(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
5 November 2015 (05.11.2015)

WIPO | PCT

(10) International Publication Number
WO 2015/167453 A1

(51) International Patent Classification:
G06F 13/14 (2006.01) *G06F 12/00* (2006.01)

(21) International Application Number:
PCT/US2014/035829

(22) International Filing Date:
29 April 2014 (29.04.2014)

(25) Filing Language: English

(26) Publication Language: English

(71) Applicant: **HEWLETT-PACKARD DEVELOPMENT COMPANY, L.P.** [US/US]; 11445 Compaq Center Drive W., Houston, Texas 77070 (US).

(72) Inventors: **CHEN, Yuan**; 1501 Page Mill Road, Palo Alto, California 94304-1100 (US). **GMACH, Daniel, Juergen**; 1501 Page Mill Road, Palo Alto, California 94304-1100 (US). **MILOJICIC, Dejan, S.**; 1501 Page Mill Road, Palo Alto, California 94304-1100 (US). **TALWAR, Vanish**; 1501 Page Mill Road, Palo Alto, California 94304-1100 (US). **WANG, Zhikui**; 1501 Page Mill Road, Palo Alto, California 94304-1100 (US).

(74) Agents: **HAQ, M., Aamir** et al.; Hewlett-packard Company, Intellectual Property Administration, Mail Stop 35 3404 E. Harmony Road, Fort Collins, Colorado 80528 (US).

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

(84) Designated States *(unless otherwise indicated, for every kind of regional protection available)*: ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, RU, TJ, TM), European (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, KM, ML, MR, NE, SN, TD, TG).

Declarations under Rule 4.17:

— *as to the identity of the inventor (Rule 4.17(i))*

— *as to applicant's entitlement to apply for and be granted a patent (Rule 4.17(ii))*

[Continued on next page]

(54) Title: COMPUTING SYSTEM MANAGEMENT USING SHARED MEMORY

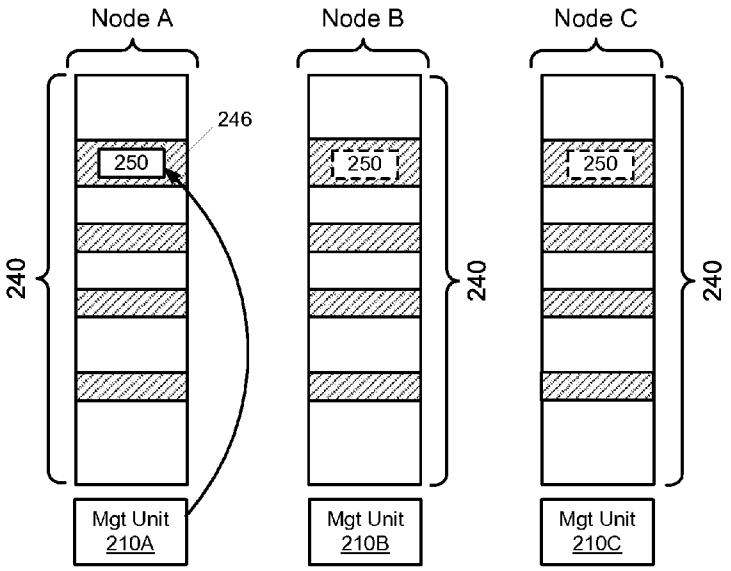

FIG. 3A

(57) Abstract: A system management command is stored in a management partition of a global memory by a first node of a multi-node computing system. The global memory is shared by each node of the multi-node computing system. In response to an indication to access the management partition, the system management command is accessed from the management partition by a second node of the multi-node computing system. The system management command is executed by the second node. Executing the system management command includes managing the second node.

WO 2015/167453 A1

Published:

— *with international search report (Art. 21(3))*

COMPUTING SYSTEM MANAGEMENT USING SHARED MEMORY

Background

[0001] Some computing systems include a group of nodes working together as a single system. Such systems may be referred to as "multi-node systems." Each node can be a computing device capable of functioning as an independent unit. The nodes may be interconnected to share data and/or resources. In addition, the nodes may communicate by passing messages to each other.

Brief Description Of The Drawings

[0002] Some implementations are described with respect to the following figures.

[0003] Fig. 1A is a schematic diagram of an example multi-node system, in accordance with some implementations.

[0004] Fig. 1B is a schematic diagram of an example node, in accordance with some implementations.

[0005] Fig. 2 is an illustration of shared memory mapping according to an example implementation.

[0006] Figs. 3A-3C are illustrations of an internode management operation according to an example implementation.

[0007] Fig. 4 is a flow diagram of a process according to some implementations.

[0008] Fig. 5 is a flow diagram of a process according to some implementations.

Detailed Description

[0009] In a multi-node computing system, each node can be a computer including hardware resources such as processor(s), memory, storage, etc. Further, each node can include software resources such as an operating system, an application, a virtual machine, data, etc. In some implementations, a multi-node computing system may be a cluster that can

be used as a single computing device. For example, a cluster may utilize clustering middleware to orchestrate the activities of each node (e.g., assigning tasks of a single application for execution on different nodes).

[0010] In accordance with some implementations, techniques or mechanisms are provided to allow for communication between nodes of a multi-node computing system using shared memory. Such communication can include passing of system management commands, thereby enabling one node to manage other nodes (referred to as "inter-node management"). The shared memory can also enable a node to access and analyze monitoring data generated by other nodes.

[0011] Fig. 1A is a schematic diagram of an example multi-node system 100, in accordance with some implementations. As shown, the multi-node system 100 can include any number of computing nodes 110. The nodes 110 are coupled by a network 115 (e.g., a high-speed cluster interconnection or system fabrics). In some implementations, the nodes 110 may be configured for inter-node management and data monitoring. The configuration of the nodes 110 is described below with reference to Fig. 1B.

