



US005944780A

United States Patent [19]

Chase et al.

[11] Patent Number: 5,944,780

[45] Date of Patent: Aug. 31, 1999

[54] NETWORK WITH SHARED CACHING

[75] Inventors: **Jeffrey Scott Chase**; Syam Gadde, both of Durham, N.C.; **Michael Rabinovich**, Gillette, N.J.[73] Assignee: **AT&T Corp.**, New York, N.Y.[21] Appl. No.: **08/850,411**[22] Filed: **May 5, 1997**[51] Int. Cl.⁶ **G06F 15/16**[52] U.S. Cl. **709/201**; 709/212; 709/213

[58] Field of Search 395/200.43, 200.46, 395/200.47, 200.56, 200.57, 200.31, 200.33, 200.42; 711/147, 148; 707/10

[56] **References Cited****U.S. PATENT DOCUMENTS**

5,222,242	6/1993	Choi et al.	395/200.57
5,276,848	1/1994	Gallagher et al.	711/121
5,511,208	4/1996	Boyles et al.	395/800
5,715,395	2/1998	Brabson et al.	395/200.56

OTHER PUBLICATIONS

Radhika Malpani et al, "Making World Wide Web Caching Servers Cooperate," <http://www.w3.org/Conferences/WWW4/Papers/59/> Fourth International WWW Conference held in Boston, Mass., Dec. 1995.

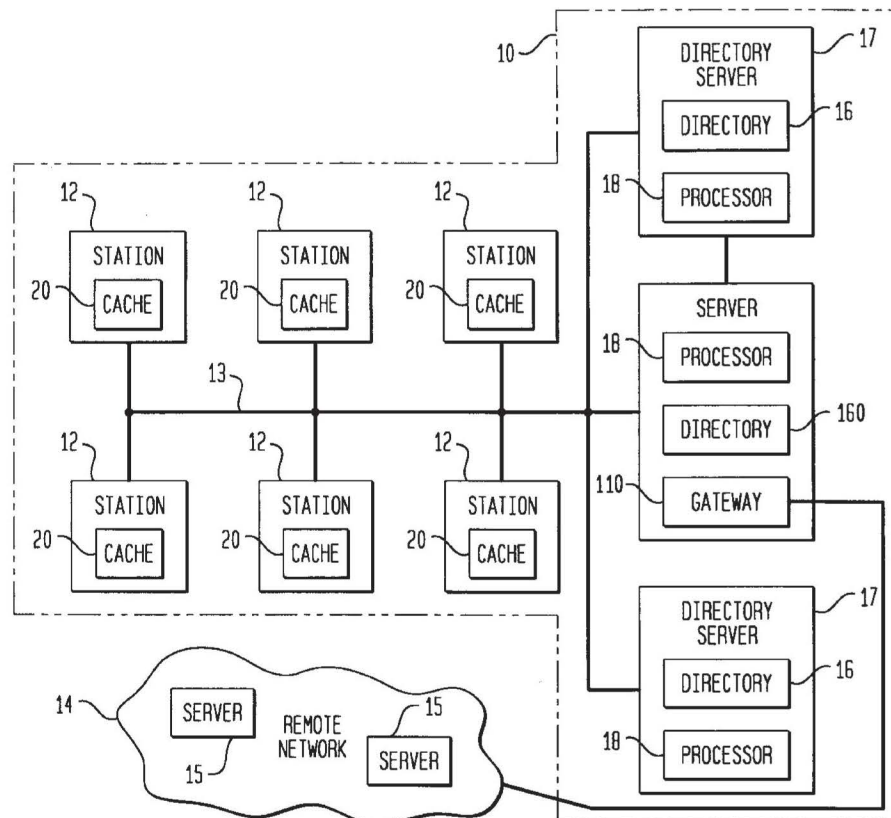
"Reduce, Reuse, Recycle: An Approach to Building large Internet Caches" by Syam Gadde, Michael Rabinovich and Jeff Chase, Jan. 15, 1997. The Sixth Workshop on Hot Topics in Operating Systems, IEEE Conference Proceedings, Cape Cod, MA, USA, 1997, 970505-970506.

"GeoPlex Convergence Software for Integrated Global and Enterprise Networks", Geosphere Communications, Inc., Technology Backgrounder, Sep., 1996.

