



- (51) **International Patent Classification:**  
*G06F 17/40* (2006.01) *G06F 3/06* (2006.01)
- (21) **International Application Number:** PCT/US2016/024709
- (22) **International Filing Date:** 29 March 2016 (29.03.2016)
- (25) **Filing Language:** English
- (26) **Publication Language:** English
- (30) **Priority Data:** 201641004903 11 February 2016 (11.02.2016) IN
- (71) **Applicant:** HEWLETT PACKARD ENTERPRISE DEVELOPMENT LP [US/US]; 11445 Compaq Center Drive West, Houston, TX 77070 (US).
- (72) **Inventors:** MOHANTA, Taranisen; Sy. No. 192, Whitefield Road, Mahadevapura Post, Bangalore 560 048, Karnataka (IN). MUDDI, Leena; Sy. No. 192, Whitefield Road, Mahadevapura Post, Bangalore 560 048, Karnataka (IN). UMESH, Abhijith; Sy. No.192, Whitefield Road, Mahadevapura Post, Bangalore 560 048, Karnataka (IN). MAHESH, Keshetti; Sy. No.192, Whitefield Road, Mahadevapura Post, Bangalore 560 048, Karnataka (IN).
- (74) **Agents:** STACY, Nathan, E. et al.; Hewlett Packard Enterprise, Mail Stop 79, 3404 E. Harmony Road, Fort Collins, CO 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, 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, ST, 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).

[Continued on next page]

## (54) Title: DISTRIBUTED STORAGE SYSTEM

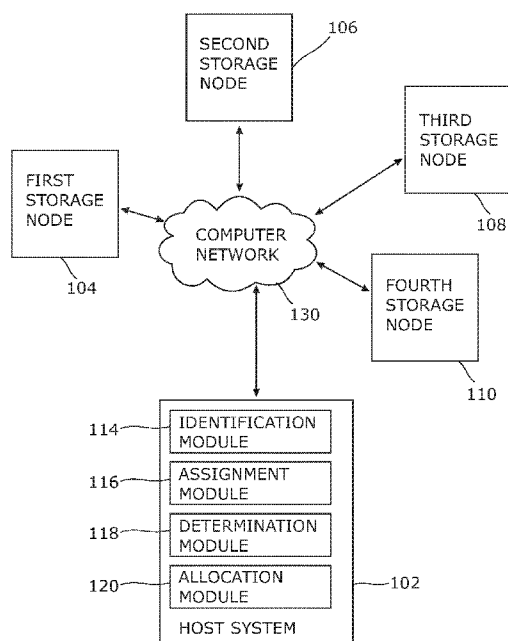


Fig. 1

100

(57) **Abstract:** Some examples relate to a distributed storage system. In an example, a plurality of factors related to storage volumes distributed across a plurality of storage nodes may be identified. A value may be assigned to each of the factors for each of the storage volumes. A priority may be determined for each of the storage volumes, based on the value assigned to each of the factors of respective storage volumes. Data stored on the storage volumes may be dynamically allocated across the plurality of storage nodes, based on respective priority of the storage volumes.

**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))*

**Published:**

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

## DISTRIBUTED STORAGE SYSTEM

Background

**[001]** Storage systems may be considered as an integral part of modern day computing. Whether it is a small start-up or a large enterprise, organizations these days may need to deal with a vast amount of data that could range from a few terabytes to multiple petabytes. Storage systems or devices provide a useful way of storing and organizing such large amounts of data. Enterprises may be looking at more efficient ways of utilizing their storage resources.

Brief Description of the Drawings

**[002]** For a better understanding of the solution, examples will now be described, purely by way of example, with reference to the accompanying drawings, in which:

**[003]** FIG. 1 is a block diagram of an example computing environment for a distributed storage system;

**[004]** FIG. 2 is a block diagram of an example system for a distributed storage system;

**[005]** FIG. 3 is a flowchart of an example method of a distributed storage system; and

**[006]** FIG. 4 is a block diagram of an example system for a distributed storage system.

Detailed Description

**[007]** Efficient data management may be desirable for the success of an organization. Whether it is a private company, a government undertaking, an educational institution, or a new start-up, managing data (for example, customer data, vendor data, patient

data, etc.) in an appropriate manner may be crucial for the existence and growth of an enterprise. Storage systems may play a useful role in this regard. A storage system allows an enterprise to store and organize data, which may be analyzed to derive useful information for a user.

**[008]** Typically, in a distributed storage system, multiple storage nodes may be interconnected with each other. Data of volumes created on a distributed storage system may be spread across multiple storage nodes, which may include various types of storage devices. These storage devices may differ from each other in various ways, for example, storage capacity, cost, availability, performance, interoperability, and scalability. Storing volume data across these varied storage nodes regardless of the data stored on the storage volumes and characteristics of a member storage device(s) of a storage node may not be an efficient use of the storage devices of a distributed storage system. It is desirable to have an auto-tiering solution that may take into account a factor(s) associated with storage volumes of a distributed storage system, and dynamically assigns data stored on these storage volumes to a plurality of storage nodes based on a characteristic(s) of the latter.

**[009]** To address this issue, the present disclosure describes various examples for auto-tiering in a distributed storage system. In an example, a plurality of factors related to storage volumes distributed across a plurality of storage nodes may be identified. A value may be assigned to each of the factors for each of the storage volumes. A priority may be determined for each of the storage volumes, based on the value assigned to each of the factors of respective storage volumes. Data stored on the storage volumes may be dynamically allocated across the plurality of storage nodes, based on respective priorities of the storage volumes.

**[0010]** FIG. 1 is a block diagram of an example computing environment 100 for a distributed storage system. In one example, computing environment 100 is configured for auto-tiering for a distributed storage system. In an example, computing environment

100 may include a computing device 102, a first storage node 104, a second storage node 106, a third storage node 108, and a fourth storage node 110. Although one computing device and four storage nodes are shown in FIG. 1, other examples of this disclosure may include more than one computing device, and more or less than four storage nodes.

