US006711562B1

(12) **United States Patent**
Ross et al.

(10) **Patent No.: US 6,711,562 B1**
(45) **Date of Patent: Mar. 23, 2004**

(54) **CACHE SENSITIVE SEARCH (CSS) TREE INDEXING SYSTEM AND METHOD**

(75) Inventors: **Kenneth A. Ross**, New York, NY (US); **Jun Rao**, New York, NY (US)

(73) Assignee: **The Trustees of Columbia University in the City of New York**, New York, NY (US)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 0 days.

(21) Appl. No.: **09/600,266**

(22) PCT Filed: **Dec. 1, 1999**

(86) PCT No.: **PCT/US99/28430**

§ 371 (c)(1),
(2), (4) Date: **Feb. 27, 2002**

(87) PCT Pub. No.: **WO01/40996**

PCT Pub. Date: **Jun. 7, 2001**

(51) **Int. Cl.**[7] **G06F 17/30**

(52) **U.S. Cl.** **707/3**; 707/2; 707/200
(58) **Field of Search** 707/1, 2, 3, 100, 707/200, 205; 711/216, 100, 133, 141

(56) **References Cited**

U.S. PATENT DOCUMENTS

5,283,894 A		2/1994	Deran	707/1
5,664,184 A	*	9/1997	Ferguson et al.	707/103 R
5,668,987 A	*	9/1997	Schneider	707/3
5,758,356 A	*	5/1998	Hara et al.	707/202
5,822,749 A		10/1998	Agarwal	707/2
5,826,253 A		10/1998	Bredenberg	707/2
5,940,838 A		8/1999	Schmuck et al.	707/200
6,047,280 A		4/2000	Ashby et al.	707/2
6,061,678 A		5/2000	Klein et al.	707/3
6,266,660 B1	*	7/2001	Liu et al.	707/3
6,408,362 B1	*	6/2002	Arimilli et al.	711/133
6,578,131 B1	*	6/2003	Larson et al.	711/216

OTHER PUBLICATIONS

"Cache Conscious Indexing for Decision–Support in Main Memory", Proceedings of the 25th International Conference on Very Large Data Bases, Jun Rao and Kenneth A. Ross, Sep. 7–10, 1999, Edinburgh, Scotland, UK, pp. 78–89.
"Cache Conscious Indexing for Decision–Support in Main Memory", Columbia University Technical Report CUCS–019–98, Jun Rao and Kenneth A. Ross, Dec. 1, 1998, pp. 0–17.
"Making B^+–Tree Cache Conscious in Main Memory", ACM SIGMOD 2000, May 2000, Jun Rao and Kenneth Ross, pp. 475–486.
Lehman et al., "A Recovery Algorithm for A High–Performance Memory–Resident Database System", Proceedings of the ACM SIGMOD Conference, pp. 104–117, 1987.

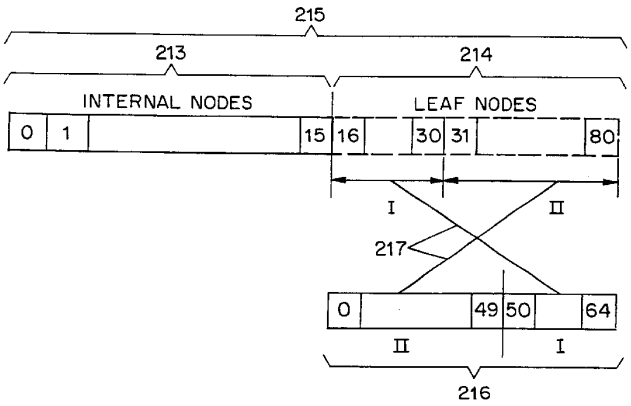
(List continued on next page.)

Primary Examiner—Charles Rones
Assistant Examiner—Neveen Abel-Jalil
(74) *Attorney, Agent, or Firm*—Baker Botts LLP

(57) **ABSTRACT**

Cache sensitive search tree (CSS-tree) index structures for providing improved search and lookup performance compared with conventional searching schemes. The CSS-tree index structures include a directory tree structure which is stored in an array **(216)** and serves as an index for a sorted array of elements. The nodes **(215)** in the directory tree structure may be of sizes selected to correspond to the cache line size in the computer system utilizing the CSS-tree index structures. Child nodes **(213)** within the directory tree structure are located by performing arithmetic operations on array offsets. Thus, it is not necessary to store internal child node pointers, thereby reducing memory storage requirements. In addition, the CSS-tree index structures are organized so that traversing each level in the tree yields good data reference locality, and therefore relatively few cache misses. Thus, the CSS-tree index structures consider cache-related parameters such as reference locality and cache behavior, without requiring substantial additional amounts of memory.

