US006510498B1

(54) **METHOD AND APPARATUS FOR MEMORY ALLOCATION IN A MULTI-THREADED VIRTUAL MACHINE**

(75) Inventors: **Urs Hölzle**, Goleta, CA (US); **Steffen Grarup**, Palo Alto, CA (US)

(73) Assignee: **Sun Microsystems, Inc.**, Palo Alto, CA (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.

This patent is subject to a terminal disclaimer.

(21) Appl. No.: **09/724,737**

(22) Filed: **Nov. 28, 2000**

Related U.S. Application Data

(63) Continuation of application No. 09/108,047, filed on Jun. 30, 1998, now Pat. No. 6,209,066.

(51) Int. Cl.^{7} **G06F 12/00**
(52) U.S. Cl. **711/153**; 711/170; 711/173
(58) Field of Search 711/153, 170, 711/173; 709/104, 105, 107, 108, 312; 712/206, 215, 245, 246

(56) **References Cited**

U.S. PATENT DOCUMENTS

5,247,634 A		9/1993	Cline et al.
5,535,361 A		7/1996	Hirata et al.
5,600,596 A		2/1997	Shirakihara
5,727,178 A	*	3/1998	Pletcher et al. 711/202
5,893,159 A	*	4/1999	Schneider 711/150

OTHER PUBLICATIONS

Robert H. Halstead, Jr., "Multilisp: A Language for Concurrent Symbolic Computation," Oct. 1985, ACM Transactions on Programming Languages and Systems, vol. 7, No. 4.

* cited by examiner

Primary Examiner—Matthew Kim
Assistant Examiner—Pierre M. Vital
(74) *Attorney, Agent, or Firm*—Beyer Weaver & Thomas

(57) **ABSTRACT**

Methods and apparatus for the efficient allocation of shared memory in a multi-threaded computer system are disclosed. In accordance with one embodiment of the present invention, a computer-implemented method for allocating memory shared by multiple threads in a multi-threaded computing system includes partitioning the shared memory into a plurality of blocks, and grouping the multiple threads into at least a first group and a second group. A selected block is allocated to a selected thread which may attempt to allocate an object in the selected block. The allocation of the selected block to the selected thread is based at least partially upon whether the selected thread is a part of the first group or the second group. In one embodiment, grouping the multiple threads into the first group and the second group includes identifying a particular thread and determining whether the particular thread is a fast allocating thread. In such an embodiment, when the particular thread is fast allocating, the particular thread is grouped into the first group.
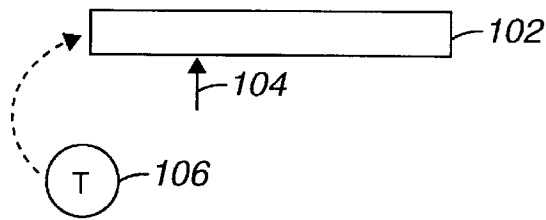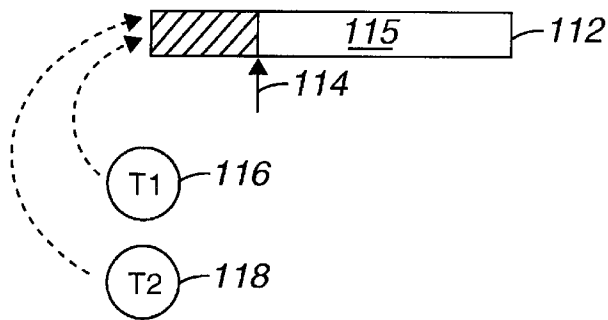
19 Claims, 14 Drawing Sheets
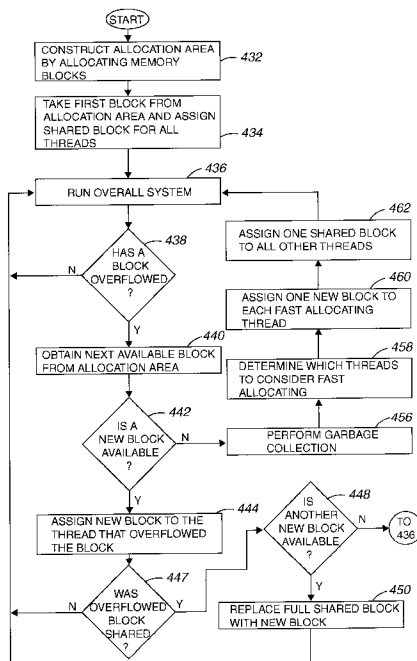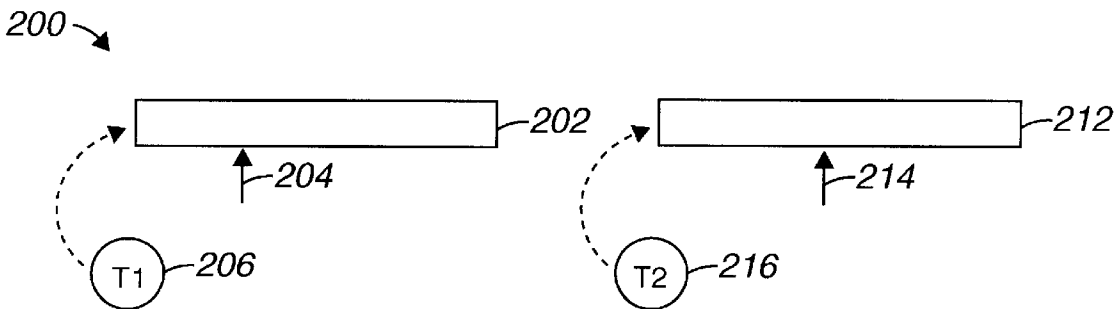
Figure 1a
(Prior Art)

