



(51) International Patent Classification:
G06F 9/50 (2006.01)

(21) International Application Number:

PCT/CN2018/092632

(22) International Filing Date:

25 June 2018 (25.06.2018)

(25) Filing Language:

English

(26) Publication Language:

English

(71) Applicant: **ALIBABA GROUP HOLDING LIMITED**
[GB/CN]; Fourth Floor, One Capital Place, P.O. Box 847,
George Town, Grand Cayman (KY).

(72) Inventors: **QIU, Sheng**; 4476 Mazzoni Terrace, Dublin,
California 94568 (US). **DU, Yu**; Building 1, No.969 West
Wen Yi Road, Yu Hang District, Hangzhou, Zhejiang (CN).

LIU, Fei; 1447 Capuchino Ave., Burlingame, California
94010 (US). **LI, Shu**; 19619 25th Dr. SE, Bothell, Wash-
ington 98012 (US).

(74) Agent: **BEIJING TSINGYUANHUI INTELLECTUAL
PROPERTY LAW FIRM**; Room 362, 3rd Floor, Suzhou
Street No.1, Haidian District, Beijing 100080 (CN).

(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, DJ, DK, DM, DO,
DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN,
HR, HU, ID, IL, IN, IR, IS, JO, JP, KE, KG, KH, KN, KP,
KR, KW, 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,

(54) Title: SYSTEM AND METHOD FOR MANAGING RESOURCES OF A STORAGE DEVICE AND QUANTIFYING THE
COST OF I/O REQUESTS

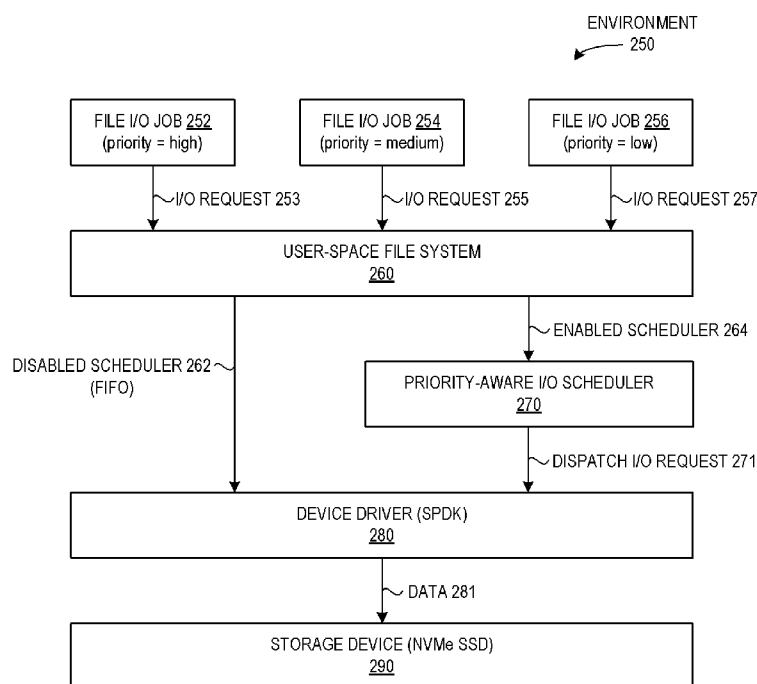


FIG. 2B

(57) Abstract: One embodiment facilitates measurement of a performance of a storage device. During operation, the system determines a normalized cost for an I/O request, wherein the normalized cost is independent of an access pattern and a type of the I/O request, wherein the normalized cost is indicated by a first number of virtual I/O operations consumed by the I/O request, and wherein a virtual I/O operation is used as a logical unit of cost associated with physical I/O operations. The system identifies a performance metric for the storage device by calculating a second number of virtual I/O operations per second which can be executed by the storage device. The system allocates incoming I/O requests to the storage device based on the performance metric, e. g., to satisfy a Quality of Service requirement, thereby causing an enhanced measurement of the performance of the storage device.



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).

Published:

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

5

SYSTEM AND METHOD FOR MANAGING RESOURCES OF A STORAGE DEVICE AND QUANTIFYING THE COST OF I/O REQUESTS

10

Inventors: Sheng Qiu, Yu Du, Fei Liu, and Shu Li

15

BACKGROUND

Field

[0001] This disclosure is generally related to the field of data storage. More specifically, this disclosure is related to a system and method for managing the resources of a storage device and for quantifying the cost of input/output (I/O) requests.