Primary Examiner—Mehmet B. Geckil

[57] **ABSTRACT**

In a computer network system, the caches at individual stations are available to other stations. A central cache directory is maintained at a network server. Each time a station caches a data object received from a remote network, it informs the central cache directory. When a station comes online, it is asked to send a list of the contents of its cache. Whenever a station seeks an object from the remote network, the local network server first checks the central directory cache to see if the request can be satisfied at one of the local stations. Only if it cannot is the requested object retrieved from the remote network.

15 Claims, 3 Drawing Sheets

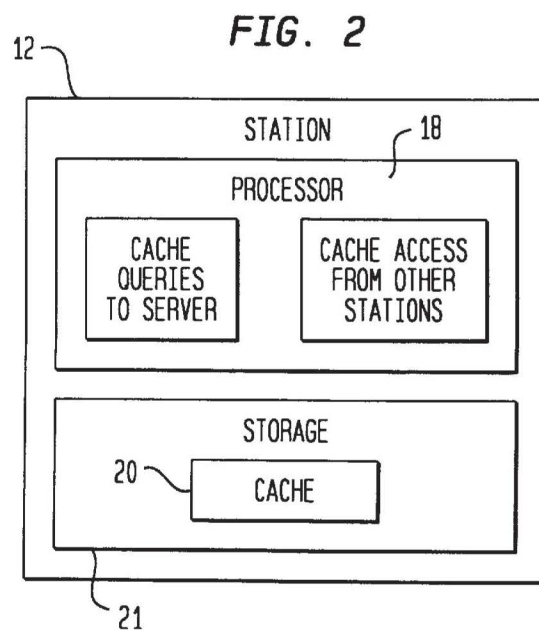
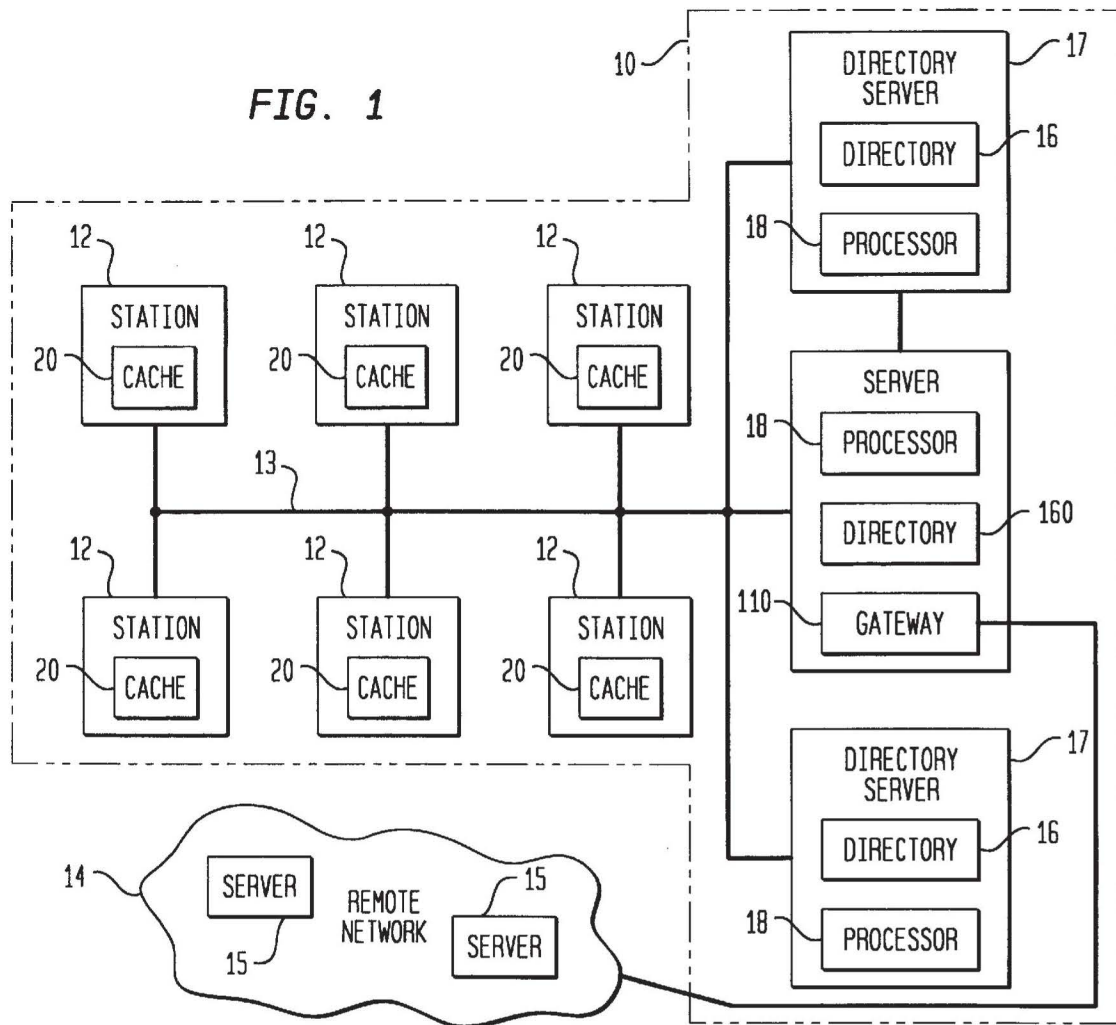


