other active ghosts. An additional feature of the buddy system is that fault tolerance is provided without having to wait for long timeout periods.

As yet another safety feature of the global hosting system, a gating mechanism can be used to keep the overall traffic for certain objects within specified limits. One embodiment of the gating mechanism works as follows. When the number of requests for an object exceeds a certain specified threshold, then the server can elect to not serve the object. This can be very useful if the object is very large. Instead, the client can be served a much smaller object that asks the client to return later. Or, the client can be redirected. Another method of implementing a gate is to provide the client with a "ticket" that allows the client to receive the object at a prespecified future time. In this method, the ghost server needs to check the time on the ticket before serving the object.

The inventive global hosting scheme is a way for global ISPs or conglomerates of regional ISPs to leverage their network infrastructure to generate hosting revenue, and to save on network bandwidth. An ISP offering the inventive global hosting scheme can give content providers the ability to distribute content to their users from the closest point on the ISPs network, thus ensuring fast and reliable access. Guaranteed web site performance is critical for any web-based business, and global hosting allows for the creation of a service that satisfies this need.

Global hosting according to the present invention also allows an ISP to control how and where content traverses its network. Global hosting servers can be set up at the edges of the ISP's network (at the many network exchange and access points, for example). This enables the ISP to serve content for sites that it hosts directly into the network exchange points and access points. Expensive backbone links no longer have to carry redundant traffic from the content provider's site to the network exchange and access points. Instead, the content is served directly out of the ISP's network, freeing valuable network resources for other traffic.

Although global hosting reduces network traffic, it is also a method by which global ISPs may capture a piece of the rapidly expanding hosting market, which is currently estimated at over a billion dollars a year.

The global hosting solution also provides numerous advantages to Content Providers, and, in particular, an efficient and cost-effective solution to improve the performance of their Web sites both domestically and internationally. The inventive hosting software ensures Content Providers with fast and reliable Internet access by providing a means to distribute content to their subscribers from the closest point on an ISP's network. In addition to other benefits described in more detail below, the global hosting solution also provides the important benefit of reducing network traffic.

Once inexpensive global hosting servers are installed at the periphery of an ISP's network (i.e., at the many network exchange and access points), content is served directly into network exchange and access points. As a result of this efficient distribution of content directly from an ISP's network, the present invention substantially improves Web site performance. In contrast to current content distribution systems, the inventive global hosting solution does not require expensive backbone links to carry redundant traffic from the Content Provider's Web site to the network exchange and access points.

A summary of the specific advantages afforded by the inventive global hosting scheme are set forth below:

1. Decreased Operational Expenses for Content Providers:

Most competing solutions require Content Providers to purchase servers at each Web site that hosts their content. As a result, Content Providers often must negotiate separate contracts with different ISPs around the world. In addition, Content Providers are generally responsible for replicating the content and maintaining servers in these remote locations.

With the present invention, ISPs are primarily responsible for the majority of the aspects of the global hosting. Content Providers preferably maintain only their single source server. Content on this server is automatically replicated by software to the locations where it is being accessed. No intervention or planning is needed by the Provider (or, for that matter, the ISP). Content Providers are offered instant access to all of the servers on the global network; there is no need to choose where content should be replicated or to purchase additional servers in remote locations.

2. Intelligent and Efficient Data Replication:

Most competing solutions require Content Providers to replicate their content on servers at a commercial hosting site or to mirror their content on geographically distant servers. Neither approach is particularly efficient. In the former situation, content is still located at a single location on the Internet (and thus it is far away from most users). In the latter case, the entire content of a Web site is copied to remote servers, even though only a small portion of the content may actually need to be located remotely. Even with inexpensive memory, the excessive cost associated with such mirroring makes it uneconomical to mirror to more than a few sites, which means that most users will still be far away from a mirror site. Mirroring also has the added disadvantage that Content Providers must insure that all sites remain consistent and current, which is a nontrivial task for even a few sites.

With the present invention, content is automatically replicated to the global server network in an intelligent and efficient fashion. Content is replicated in only those locations where it is needed. Moreover, when the content changes, new copies preferably are replicated automatically throughout the network.