44 Claims, 9 Drawing Sheets



OTHER PUBLICATIONS

LaMarca et al., "The Influence of Caches on the Performance of Heaps", ACM Journal of Experimental Algorithmics, 1996.

COMPAQ, "InfoCharger Engine: Optimization for Decision Support Solutions", 1998.

Chilimbi et al., "Improving Pointer–Based Codes Through Cache–Conscious Data Placement"., Technical Report '98, University of Wisconsin–Madison, Computer Science Department, 1998.

TimesTen Performance Software White Paper, "Architected for Real–Time Data Management; TimesTen's Core In–Memory Database Technology", (Revised Version) 1997.

Sun Microsystems, "UltraSPARC™ User's Manual", UltraSPARC–1 UltraSPARC–II, Jul. 1997.

Phil Bernstein et al., "The Asilomar Report on Database Research," *ACM Sigmod Record*, 27 (4), 1998.

Sybase Corporation, *Sybase I.Q. 11.2.1*, 1997.

* cited by examiner

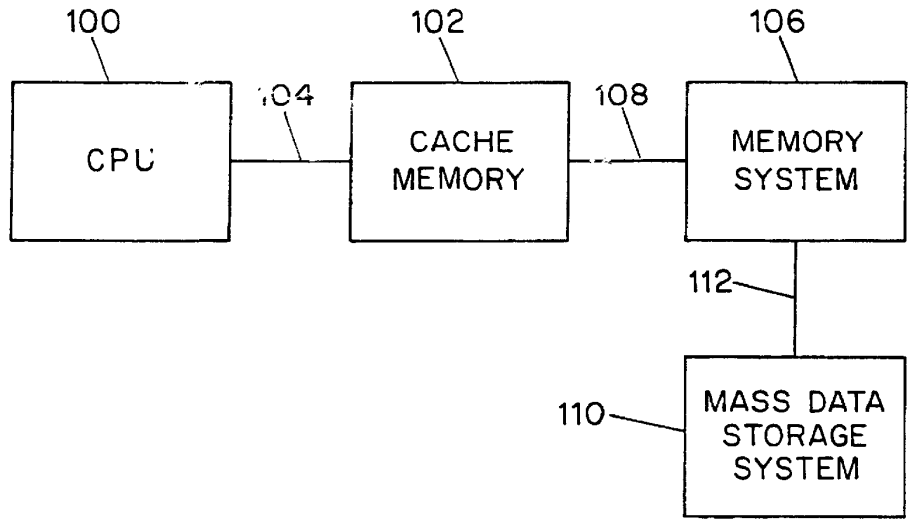


FIG. 1
(PRIOR ART)