20

Related Art

[0002] The proliferation of the Internet and e-commerce continues to create a vast amount of digital content. Various storage systems and servers have been created to access and store such digital content. In cloud or clustered storage systems, multiple applications may share the underlying system resources (e.g., of a storage device). Managing the resources of a storage device is critical both for the performance of the system, and to satisfy any Quality of Service (QoS)

25

requirements (e.g., in a service level agreement). A metric is needed which can quantify the performance capability of a storage device. Conventional performance metrics such as input/output operations per second (IOPS) or throughput may be useful for quantifying the performance capability of a storage device under specific workloads, such as a workload with all read requests or all write requests).

[0003] However, in mixed workloads (e.g., with a dynamic ratio of read requests versus write requests distributed across a varying range of request sizes), neither the read nor the write bandwidth may be directly used to indicate the performance capability of a storage device. Furthermore, in a storage device such as a solid state drive (SSD), the read and write performance may be asymmetric because of the characteristics of the NAND media and the internal garbage collection mechanism. Thus, conventional performance metrics cannot effectively evaluate the performance capability of a storage device across a wide range of mixed workloads and request sizes. This limitation on a versatile performance metric can create result in a decreased accuracy in identifying the performance capabilities of a storage device, which can result in an ineffective management of the underlying resources of the storage device. This in turn can result in a reduced performance in the overall efficiency of a cloud or a clustered storage system, and may also impact the scalability of a distributed storage system.

SUMMARY

[0004] One embodiment facilitates measurement of a performance of a storage device. During operation, the system determines a normalized cost for an I/O request, wherein the normalized cost is independent of an access pattern and a type of the I/O request, wherein the normalized cost is indicated by a first number of virtual I/O operations consumed by the I/O request, and wherein a virtual I/O

operation is used as a logical unit of cost associated with physical I/O operations. The system identifies a performance metric for the storage device by calculating a second number of virtual I/O operations per second which can be executed by the storage device. The system allocates incoming I/O requests to the storage device
5 based on the performance metric, e.g., to satisfy a Quality of Service requirement, thereby causing an enhanced measurement of the performance of the storage device.

[0005] In some embodiments, determining the normalized cost comprises determining a cost factor for the I/O request, which involves: selecting a set of
10 test cost factor values; executing varied workloads on the storage device based on varying I/O request sizes and a varying ratio of read I/O requests to write I/O requests; measuring the performance of the storage device based on a third number of virtual I/O operations per second executed on the storage device for each workload; determining a test cost factor value which projects a most constant
15 virtual I/O operations per second performance across all the workloads; and setting the cost factor for the I/O request to the determined test cost factor value.

[0006] In some embodiments, determining the normalized cost further comprises: determining a first ratio of a size of the I/O request to a predetermined size for a base unit; and scaling the first ratio by the determined cost factor for the
20 I/O request.

[0007] In some embodiments, the system generates a data structure which maps a range of latency values to a fourth number of virtual I/O operations per second that can be executed by the storage device.

[0008] In some embodiments, the system obtains the data structure in
25 response to one or more of: initiating a startup or a configuration procedure; and requesting, by the storage device or by a user or application associated with the storage device, the data structure.

[0009] In some embodiments, the system determines a specific read latency or a specific write latency based on a size of a specific I/O request. The system performs a lookup in the data structure, based on the specific read latency or the specific write latency, to obtain a fifth number of virtual I/O operations per second mapped to a second range of latency values which include the specific read latency or the specific write latency.

[0010] In some embodiments, the system assigns portions of the second number of virtual I/O operations per second to one or more applications. The system receives a second I/O request from an application. The system schedules the second I/O request based on a respective portion of the second number of virtual I/O operations per second assigned to the application.

[0011] In some embodiments, assigning the portions of the second number of virtual I/O operations per second is based on one or more of: a priority associated with a respective application; a user associated with the respective application; and a predetermined time period which, when exceeded, causes a refresh or reassigning of the portions of the second number of virtual I/O operations per second to the one or more application.

[0012] In some embodiments, scheduling the second I/O request is further based on: a priority associated with the second I/O request; a priority associated with the respective application; and a user associated with the respective application.

BRIEF DESCRIPTION OF THE FIGURES

[0013] FIG. 1 illustrates an exemplary environment that facilitates measurement of the performance of a storage device, in accordance with an embodiment of the present application.

