

(19) World Intellectual Property
Organization
International Bureau



(43) International Publication Date
8 December 2005 (08.12.2005)

PCT

(10) International Publication Number
WO 2005/116828 A2

(51) International Patent Classification⁷: **G06F 9/455**

(21) International Application Number:
PCT/US2005/014269

(22) International Filing Date: 25 April 2005 (25.04.2005)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
60/573,578 21 May 2004 (21.05.2004) US

(71) Applicant (for all designated States except US): **COM-
PUTER ASSOCIATES THINK, INC.** [US/US]; One
Computer Associates Plaza, Islandia, NY 11749 (US).

(72) Inventors; and

(75) Inventors/Applicants (for US only): **ESFAHANY,
Kouros, H.** [US/US]; 40 Briarwood Drive, Hunting-
ton, NY 11743 (US). **CHIARAMONTE, Michael, R.**
[US/US]; 92 Wood Road, Centereach, NY 11720 (US).
NARAYANAN, Karthik [IN/US]; 38D Country Club
Drive, Coram, NY 11727 (US).

(74) Agents: **JAWORSKI, Richard, F.** et al.; Cooper & Dun-
ham LLP, 1185 Avenue of the Americas, New York, NY
10036 (US).

(81) Designated States (unless otherwise indicated, for every
kind of national protection available): AE, AG, AL, AM,
AT, AU, AZ, BA, BB, BG, BR, BW, BY, BZ, CA, CH, CN,
CO, CR, CU, CZ, DE, DK, DM, DZ, EC, EE, EG, ES, FI,
GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE,
KG, KM, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA,
MD, MG, MK, MN, MW, MX, MZ, NA, NI, NO, NZ, OM,
PG, PH, PL, PT, RO, RU, SC, SD, SE, SG, SK, SL, SM, SY,
TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, YU,
ZA, ZM, ZW.

(84) Designated States (unless otherwise indicated, for every
kind of regional protection available): ARIPO (BW, GH,
GM, KE, LS, MW, MZ, NA, SD, SL, SZ, TZ, UG, ZM,
ZW), Eurasian (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM),
European (AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI,
FR, GB, GR, HU, IE, IS, IT, LT, LU, MC, NL, PL, PT, RO,
SE, SI, SK, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN,
GQ, GW, ML, MR, NE, SN, TD, TG).

Published:

— without international search report and to be republished
upon receipt of that report

For two-letter codes and other abbreviations, refer to the "Guid-
ance Notes on Codes and Abbreviations" appearing at the begin-
ning of each regular issue of the PCT Gazette.

(54) Title: METHOD AND APPARATUS FOR DYNAMIC MEMORY RESOURCE MANAGEMENT

(57) Abstract: A method and an apparatus for dynamic memory resource management are provided. Memory activity information is collected for a virtual machine. Additional memory shares are dynamically allocated to the virtual machine, as needed, based on the memory activity information for the virtual machine and based on a specified priority of the virtual machine in order to properly adjust the total physical memory used by the virtual machine.



WO 2005/116828 A2

METHOD AND APPARATUS FOR DYNAMIC MEMORY RESOURCE MANAGEMENT

TECHNICAL FIELD

5 The present disclosure relates generally to systems management and more particularly to a method and apparatus for dynamic memory resource management.

DESCRIPTION OF THE RELATED ART

 Systems management involves the supervision and management of
10 information technology resources in an enterprise (or other organization). For example, many systems management software include tools for monitoring and collecting information regarding resource usage, such as memory activity.

 As enterprises grow, their needs for information technology resources can change rapidly, as demands for performance and reliability tend to increase. The
15 typical approach for addressing the growing demands is to consolidate the servers in order to maximize resources available to a user. For example, applications and infrastructure services running on diverse operating systems can be consolidated on a reduced number of high performance servers, or even a single high performance server, running multiple virtual machines.