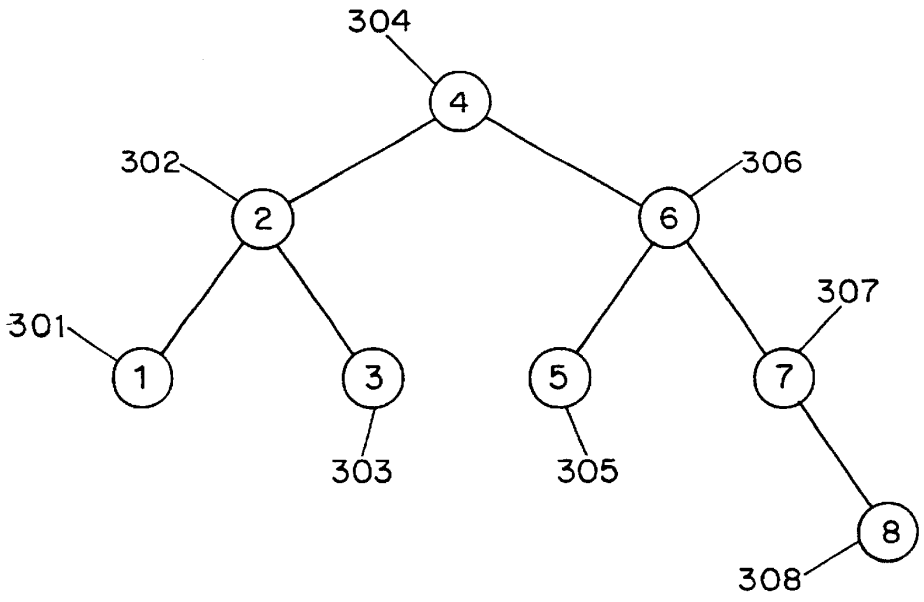


FIG. 3

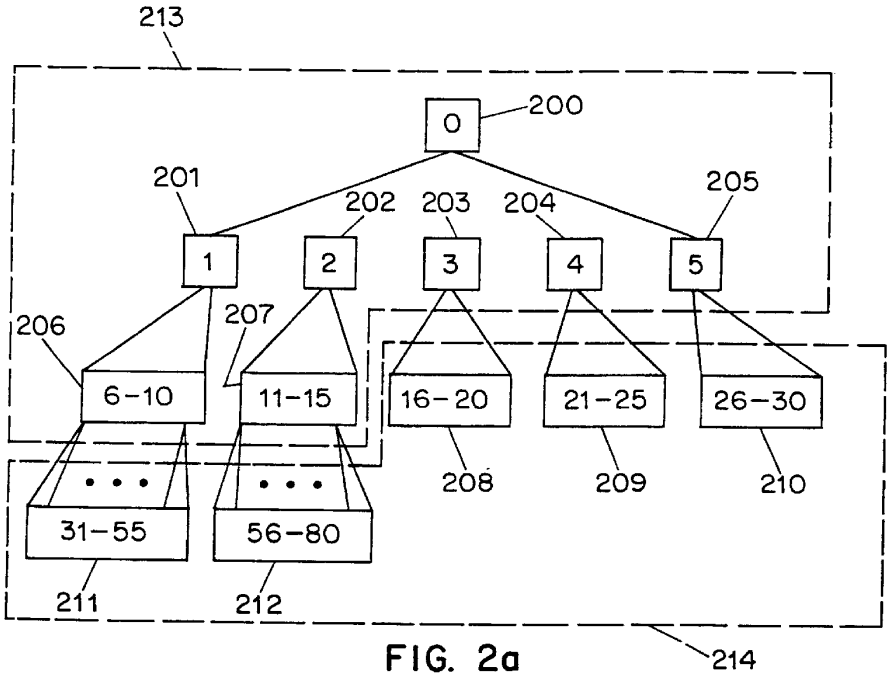


FIG. 2a

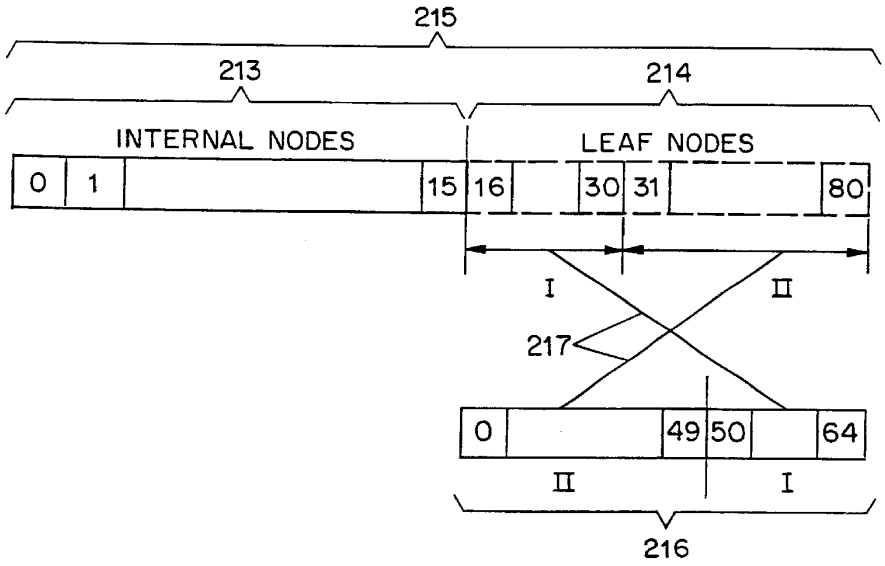


FIG. 2b