**[0011]** Computing device (or host system) 102 may represent any type of computing system capable of reading machine-executable instructions. Examples of computing device 102 may include, without limitation, a server, a desktop computer, a notebook computer, a tablet computer, a thin client, a mobile device, a personal digital assistant (PDA), a phablet, and the like.

**[0012]** Storage nodes 104, 106 108, and 110 may each be a storage device. The storage device may be an internal storage device, an external storage device, or a network attached storage device. Some non-limiting examples of the storage device may include a hard disk drive, a storage disc (for example, a CD-ROM, a DVD, etc.), a storage tape, a solid state drive (SSD), a USB drive, a Serial Advanced Technology Attachment (SATA) disk drive, a Fibre Channel (FC) disk drive, a Serial Attached SCSI (SAS) disk drive, a magnetic tape drive, an optical jukebox, and the like. In an example, storage nodes 104, 106 108, and 110 may each be a Direct Attached Storage (DAS) device, a Network Attached Storage (NAS) device, a Redundant Array of Inexpensive Disks (RAID), a data archival storage system, or a block-based device over a storage area network (SAN). In another example, storage nodes 104, 106 108, and 110 may each be a storage array, which may include a storage drive or plurality of storage drives (for example, hard disk drives, solid state drives, etc.).

**[0013]** In an example, storage nodes 104, 106 108, and 110 may be part of a distributed storage system. Storage nodes may be in communication with each other, for example, via a computer network 130. Such a computer network 130 may be a wireless or wired network. Such a computer network 130 may include, for example, a Local Area Network

(LAN), a Wireless Local Area Network (WAN), a Metropolitan Area Network (MAN), a Storage Area Network (SAN), a Campus Area Network (CAN), or the like. Further, such a computer network 130 may be a public network (for example, the Internet) or a private network (for example, an intranet). Computing device 102 may be in communication with any or all of the storage nodes, for example, via a computer network 130. Such a computer network may be similar to the computer network described above.

**[0014]** Storage nodes 104, 106 108, and 110 may communicate with computing device 102 via a suitable interface or protocol such as, but not limited to, Fibre Channel, Fibre Connection (FICON), Internet Small Computer System Interface (iSCSI), HyperSCSI, and ATA over Ethernet.

**[0015]** In an example, physical storage space provided by storage nodes 104, 106 108, and 110 may be presented as a logical storage space to computing device 102. Such logical storage space (also referred as “logical volume”, “virtual disk”, or “storage volume”) may be identified using a “Logical Unit Number” (LUN). In another example, physical storage space provided by storage nodes may be presented as multiple logical volumes to computing device 102. In such case, each of the logical storage spaces may be referred to by a separate LUN. In an example, a storage volume(s) may be distributed across a plurality of storage nodes.

**[0016]** In an example, storage nodes may be graded or ranked based on a factor or a set of factors. Some non-limiting examples of the factors may include cost of a storage node, capacity of a storage node, and a performance-related parameter of a storage node. Thus, by way of an example, if cost of a storage node is a factor, then storage nodes may be graded accordingly, wherein a storage node comprising of relatively high cost storage devices may be ranked higher compared to a storage node comprising of low cost storage devices. Likewise, another factor(s) may be considered to grade the storage nodes. This may thus result into formation of multiple tiers of storage nodes.

**[0017]** In an example, storage nodes may be divided into a plurality of groups based a factor or set of factors. Some non-limiting examples of the factors may include cost of a storage node, capacity of a storage node, and a performance-related parameter of a storage node. Thus, by way of an example, if cost of a group of storage nodes is a factor, then each group of storage nodes may be graded accordingly, wherein a group of storage nodes comprising of relatively high cost storage devices may be ranked higher compared to a group of storage nodes comprising of low cost storage devices. Likewise, another factor(s) may be considered to grade each group of storage nodes. This may result into formation of multiple tiers of storage nodes (for example, tier 0, tier 1, tier 2, and so on), wherein each tier includes a group of storage nodes.

**[0018]** In an example, computing device 102 may include an identification module 114, an assignment module 116, a determination module 118, and an allocation module 120. The term “module” may refer to a software component (machine readable instructions), a hardware component or a combination thereof. A module may include, by way of example, components, such as software components, processes, tasks, co-routines, functions, attributes, procedures, drivers, firmware, data, databases, data structures, Application Specific Integrated Circuits (ASIC) and other computing devices. A module may reside on a volatile or non-volatile storage medium and interact with a processor of a computing device (e.g. 102).

**[0019]** Some of the example functionalities that may be performed by identification module 114, assignment module 116, determination module 118, and allocation module 120 are described in reference to FIG. 2 below.

**[0020]** FIG. 2 is a block diagram of an example system 200 for a distributed storage system. In one example, system 200 is configured for auto-tiering for a distributed storage system. In an example, system 200 may be analogous to computing device 102 of FIG. 1, in which like reference numerals correspond to the same or similar, though perhaps not identical, components. For the sake of brevity, components or reference

numerals of FIG. 2 having a same or similarly described function in FIG. 1 are not being described in connection with FIG. 2. The components or reference numerals may be considered alike.

**[0021]** System 200 may represent any type of computing system capable of reading machine-executable instructions. Examples of computing device 102 may include, without limitation, a server, a desktop computer, a notebook computer, a tablet computer, a thin client, a mobile device, a personal digital assistant (PDA), a phablet, and the like. In an example, system 200 may include an identification module 114, an assignment module 116, a determination module 118, and an allocation module 120.

**[0022]** Identification module 114 may be used to identify a plurality of factors related to a storage volume(s) that may be present on a storage node or a plurality of storage nodes (for example, 104, 106 108, and 110). Assignment module 116 may be used to assign a value (numerical or non-numerical) to each of the plurality of factors for each of the storage volumes.