20 A virtual machine is typically a software based concept or logical entity that is implemented over a hardware platform and operating system and can use multiple devices (such as memory, processors, other hardware resources, network systems, etc.) to create multiple virtual systems, each of which can run independently as a copy of the operating system. In other words, a virtual machine can be thought of as a
25 computer that operates inside, for example, a server, as an entity separate from other

virtual machines. A virtual machine can allow for flexibility across platforms and can provide performance optimization by allowing time-expensive hardware to be shared.

Virtual machine software, such as VMware ESX Server ("ESX"), can be used to consolidate systems in advanced environments. System(s) as herein referred to
5 may include(s) individual computers, servers, computing resources, and/or networks, etc. For example, ESX can provide a virtualization software tool that deploys multiple, secure isolated virtual machines, with respective allocated memory shares, on a single server system where system resources can be allocated to any virtual machine based on need. However, these system resource allocations can only take
10 place statically or manually, and can prove to be problematic. For example, if a network administrator over-provisions resources based on a worst case scenario, under-utilization of system resources most likely results. On the other hand, under-provisioning of resources can be equally problematic.

It is desirable for users, such as network administrators, to have reliable and
15 effective tools for dynamically managing (in addition to monitoring) system resources based on real time requirements of, for example, virtual machines in virtual environments, in order to meet growing business demands.

SUMMARY

20 This application describes methods and apparatuses for dynamic memory resource management. A method for dynamic memory resource management, according to an embodiment of the present disclosure, includes collecting memory activity information for a virtual machine, and dynamically changing memory shares

(for example, as a proportional resource unit used to determine how much memory the virtual machine should use compared with other running virtual machines). The memory shares of the virtual machine are dynamically changed, as needed, based on the memory activity information for the virtual machine and based on a specified
5 priority of the virtual machine. The method may further include analyzing the collected memory activity information for the virtual machine, to determine memory usage of the virtual machine, and increasing the memory shares of the virtual machine, if it is determined that the memory usage of the virtual machine is equal to or higher than the allocated memory share of the virtual machine. On the other hand,
10 the memory shares of the virtual machine may also be decreased, if it is determined that the memory usage of the virtual machine is substantially lower than the amount currently allocated to the virtual machine, thereby freeing resources for use by other virtual machines.

According to another embodiment, a method for dynamic memory resource
15 management, includes collecting memory related information and user defined criteria for one or more virtual machines, determining the memory state for each of the virtual machines, computing a dynamic priority for each of the virtual machines, and determining whether to increase, decrease, or leave as is memory shares for each of the virtual machines.

20 An apparatus for dynamic memory resource management, according to one embodiment, includes means for collecting memory activity information for a virtual machine, and means for dynamically changing memory shares of the virtual machine, as needed, based on the memory activity information for the virtual machine and

based on a specified priority of the virtual machine.

According to another embodiment, an apparatus for dynamic memory resource management includes means for collecting memory related information and user defined criteria for one or more virtual machines, means for determining the memory
5 state for each of the virtual machines, means for computing a dynamic priority for each of the virtual machines, and means for determining whether to increase, decrease, or leave as is memory shares for each of the virtual machines.

The methods and apparatuses of this disclosure may be embodied in one or more computer programs stored on a computer readable medium or program storage
10 device and/or transmitted via a computer network or other transmission medium. For example, a computer storage medium including computer executable code for dynamic memory resource management, according to one embodiment of the present disclosure, includes code for collecting memory related information and user defined criteria for one or more virtual machines, code for determining the memory state for
15 each of the virtual machines, code for computing a dynamic priority for each of the virtual machines, and code for determining whether to increase, decrease, or leave as is memory shares for each of the virtual machines.

BRIEF DESCRIPTION OF THE DRAWINGS

20 The features of the present application can be more readily understood from the following detailed description with reference to the accompanying drawings wherein:

FIG. 1 shows a block diagram of an exemplary computer system capable of

implementing the methods and apparatuses of the present disclosure;

FIG. 2A shows a flow chart illustrating a method for dynamic memory resource management, according to an exemplary embodiment of the present disclosure;

5 FIG. 2B shows a flow chart illustrating a method for dynamic memory resource management, according to another embodiment;

FIG. 2C shows a flow chart illustrating a method for dynamic memory resource management, according to another embodiment; and

FIG. 3 shows a state transition diagram illustrating the possible memory states
10 for a virtual machine, according to an embodiment of the present disclosure.