Figure 1b
(Prior Art)

Figure 2a
(Prior Art)
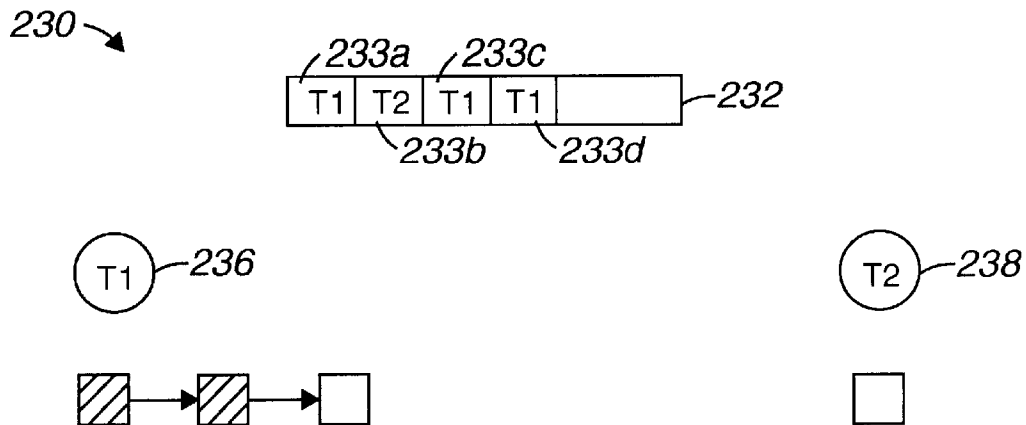
Figure 2b
(Prior Art)