**[0023]** In an example, the factor may include a priority assigned by a user to respective storage volumes. A user (for example, a system administrator) may assign a priority to respective storage volumes based on some criterion. In an example, the criterion may include the type or value of data stored in a storage volume. There may be a scenario where data stored on a storage volume may be relatively more valuable compared to data stored on another storage volume. In such case, the latter storage volume may be assigned a lower priority compared to the earlier storage volume.

**[0024]** In another example, the factor may include the configuration status of respective storage volumes to handle “high” I/O (input/output) operations. A user (or system) may define a threshold value for number of I/O operations that may be allowed to be performed on a storage volume. In an example, this threshold value may be “enabled” or “disabled” for a storage volume. If the threshold value is enabled for a storage

volume, it may indicate that the storage volume is capable of handling I/O (input/output) operations beyond the defined threshold value i.e. the storage volume may be capable of handling a high number of I/O operations. On the other hand, if the threshold value is disabled for a storage volume, it may indicate that the storage volume is incapable of handling I/O (input/output) operations beyond the threshold value.

**[0025]** In another example, the factor may include the status of Write Amplification Factor (WAF) of respective storage volumes. Write amplification is an event that occurs when the actual amount of written physical data in a SSD is more than the amount of logical data that is written by the host computer. WAF is a numerical value that represents the ratio of physical writes to logical writes. A user may define a WAF policy wherein a user may set an endurance threshold value (or WAF value) beyond which a write may not take place on a SSD. This value may be in percentage (for example, 90%). In an example, this threshold value may be “enabled” or “disabled” for a storage volume. If the threshold value is enabled for a storage volume, it may indicate that once the storage volume reaches the WAF value, no further write may take place on the SSD device that hosts the storage volume. On the other hand, if the threshold value is disabled for a storage volume, it may indicate that a further write may take place on the SSD device that hosts the storage volume.

**[0026]** In another example, the factor may include volume replication factor (RF) of respective storage volumes. RF defines the number of raw data files that may be kept for a particular data. In other words, a replication factor may refer to the total number of replicas of a data file across a distributed storage system. A relatively higher replication factor may be defined for a storage volume that stores valuable data compared to a storage volume that stores relatively less valuable data. In an example, the RF factor may receive either of the following values: performance or capacity. If the RF value is enabled for “capacity”, it may indicate that the data in the storage volume may be stored in a lower ranked storage node. On the other hand, if the RF value is

enabled for “performance”, it may indicate that the data in the storage volume may be stored in a higher ranked storage node.

**[0027]** In another example, the factor may include the type of volume write workload of respective storage volumes. For example, whether the write workload of a storage volume includes a sequential write workload or random write workload. In another example, the factor may include the type of volume read workload of respective storage volumes. For example, whether the read workload of a storage volume includes a sequential read workload or random read workload.

**[0028]** In another example, the factor may include the “preferred node” status of a storage volume. A user (or system) may define a “preferred node” status of a storage volume that may indicate the status of the storage node to preferentially perform a data read or write operation vis-à-vis other storage nodes in a distributed storage system. In an example, this status may be “enabled” or “disabled” for a storage volume. If the “preferred node” status is enabled for a storage volume, it may indicate that the storage volume is a preferred node for a data read or write operation vis-à-vis other storage nodes in a distributed storage system. On the other hand, if the “preferred node” status is disabled for a storage volume, it may indicate that the storage volume is not a preferred node for a data read or write operation vis-à-vis other storage nodes in a distributed storage system.

**[0029]** In another example, the factor may include the encryption status of respective storage volumes. In an example, this status may be “enabled” or “disabled” for a storage volume. If the encryption status is enabled for a storage volume, it may indicate that the storage volume is a preferred node for a data read or write operation vis-à-vis an unencrypted storage node(s) in a distributed storage system.

**[0030]** In another example, the factor may include the compression status of respective storage volumes. In an example, this status may be “enabled” or “disabled” for a

storage volume. If the compression status is enabled for a storage volume, it may indicate that the storage volume is a preferred node for a data read or write operation vis-à-vis an uncompressed storage node(s) in a distributed storage system.

**[0031]** Determination module 118 may be used to determine a priority for each of the storage volumes, based on the value assigned to each of the factors related to respective storage volumes. Thus, based on the values assigned to the factors related to a storage volume, determination module 118 may determine a priority for the storage volume. Likewise, the same determination may be made for other storage volumes in the storage system.

**[0032]** In an example, relative priorities (for example, non-numerical values such as High, Medium, and Low, or numerical values such as 1, 2, 3, and 4) may be assigned to the storage volumes.

**[0033]** A relatively higher priority may be assigned to a storage volume in certain scenarios. For example, if “high” priority is assigned by a user to a storage volume; if RAF value for a storage volume is for “performance”; if the volume read workload type is of sequential type; and if the preferred node status of a storage volume is enabled.

**[0034]** Allocation module 120 may be used to dynamically allocate data stored on a storage volume to a storage node(s), based on the priority determined for the storage volume. Likewise, data stored on other storage volumes may be stored across the plurality of storage nodes, based on respective priorities of the storage volumes. As mentioned earlier, storage nodes may be graded or ranked based on a factor or a set of factors. Some non-limiting examples of the factors may include cost of a storage node, capacity of a storage node, and a performance-related parameter of a storage node. Thus, by way of an example, if cost of a storage node is a factor then storage nodes may be graded accordingly, wherein a storage node comprising of relatively high cost storage devices may be ranked higher compared to a storage node comprising of low cost

storage devices. Likewise, another factor(s) may be considered to grade the storage nodes. This may thus result into formation of multiple tiers of storage nodes.

**[0035]** In an example, allocation module 120 may be used to dynamically allocate data stored on the storage volumes across the plurality of storage nodes, based on respective priorities of the storage volumes. Thus, data stored on a storage volume having a specific priority may be stored in a correspondingly ranked storage node. For example, data of a higher priority storage volume may be stored on a higher priority storage node. Likewise, data of a relatively lower priority storage volume may be stored on a relatively lower priority storage node.