[0012] As shown in Fig. 1B, a node 110 can include processor(s) 120, local memory 130, machine-readable storage 140, and an inter-node management unit 160. In some implementations, the inter-node management unit 160 can enable the node 110 to access a global memory 170 shared with other nodes. In some implementations, the global memory 170 can be non-volatile memory, and may not lose its state when powered off.

[0013] The processor(s) 120 can include a microprocessor, microcontroller, processor module or subsystem, programmable integrated circuit, programmable gate array, multiple processors, a microprocessor including multiple processing cores, or another control or computing device. The local memory 130 and the global memory 170 can be any type of computer memory (e.g., dynamic random access memory (DRAM), static random-access memory (SRAM), non-volatile memory (NVM), a combination of DRAM and NVM, etc.). The machine-readable storage 140 can include non-transitory storage media such as hard drives, flash storage, optical disks, non-volatile memory, etc.

[0014] The global memory 170 can be obtained using memory sharing techniques such as cache-coherent shared memory, Remote Direct Memory Access (RDMA), and so forth. As shown, in some implementations, the global memory 170 may be shared memory that is external to the nodes 110. In other implementations, the global memory 170 may include some or all of the local memories 130 of all of the nodes 110, or may be some combination of external memory and local memory 130.

[0015] In some implementations, the inter-node management unit 160 can provide inter-node management using the global memory 170. Specifically, the inter-node management unit 160 may enable a node 110 to manage other nodes 110 by writing and/or reading system management commands in the global memory 170. The system management commands may be written to memory structures configured to store management information such as instruction code, function identifiers, function invocations, etc. Further, the system management commands can have a specific format. For example, one format of a system management command may include: a function identifier, the memory address of input parameters, the memory address of the destination data to store, and the function type to invoke.

[0016] In some implementations, the system management commands can provide system management of nodes. For example, the system management commands can control the power state of a node (e.g., shut down the node, turn on/off a processor or core of the node, adjust clock speed and/or voltage, etc.). The system management commands can also cause a node to reboot, and/or specify an image of machine-readable instructions to be loaded on the node after reboot. Further, the system management commands can cause a node to enter/exit a management mode (i.e., for managing other nodes), and specify an image of machine-readable instructions to be executed on the node during the management mode. In addition, the system management commands can control the monitoring state of a node (e.g., begin or end monitoring a specific node, a group of nodes, any nodes having a specific status, a specific application running on a node, etc.), and/or cause the node to analyze monitoring data of some or all nodes.

[0017] In some implementations, the inter-node management unit 160 of each node may send an indication to other nodes to check the global memory 170 for new system

management commands. For example, the indication may be an inter-node signal such as an interrupt, a message, etc. In other implementations, the inter-node management unit 160 of each node may check the global memory 170 in response to a time-based indication (e.g., expiration/restart of a repeating cycle, a scheduled event, a periodic polling, etc.).

[0018] In some implementations, the inter-node management unit 160 may share monitoring data with other nodes by storing the monitoring data in the global memory 170. Examples of monitoring data may include error logs, performance metrics, and operation states. The stored monitoring data may be organized as key-value pairs. The key for the monitoring data can be mapped to a memory address via hash function or table. The inter-node management unit 160 of another node can then access the monitoring data via key mapping and memory access. In some implementations, the monitoring data can be opportunistically stored in a location that is physically local or close to the node that generates or consumes the monitoring data. Further, the monitoring data can be cached, replicated and/or synchronized across nodes.

[0019] The inter-node management unit 160 may also enable analysis of monitoring data stored in the global memory 170. For example, the inter-node management unit 160 may collect and analyze monitoring data locally, and then copy only the analysis results to global memory 170. In some implementations, a ring buffer can be implemented in a distributed memory manner, and can be utilized to store last N instances of collected monitoring data. Thus, in the event of a need to search the distributed memory ring buffers, such search can be performed by searching individual ring buffers in a distributed manner, and then storing the end result in the requesting node's memory. Further, the inter-node management unit 160 can utilize probabilistic graphical models such as Bayesian networks to capture correlations and dependencies among the collected monitoring data.

[0020] In some implementations, the inter-node management unit 160 includes an Application Programming Interface (API) for external system management systems (e.g., in-band management services, deployment services, out-of-band management services, etc.). This API can enable an external system management system to manage nodes using system management commands written to the global memory 170. Further, the API can enable the external system management system to access the monitoring data of the nodes.

[0021] Various tasks of the inter-node management unit 160 are discussed further below with reference to Figs. 2, 3A-3C, 4, and 5. Note that any of the features described herein in relation to the inter-node management unit 160 can be implemented in any suitable manner. For example, any of these features can be hard-coded as circuitry in the inter-node management unit 160. In other examples, the machine-readable storage 140 can include inter-node management instructions 165 that can be loaded and executed by the processor(s) 120 and/or the inter-node management unit 160 to implement the features of the inter-node management unit 160 described herein. In further examples, all or a portion of the machine-readable storage 140 can be embedded within the inter-node management unit 160. In still further examples, the inter-node management instructions 165 can be stored in an embedded storage medium of the inter-node management unit 160, while other information is stored in the machine-readable storage 140 that is external to the inter-node management unit 160. In yet further example, the inter-node management instructions 165 can be stored in the global memory 170.

[0022] Referring to Fig. 2, shown is an illustration of shared memory mapping according to an example implementation. Specifically, Fig. 2 corresponds to an example multi-node system including three nodes, namely Node A, Node B, and Node C. As shown, each node may be associated with a management unit 210. The management unit 210 may correspond generally to the inter-node management unit 160 shown in Fig. 1B. Further, the Nodes A, B, and C may share the global address space 240. The global address space 240 may be a virtual address space for a global memory shared by Nodes A, B, and C. For example, the global address space 240 may correspond to a global memory that is external to nodes, to a global memory including some or all of the local memories of the nodes, to a combination of external and local memories, and so forth.