3. Automatic Content Management:

Many existing solutions require active management of content distribution, content replication and load balancing between different servers. In particular, decisions about where content will be hosted must be made manually, and the process of replicating data is handled in a centralized push fashion. On the contrary, the invention features passive management. Replication is done in a demand-based pull fashion so that content preferably is only sent to where it is truly needed. Moreover, the process preferably is fully automated; the ISP does not have to worry about how and where content is replicated and/or the content provider.

4. Unlimited, Cost Effective Scalability:

Competing solutions are not scalable to more than a small number of sites. For example, solutions based on mirroring are typically used in connection with at most three or four sites. The barriers to scaling include the expense of replicating the entire site, the cost of replicating computing

resources at all nodes, and the complexity of supporting the widely varying software packages that Content Providers use on their servers.

The unique system architecture of the present invention is scaleable to hundreds, thousands or even millions of nodes. Servers in the hosting network can malfunction or crash and the system's overall function is not affected. The global hosting framework makes efficient use of resources; servers and client software do not need to be replicated at every node because only the hosting server runs at each node. In addition, the global hosting server is designed to run on standard simple hardware that is not required to be highly fault tolerant.

5. Protection against Flash Crowds:

Competing solutions do not provide the Content Provider with protection from unexpected flash crowds. Although mirroring and related load-balancing solutions do allow a Content Provider to distribute load across a collection of servers, the aggregate capacity of the servers must be sufficient to handle peak demands. This means that the Provider must purchase and maintain a level of resources commensurate with the anticipated peak load instead of the true average load. Given the highly variable and unpredictable nature of the Internet, such solutions are expensive and highly wasteful of resources.

The inventive hosting architecture allows ISPs to utilize a single network of hosting servers to offer Content Providers flash crowd insurance. That is, insurance that the network will automatically adapt to and support unexpected higher load on the Provider's site. Because the ISP is aggregating many Providers together on the same global network, resources are more efficiently used.

6. Substantial Bandwidth Savings:

Competing solutions do not afford substantial bandwidth savings to ISPs or Content Providers. Through the use of mirroring, it is possible to save bandwidth over certain links (i.e., between New York and Los Angeles). Without global hosting, however, most requests for content will still need to transit the Internet, thus incurring bandwidth costs. The inventive hosting framework saves substantial backbone bandwidth for ISPs that have their own backbones. Because content is distributed throughout the network and can be placed next to network exchange points, both ISPs and Content Providers experience substantial savings because backbone charges are not incurred for most content requests.

7. Instant Access to the Global Network:

Competing solutions require the Content Provider to choose manually a small collection of sites at which content will be hosted and/or replicated. Even if the ISP has numerous hosting sites in widely varied locations, only those sites specifically chosen (and paid for) will be used to host content for that Content Provider.

On the contrary, the global hosting solution of the present invention allows ISPs to offer their clients instant access to the global network of servers. To provide instant access to the global network, content is preferably constantly and dynamically moved around the network. For example, if a Content Provider adds content that will be of interest to customers located in Asia, the Content Provider will be assured that its content will be automatically moved to servers that are also located in Asia. In addition, the global hosting framework allows the content to be moved very close to end users (even as close as the user's building in the case of the Enterprise market).

8. Designed for Global ISPs and Conglomerates:

Most competing solutions are designed to be purchased and managed by Content Providers, many of whom are already consistently challenged and consumed by the administrative and operational tasks of managing a single server. The inventive hosting scheme may be deployed by a global ISP, and it provides a new service that can be offered to Content Providers. A feature of the service is that it minimizes the operational and managerial requirements of a Content Provider, thus allowing the Content Provider to focus on its core business of creating unique content.

9. Effective Control of Proprietary Database s a nd Confidential Information:

Many competing solutions require Content Providers to replicate their proprietary databases to multiple geographically distant sites. As a result, the Content Provider effectively loses control over its proprietary and usually confidential databases. To remedy these problems, the global hosting solution of the present invention ensures that Content Providers retain complete control over their databases. As described above, initial requests for content are directed to the Content Provider's central Web site, which then implements effective a nd controlled database access. Preferably, high-bandwidth, static parts for page requests are retrieved from the global hosting network.