[0014] FIG. 2A illustrates an exemplary environment for facilitating measurement of the performance of a storage device, in accordance with an embodiment of the present application.

5 [0015] FIG. 2B illustrates an exemplary environment for facilitating measurement of the performance of a storage device, in accordance with an embodiment of the present application.

[0016] FIG. 3A presents a flowchart illustrating a method for facilitating measurement of the performance of a storage device, in accordance with an embodiment of the present application.

10 [0017] FIG. 3B presents a flowchart illustrating a method for determining the cost factor for an I/O request, in accordance with an embodiment of the present application.

15 [0018] FIG. 4 presents a diagram illustrating exemplary test results of measuring VIOPS over varying ratios of read to write requests on a given storage device, based on various test cost factor values, in accordance with an embodiment of the present application.

[0019] FIG. 5 presents a flowchart illustrating a method for scheduling I/O requests based on priority and allocated VIOPS, in accordance with an embodiment of the present application.

20 [0020] FIG. 6 presents exemplary pseudocode for scheduling or dispatching an I/O request based on priority and allocated VIOPS, in accordance with an embodiment of the present application.

25 [0021] FIG. 7A presents a flowchart illustrating a method for looking up VIOPS capability in a given storage device based on a specific target latency value, in accordance with an embodiment of the present application.

[0022] FIG. 7B illustrates an exemplary data structure mapping target latency range values to VIOPS capability in a given storage device, in accordance with an embodiment of the present application.

5 [0023] FIG. 8 illustrates an exemplary computer system that facilitates measurement of the performance of a storage device, in accordance with an embodiment of the present application.

[0024] FIG. 9 illustrates an exemplary apparatus that facilitates measurement of the performance of a storage device, in accordance with an embodiment of the present application.

10 [0025] In the figures, like reference numerals refer to the same figure elements.

DETAILED DESCRIPTION

[0026] The following description is presented to enable any person skilled in the art to make and use the embodiments, and is provided in the context of a particular application and its requirements. Various modifications to the disclosed embodiments will be readily apparent to those skilled in the art, and the general principles defined herein may be applied to other embodiments and applications without departing from the spirit and scope of the present disclosure.

15 in the art to make and use the embodiments, and is provided in the context of a particular application and its requirements. Various modifications to the disclosed embodiments will be readily apparent to those skilled in the art, and the general principles defined herein may be applied to other embodiments and applications without departing from the spirit and scope of the present disclosure.

20 Thus, the embodiments described herein are not limited to the embodiments shown, but are to be accorded the widest scope consistent with the principles and features disclosed herein.

Overview

25 [0027] The embodiments described herein provide a system which solves the problem of providing a metric to quantify the performance of a storage device

given a workload which consists of varying types of I/O requests (e.g., a dynamic ratio of read requests versus write requests).

[0028] In cloud or clustered storage systems, multiple applications may share the underlying system resources (e.g., of a storage device). Managing the resources of a storage device is critical both for the performance of the system, and to satisfy any Quality of Service (QoS) requirements (e.g., in a service level agreement). A metric is needed which can quantify the performance capability of a storage device. Conventional performance metrics such as input/output operations per second (IOPS) or throughput may be useful for quantifying the performance capability of a storage device under specific workloads, such as a workload with all read requests or all write requests).

[0029] However, in mixed workloads (e.g., with a dynamic ratio of read requests versus write requests distributed across a varying range of request sizes), neither the read nor the write bandwidth may be directly used to indicate the performance capability of a storage device. Furthermore, in a storage device such as an SSD, the read and write performance may be asymmetric because of the characteristics of the NAND media and the internal garbage collection mechanism. Thus, conventional performance metrics cannot effectively evaluate the performance capability of a storage device across a wide range of mixed workloads and request sizes. This limitation on a versatile performance metric can create result in a decreased accuracy in identifying the performance capabilities of a storage device, which can result in an ineffective management of the underlying resources of the storage device. This in turn can result in a reduced performance in the overall efficiency of a cloud or a clustered storage system, and may also impact the scalability of a distributed storage system.

[0030] The embodiments described herein address these limitations by providing a system which uses a new performance metric – “virtual I/O

operations per second” or “VIOPS” – to generalize the concept of bandwidth in a storage device (e.g., an SSD). The system can normalize the resource cost of varying I/O requests on a target storage device, independent of the access pattern (e.g., random or sequential) and the type (e.g., read or write) of the I/O request.