**[0036]** Likewise, as mentioned earlier, storage nodes may be divided into a plurality of groups based a factor or set of factors, wherein each group of storage nodes may be graded depending on the factor(s). This may result into formation of multiple tiers of storage nodes (for example, tier 0, tier 1, tier 2, and so on), wherein each tier includes a group of storage nodes. In this context, allocation module may be used to dynamically allocate data stored on the storage volumes across the plurality of groups of storage nodes, based on respective priorities of the storage volumes.

**[0037]** In an example, allocation module may be used to dynamically allocate data stored on the storage volumes across the plurality of groups of storage nodes, based on a match between the respective priorities of the storage volumes and respective rankings of the storage nodes. Thus, data stored on a storage volume having a specific priority may be stored in a corresponding ranked group of storage nodes. For example, data of a higher priority storage volume may be stored on a higher priority group of storage nodes. Likewise, data of a relatively lower priority storage volume may be stored on a relatively lower priority group of storage nodes.

**[0038]** Table 1 illustrates distribution of data of storage volumes across two groups of storage nodes (i.e. Tier 0 and Tier 1, where Tier 0 may represent a higher tier), according

to an example. A priority is determined for each of the storage volumes based on the values assigned to the factors associated with respective storage volumes. Based on respective priority of the storage volumes, data stored on the storage volumes is allocated to either of the two tiers.

[0039] Table 1

| User assigned Volume Priority | High I/O | WAF Policy | Volume replication factor | Volume Write workload type | Volume read workload type | Volume preferred node | Priority Based policies on |
|-------------------------------|----------|------------|---------------------------|----------------------------|---------------------------|-----------------------|----------------------------|
| high                          | enabled  | enabled    | performance               | random                     | sequential                | enabled               | Tier 0                     |
| high                          | disabled | enabled    | performance               | sequential                 | sequential                | disabled              | Tier 0                     |
| low                           | enabled  | enabled    | capacity                  | random                     | sequential                | enabled               | Tier 0                     |
| high                          | disabled | disabled   | performance               | sequential                 | random                    | disabled              | Tier 1                     |
| low                           | enabled  | disabled   | capacity                  | random                     | sequential                | disabled              | Tier 1                     |
| low                           | disabled | disabled   | capacity                  | sequential                 | random                    | disabled              | Tier 1                     |
| high                          | disabled | disabled   | performance               | random                     | sequential                | disabled              | Tier 0                     |
| high                          | disabled | disabled   | performance               | random                     | random                    | disabled              | Tier 1                     |
| low                           | enabled  | disabled   | performance               | random                     | sequential                | enabled               | Tier 0                     |
| low                           | disabled | disabled   | capacity                  | random                     | sequential                | disabled              | Tier 1                     |
| low                           | disabled | disabled   | performance               | sequential                 | random                    | enabled               | Tier 0                     |
| high                          | enabled  | disabled   | performance               | sequential                 | random                    | disabled              | Tier 0                     |
| high                          | disabled | disabled   | performance               | random                     | sequential                | enabled               | Tier 1                     |
| high                          | enabled  | enabled    | capacity                  | random                     | sequential                | disabled              | Tier 1                     |

**[0040]** In an example, a value assigned to a factor related to a storage volume may change over a period of time. In this case, determination module 118 may redetermine the priority of the storage volume. The allocation module 120 may allocate data stored on the storage volume to a storage node based on the redetermined priority of the storage volume.

**[0041]** FIG. 3 is a flowchart of an example method 300 for a distributed storage system. In one example, method 300 is configured for auto-tiering for a distributed storage system. The method 300, which is described below, may at least partially be executed on a computer system, for example, computing device 100 of FIG. 1 or system 200 of FIG. 2. However, other computing devices may be used as well. At block 302, a plurality of factors related to storage volumes distributed across a plurality of storage nodes may be identified. At block 304, a value may be assigned to each of the factors for each of the storage volumes. At block 306, a priority may be determined for each of the storage volumes, based on the value assigned to each of the factors of respective storage volumes. At block 308, data stored on the storage volumes may be dynamically allocated across the plurality of storage nodes, based on respective priorities of the storage volumes.

**[0042]** FIG. 4 is a block diagram of an example system 400 for a distributed storage system. In one example, system 400 is configured for auto-tiering for a distributed storage system. System 400 includes a processor 402 and a machine-readable storage medium 404 communicatively coupled through a system bus. In an example, system 400 may be analogous to computing device 100 of FIG. 1 or system 200 of FIG. 2. Processor 402 may be any type of Central Processing Unit (CPU), microprocessor, or processing logic that interprets and executes machine-readable instructions stored in machine-readable storage medium 404. Machine-readable storage medium 404 may be a random access memory (RAM) or another type of dynamic storage device that may store information and machine-readable instructions that may be executed by processor 402. For example, machine-readable storage medium 404 may be Synchronous DRAM

(SDRAM), Double Data Rate (DDR), Rambus DRAM (RDRAM), Rambus RAM, etc. or a storage memory media such as a floppy disk, a hard disk, a CD-ROM, a DVD, a pen drive, and the like. In an example, machine-readable storage medium 404 may be a non-transitory machine-readable medium. Machine-readable storage medium 404 may store instructions 406, 408, 410, and 412. In an example, instructions 406 may be executed by processor 402 to identify a plurality of factors related to storage volumes present across a plurality of storage devices of different types. Instructions 408 may be executed by processor 402 to assign a value to each of the factors for each of the storage volumes. Instructions 410 may be executed by processor 402 to determine a priority for each of the storage volumes, based on the value assigned to each of the factors of respective storage volumes. Instructions 412 may be executed by processor 402 to dynamically allocate data stored on the storage volumes present across the plurality of storage devices of different types, based on respective priorities of the storage volumes.