DETAILED DESCRIPTION

The present disclosure provides tools (in the form of methodologies, apparatuses, and systems) for dynamic memory resource management (also referred
15 to herein as “DMRM”). DMRM according to this disclosure can be utilized to dynamically manage memory utilization and allocation in virtual machines located in virtual environments. Based on certain predefined or specified criteria and priorities, DMRM can continuously adjust and balance the memory utilization across a virtual environment, to meet the resource needs of each virtual machine in the environment,
20 while simultaneously following the user-defined priorities in real time. DMRM may employ dynamic programming techniques and demand-based usage balancing of memory in order to detect possible peak scenarios of memory resource utilizations based on given criteria. The criteria can be established by users (for example,

network administrators) and/or dynamically determined as time progresses in order to satisfy real time requirements.

The following exemplary embodiments are set forth to aid in an understanding of the subject matter of this disclosure, but are not intended, and should not be construed, to limit in any way the claims which follow thereafter. Therefore, while
5 specific terminology is employed for the sake of clarity in describing some exemplary embodiments, the present disclosure is not intended to be limited to the specific terminology so selected, and it is to be understood that each specific element includes all technical equivalents which operate in a similar manner.

10 FIG. 1 shows an example of a computer system **100** which may implement the methods and apparatuses of the present disclosure. The apparatus and method of the present disclosure may be implemented in the form of a software application running on a computer system, for example, a mainframe, personal computer (PC), handheld computer, server, etc. The software application may be stored on computer readable
15 media or program storage device locally accessible by the computer system, for example, floppy disk, compact disk, hard disk, etc., or may be remote from the computer system and accessible via a hard wired or wireless connection to a computer network (for example, a local area network, the Internet, etc.) or another transmission medium.

20 The computer system **100** can include a central processing unit (CPU) **102**, program and data storage devices **104**, a printer interface **106**, a display unit **108**, a (LAN) local area network data transmission controller **110**, a LAN interface **112**, a network controller **114**, an internal bus **116**, and one or more input devices **118** (for

example, a keyboard, mouse etc.). As shown, the system 100 may be connected to a database 120, via a link 122.

Methods for dynamic memory resource management are discussed below.

According to an exemplary embodiment (FIG. 2A), a method for dynamic
5 memory resource management includes collecting memory activity information for a virtual machine (step S11), and dynamically changing memory shares of the virtual machine, as needed, based on the memory activity information for the virtual machine and based on a specified priority of the virtual machine (step S12). The method may further comprise analyzing the collected memory activity information for the virtual
10 machine, to determine memory usage of the virtual machine, and then increasing the memory share of the virtual machine if it is determined that the memory usage of the virtual machine is equal to or higher than the allocated memory of the virtual machine, or decreasing the memory share of the virtual machine if it is determined that the memory usage of the virtual machine is substantially lower than the allocated
15 memory of the virtual machine.

A method for dynamic memory resource management, according to another exemplary embodiment (FIG. 2B), will be discussed below. Memory activity information is collected for a virtual machine (step S101). Memory shares (for example, proportional amount of memory in comparison to other virtual machines) of
20 the virtual machine are dynamically changed, as needed, based on the memory activity information for the virtual machine and based on a specified priority of the virtual machine. For example, the collected memory activity information for the virtual machine is analyzed to determine memory usage of the virtual machine (step

S102). If it is determined that the memory usage of the virtual machine is equal to or higher than the allocated memory of the virtual machine (step S103, "YES"), the memory shares of the virtual machine are increased (step S104). On the other hand, if it is determined that the memory usage of the virtual machine is substantially lower than the allocated memory of the virtual machine (step S103, "NO", and step S105, "YES"), the memory shares of the virtual machine are decreased (step S106).

