



[54] **METHOD AND APPARATUS FOR FAST HIERARCHICAL ADDRESS LOOKUP USING CONTROLLED EXPANSION OF PREFIXES**

Mills, Don, *The Art of Computer Programming, Volume 3/Sorting and Searching*, pp. 481-499.

[75] Inventors: **George Varghese**, St. Louis;
Srinivasan Venkatachary, University
City, both of Mo.

Primary Examiner—Min Jung
Attorney, Agent, or Firm—Howell & Haferkamp, L.C.

[73] Assignee: **Washington University**, St. Louis, Mo.

[57] **ABSTRACT**

[21] Appl. No.: **08/821,100**

[22] Filed: **Mar. 20, 1997**

[51] **Int. Cl.**⁷ **H04L 12/46**

[52] **U.S. Cl.** **370/392; 370/401; 395/200.75**

[58] **Field of Search** 370/392, 393,
370/401, 402, 403, 404, 405, 466, 467;
395/200.68, 200.75

Many network protocols, including the Internet, have addresses that are structured hierarchically. The hierarchy is expressed by an address prefix P that represents all addresses in the given hierarchical level that starts with prefix P. The hierarchy is not strict and can be overridden by more inclusive hierarchies. This is achieved by having network routers find the longest prefix that matches a destination address in a message.

[56] **References Cited**

U.S. PATENT DOCUMENTS

| | | | |
|-----------|---------|-----------------------|---------|
| 4,897,814 | 1/1990 | Clark | 365/49 |
| 5,247,620 | 9/1993 | Fukuzawa et al. | 370/402 |
| 5,261,090 | 11/1993 | Lien | 395/600 |
| 5,313,465 | 5/1994 | Perlman et al. | 370/403 |
| 5,386,413 | 1/1995 | McAuley et al. | 370/54 |
| 5,414,704 | 5/1995 | Spinney | 370/60 |
| 5,446,887 | 8/1995 | Berkowitz | 395/600 |
| 5,459,717 | 10/1995 | Mullan et al. | 370/54 |
| 5,761,440 | 6/1998 | De Marco et al. | 370/392 |
| 5,781,772 | 7/1998 | Wilkinson, III et al. | 370/229 |
| 5,842,224 | 11/1998 | Fenner | 370/392 |
| 5,856,974 | 1/1999 | Gervais et al. | 370/401 |

The disclosed invention describes a method and apparatus for implementing controlled expansion: for expanding a set of prefixes into an equivalent (possibly larger) set of prefixes that have a smaller set of prefix lengths. The new set of prefixes can then be looked up significantly faster using any technique whose speed improves by reducing the number of prefix lengths. Our invention also incorporates fast techniques to insert and delete new prefixes, and a technique of pointer hoisting to halve the memory READs needed for trie search. Our solution to longest matching prefix also applies to other routing protocols such as OSI Routing, call routing in telephone networks, and to string matching problems.

OTHER PUBLICATIONS

Perlman, Radia, 9.7 Address Matching, *Interconnection Bridges and Routers*, pp. 233-239.

Wright, Gary R., Radix Tree Routing Tables, *TCP/IP Illustrated, Volume 2, The Implementation*, pp. 559-569.

20 Claims, 10 Drawing Sheets

| | Prefix | Link |
|-----------|---------------|-------------|
| P1 | 01* | L1 |
| P2 | 1* | L2 |
| P3 | 10* | L3 |
| P4 | 101 * | L2 |
| P5 | 100 * | L6 |
| P6 | 10111 * | L2 |

| Old Prefix | New Prefix | Link |
|-------------------|-------------------|-------------|
| P2 | 110* | L2 |
| P2 | 111* | L2 |
| P1 | 010* | L1 |
| P1 | 011* | L1 |
| P4 | 101* | L2 |
| P5 | 100* | L6 |
| P6 | 101110* | L7 |
| P6 | 101111* | L7 |