**[0043]** For the purpose of simplicity of explanation, the example method of FIG. 3 is shown as executing serially, however it is to be understood and appreciated that the present and other examples are not limited by the illustrated order. The example systems of FIGS. 1, 2, and 4, and method of FIG. 3 may be implemented in the form of a computer program product including computer-executable instructions, such as program code, which may be run on any suitable computing device in conjunction with a suitable operating system (for example, Microsoft Windows, Linux, UNIX, and the like). Examples within the scope of the present solution may also include program products comprising non-transitory computer-readable media for carrying or having computer-executable instructions or data structures stored thereon. Such computer-readable media can be any available media that can be accessed by a general purpose or special purpose computer. By way of example, such computer-readable media can comprise RAM, ROM, EPROM, EEPROM, CD-ROM, magnetic disk storage or other storage devices, or any other medium which can be used to carry or store desired program code in the form of computer-executable instructions and which can be accessed by a general

purpose or special purpose computer. The computer readable instructions can also be accessed from memory and executed by a processor.

**[0044]** It should be noted that the above-described examples of the present solution is for the purpose of illustration. Although the solution has been described in conjunction with a specific example thereof, numerous modifications may be possible without materially departing from the teachings of the subject matter described herein. Other substitutions, modifications and changes may be made without departing from the spirit of the present solution. All of the features disclosed in this specification (including any accompanying claims, abstract and drawings), and/or all of the parts of any method or process so disclosed, may be combined in any combination, except combinations where at least some of such features and/or parts are mutually exclusive.

Claims:

## 1. A method comprising:

identifying a plurality of factors related to storage volumes distributed across a plurality of storage nodes;

assigning a value to each of the factors for each of the storage volumes;

determining a priority for each of the storage volumes, based on the value assigned to each of the factors of respective storage volumes; and

dynamically allocating data stored on the storage volumes across the plurality of storage nodes, based on respective priorities of the storage volumes.

2. The method of claim 1, wherein the factors include one of a priority assigned by a user to respective storage volumes, configuration status of respective storage volumes to handle high I/O operations, status of Write Amplification Factor (WAF) of respective storage volumes, volume replication factor of respective storage volumes, type of volume write workload of respective storage volumes, type of volume read workload of respective storage volumes, preferred node status of respective storage volumes, encryption status of respective storage volumes, and compression status of respective storage volumes.

3. The method of claim 1, further comprising ranking each of the storage nodes based on a factor.

4. The method of claim 3, wherein the factor includes one of cost of respective storage volumes, capacity of respective storage volumes, and a performance-related parameter of respective storage volumes.

5. The method of claim 3, wherein the allocating comprises dynamically allocating data stored on the storage volumes across the plurality of storage nodes, based on a match

between the respective priorities of the storage volumes and respective rankings of the storage nodes.

6. A system comprising:

- an identification module to identify a plurality of factors related to storage volumes distributed across a plurality of graded storage devices;

- an assignment module to assign a value to each of the factors for each of the storage volumes;

- a determination module to determine a priority for each of the storage volumes, based on the value assigned to each of the factors of respective storage volumes; and

- an allocation module to dynamically allocate data stored on the storage volumes across the plurality of graded storage devices, based on respective priorities of the storage volumes.

7. The system of claim 6, wherein the storage devices include a solid-state device (SSD).

8. The system of claim 6, wherein the storage devices include different types of storage devices.

9. The system of claim 6, wherein:

- the plurality of graded storage devices are divided into groups; and

- the allocation module to dynamically allocate data stored on the storage volumes across the groups of graded storage devices, based on respective priorities of the storage volumes.

10. The system of claim 6, wherein:

- the assignment module to assign a ranking to each of the groups of graded storage devices; and

the allocation module to dynamically allocate data stored on the storage volumes across each of the groups of graded storage devices, based on respective rankings of the groups of graded storage devices.

11. A non-transitory machine-readable storage medium comprising instructions for a distributed storage system, the instructions executable by a processor to:

- identify a plurality of factors related to storage volumes present across a plurality of storage devices of different types;

- assign a value to each of the factors for each of the storage volumes;

- determine a priority for each of the storage volumes, based on the value assigned to each of the factors of respective storage volumes; and

- dynamically allocate data stored on the storage volumes present across the plurality of storage devices of different types, based on respective priorities of the storage volumes.

12. The storage medium of claim 11, wherein the factors include one of a priority assigned by a user to respective storage volumes, configuration status of respective storage volumes to handle high I/O operations, status of Write Amplification Factor (WAF) of respective storage volumes, volume replication factor of respective storage volumes, type of volume write workload of respective storage volumes, type of volume read workload of respective storage volumes, preferred node status of respective storage volumes, encryption status of respective storage volumes, and compression status of respective storage volumes.

13. The storage medium of claim 11, further comprising instructions to:

- determine that a value assigned to a factor among the plurality of factors has changed for a storage volume;

- in response to the determination, redetermine the priority for the storage volume; and

allocate data stored on the storage volume to a storage device among the plurality of storage devices, based on the redetermined priority of the storage volume.

14. The storage medium of claim 11, wherein:

the plurality of storage devices are ranked into groups based on a factor; and

the allocation module to dynamically allocate data stored on the storage volumes across the groups of storage devices, based on respective rankings of the groups.

15. The storage medium of claim 14, wherein the factor includes one of cost of respective storage devices, capacity of respective storage devices, and a performance-related parameter of respective storage devices.