A method according to another embodiment for dynamic memory resource management is illustrated in FIG. 2C. Memory related information and user defined criteria are collected for one or more virtual machines during an analysis interval (Step S201). The memory state is then determined (Step S202) with respect to the user defined criteria set internal to the engine and a dynamic priority is computed (Step S203), for each virtual machine. An analytical engine makes a determination, based on the memory state, to increase, decrease or leave as is, the memory shares for the virtual machine (Step S204). For example, if the allocated physical memory for a virtual machine is less than what the virtual machine actually needed, used, etc., then the analytical engine can increase the memory shares for that virtual machine. An increase in the memory shares for a virtual machine indicates to the server, to allocate more physical memory for that virtual machine. On the other hand, if the allocated physical memory for a virtual machine is substantially more than what the virtual machine actually needed, used, etc., then the analytical engine can decrease the memory shares for that particular machine. A decrease in the memory share for a particular virtual machine can indicate to the server, to reduce the physical memory allocated for that virtual machine. Based on the determination in Step S204, the

memory share for a virtual machine can be updated accordingly. The engine can repeat the entire process in the next analysis interval (Steps **S201-S204**).

The dynamic priority for a virtual machine can be computed using the user defined criteria. The user defined criteria can comprise one or more management
5 criteria, memory criteria, a virtual machine priority, and a memory coefficient for each of the virtual machines. The user defined criteria can also comprise a poll interval and an analysis interval for each server. The management criteria can comprise information indicating the virtual machines to be managed, not managed or monitored. The memory criteria can comprise information indicating memory needs
10 of the virtual machines to be managed, not managed or monitored. The analysis interval can be the period of time during which the memory related information is collected. The poll interval can be the period of time between collections of memory related information. The memory related information is preferably analyzed in real time.

15 FIG. 3 is a schematic diagram illustrating the possible memory states for a virtual machine, according to an embodiment of the present disclosure. The memory states of a virtual machine can include the following: Normal **301**, Repaired **302**, Stressed **303**, and Improperly Repaired **304**. A virtual machine starts in a Normal **301** state, and can go from the Normal **301** state to the Stressed **303** state, if the physical
20 size of the allocated memory for the virtual machine is less than the memory used. Similarly, a virtual machine can go from the Repaired **302** state to the Stressed **303** state, if the physical size of the allocated memory is less than the memory used by the virtual machine. A virtual machine can return to the Normal **301** state from the

Repaired **302** state, if the dynamic priority reaches a predefined minimum value. A virtual machine can go from the Stressed **303** state to the Improperly Repaired **304** state, if the physical size of the memory allocated to the virtual machine is greater than 95% of the maximum size and is not equal to the memory used. A virtual machine can return from the Stressed **303** state back to the Repaired **302** state, if the actual physical size of the memory is greater than the memory used. A virtual machine can go from the Improperly Repaired **304** state to the Stressed **303** state, if the actual physical size of the memory is less than the memory used or can go to the Repaired **302** state, if the actual physical size of the memory is greater than the memory used.

The virtual machines can reside, according to one exemplary embodiment, in a client-server system with one or more management stations. Memory activity in the system, and in particular for each virtual machine, is monitored, and then can be reported using a reporting tool. The server can be supplemented with additional instrumentation providing the management stations with a comprehensive, real time view of the server and each virtual machine. The management stations can pinpoint status at various levels, such as at the server level or at the virtual machine level. The status of the server and virtual machines can be shown through the reporting tool. Events can be automatically forwarded to event management components, which can further propagate the events to a central management station.

A user, such as a network administrator, can specify whether he/she wants to monitor, manage or not manage memory related information for individual virtual machines. The user can also define a virtual machine priority, and a memory

coefficient, for each of the virtual machines located on a server. The user can also define a poll interval and an analysis interval for each server. Memory related information can also be provided, for each of the virtual machines, to allow an analytical engine to dynamically determine memory status of a virtual machine, and
5 whether to allocate additional physical memory to a particular virtual machine.

