

EXHIBIT C8

Summary of Invalidity Analysis of U.S. Patent No. 6,771,646 (“’646 Patent”) in view of U.S. Patent No. 6,412,000 (“Riddle”), further in view of WO 92/19054 (“Ferdinand”), further in view of U.S. Patent No. 5,740,175 (“Wakeman”), and further in view of RFC 1945 - Hypertext Transfer Protocol -- HTTP/1.0 (“RFC1945”)

U.S. Patent No. 6,412,000, issued on June 25, 2002, qualifies as prior art to the ’646 Patent under at least 102(e) because it was filed on November 23, 1998, before the June 30, 1999 filing date of the provisional application to which the ’646 Patent claims priority. Riddle further qualifies as prior art to the ’646 Patent under at least Pre-AIA 35 U.S.C. § 102(e) because U.S. patent has an effective prior art date under pre-AIA 35 U.S.C. § 102(e) based on the filing date of an earlier-filed application if the patent’s relevant subject matter is described in the earlier-filed application, and at least one of the claims is supported by the earlier-filed application’s written description in compliance with pre-AIA 35 U.S.C. § 112, first sentence. Riddle was the first application that issued as Riddle was filed on November 23, 1998. Riddle claims priority to U.S. Provisional Patent Application No. 60/066,864 (“’864 Provisional”), which was filed on November 25, 1997.

Riddle and the related ’864 Provisional incorporate-by-reference the following patent applications in their entirety:

- U.S. Patent Application No. 09/198,051 (“’051 Application”);
- U.S. Patent Application No. 08/762,828, issued as U.S. Patent No. 5,802,106;
- U.S. Patent Application No. 08/977,642 (“Packer Application”), having attorney docket number 08-00000001, issued as U.S. Patent No. 6,046,980 (“Packer”); and
- U.S. Patent Application No. 08/742,994, issued as U.S. Patent No. 6,038,216.

WO 92/19054 (“Ferdinand”), published on October 29, 1992, qualifies as prior art to the ’646 Patent under at least Pre-AIA 35 U.S.C. § 102(b) because it was published more than one year before the June 30, 1999 filing date of the provisional application to which the ’646 Patent claims priority.

U.S. Patent No. 5,740,175 (“Wakeman”), published on April 14, 1998, qualifies as prior art to the ’646 Patent under at least Pre-AIA 35 U.S.C. § 102(b) because it was published more than one year before the June 30, 1999 filing date of the provisional application to which the ’646 Patent claims priority.

RFC 1945 - Hypertext Transfer Protocol -- HTTP/1.0 (“RFC1945”), published in March 1996, qualifies as prior art to the ’646 Patent under at least Pre-AIA 35 U.S.C. § 102(b) because it was published more than one year before the June 30, 1999 filing date of the provisional application to which the ’646 Patent claims priority.

EXHIBIT C8

Invalidity of U.S. PATENT NO. 6,771,646 in view of Riddle et al.	
CLAIM LANGUAGE	Exemplary Citations to Riddle et al.
INDEPENDENT CLAIM 1	
1	<p>A packet monitor for examining packets passing through a connection point on a computer network, each packet conforming to one or more protocols, the monitor comprising:</p> <p>U.S. Patent No. 6,412,000 (“Riddle”) discloses a packet monitor for examining packets passing through a connection point on a computer network, each packet conforming to one or more protocols.</p> <p>For example:</p> <p>“In a packet communication environment, a method is provided for automatically classifying packet flows for use in allocating bandwidth resources and the like by a rule of assignment of a service level. The method comprises applying individual instances of traffic classification paradigms to packet network flows based on selectable information obtained from a plurality of layers of a multi-layered communication protocol in order to define a characteristic class, then mapping the flow to the defined traffic class. It is useful to note that the automatic classification is sufficiently robust to classify a complete enumeration of the possible traffic.” Riddle, 4:6-17.</p> <p>“3.2 Automatic Traffic Classification Processing FIG. 3 depicts components of a system for automatically classifying packet flows for use in allocating bandwidth resources and the like by a rule of assignment of a service level. A traffic tree 302 in which new traffic will be added to a particular member class node. A traffic classifier 304 detects incoming traffic. Alternatively, the classifier may start with a service assignment and use it. A knowledge base 306 contains heuristics for determining a traffic class. The knowledge base may be embodied in a file or a relational database.” Riddle, 4:6-17.</p>

EXHIBIT C8

embodiment, the knowledge is contained within a data structure. A plurality of saved lists 308 stores classified traffic pending in a tree 302. In select embodiments, entries for each instance of an alternate embodiment, a copy of an entry and a count of duplicates is maintained.” Riddle, 12:27-41.