10. Compatibility with Content Provider Software:

Many competing solutions require Content Provider s to utilize a specific set of servers and databases. These particular, non-uniform requirements constrain the Content Provider's ability to most effectively use new technologies, and may require expensive changes to a Content Provider's existing infrastructure. By eliminating these problems, the inventive global hosting architecture effectively interfaces between the Content Provider and the ISP, and it does not make any assumptions about the systems or servers used by the Content Provider. Furthermore, the Content Provider's systems can be upgraded, changed or completely replaced without modifying or interrupting the inventive architecture.

11. No Interference with Dynamic Content, Personalized Advertising or E-Commerce, and No stale content:

Many competing solutions (such as naive caching of all content) can interfere with dynamic content, personalized advertising and E-commerce and can serve the user with stale content. While other software companies have attempted to partially eliminate these issues (such as keeping counts on hits for all cached copies), each of these solutions causes a partial or complete loss of functionality (such as the ability to personalize advertising). On the contrary, the global hosting solution does not interfere with generation of dynamic content, personalized advertising or E-commerce, because each of these tasks preferably is handled by the central server of the Content Provider.

12. Designed for the Global Network:

The global hosting architecture is highly scaleable and thus may be deployed on a world-wide network basis.

The above-described functionality of each of the components of the global hosting architecture preferably is implemented in software executable in a processor, namely, as a set of instructions or program code in a code module resident in the random access memory of the computer. Until required by the computer, the set of instructions may be stored in another computer memory, for example, in a hard disk drive, or in a removable memory such as an optical disk (for eventual use in a CD ROM) or floppy disk (for eventual use in a floppy disk drive), or downloaded via the Internet or other computer network.

In addition, although the various methods described are conveniently implemented in a general purpose computer selectively activated or reconfigured by software, one of ordinary skill in the art would also recognize that such

methods may be carried out in hardware, in firmware, or in more specialized apparatus constructed to perform the required method steps.

Further, as used herein, a Web "client" should be broadly construed to mean any computer or component thereof directly or indirectly connected or connectable in any known or later-developed manner to a computer network, such as the Internet. The term Web "server" should also be broadly construed to mean a computer, computer platform, an adjunct to a computer or platform, or any component thereof. Of course, a "client" should be broadly construed to mean one who requests or gets the file, and "server" is the entity which downloads the file.

Having thus described our invention, what we claim as new and desire to secure by Letters Patent is set forth in the following claims:

1. A distributed hosting framework operative in a computer network in which users of client machines connect to a content provider server, the framework comprising:

a routine for modifying at least one embedded object URL of a web page to include a hostname pretended to a domain name and path;

a set of content servers, distinct from the content provider server, for hosting at least some of the embedded objects of web pages that are normally hosted by the content provider server;

at least one first level name server that provides a first level domain name service (DNS) resolution; and

at least one second level name server that provides a second level domain name service (DNS) resolution;

wherein in response to requests for the web page, generated by the client machines the web page including the modified embedded object URL is served from the content provider server and the embedded object identified by the modified embedded object URL is served from a given one of the content servers as identified by the first level and second level name servers.

2. The hosting ramework as described in claim 1 further including a redundant first level name server.

3. The hosting framework as described in claim 1 further including a redundant second level name server.

4. The hosting framework as described in claim 1 wherein a given one of the set of servers includes a buddy server for assuming the hosting responsibilities of the given one of the set of servers upon a given failure condition.

5. The hosting framework as described in claim 1 wherein the second level name server includes a load balancing mechanism that balances loads across a subset of the set of servers.

6. The hosting framework as described in claim 5 wherein the load balancing mechanism minimizes the amount of replication required for the embedded objects while not exceeding a capacity of any of the set of servers.

7. The hosting framework as described in claim 1 further including an overflow control mechanism for minimizing an overall amount of latency experienced by client machines while not exceeding the capacity of any given subset of the set of servers.

8. The hosting framework as described in claim 7 wherein the overflow control mechanism includes a min-cost multi-commodity flow algorithm.

9. The hosting framework as described in claim 1 wherein the first level name server includes a network map for use in directing a request for the embedded object generated by a client.