A front end graphical user interface (for example, implemented in Java) can be provided to configure the user-defined criteria. The parameters entered by the user can be stored in the reporting tool. The analytical engine can then read those stored values from the reporting tool and use them when it is dynamically reconfiguring
10 memory allocations.

An analysis interval may include the period of time, for example, in seconds, during which memory related information can be collected for analysis of virtual machine memory status. A poll interval may include the amount of time, for example, in seconds, between collections of memory related information. The
15 analysis interval preferably is not less than the poll interval. For example, the default value of the analysis interval can be set to twice the amount of the poll interval. If the poll interval is, for example, 60 seconds, the analysis interval can be 120 seconds.

An analytical engine can perform dynamic memory reconfiguration based on memory related information and user-defined criteria. An engine service can also be
20 provided to assist the analytical engine in managing the virtual machines. The engine service can be, for example, a service on a Windows platform or a daemon on a Linux platform, and can spawn the analytical engine responsible for managing a server. For example, the engine service can read the discovered server objects, such as the virtual

machines running on the server, and then spawn the analytical engine to take care of dynamic memory reconfiguration.

The specific embodiments described herein are illustrative, and many additional modifications and variations can be introduced on these embodiments
5 without departing from the spirit of the disclosure or from the scope of the appended claims. Elements and/or features of different illustrative embodiments may be combined with each other and/or substituted for each other within the scope of this disclosure and appended claims.

Additional variations may be apparent to one of ordinary skill in the art from
10 reading U.S. provisional application Serial No. 60/573,578, filed May 21, 2004 and entitled "METHOD AND APPARATUS FOR DYNAMIC MEMORY RESOURCE MANAGEMENT", the entire contents of which are incorporated herein by reference.

What is claimed is:

1. A method for dynamic memory resource management, comprising:
collecting memory activity information for a virtual machine; and
dynamically changing memory shares of the virtual machine, as needed, based
5 on the memory activity information for the virtual machine and based on a specified
priority of the virtual machine.

2. The method of claim 1, further comprising:
analyzing the collected memory activity information for the virtual machine,
10 to determine memory usage of the virtual machine; and
increasing the memory share of the virtual machine, if it is determined that the
memory usage of the virtual machine is equal to or higher than the allocated memory
of the virtual machine.

15 3. The method of claim 1, further comprising:
analyzing the collected memory activity information for the virtual machine,
to determine memory usage of the virtual machine; and
decreasing the memory shares of the virtual machine, if it is determined that
the memory usage of the virtual machine is substantially lower than the allocated
20 memory of the virtual machine.

4. A computer system comprising:
a processor; and

a program storage device readable by the computer system, tangibly embodying a program of instructions executable by the processor to perform the method claimed in claim 1.

5 5. A program storage device readable by a machine, tangibly embodying a program of instructions executable by the machine to perform the method claimed in claim 1.

6. A computer data signal transmitted in one or more segments in a
10 transmission medium which embodies instructions executable by a computer to perform the method claimed in claim 1.

7. An apparatus for dynamic memory resource management, comprising:
means for collecting memory activity information for a virtual machine; and
15 means for dynamically changing memory shares of the virtual machine, as needed, based on the memory activity information for the virtual machine and based on a specified priority of the virtual machine.

8. A method for dynamic memory resource management, comprising:
20 collecting memory related information and user defined criteria for one or more virtual machines;
determining the memory state for each of the one or more virtual machines;
computing a dynamic priority for each of the one or more virtual machines;

and

determining for each of the one or more virtual machines whether to increase, decrease, or maintain the memory shares of the virtual machine, based on the memory related information, memory state and dynamic priority of the virtual machine.

5

9. The method of claim 8, further comprising analyzing the memory related information in real time.

10. The method of claim 8, wherein the memory shares of the virtual machine
10 are increased, if it is determined that actual memory usage of the virtual machine is higher than an allocated physical memory of the virtual machine.

11. The method of claim 8, wherein the memory shares of the virtual machine
are decreased if it is determined that actual memory usage of the virtual machine is
15 lower than an allocated physical memory of the virtual machine.