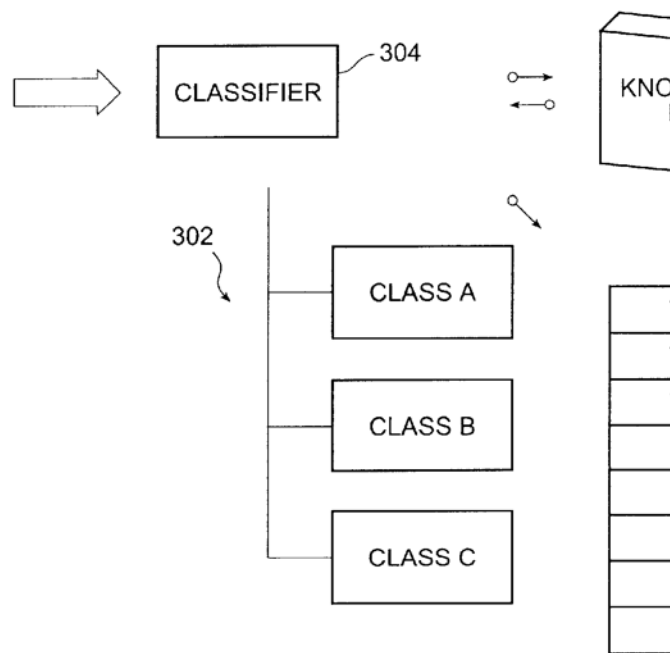


FIG. 3

Riddle, Fig. 3.

“The method for automatically classifying heterogeneous packet traffic in a telecommunications environment of the present invention is implemented in a programming language and is operational on a computer system 1A. This invention may be implemented in a client-server environment. A client-server environment is not essential. This figure shows a conventional

EXHIBIT C8

computer system which includes a server 20 and numerous clients shown as client 25. The use of the term "server" is used in the context wherein the server receives queries from (typically remote) clients and performs all the processing necessary to formulate responses to the queries and return responses to the clients. However, server 20 may itself act as a client when it accesses remote databases located at another node of the network.

The hardware configurations are in general standard and will be understood by those skilled in the art. In accordance with known practice, server 20 includes one or more processors which communicate with a number of peripheral devices via a system bus. These peripheral devices typically include a Storage Subsystem, a memory subsystem 35a and a file storage subsystem 35b holding programs (e.g., code or instructions) and data, a set of user interface input devices 37 and an interface to outside networks, which may employ Ethernet, IEEE 802.3, ITU X.25, Serial Link Internet Protocol (SLIP) or a telephone network. This interface is shown schematically as block 40. It is coupled to corresponding interface devices in other computers in the network connection 45." Riddle, 5:53-6:15.

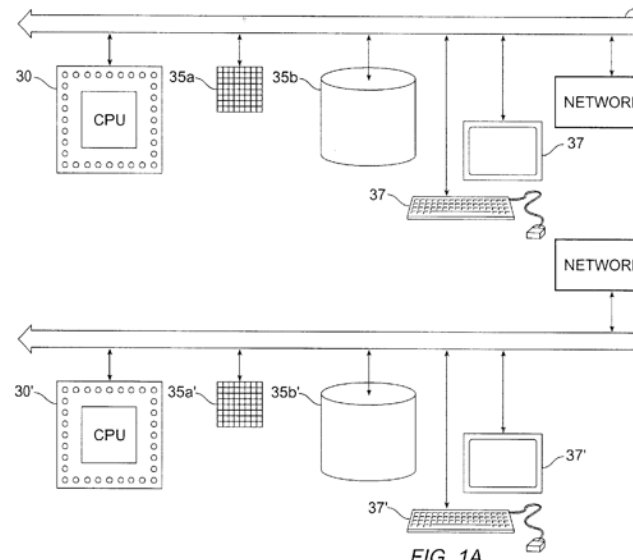


FIG. 1A (PRIOR ART)

EXHIBIT C8

Riddle, Fig. 1A.

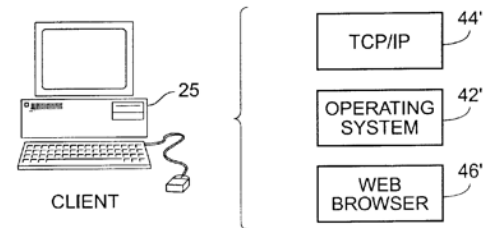
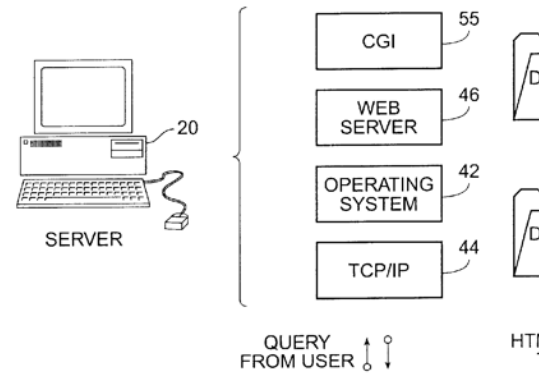


FIG. 1B
(PRIOR ART)

Riddle, Fig. 1B.

“FIG. 1C is illustrative of the internetworking of a plurality of servers such as server 20 of FIGS. 1A and 1B and a plurality of clients such as client 25 of FIGS. 1A and 1B and a plurality of servers such as server 20 described herein above. In FIG. 1C, network 60 is an example of a network that connects the servers and clients.”

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.