10. The hosting framework as described in claim 1 wherein a server in the set of servers includes a gating mechanism for maintaining overall traffic for a given embedded object within specified limits.

11. The hosting framework as described in claim 10 wherein the gating mechanism comprises:

means for determining whether a number of requests for the given embedded object exceeds a given threshold; and

means responsive to the determining means for restricting service of the given embedded object.

12. The hosting framework as described in claim 11 wherein the restricting means comprises means for serving an object that is smaller than the given embedded object.

13. The hosting framework as described in claim 11 wherein the object is a ticket that allows a client to receive the given embedded object at a later time.

14. A method of serving a page supported at a content provider server, the page comprising a markup language base document having associated therewith a set of embedded objects, each embedded object identified by a URL, comprising the steps of:

rewriting the URL of an embedded object to generate a modified URL, the modified URL including a new hostname prepended to an original hostname, wherein the original hostname is maintained as part of the modified URL for use in retrieving the embedded object whenever a cached copy of the embedded object is not available;

in response to a request to serve the page received at the content provider site, serving the page with the modified URL;

attempting to serve the embedded object from a content server other than the content provider server as identified by the new hostname; and

if the cached copy of the embedded object is not available from the content server, serving the embedded object from the content provider server.

15. A method of serving a page and an associated page object, wherein the page is stored on a content provider server and copies of the page object are stored on a set of content servers distinct from the content provider server, comprising the steps of:

(a) modifying a URL for the page object to include a hostname prepended to a content provider-supplied domain name and path;

(b) serving the page from the content provider server with the modified URL;

(c) responsive to a browser query to resolve the hostname, identifying a given one of the set of content servers from which the object may be retrieved; and

(d) returning to the browser an IP address of the identified content server to enable the browser to attempt to retrieve the object from that content server.

16. The method as described in claim 15 wherein the copies of the page object are stored on a subset of the set of content servers.

17. A content delivery method, comprising:

tagging an embedded object in a page to resolve to a domain other than a content provider domain by prepending given data to a content provider-supplied URL to generate an alternate resource locator (ARL);

serving the page from a content provider server with the ARL; and

resolving the ARt to identify a content server in the domain; and

serving the embedded object from the identified content server.

18. The method as described in claim 17 wherein the step of resolving the ARL comprises:

utilizing a requesting user's location and data identifying then-current Internet traffic conditions to identify the content server.

19. A content delivery service, comprising:

replicating a set of page objects across a wide area network of content servers managed by a domain other than a content provider domain;

for a given page normally served from the content provider domain, tagging the embedded objects of the page so that requests for the page objects resolve to the domain instead of the content provider domain;

responsive to a request for the given page received at the content provider domain, serving the given page from the content provider domain; and

serving at least one embedded object of the given page from a given content server in the domain instead of from the content provider domain.

20. The content delivery method as described in claim 19 wherein the serving step comprises:

for each embedded object, identifying one or more content servers from which the embedded object may be retrieved.

21. The method as described in claim 20 wherein the identifying step comprises:

resolving a request to the domain as a function of a requesting user's location.

22. The method as described in claim 21 wherein the identifying step comprises:

resolving a request to the domain as a function of a requesting user's location and then-current Internet traffic conditions.

23. A method for Internet content delivery, comprising:

at the content provider server, modifying at least one embedded object URL of a page to include a hostname prepended to a domain name and a path normally used to retrieve the embedded object;

responsive to a request for the page issued from a client machine, serving the page with the modified embedded object URL to the client machine from the content provider server;

responsive to a request for the embedded object, resolving the hostname to an IP address of a content server, other than the content provider server, that is likely to host the embedded object; and

attempting to serve the embedded object to the client from the content server.

24. The method as described in claim 23 wherein the hostname includes a value generated by applying a given function to the embedded object.

25. The method as described in claim 24 wherein the value is generated by encoding given information, the given information selected from a group of information consisting essentially of: size data, popularity data, creation data and object type data.

26. The method as described in claim 4 wherein the given function randomly associates the embedded object with a virtual content bucket.

27. The method as described in claim 26 wherein the given function is an encoding function.

28. The method as described in claim 26 wherein the given function is a hash function.

29. The method as described in claim 23 wherein the modified URL also includes a fingerprint value generated by applying a given function to the embedded object.