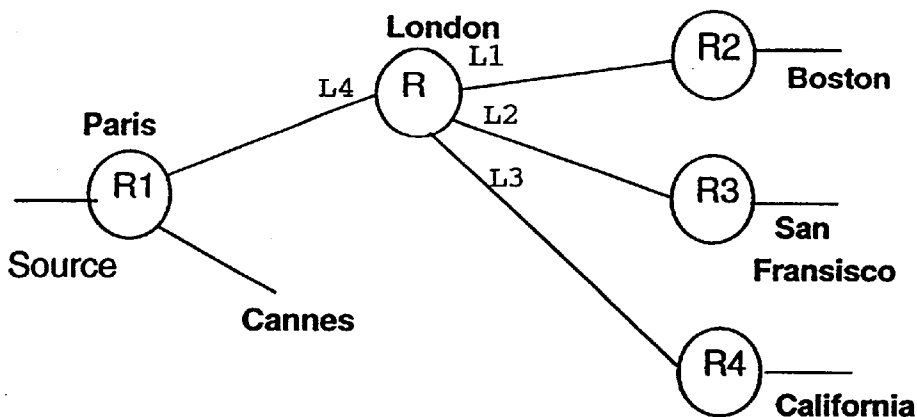


FIG. 1

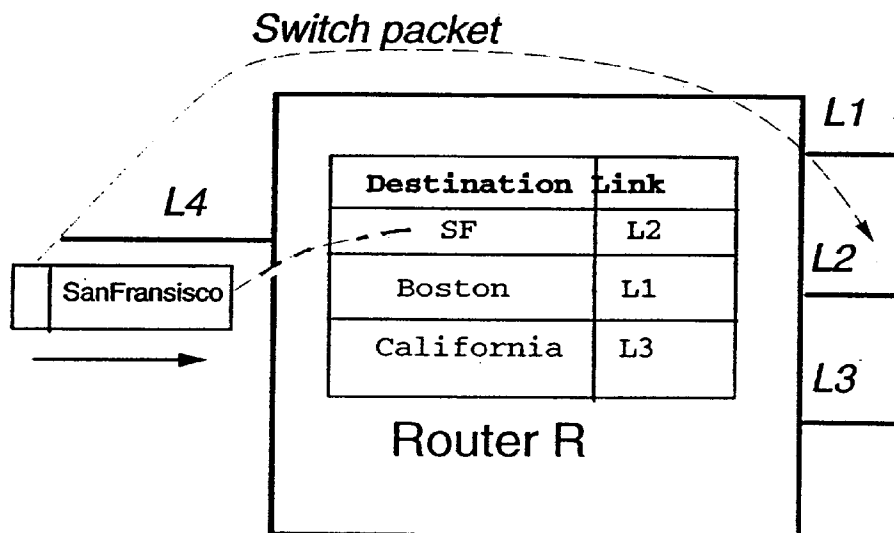


FIG. 2

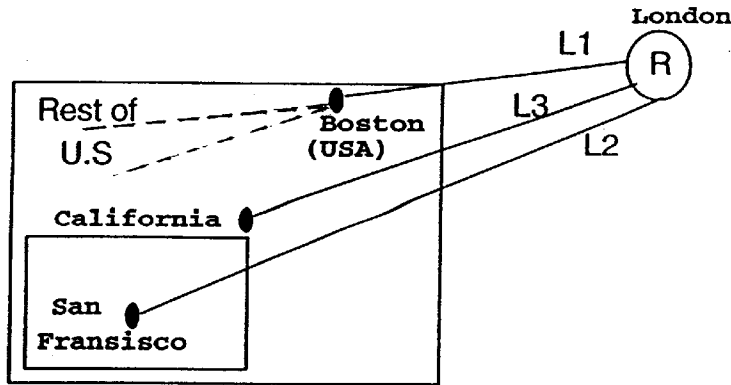


FIG. 3

| Destination | Link |
|---------------|------|
| USA.CA.SF | L2 |
| USA | L1 |
| USA.CA | L3 |
| USA.MA.Boston | L2 |

FIG. 4

| | Prefix | Link |
|----|---------|------|
| P1 | 01* | L1 |
| P2 | 1* | L2 |
| P3 | 10* | L3 |
| P4 | 101 * | L2 |
| P5 | 100 * | L6 |
| P6 | 10111 * | L2 |

FIG. 5

| | | |
|----------|-----------|---------|
| Length 1 | 1* L2 | |
| Length 2 | 10* L3 | 01* L1 |
| Length 3 | 101* L2 | 100* L2 |
| Length 5 | 10111* L7 | |