[0023] In some implementations, the global address space 240 can include memory partitions reserved for inter-node management or data monitoring, which may be referred to as "management partitions." For example, as shown in Fig. 2, the global address space 240 includes the management partitions 242, 243, 244, and 246, which can be accessed by the Nodes A, B, and C.

[0024] In some implementations, any of the management partitions can be used to manage any or all of the nodes. In other implementations, some management partitions may be reserved for managing a particular node or group of nodes. For example, referring to Fig. 2, the management partition 246 may be reserved for managing Node C. In some implementations, the association between a management partition and a node may be established by firmware associated with a bootstrap process.

[0025] In some implementations, the remaining portions 248 of the global address space 240 that are not used as management partitions can be used as functional memory of the multi-node system 100. For example, each functional memory portion 248 can be used to store variable values, program instructions, static data, etc.

[0026] In some implementations, each management partition may be specified as a virtual or physical address (or range of addresses). Further, each management partition can be associated with security permissions assigned to the inter-node management units 160, thereby reserving the management partition for inter-node management or data monitoring purposes.

[0027] Figs. 3A-3C are illustrations of an example internode management operation according to some implementations. Specifically, Figs. 3A-3C illustrate three states or stages of an example internode management operation using the shared memory mapping illustrated in Fig. 2. Assume that, in this example, Node A is a manager node of the multi-node system including Node A, Node B, and Node C. Assume further that Node A determines that Node C should be rebooted, and thus initiates an internode management operation to reboot Node C. Assume also that the management partition 246 is associated with Node C (e.g., is reserved for managing Node C).

[0028] Referring now to Fig. 3A, the management unit 210A of Node A can initiate the internode management operation by writing a system management command 250 into the management partition 246 associated with Node C. In this example, the system management command 250 is a reboot instruction.

[0029] Referring now to Fig. 3B, the management unit 210A of Node A can send a signal 214 to the management unit 210C of Node C to check the management partition 246 associated with Node C. In some implementations, the signal 214 can be an inter-node signal such as an interrupt, a message, etc.

[0030] Referring now to Fig. 3C, in response to the signal 214, the management unit 210C of Node C can check the management partition 246. Further, the management unit 210C can access the system management command 250 stored in the management partition 246. The system management command 250 can then be executed by Node C, thereby causing Node C to be rebooted.

[0031] Note that, while Fig. 3B shows the management unit 210A sending the signal 214 to cause the management unit 210C to check the management partition 246, other implementations are possible. For example, in some implementations, each management unit 210 can automatically check a management partition according to a predefined period or schedule, in response to an event, etc.

[0032] Note also that, while Figs. 3A-3C show an example in which one node controls another node, other implementations are possible. For example, in some implementations, all nodes (or a subset of nodes) can be signaled to check a particular management partition storing a system management command. In this manner, a manager node can broadcast a command to a group of nodes. Further, in some implementations, the nodes can be divided into hierarchical levels, with nodes on a first level having control over nodes on a second level. The nodes can thus be arranged in hierarchical trees, thereby enabling varying degrees of control within the multi-node system.

[0033] In some implementations, the nodes may use a management partition to communicate both a request and a response. For example, assume that, after reading the system management command 250 from the management partition 246, the management unit 210C writes a command acknowledgement (not shown) into the management partition 246. In another example (not shown), a first node may write a request into a management partition for a second node to provide data, and the second node may write the requested data into the same management partition for access by the first node.

[0034] Referring now to Fig. 4, shown is a process 400 for internode management in accordance with some implementations. The process 400 may be performed by the processor(s) 120 shown in Fig. 1B, the inter-node management unit 160 shown in Fig. 1B, and/or the management units 210 shown in Figs. 2 and 3A-3C. The process 400 may be implemented in hardware or machine-readable instructions (e.g., software and/or firmware). The machine-readable instructions are stored in a non-transitory computer readable medium, such as an optical, semiconductor, or magnetic storage device. For the sake of illustration, details of the process 400 may be described below with reference to Figs. 1A-1B, 2, and 3A-3C, which show examples in accordance with some implementations. However, other implementations are also possible.

[0035] At 410, a first node of a multi-node system may store a system management command in a management partition of a global memory. The global memory can be shared by each node of a multi-node computing system. For example, referring to Fig. 3A, the management unit 210A of Node A writes the system management command 250 to the management partition 246.

[0036] At 420, in response to an indication to check the management partition, a second node can access the system management command from the management partition. For example, referring to Fig. 3B, the management unit 210C of Node C can access the system management command 250 stored in the management partition 246. In some implementations, the management unit 210C can check the management partition 246 in response to receiving the signal 214 from Node A. Alternatively, in other implementations, the management unit 210C can check the management partition 246 periodically or according to a schedule, in response to an event, etc.

[0037] At 430, the system management command can be executed to manage the second node. For example, referring to Fig. 3C, Node C can execute the system management command 250 stored in the management partition 246. In some implementations, the system management command can control the power state of the second node, a management mode of the second node, a monitoring state of the second node, etc. After 430, the process 400 is completed.

[0038] Referring now to Fig. 5, shown is a process 500 for internode management and data monitoring in accordance with some implementations. The process 500 may be performed by the processor(s) 120 shown in Fig. 1B, the inter-node management unit 160 shown in Fig. 1B, and/or the management units 210 shown in Figs. 2 and 3A-3C. The process 500 may be implemented in hardware or machine-readable instructions (e.g., software and/or firmware). The machine-readable instructions are stored in a non-transitory computer readable medium, such as an optical, semiconductor, or magnetic storage device. For the sake of illustration, details of the process 500 may be described below with reference to Figs. 1A-1B, 2, and 3A-3C, which show examples in accordance with some implementations. However, other implementations are also possible.