12. The method of claim 8, wherein the user defined criteria comprises one or more management criteria, memory criteria, a virtual machine priority, and a memory coefficient, for each of the virtual machines.

20

13. The method of claim 12, wherein the management criteria comprises information indicating the virtual machines to be managed, not managed or monitored.

14. The method of claim 12, wherein the memory criteria comprises information indicating memory needs of the virtual machines to be managed, not managed or monitored.

5

15. The method of claim 8, wherein the user defined criteria comprises a poll interval and an analysis interval, for each server.

16. The method of claim 15, wherein the analysis interval is the period of
10 time during which the memory related information is collected.

17. The method of claim 15, wherein the poll interval is the period of time between collections of memory related information.

15 18. The method of claim 8, wherein the memory state is either normal, stressed, repaired or improperly repaired.

19. The method of claim 8, wherein the dynamic priority for each of the virtual machines is computed using the user defined criteria.

20

20. The method of claim 8, wherein the virtual machines are located in a client-server system.

21. A computer system comprising:

a processor; and

a program storage device readable by the computer system, tangibly
embodying a program of instructions executable by the processor to perform the
5 method claimed in claim 8.

22. A program storage device readable by a machine, tangibly embodying a
program of instructions executable by the machine to perform the method claimed in
claim 8.

10

23. A computer data signal transmitted in one or more segments in a
transmission medium which embodies instructions executable by a computer to
perform the method claimed in claim 8.

15

24. An apparatus for dynamic memory resource management, comprising:

means for collecting memory related information and user defined criteria for
one or more virtual machines;

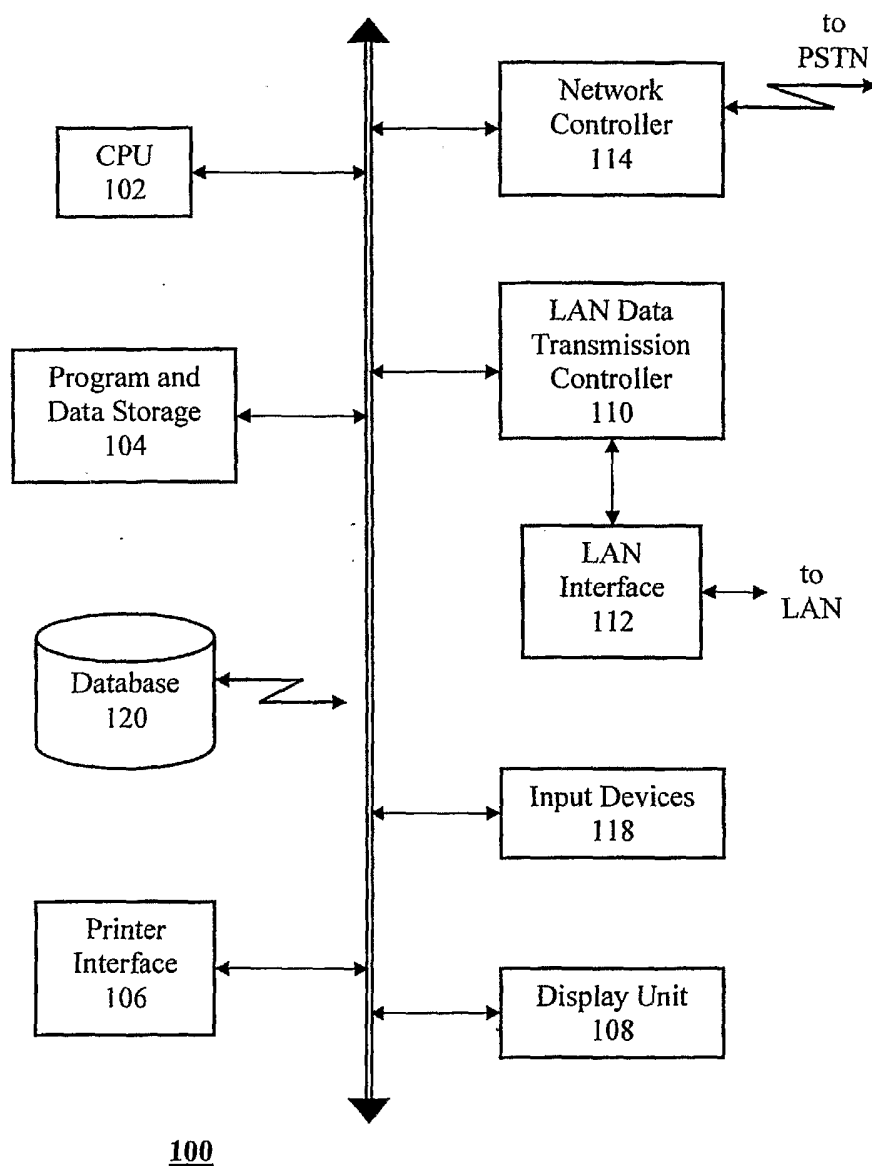
means for determining the memory state for each of the one or more virtual
machines;