1/4

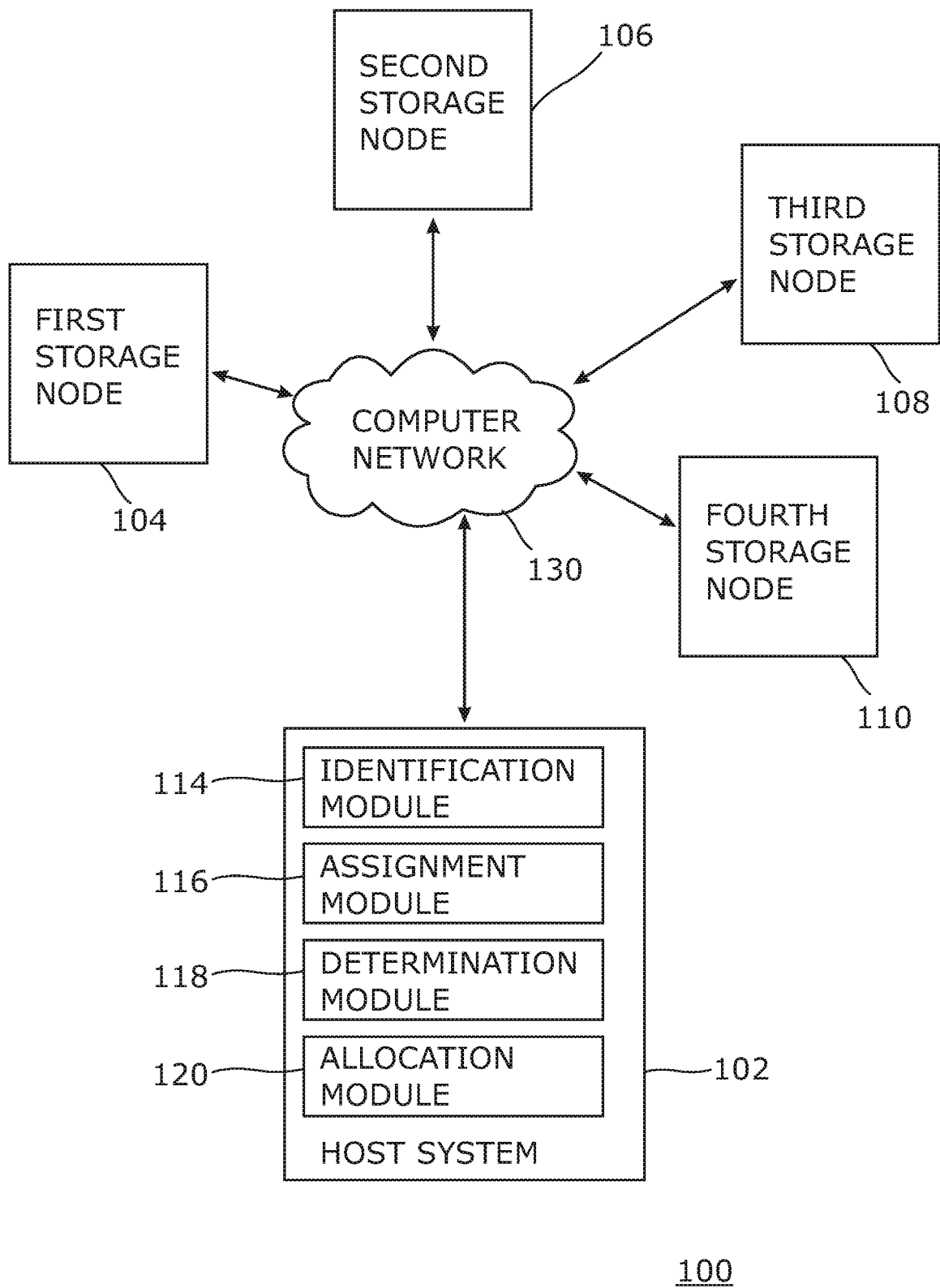


Fig. 1

2/4

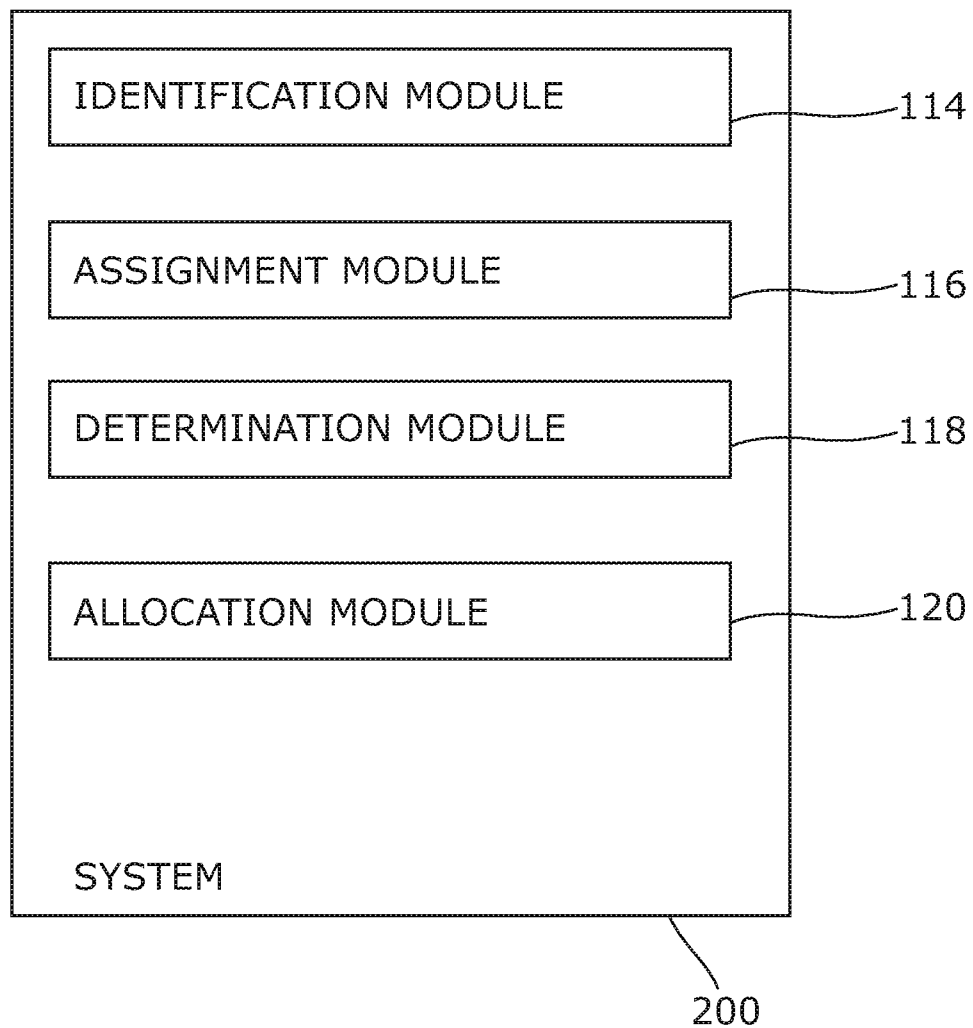


Fig. 2

3/4

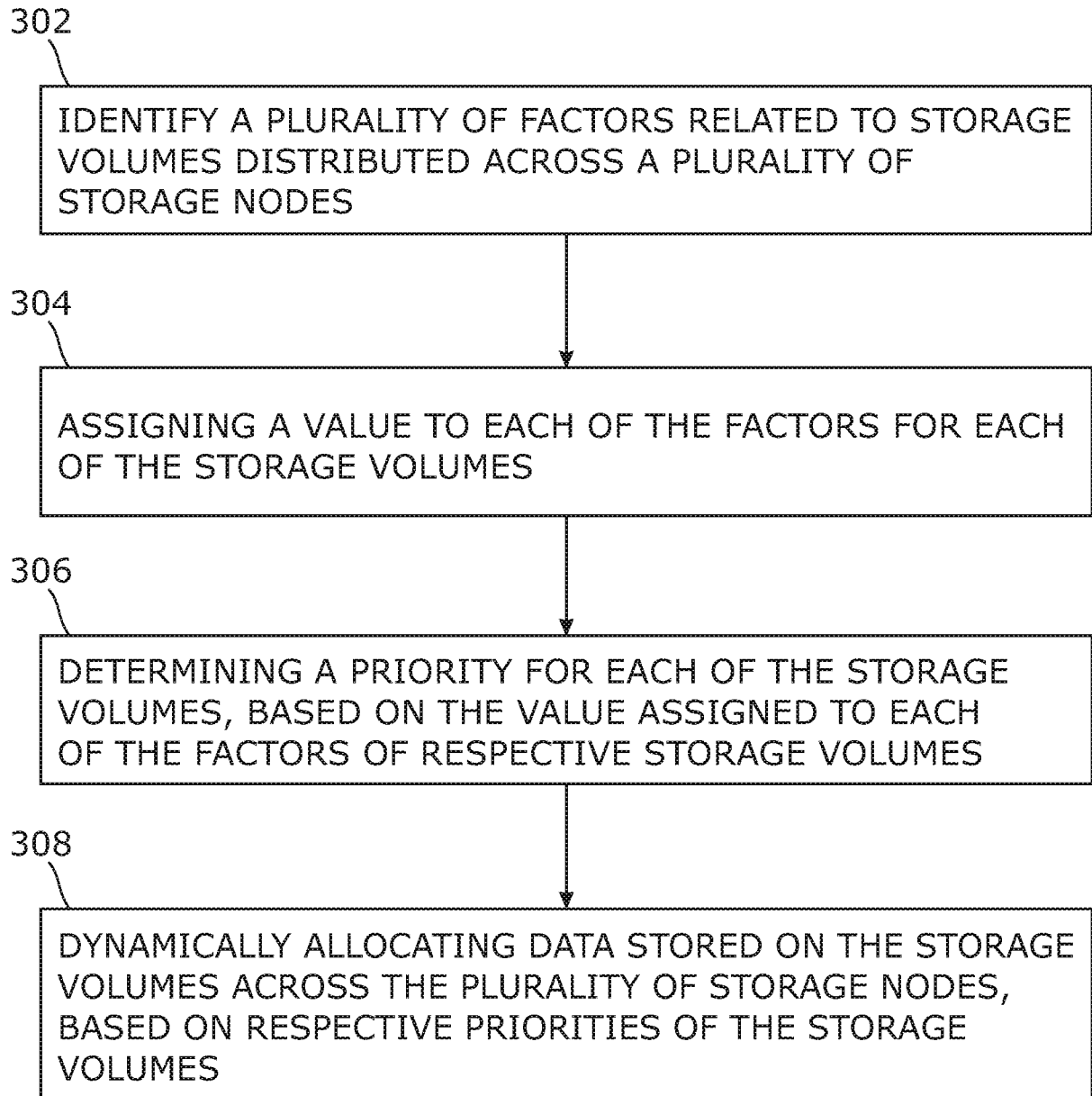
300

Fig. 3

4/4

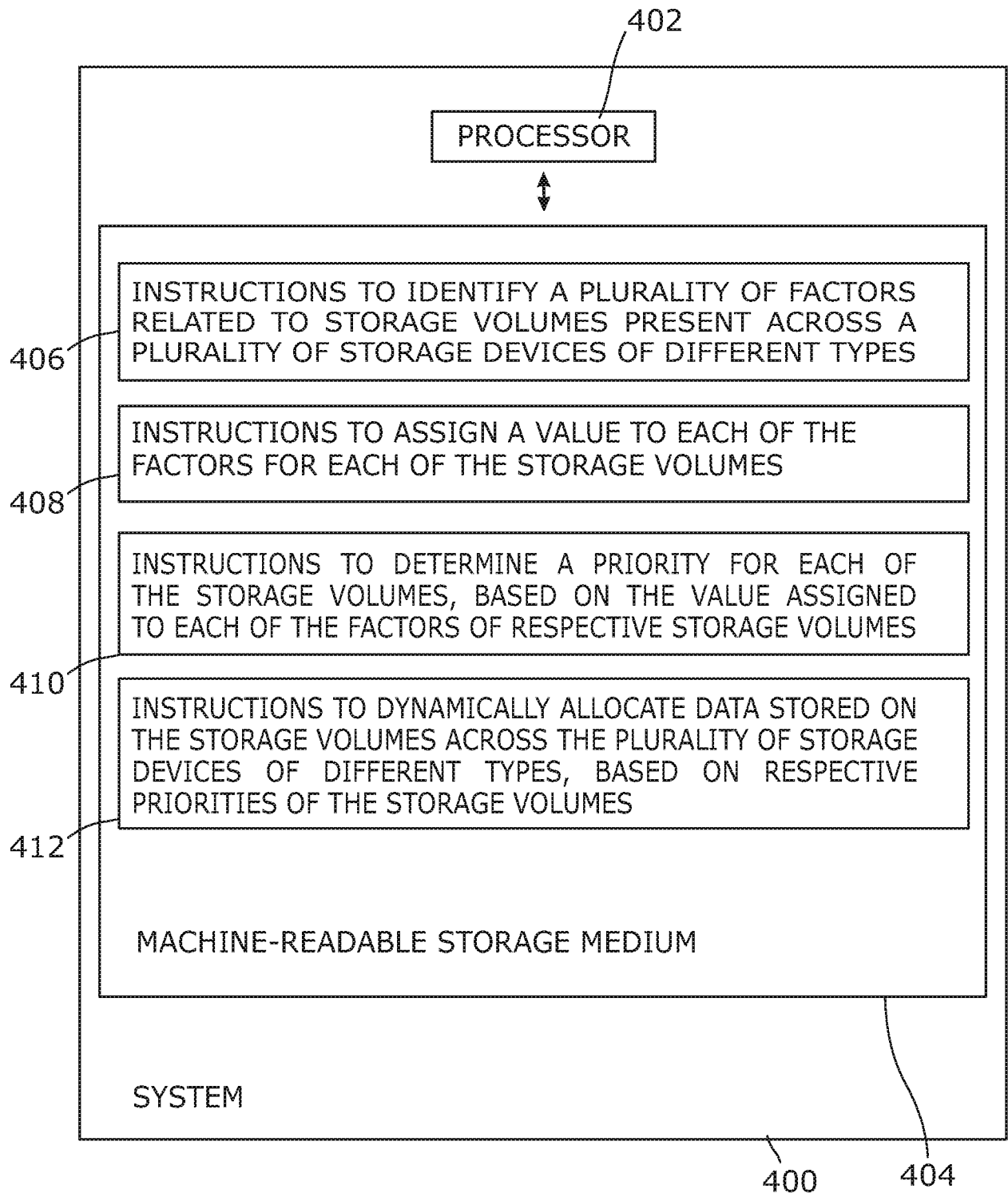


Fig. 4

## INTERNATIONAL SEARCH REPORT

International application No.  
**PCT/US2016/024709****A. CLASSIFICATION OF SUBJECT MATTER****G06F 17/40(2006.01)i, G06F 3/06(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 17/40; G06F 12/06; G06Q 99/00; G06F 17/60; G06F 12/00; G06F 13/00; G06F 12/08; G06F 3/06

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) &amp; Keywords: auto tiering, dynamically, allocate, priority, ranking, identify, factor, SSD, score, 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 2013-0024650 A1 (GOPAKUMAR AMBAT et al.) 24 January 2013<br>See paragraphs [0003]–[0004], [0008], [0010] and [0020]; and figure 1.                                                                | 1–15                  |
| Y         | US 8,972,694 B1 (SEAN DOLAN et al.) 03 March 2015<br>See column 10, lines 14–16; column 12, lines 63–67; column 30, lines 46–50;<br>column 51, lines 65–66; column 55, lines 44–62; and<br>figure 1. | 1–15                  |