[0039] At 510, multiple nodes of a multi-node system may store monitoring data in shared memory partitions(s). For example, referring to Fig. 2, the management unit 210A of Node A can write monitoring data to management partition 242, the management unit 210B of Node B can write monitoring data to management partition 244, and the management unit 210C of Node C can write monitoring data to management partition 246. In some implementations, the monitoring data can be organized as key-value pairs, and the key can be mapped to a memory address via hash function or table.

[0040] At 520, a determination is made at each node about whether an indication to analyze monitoring data has been received. For example, referring to Fig. 3A, the management unit 210C can determine whether a system management command accessed from the management partition 246 includes analyzing monitoring data from Node A, Node B, and Node C. In other examples, the management unit 210C can determine whether a policy is triggered by a period, a schedule, and/or an event.

[0041] If it is determined at 520 that an indication to analyze monitoring data has not been received by a particular node, then the process 500 can repeat at 510. However, if it is determined at 520 that an indication to analyze monitoring data has been received by the particular node, then at 530, the particular node can access the monitoring data of multiple nodes. In some implementations, the monitoring data of each node can be accessed from a management partition associated with each node. For example, referring to Fig. 2, the management unit 210C of Node C can access management partition 242 to obtain monitoring

data associated with Node A, and can access management partition 244 to obtain monitoring data associated with Node B.

[0042] At 540, the particular node can analyze the monitoring data of the multiple nodes. At 550, the particular node can provide the results of the analysis of the monitoring data. For example, referring to Fig. 2, the management unit 210C and/or Node C can analyze monitoring data associated with Node A, Node B, and Node C, and can then provide analysis results including all of these nodes. In some implementations, the analysis results can be shared by storing them in the global memory. After 550, the process 500 can repeat at 510.

[0043] Note that, while Fig. 5 shows an example in which each node stores monitoring data in an associated management partition, other implementations are possible. For example, in some implementations, all nodes (or a subset of nodes) can store monitoring data in one management partition (or a group of management partitions).

[0044] Data and instructions are stored in respective storage devices, which are implemented as one or multiple computer-readable or machine-readable storage media. The storage media include different forms of non-transitory memory including semiconductor memory devices such as dynamic or static random access memories (DRAMs or SRAMs), erasable and programmable read-only memories (EPROMs), electrically erasable and programmable read-only memories (EEPROMs) and flash memories; non-volatile memory (NVM), magnetic disks such as fixed, floppy and removable disks; other magnetic media including tape; optical media such as compact disks (CDs) or digital video disks (DVDs); or other types of storage devices.

[0045] Note that the instructions discussed above can be provided on one computer-readable or machine-readable storage medium, or alternatively, can be provided on multiple computer-readable or machine-readable storage media distributed in a large system having possibly plural nodes. Such computer-readable or machine-readable storage medium or media is (are) considered to be part of an article (or article of manufacture). An article or article of manufacture can refer to any manufactured single component or multiple components. The storage medium or media can be located either in the machine running the

machine-readable instructions, or located at a remote site from which machine-readable instructions can be downloaded over a network for execution.

[0046] In the foregoing description, numerous details are set forth to provide an understanding of the subject disclosed herein. However, implementations may be practiced without some of these details. Other implementations may include modifications and variations from the details discussed above. It is intended that the appended claims cover such modifications and variations.

What is claimed is:

1. A method comprising:

storing, by a first node of a multi-node computing system, a system management command in a management partition of a global memory, wherein the global memory is shared by each node of the multi-node computing system;

in response to an indication to access the management partition, accessing, by a second node of the multi-node computing system, the system management command from the management partition; and

executing, by the second node of multi-node system, the system management command accessed from the management partition, wherein executing the system management command comprises managing the second node.

2. The method of claim 1, wherein executing the system management command comprises managing at least one selected from a power state of the second node, a monitoring state of the second node, and a management mode of the second node.

3. The method of claim 1, wherein the management partition is reserved for inter-node management of the multi-node computing system.

4. The method of claim 1, further comprising:

storing, by each node of the multi-node computing system, monitoring data in a plurality of management partitions included in the global memory.

5. The method of claim 4, further comprising:

retrieving, by the first node, the monitoring data from the plurality of management partitions; and

analyzing the retrieved monitoring data to obtain combined results.

6. The method of claim 1, further comprising:

storing, by the second node of multi-node system, a response to the system management command into the management partition.

7. The method of claim 1, wherein the indication is one selected from a time-based indication and an inter-node signal.

8. A system comprising:

a plurality of computing nodes, each computing node comprising:

at least one processor;

a local memory; and

an inter-node management unit to:

determine whether a system management command is stored in a management partition of a global memory shared by the plurality of computing nodes, wherein the system management command is generated by a different computing node; and

in response to a determination that the system management command is stored in the management partition of the global memory, execute the system management command to manage the computing node.

9. The system of claim 8, the inter-node management unit further to:

during an initiation of the computing node, map the management partition to a portion of the global memory.

10. The system of claim 8, the inter-node management unit further to:

analyze monitoring data from a plurality of management partitions.

11. The system of claim 10, wherein the monitoring data is organized in key-value pairs.

12. The system of claim 10, each computing node of the plurality of computing nodes further comprising a ring buffer to store the last N instances of the monitoring data.

13. The system of claim 8, wherein the inter-node management unit is to determine whether the system management command is stored in the management partition in response to a signal from the different computing node.