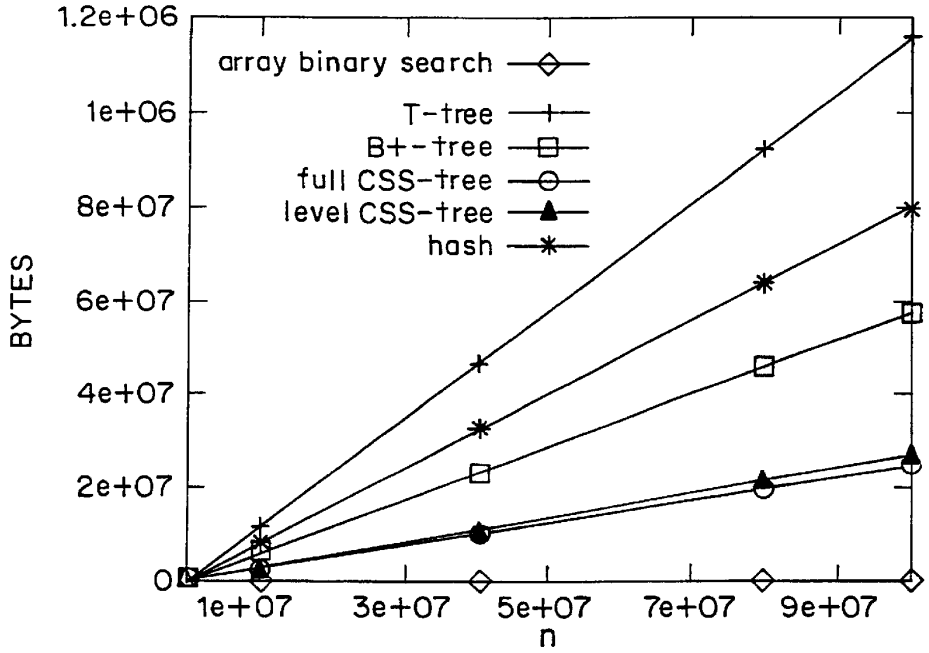


FIG. 4a

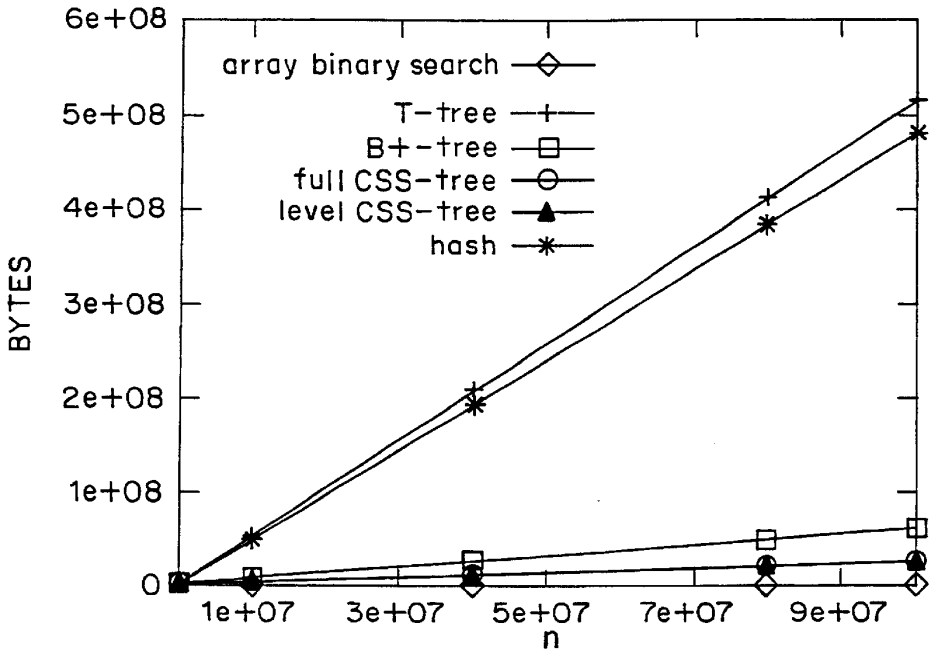


FIG. 4b

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.