FIG. 3

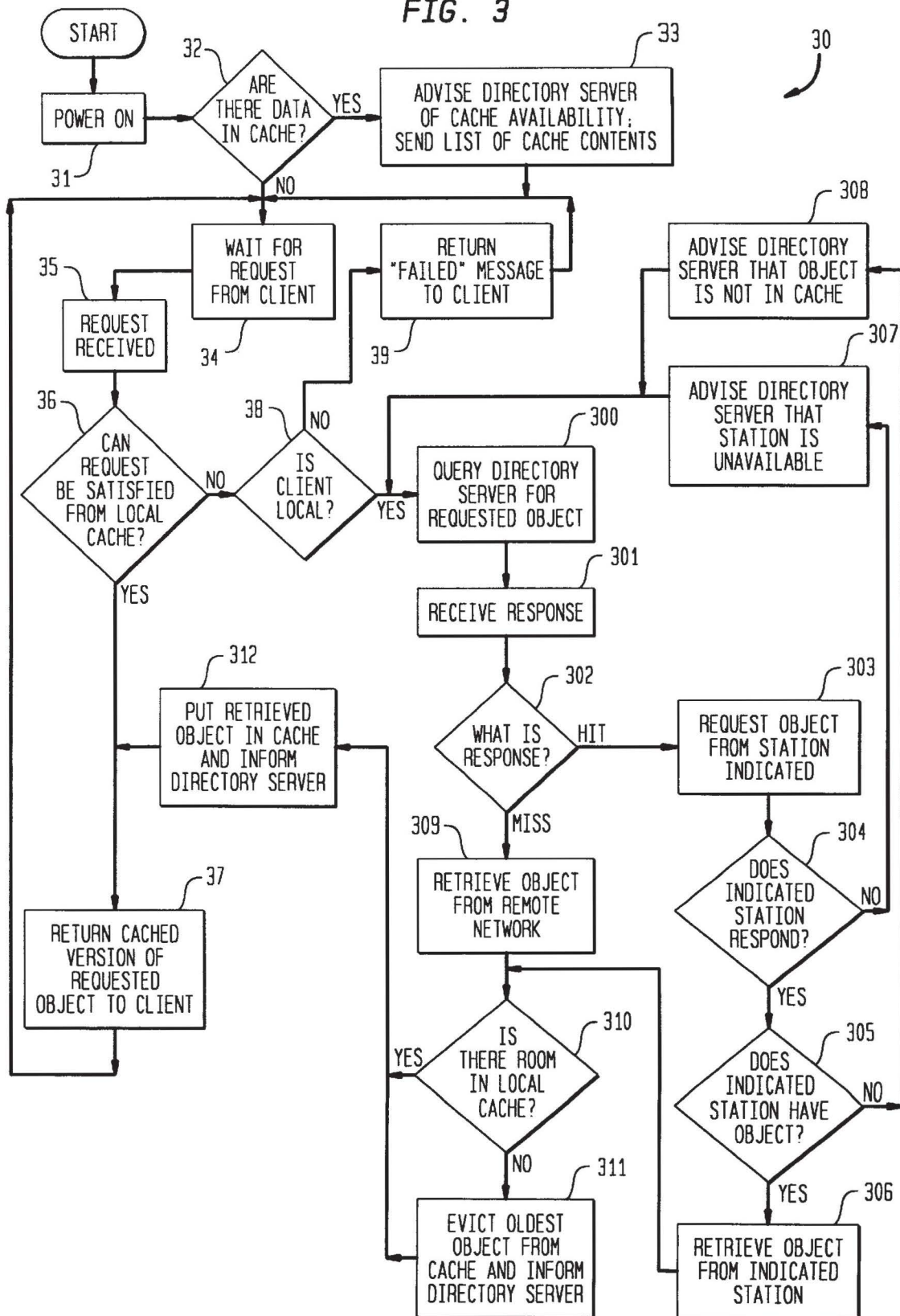
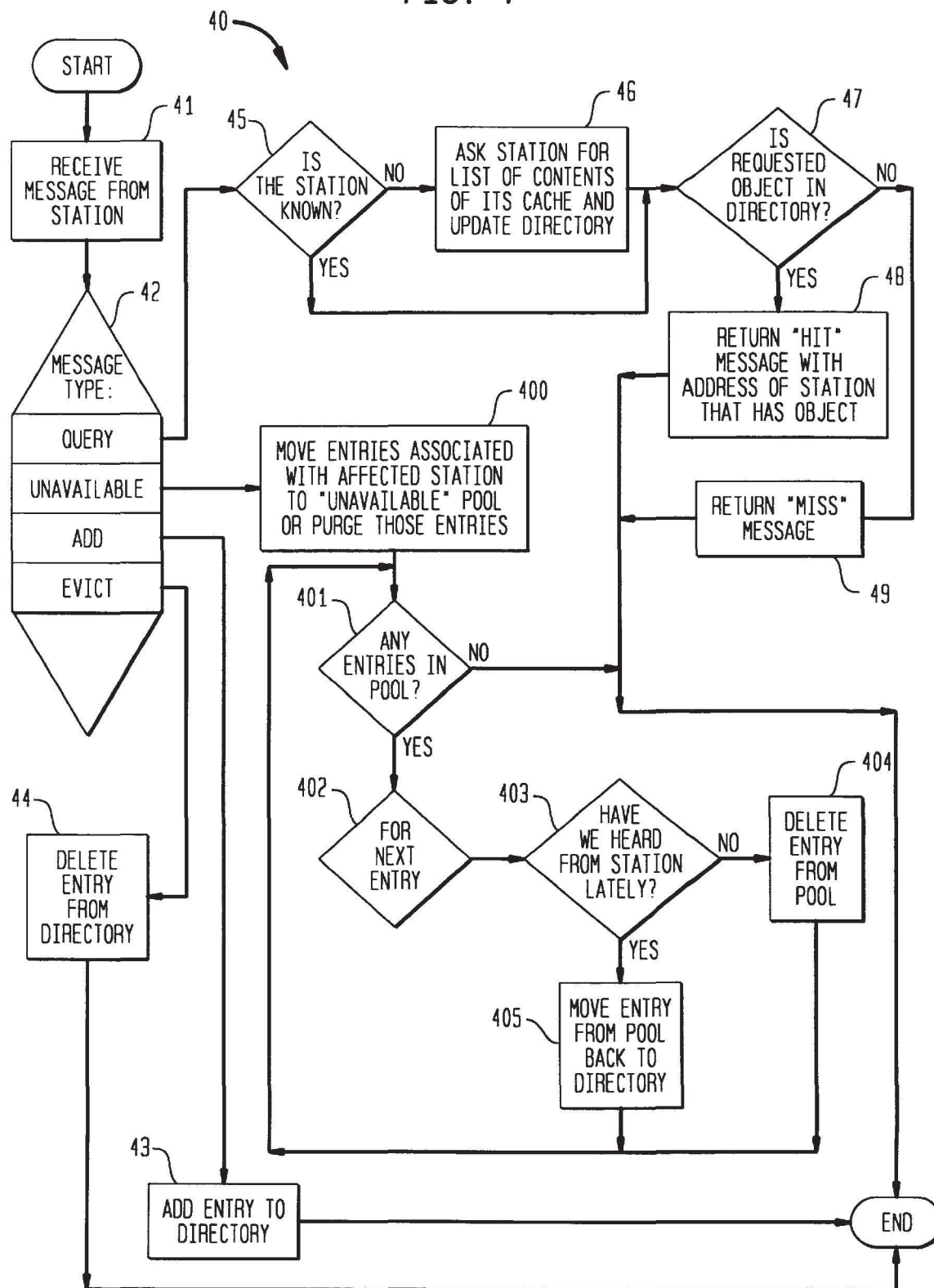


FIG. 4



NETWORK WITH SHARED CACHING

BACKGROUND OF THE INVENTION

This invention relates to computer networks in which a plurality of local stations that are networked together also communicate with an external database. More particularly, this invention relates to such a network in which each station can share data from the external database that has been cached by other stations.