14. An article comprising at least one non-transitory machine-readable storage medium storing instructions that upon execution cause a first computing node to:

receive an indication to access a shared memory partition associated with the first computing node, wherein the first computing node is one of a plurality of computing nodes included in a cluster, wherein the shared memory partition is included in a global memory;

in response to the received indication, access a system management command from the shared memory partition, wherein the system management command is generated by a second computing node of the plurality of computing nodes; and

in response to a determination that the system management command is stored in the shared memory partition, execute the system management command to manage the first computing node.

15. The article of claim 14, wherein the instructions further cause the first computing node to:

store a second system management command in a second shared memory partition, wherein the second shared memory partition is associated with a third computing node of the plurality of computing nodes, and wherein the second system management command is to manage the third computing node.

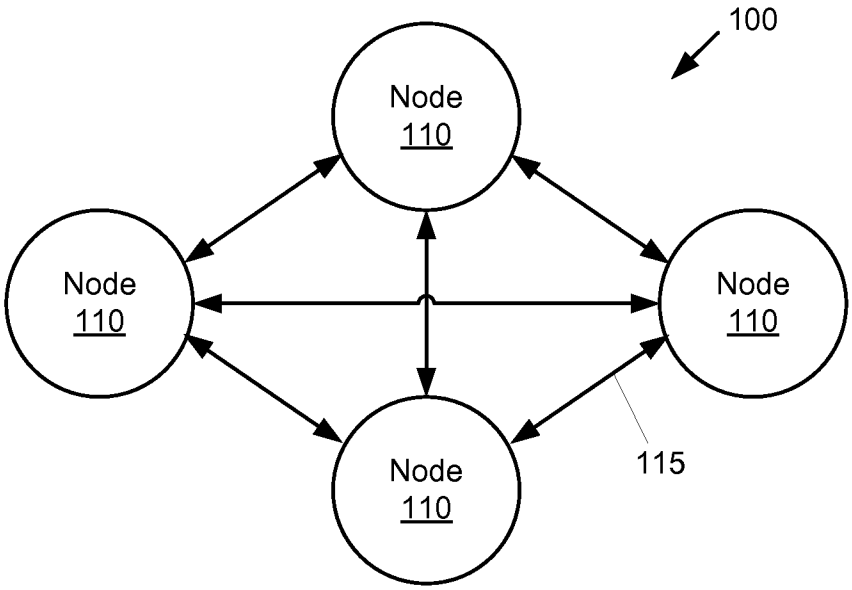
100
Node
110
Node
110
Node
110
Node
110
115

FIG. 1A

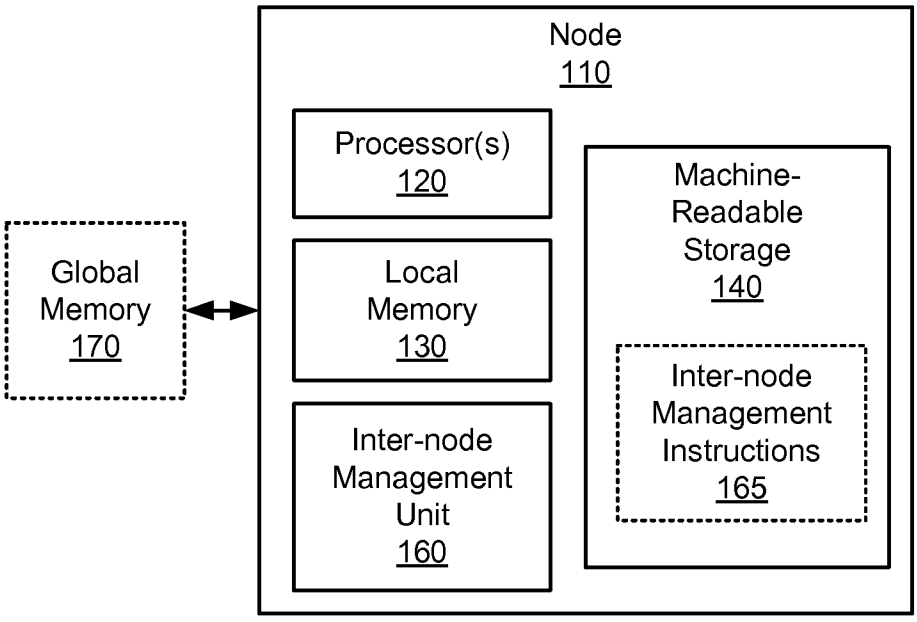
Node
110
Processor(s)
120
Local
Memory
130
Inter-node
Management
Unit
160
Machine-
Readable
Storage
140
Inter-node
Management
Instructions
165
Global
Memory
170

FIG. 1B

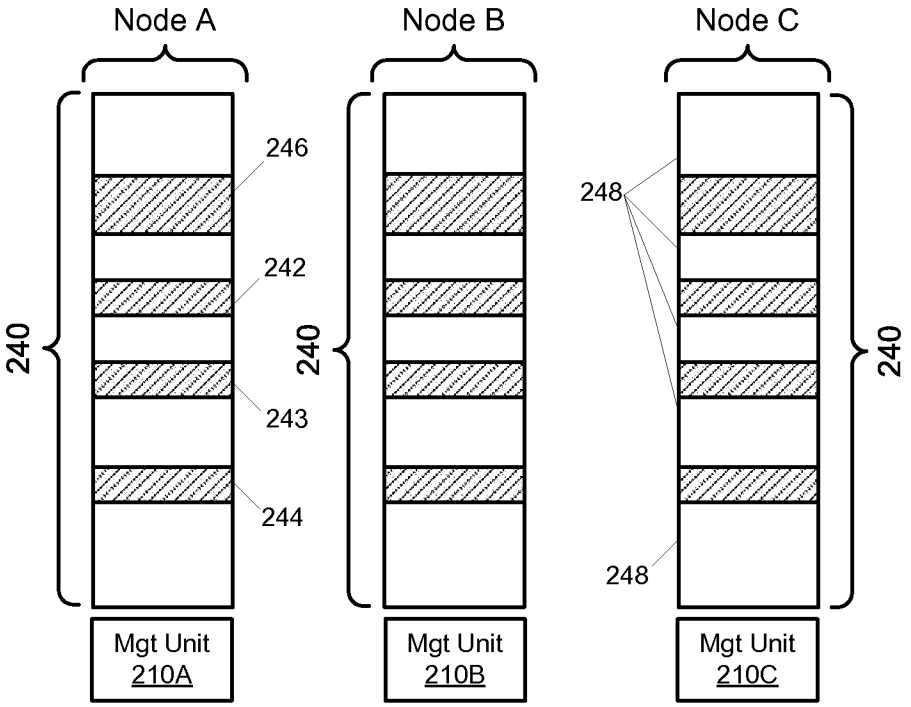
Node A
Node B
Node C
246
242
243
244
248
248
240
240
240
Mgt Unit 210A
Mgt Unit 210B
Mgt Unit 210C