20

means for computing a dynamic priority for each of the one or more virtual
machines; and

means for determining for each of the one or more virtual machines whether to
increase, decrease, or maintain a memory share of the virtual machine, based on the

memory related information, memory state and dynamic priority of the virtual machine.

FIG. 1

2/4

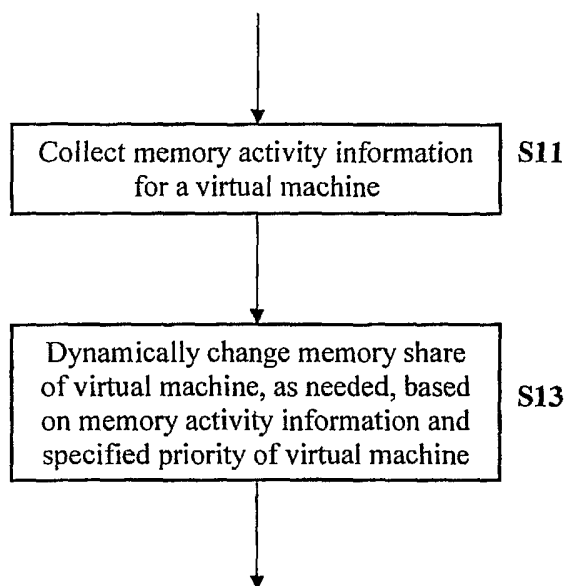
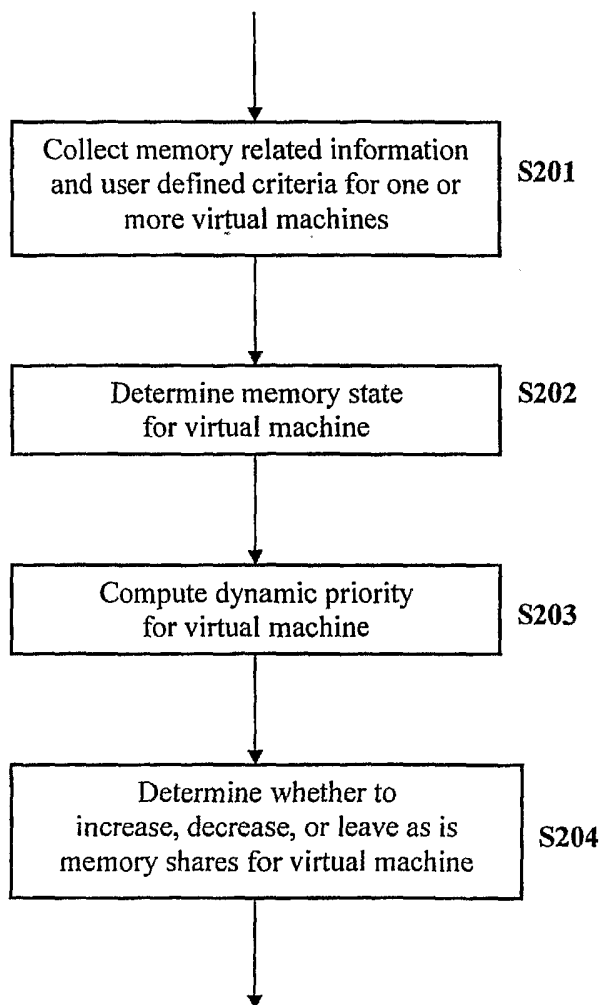
**FIG. 2A****FIG. 2C**