Figure 3

Figure 4

START

CONSTRUCT ALLOCATION AREA BY ALLOCATING MEMORY BLOCKS — 402

TAKE FIRST BLOCK FROM ALLOCATION AREA AND ASSIGN SHARED BLOCK FOR ALL THREADS — 404

RUN OVERALL SYSTEM — 406

408 — HAS A BLOCK OVERFLOWED ? — N / Y

OBTAIN NEXT AVAILABLE BLOCK FROM ALLOCATION AREA — 410

412 — IS A NEW BLOCK AVAILABLE ? — N → PERFORM GARBAGE COLLECTION — 416

Y

ASSIGN NEW BLOCK TO THE THREAD THAT OVERFLOWED THE BLOCK — 414

418 — IS ANOTHER NEW BLOCK AVAILABLE ? — N → TO 406

417 — WAS OVERFLOWED BLOCK SHARED ? — N / Y

Y → REPLACE FULL SHARED BLOCK WITH NEW BLOCK — 420

Figure 5a

START

CONSTRUCT ALLOCATION AREA BY ALLOCATING MEMORY BLOCKS — 432

TAKE FIRST BLOCK FROM ALLOCATION AREA AND ASSIGN SHARED BLOCK FOR ALL THREADS — 434

— 436

RUN OVERALL SYSTEM

438 — HAS A BLOCK OVERFLOWED? — N

Y

OBTAIN NEXT AVAILABLE BLOCK FROM ALLOCATION AREA — 440

442 — IS A NEW BLOCK AVAILABLE? — N → PERFORM GARBAGE COLLECTION — 456

Y

ASSIGN NEW BLOCK TO THE THREAD THAT OVERFLOWED THE BLOCK — 444

447 — WAS OVERFLOWED BLOCK SHARED? — N / Y