| A         | US 2014-0310463 A1 (NETAPP, INC.) 16 October 2014<br>See paragraphs [0005] and [0034]; and figure 1.                                                                                                 | 1–15                  |
| A         | US 2002-0198757 A1 (SANJAY R. HEGDE et al.) 26 December 2002<br>See paragraphs [0029] and [0057]; and figure 3.                                                                                      | 1–15                  |
| A         | EP 2244223 A1 (ROCKWELL AUTOMATION TECHNOLOGIES, INC.) 27 October 2010<br>See paragraphs [0007] and [0022]; 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

"&amp;" document member of the same patent family

Date of the actual completion of the international search

08 November 2016 (08.11.2016)

Date of mailing of the international search report

**09 November 2016 (09.11.2016)**

Name and mailing address of the ISA/KR

International Application Division  
Korean Intellectual Property Office  
189 Cheongsa-ro, Seo-gu, Daejeon, 35208, Republic of Korea

Facsimile No. +82-42-481-8578

Authorized officer

NHO, Ji Myong

Telephone No. +82-42-481-8528



**INTERNATIONAL SEARCH REPORT**

Information on patent family members

International application No.

**PCT/US2016/024709**

| Patent document<br>cited in search report | Publication<br>date | Patent family<br>member(s)           | Publication<br>date      |
|-------------------------------------------|---------------------|--------------------------------------|--------------------------|
| US 2013-0024650 A1                        | 24/01/2013          | US 8627035 B2                        | 07/01/2014               |
| US 8972694 B1                             | 03/03/2015          | None                                 |                          |
| US 2014-0310463 A1                        | 16/10/2014          | US 8769202 B1<br>US 9298616 B2       | 01/07/2014<br>29/03/2016 |
| US 2002-0198757 A1                        | 26/12/2002          | US 7197469 B2                        | 27/03/2007               |
| EP 2244223 A1                             | 27/10/2010          | CN 101872384 A<br>US 2010-0274603 A1 | 27/10/2010<br>28/10/2010 |