FIG. 2

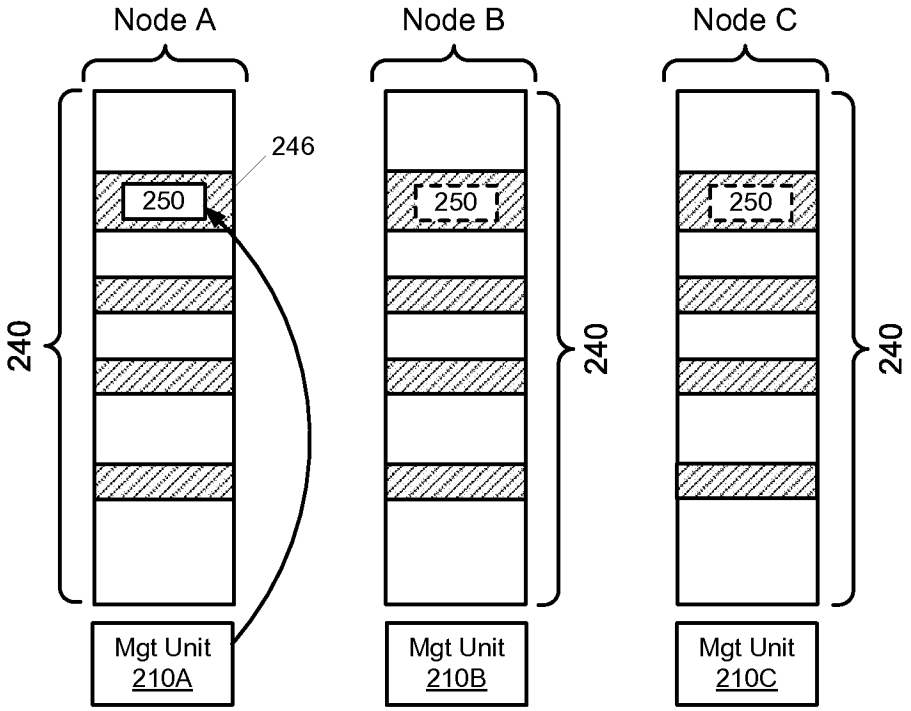
Node A
Node B
Node C
246
250
250
250
240
240
240
Mgt Unit 210A
Mgt Unit 210B
Mgt Unit 210C

FIG. 3A

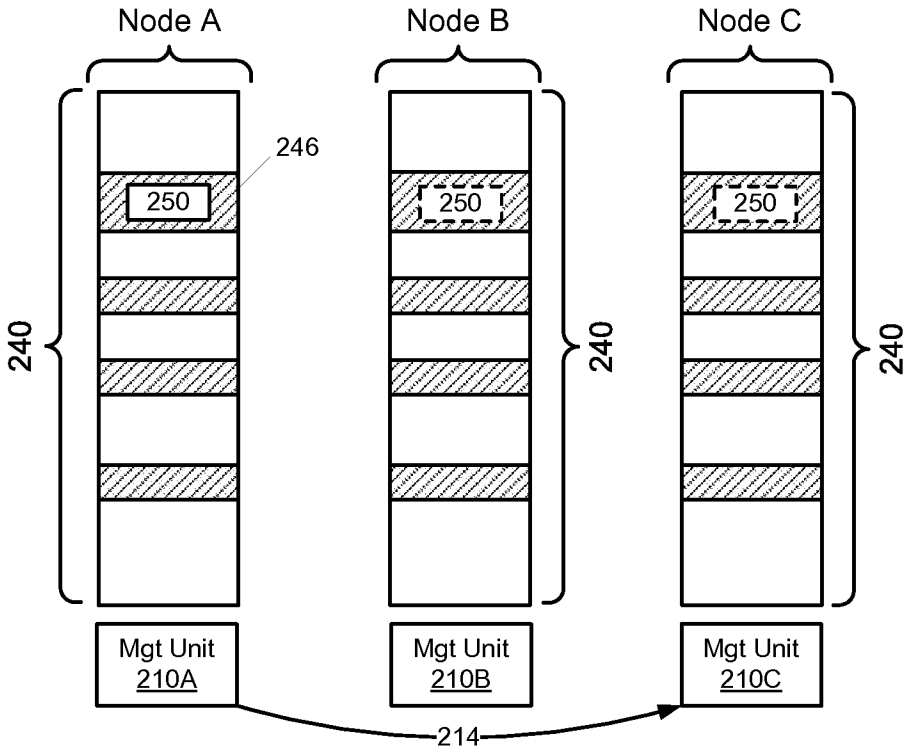
Node A
Node B
Node C
246
250
250
250
240
240
240
Mgt Unit
210A
Mgt Unit
210B
Mgt Unit
210C
214

