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.

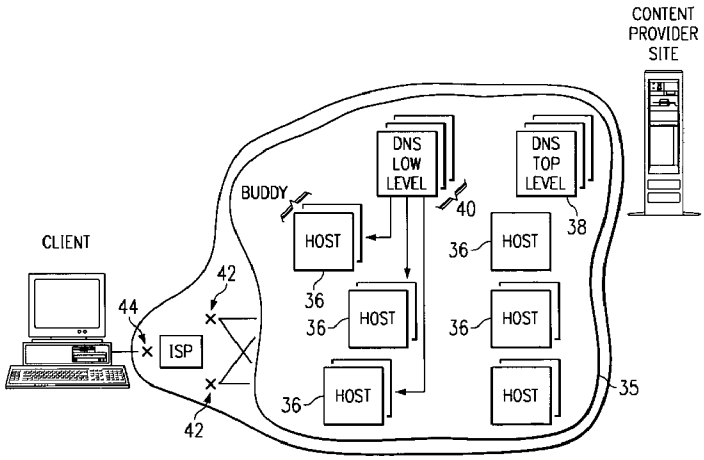
(List continued on next page.)

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.

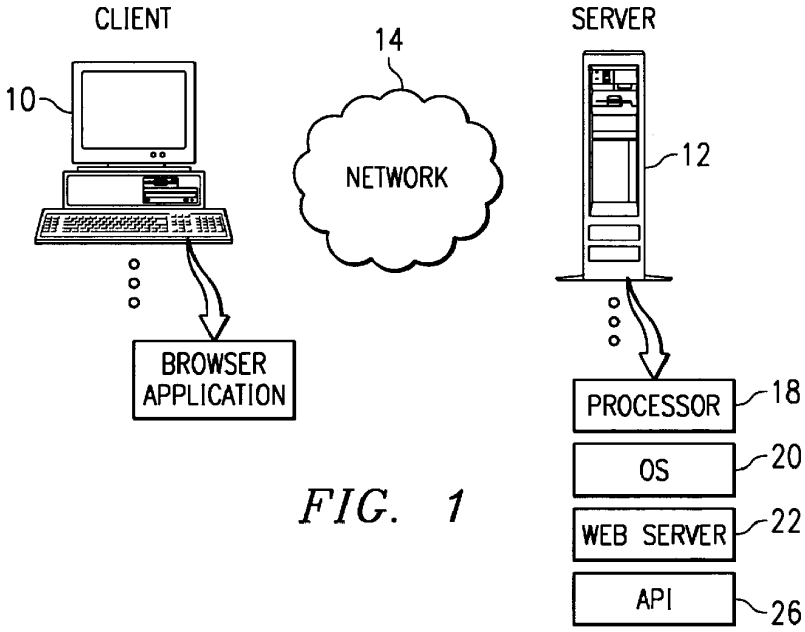


FIG. 1

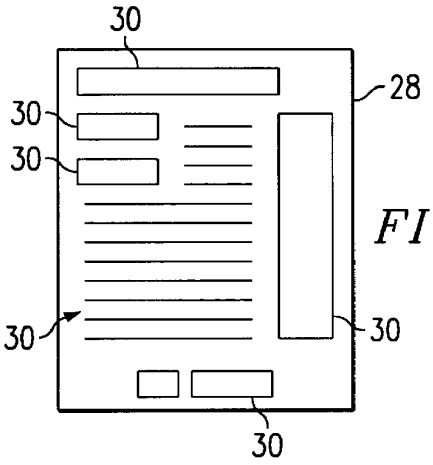


FIG. 2

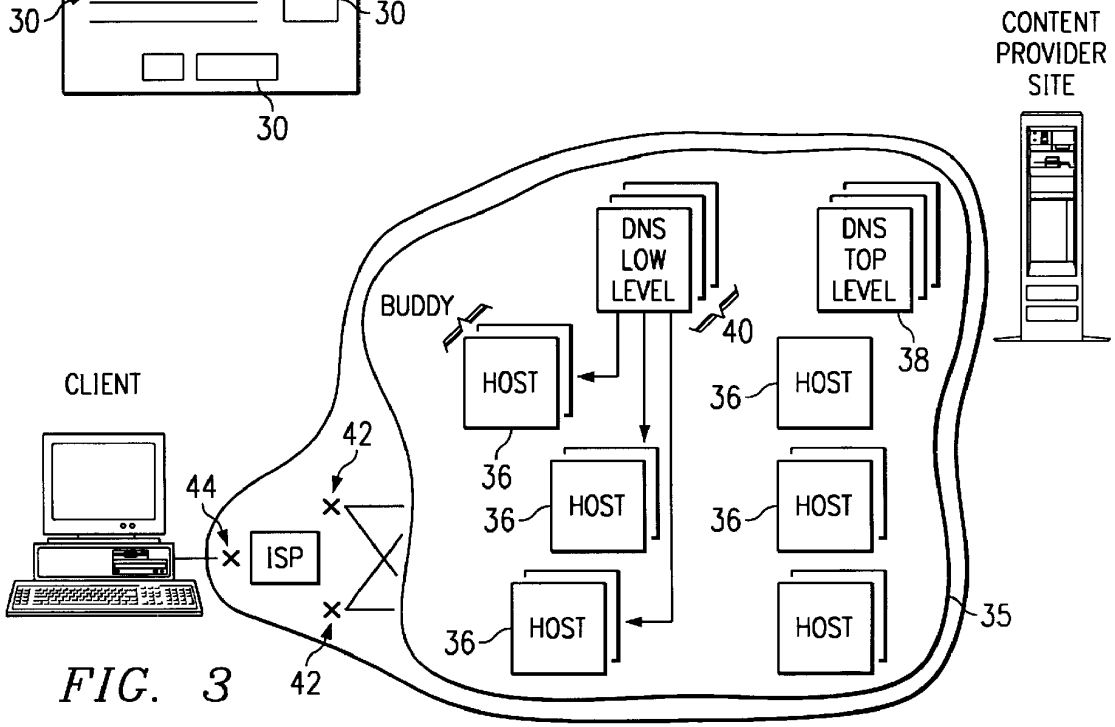


FIG. 3

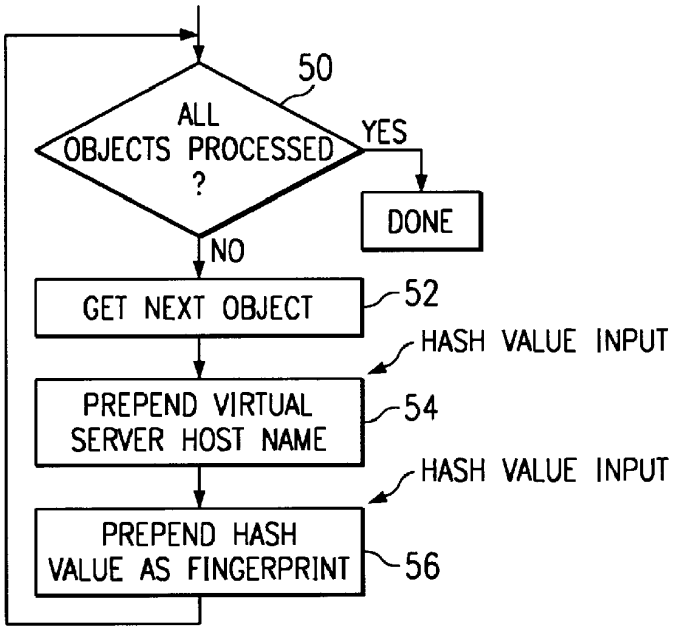


FIG. 4

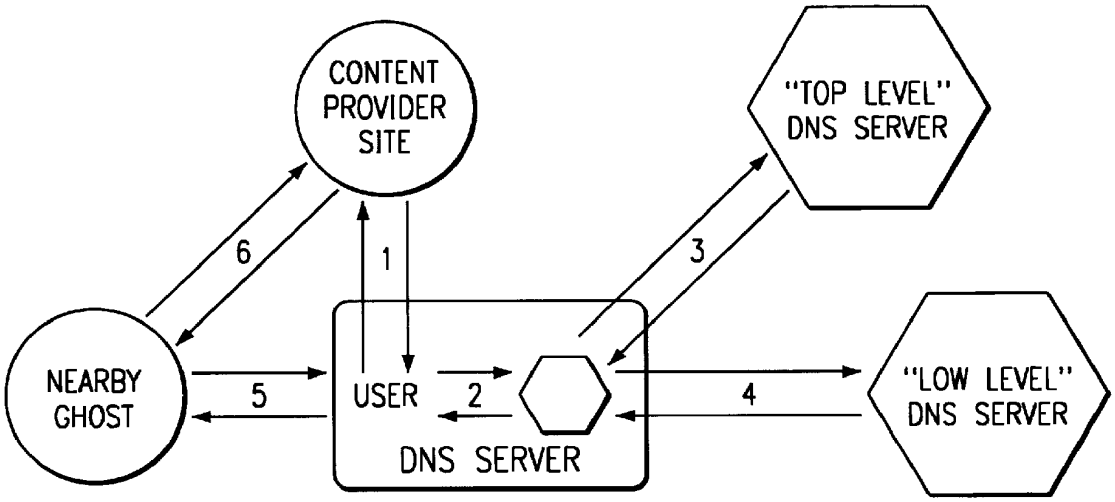


FIG. 5

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.

DOCKET ALARM

Explore Litigation Insights

Docket Alarm provides insights to develop a more informed litigation strategy and the peace of mind of knowing you're on top of things.

Real-Time Litigation Alerts

Keep your litigation team up-to-date with **real-time alerts** and advanced team management tools built for the enterprise, all while greatly reducing PACER spend.

Our comprehensive service means we can handle Federal, State, and Administrative courts across the country.

Advanced Docket Research

With over 230 million records, Docket Alarm's cloud-native docket research platform finds what other services can't. Coverage includes Federal, State, plus PTAB, TTAB, ITC and NLRB decisions, all in one place.

Identify arguments that have been successful in the past with full text, pinpoint searching. Link to case law cited within any court document via Fastcase.

Analytics At Your Fingertips

Learn what happened the last time a particular judge, opposing counsel or company faced cases similar to yours.

Advanced out-of-the-box PTAB and TTAB analytics are always at your fingertips.

API

Docket Alarm offers a powerful API (application programming interface) to developers that want to integrate case filings into their apps.

LAW FIRMS
Build custom dashboards for your attorneys and clients with live data direct from the court.

Automate many repetitive legal tasks like conflict checks, document management, and marketing.

FINANCIAL INSTITUTIONS
Litigation and bankruptcy checks for companies and debtors.

E-DISCOVERY AND LEGAL VENDORS
Sync your system to PACER to automate legal marketing.