448 — IS ANOTHER NEW BLOCK AVAILABLE? — N → TO 436

Y

REPLACE FULL SHARED BLOCK WITH NEW BLOCK — 450

ASSIGN ONE SHARED BLOCK TO ALL OTHER THREADS — 462

ASSIGN ONE NEW BLOCK TO EACH FAST ALLOCATING THREAD — 460

DETERMINE WHICH THREADS TO CONSIDER FAST ALLOCATING — 458

DOCKET ALARM
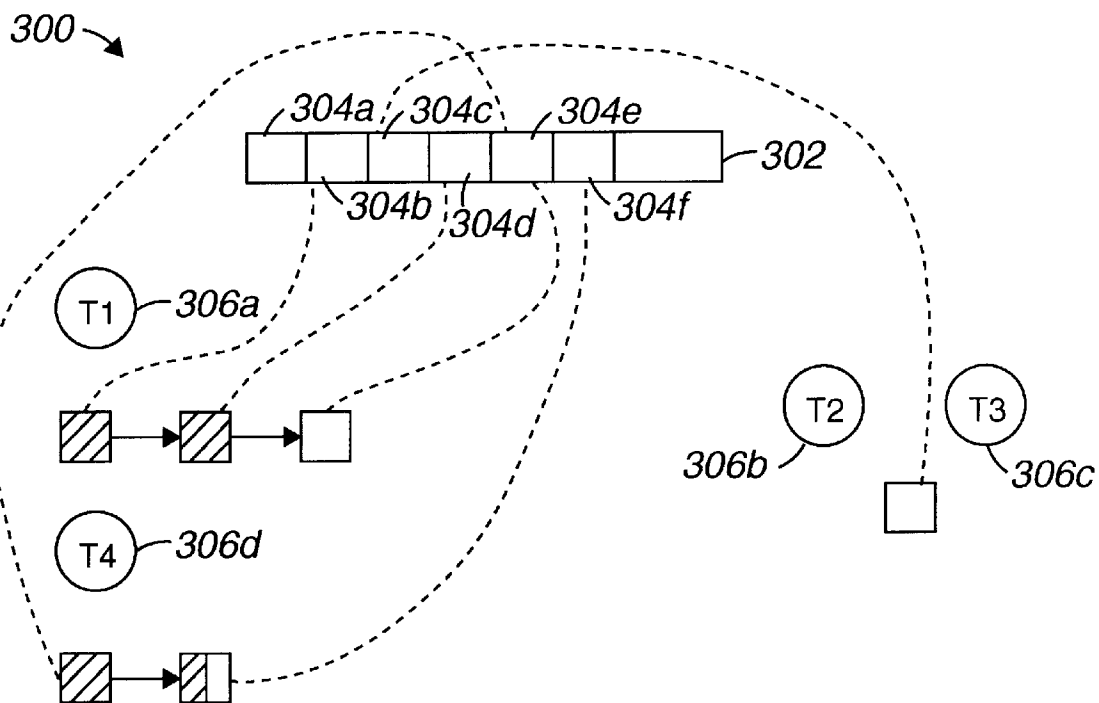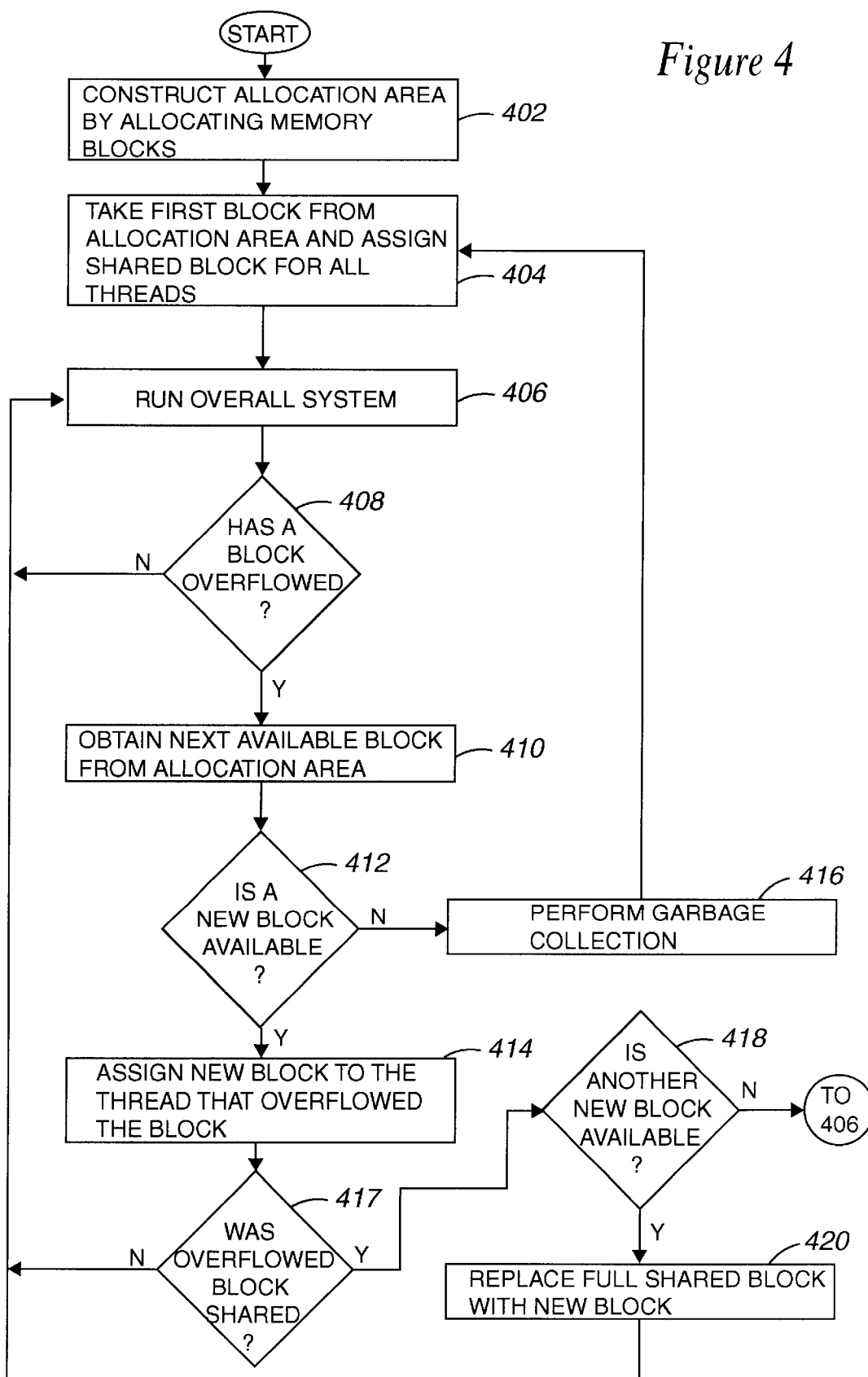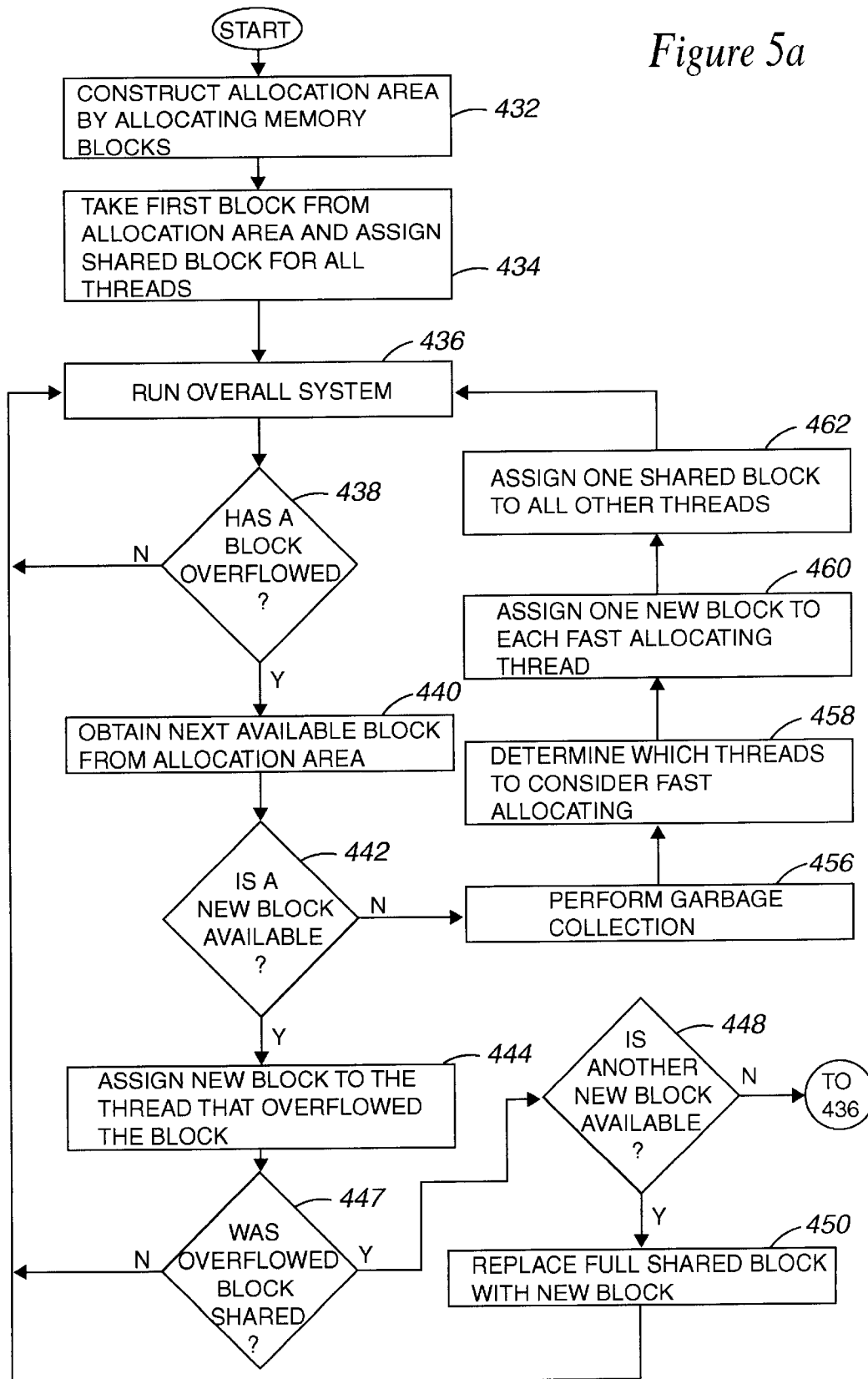
Explore Litigation Insights

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.

fastcase
Smarter legal research.