FIG. 3B

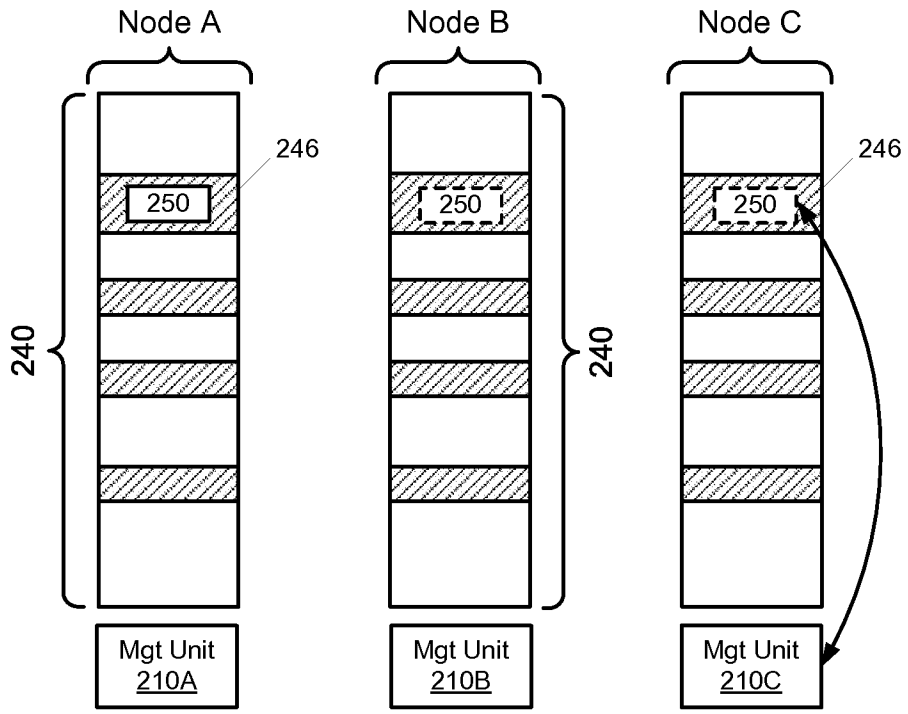
Node A
Node B
Node C
246
250
250
250
246
240
240
Mgt Unit
210A
Mgt Unit
210B
Mgt Unit
210C

FIG. 3C

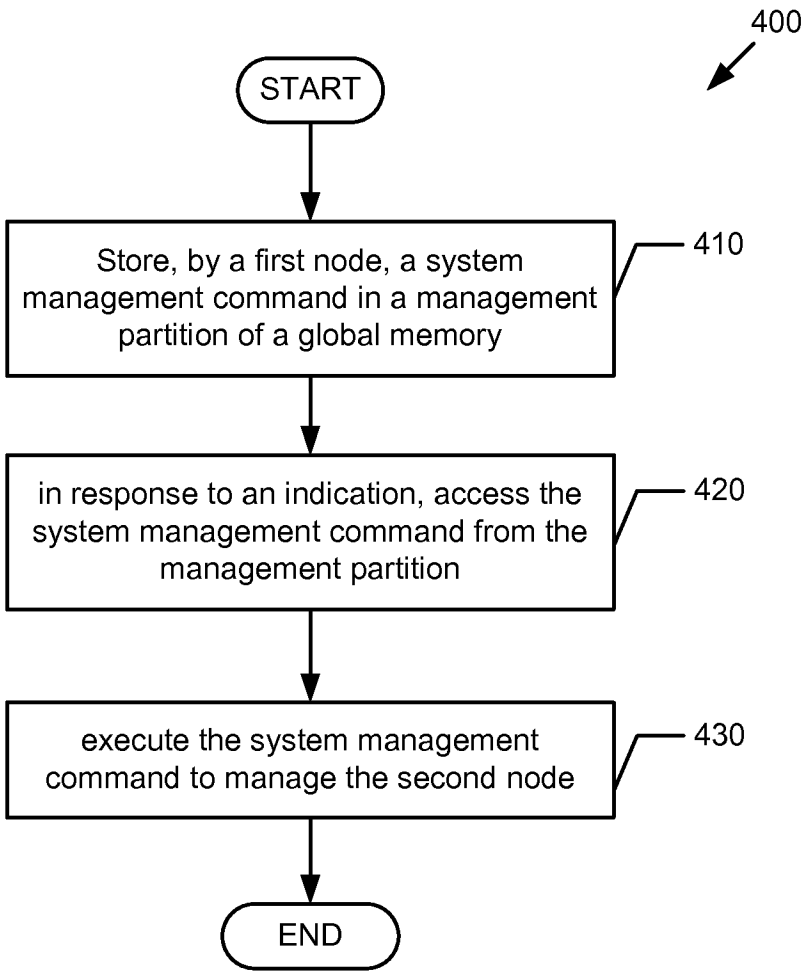
400
START
Store, by a first node, a system management command in a management partition of a global memory
410
in response to an indication, access the system management command from the management partition
420
execute the system management command to manage the second node
430
END