FIG. 6

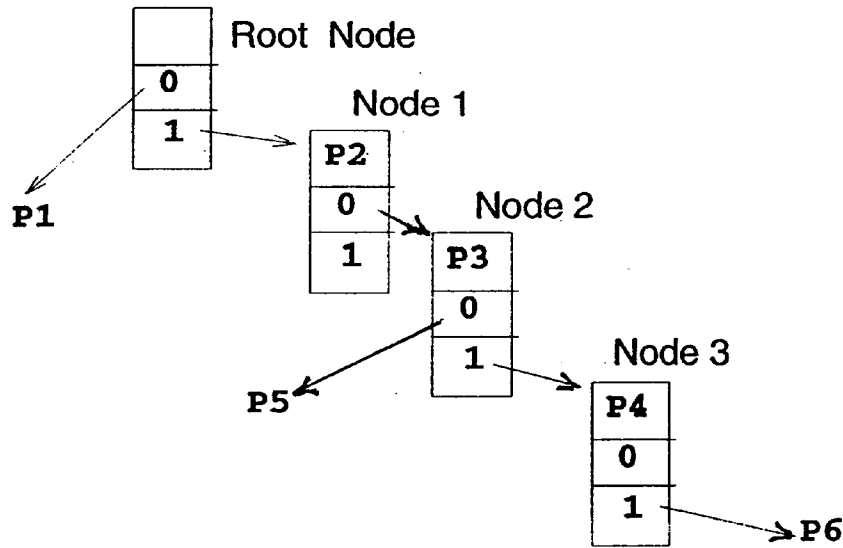


FIG. 7

| Old Prefix | New Prefix | Link |
|------------|------------|------|
| P2 | 110* | L2 |
| P2 | 111* | L2 |
| P1 | 010* | L1 |
| P1 | 011* | L1 |
| P4 | 101* | L2 |
| P5 | 100* | L6 |
| P6 | 101110* | L7 |
| P6 | 101111* | L7 |

FIG. 8

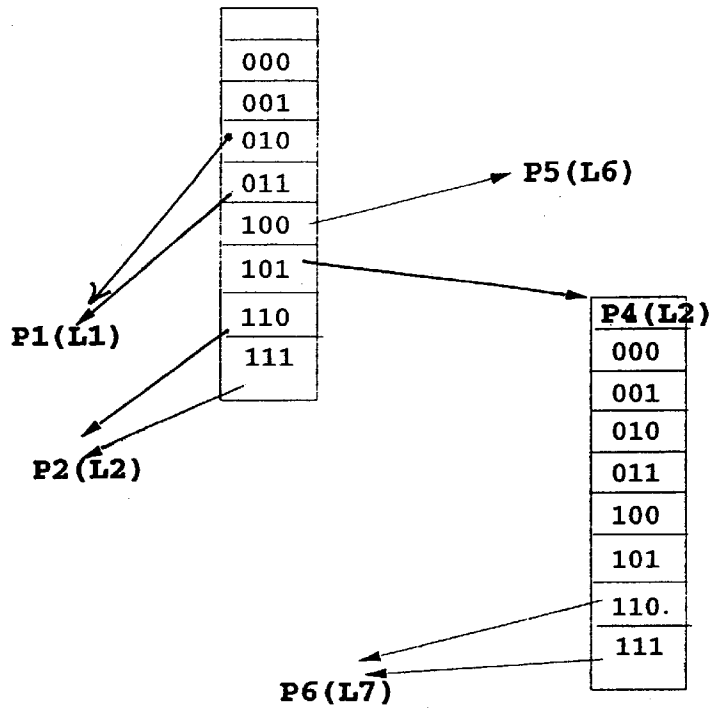


FIG. 9

Pointer BestPrefix

| | |
|-----|--------|
| 000 | |
| 001 | |
| 010 | P1(L1) |
| 011 | P1(L1) |
| 100 | P5(L6) |
| 101 | P4(L2) |
| 110 | P2(L2) |
| 111 | P2(L2) |

Pointer BestPrefix

| | |
|-----|--------|
| 000 | |
| 001 | |
| 010 | |
| 011 | |
| 100 | |
| 101 | |
| 110 | P6(L7) |
| 111 | P6(L7) |

FIG. 10

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.