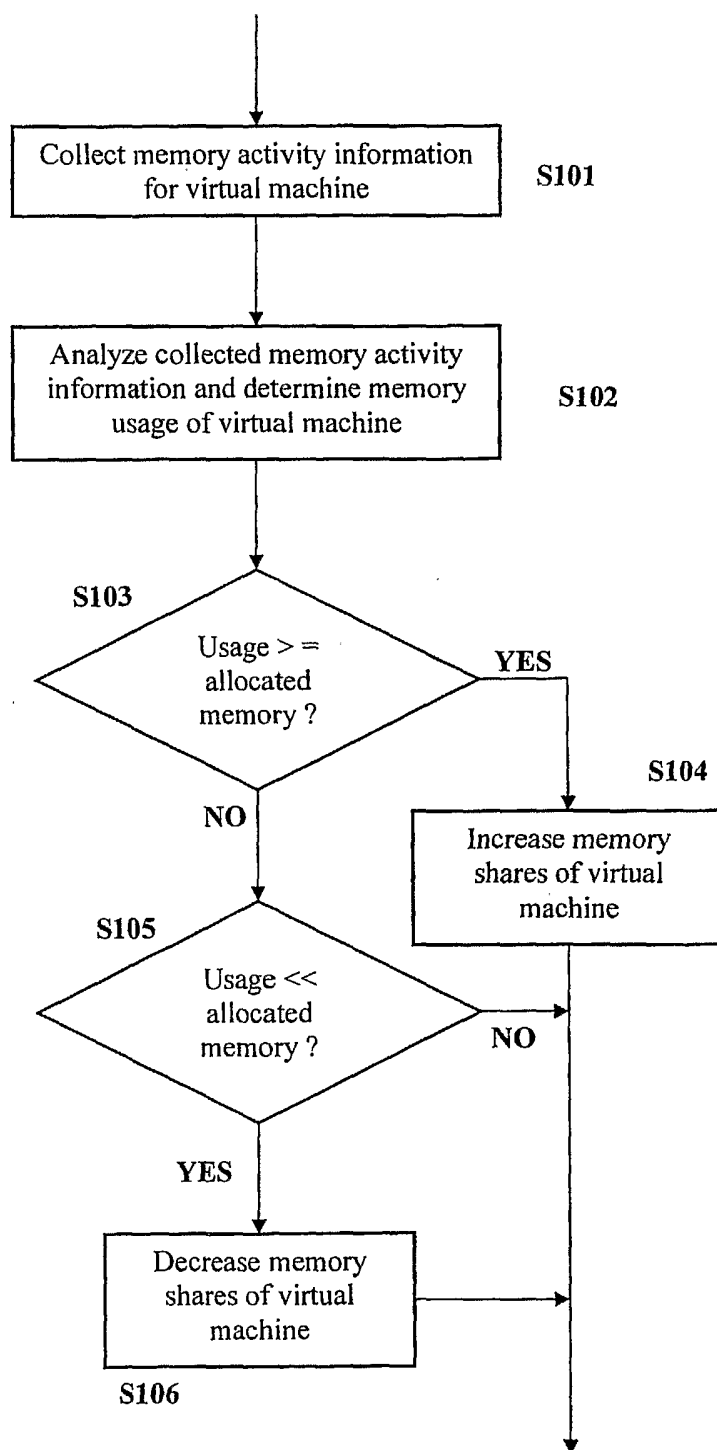
FIG. 2B

FIG. 3