30. The method as described in claim 29 wherein the value is a number generated by hashing the embedded object.

31. The method as described in claim 23 wherein the page is formatted according to a markup language.

32. The method as described in claim 23 further including the step of rewriting the embedded object URL as the content provider modifies the page.

33. The method as described in claim 23 wherein the step of resolving the hostname includes:

identifying a subset of content servers that may be available to serve the embedded object based on a location of the client machine and current Internet traffic conditions; and

identifying the content server from the subset of content servers.

34. A content delivery method, comprising:

distributing a set of page objects across a network of content servers managed by a domain other than a content provider domain, wherein the network of content servers are organized into a set of regions;

for a given page normally served from the content provider domain, tagging at least some of the embedded objects of the page so that requests for the objects resolve to the domain instead of the content provider domain;

in response to a client request for an embedded object of the page:

resolving the client request as a function of a location of the client machine making the request and current Internet traffic conditions to identify a given region; and

returning to the client an IP address of a given one of the content servers within the given region that is likely to host the embedded object and that is not overloaded.

* * * * *

UNITED STATES PATENT AND TRADEMARK OFFICE

CERTIFICATE OF CORRECTION

PATENT NO. : 6,108,703
DATED : Aug. 22, 2000
INVENTOR(S) : F. Thomson Leighton, Daniel M. Lewin

It is certified that error appears in the above-identified patent and that said Letters Patent is hereby corrected as shown below:

Column 11, line 1, delete "assachusetts" and substitute -- Massachusetts --.
In Column 11, line 2, delete "ncorporated" and substitute -- incorporated --.
In Column 11, line 3, delete "ake" and substitute -- make --.
In Column 16, line 9, delete "Database s a nd" and substitute -- Databases and --.
In Column 16, line 20, delete "a nd" and substitute -- and --.
In Column 16, line 24, delete "Provider s" and substitute -- Providers --.
In Claim 1, Column 17, line 21, delete "pretended" and substitute -- prepended --.
In Claim 1, Column 17, line 32, after "machines" insert -- , --.
In Claim 17, Column 18, line 64, delete "ARt" and substitute -- ARL --.

Signed and Sealed this

Fifteenth Day of May, 2001

Attest:

Nicholas P. Godici

NICHOLAS P. GODICI

Attesting Officer

Acting Director of the United States Patent and Trademark Office

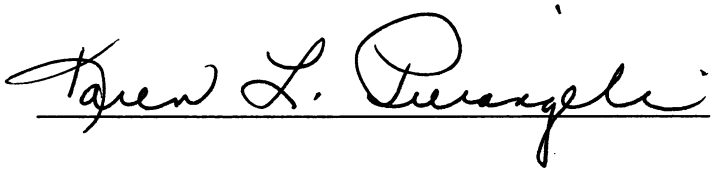
CERTIFICATE OF SERVICE

I hereby certify that on June 20, 2011, two true and correct copies of the foregoing **Principal Brief for Plaintiff-Appellant Akamai Technologies, Inc. On Rehearing En Banc** were served by the indicated means to the persons at the addresses listed:

Via Overnight Courier:

Robert G. Krupka
Kirkland & Ellis LLP
777 South Figueroa Street, Suite 3700
Los Angeles, CA 90017

Robert S. Frank, Jr.
Choate, Hall & Stewart LLP
Two International Place
Boston, MA 02110



CERTIFICATE OF COMPLIANCE

I certify that the foregoing **Principal Brief For Plaintiff-Appellant Akamai Technologies, Inc. On Rehearing En Banc** contains 13,693 words as measured by the word processing software used to prepare this brief.

Respectfully submitted,

Date: June 20, 2011

Donald R. Dunner
Kara F. Stoll
Finnegan, Henderson,
Farabow, Garrett & Dunner, LLP
901 New York Avenue, NW
Washington, DC 20001
Telephone: (202) 408-4000

Jennifer S. Swan
Finnegan, Henderson,
Farabow, Garrett & Dunner, LLP
3300 Hillview Ave.
Palo Alto, CA 94304