FIG. 4

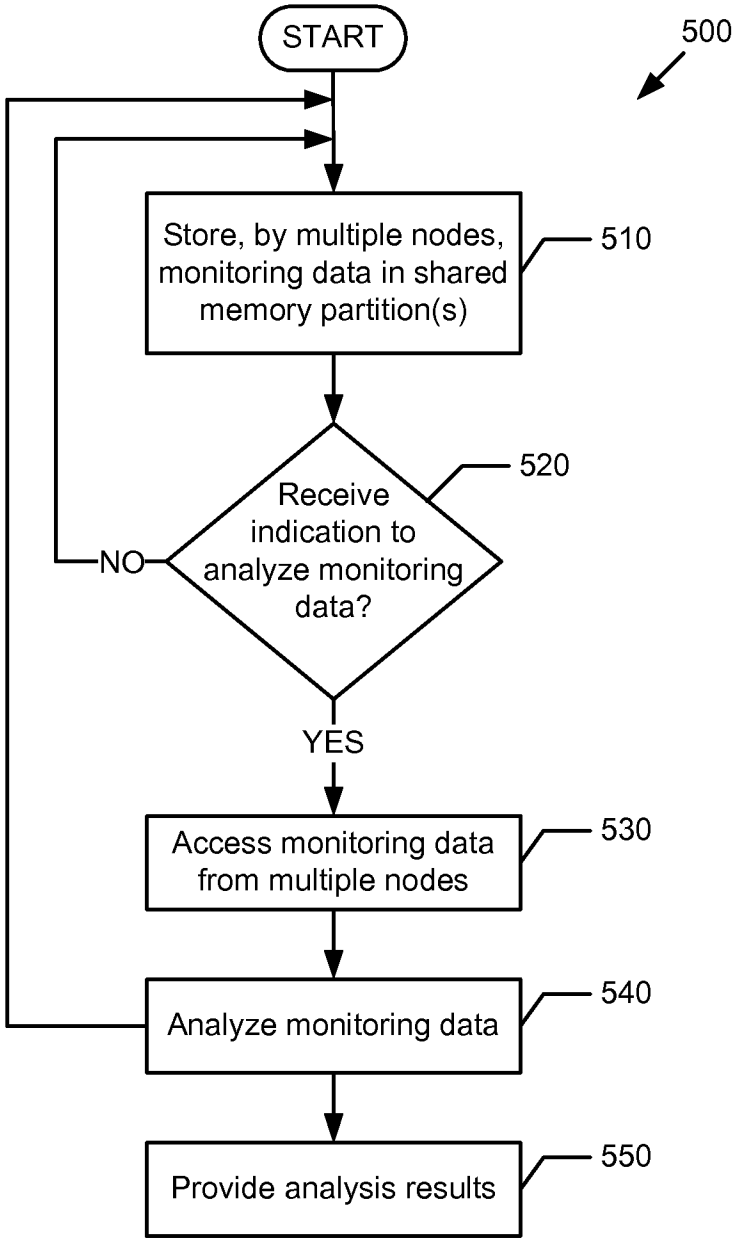
START
500
Store, by multiple nodes, monitoring data in shared memory partition(s)
510
Receive indication to analyze monitoring data?
520
NO
YES
Access monitoring data from multiple nodes
530
Analyze monitoring data
540
Provide analysis results
550

FIG. 5

INTERNATIONAL SEARCH REPORT

International application No.
PCT/US2014/035829

A. CLASSIFICATION OF SUBJECT MATTER

G06F 13/14(2006.01)i, G06F 12/00(2006.01)i

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
G06F 13/14; G11C 5/00; G06F 9/54; G06F 12/00; G06F 12/08

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched
Korean utility models and applications for utility models
Japanese utility models and applications for utility models

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)
eKOMPASS(KIPO internal) & Keywords: shared memory, global memory, multi-node, management, and similar terms

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
Y	US 2013-0339981 A1 (SHOJIRO ISHIBASHI et al.) 19 December 2013 See paragraphs [0004], [0007], [0032], [0036], [0039]-[0040], [0056], [0062]-[0063], [0065], [0080], and [0096]-[0099]; claim 1; and figures 1-2, 7, and 11-12.	1-15
Y	US 2012-0233409 A1 (JONATHAN ROSS et al.) 13 September 2012 See paragraphs [0006], [0018], [0026], and [0031]; claims 1 and 10; and figure 3.	1-15
A	US 2009-0198891 A1 (LAKSHIMARAYANA B. ARIMILLI et al.) 06 August 2009 See paragraphs [0025] and [0077]-[0081]; and figures 3A-3B.	1-15
A	US 2012-0331239 A1 (FONG PONG) 27 December 2012 See paragraphs [0004] and [0051]-[0055]; and figure 2.	1-15
A	US 2003-0115402 A1 (FREDRIK DAHLGREN et al.) 19 June 2003 See paragraphs [0022]-[0023] and [0036]-[0041]; and figure 1.	1-15

☐ Further documents are listed in the continuation of Box C.

☒ See patent family annex.

* Special categories of cited documents:
"A" document defining the general state of the art which is not considered to be of particular relevance
"E" earlier application or patent but published on or after the international filing date
"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)
"O" document referring to an oral disclosure, use, exhibition or other means
"P" document published prior to the international filing date but later than the priority date claimed
"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents,such combination being obvious to a person skilled in the art
"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
30 January 2015 (30.01.2015)	**30 January 2015 (30.01.2015)**

Name and mailing address of the ISA/KR
International Application Division
Korean Intellectual Property Office
189 Cheongsa-ro, Seo-gu, Daejeon Metropolitan City, 302-701,
Republic of Korea
Facsimile No. ++82 42 472 3473

Authorized officer
NHO, Ji Myong
Telephone No. +82-42-481-8528

INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No.
PCT/US2014/035829

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 2013-0339981 A1	19/12/2013	CA 2826100 A1	09/08/2012
		JP 2012-159988 A	23/08/2012
		WO 2012-105560 A1	09/08/2012
US 2012-0233409 A1	13/09/2012	US 8832388 B2	09/09/2014
US 2009-0198891 A1	06/08/2009	US 7966454 B2	21/06/2011
US 2012-0331239 A1	27/12/2012	US 2008-301379 A1	04/12/2008
		US 2011-167226 A1	07/07/2011
		US 7941613 B2	10/05/2011
		US 8281081 B2	02/10/2012
		US 8812795 B2	19/08/2014
US 2003-0115402 A1	19/06/2003	None	