In data networks such as the Internet, data is stored on servers interconnected by high-speed connections. Such networks support protocols, such as the Hypertext Transfer Protocol ("HTTP") used in the popular World Wide Web portion of the Internet, in which data is transmitted to users in a format known as a "page." Under the HTTP protocol, the user interface software (known as a "browser") cannot begin to display a page until a significant portion of the page has been received, and clearly cannot fully display the page until the entire page has been received. The resulting delays are referred to as "latency."

Unfortunately, many Internet users are connected to the Internet by relatively slow connections. Even where users are connected to a fast local-area network—e.g., a corporate "intranet"—the local-area network may be connected to the Internet using modems and standard telephone lines. Even the fastest commercially available telephone modems are limited to speeds of between 28.8 kilobits per second ("kbps") and 57.6 kbps. This limits the speed at which a World Wide Web page can be transmitted to a user and displayed by the user's browser. In addition, heavy user traffic, particularly heavy access by other users to the same server, also slow down the apparent speed of the World Wide Web. As a result, many users complain about the slow speed of the Internet in general, and the World Wide Web in particular. In fact, much of the latency perceived by users is the result of their relatively slow connection to, and heavy traffic on, what inherently ought to be a very fast network.

Currently available browser software makes some attempts to eliminate delays in receiving World Wide Web pages. For example, most browsers will store received pages in a disk cache. If the user asks for a page within a short time after having asked for it previously, the browser will retrieve the page from the cache. However, under the HTTP protocol, certain World Wide Web pages may not be cached, such as those that are dynamically generated. Therefore, current caching techniques are of limited usefulness in solving the latency problem.

More sophisticated, and therefore more useful, caching techniques can be employed in environments in which multiple users are connected to a local network that is connected to the Internet or other remote network—e.g., in a corporate in-house network or "intranet" that has a gateway to the Internet. In such environments it is known to have a central cache, either at the gateway or at a separate cache server. The central cache caches all pages or other data received from the remote network in response to a query by any user on the local network. If another user (or the same user again if for some reason the data are not cached at the user's station) requests the same data, the data can be delivered to that user from the central cache of the local network, without having to be retrieved from the remote network. Such an arrangement enhances the benefits of caching by making every user's cached data available to all other users on the local network, instead of being available only to the user who previously requested it.

However, arrangements such as that described have disadvantages. In order for the central cache to be useful, each

user request for data from the remote network must be routed through the central cache to see if it can be satisfied there. The central cache can thus become a bottleneck, slowing down each user's queries as they wait behind other users' queries while the central cache searches to see if it has the requested data for each query. In fact, an overloaded central cache may even drop user requests without servicing them.

As a further refinement of a system having a central cache, systems are known in which different groups of users are served by their own central caches, but the central caches cooperate. Thus, not only is the central cache queried when a user makes a request, but also, if the central cache associated with that user does not have the requested data, then before the data are requested from their home site on the remote network, the other central caches serving other groups in the system are queried first. Although in such a system, the likelihood is greater that some cache will contain the requested data, avoiding the need to retrieve them from their home site, at some point it ceases to be efficient if too many cache servers have to be queried. Moreover, each time a cache server receives a query from another cache server, it adds to the delay in processing requests at the first cache server from its own associated users.

In another known system, objects in a network file system are cached at individual user stations of the network. However, such a system, designed for data that reside within the local-area network and that may be modified by users, entails complex mechanisms to maintain coherency of cached data and data availability in the case of failure of a station. These mechanisms provide only limited performance improvement (and may even degrade performance), and impose significant overhead that is unnecessary in an Internet access environment.

It would be desirable to be able to provide a system having the benefits of central caching without the bottleneck caused by a central cache server.

SUMMARY OF THE INVENTION

It is an object of the present invention to provide a system having the benefits of central caching without the bottleneck caused by a central cache server.

In accordance with this invention, there is provided a computer system having a plurality of local stations, a communications channel interconnecting the plurality of local stations, and a communications link connecting the communications channel to an external database. Each respective one of at least a first subset of the local stations has its own respective cache memory for caching data retrieved by that respective local station from the external database. The computer system further has a central directory unit for maintaining a directory of data cached in the cache memories of respective ones of the first subset of local stations. When a respective one of a second subset of local stations requires data from the external database, a cache query unit in that respective one of the second subset of the local stations queries the directory to determine whether the required data are cached in a respective cache memory of the first subset of local stations. A cache access unit in each respective one of the first subset of the local stations allows access to the respective cache memories by respective ones of the second subset of local stations.

BRIEF DESCRIPTION OF THE DRAWINGS

The above and other objects and advantages of the invention will be apparent upon consideration of the fol-

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.