- 5 The system measures the cost of an individual I/O request by using the consumed number of virtual I/Os.

[0031] I/O requests can vary based on various characteristics or features of the request, including, e.g., the request type (read or write), the request size, and the access pattern (random or sequential). The number of VIOs consumed by an I/O request depends on the various characteristics of the I/O request. For example, a 4 KB random read may consume one virtual I/O, while a 4KB random write may consume more than one virtual I/O. A virtual I/O operation is used as a logical unit of cost associated with physical I/O operations. The system can use a formula to calculate the number of virtual I/Os consumed by serving any random storage request (i.e., a read or a write request of any size), for a given storage device. Thus, the cost of an actual I/O request is converted into the consumed number of virtual I/Os on a target storage device. Determining the cost of random I/O requests is described below in relation to FIGs. 3B and 4.

[0032] By normalizing the cost of varying I/O requests based on the VIOs and independent of the access pattern or type of a given I/O request, the system can evaluate the performance of the storage device based on the VIOPS, i.e., the number of virtual I/Os served per second. Furthermore, a priority-based I/O scheduler component can leverage the VIOPS model to accurately control service resources among different I/O services or applications as described below in relation to FIG. 5. In a distributed storage system with multiple storage servers and storage devices, incoming I/O requests may be allocated to a particular storage device based on the calculated VIOPS for the particular storage device.

That is, the scheduler component can determine, based on the calculated VIOPS for each of the multiple storage devices in the distributed storage system, how to allocate, distribute, dispatch, and schedule the incoming I/O requests among the multiple storage devices.

5 **[0033]** Thus, the embodiments described herein provide a system which improves and enhances the efficiency and performance of a storage system. By measuring the cost of any random I/O request using VIOs, the system can normalize the cost of varying I/O requests (by converting the cost of an I/O request to a number of VIOs consumed by a given storage device). The system
10 can also identify a new performance metric with which to evaluate the storage device (by calculating a number of VIOPS served by the storage device). By using the VIOPS as the performance metric for the storage device, the system can overcome the deficiencies of the conventional metrics, such as IOPS or throughput, which are limited to quantifying performance under specific
15 workloads. The embodiments described herein can thus more accurately provide a measurement of the performance of a storage device by accounting for mixed workloads, e.g., with a dynamic ratio between read requests and write requests, and varying sizes of I/O requests. In this way, and unlike conventional performance metrics, the system enhances the measurement of the performance of
20 the storage device. Furthermore, by allocating incoming I/O requests based on the more accurate and flexible performance metric of VIOPS, the embodiments described herein enhance the overall efficiency and performance of a distributed storage system.

25 **Exemplary Environment and Network**

[0034] FIG. 1 illustrates an exemplary environment 100 that facilitates measurement of the performance of a storage device, in accordance with an

embodiment of the present application. Environment 100 can include a computing device 102 and an associated user 104. Computing device 102 can communicate via a network 110 with storage servers 112, 114, and 116, which can be part of a distributed storage system and accessed via client servers (not shown). A storage server can include multiple storage devices, and each storage device can include a controller and multiple physical media for data storage. For example, server 116 can include: a network interface card (NIC) 122; a CPU 124; a DRAM DIMM 126; hard disk drives (HDDs) 132 and 136 (with, respectively, controllers 134 and 138); and SSDs 140 and 144 (with, respectively, controllers 142 and 146).

[0035] A controller can include interfaces to a host and to a non-volatile memory. For example, SSD 140 can include SSD controller 142. Controller 142 can include: a host interface 150; an embedded processor 152; and a channel management 158. SSD controller 142 can communicate with a host (e.g., via host interface 150 and a communication to/from host 192). SSD controller 142 can also communicate with the non-volatile memory (via channel management 158). The non-volatile memory can be accessed via multiple channels. For example, NAND dies 172, 174, and 176 may be accessed via a channel 170, and NAND dies 182, 184, and 186 may be accessed via a channel 180.

[0036] During operation, in the embodiments described herein, SSD 140, via SSD controller 142, may receive various I/O requests (e.g., via communication 192). The cost of an individual I/O request (i.e., VIO) and the identification of a performance metric (i.e., VIOPS) for SSD 140 may be determined by a component or module not residing in SSD 140. A priority-aware I/O scheduler, as described below in relation to FIGs. 2A and 2B), can use the determined cost and performance metric to dispatch incoming I/O requests which

have different priorities, characteristics (e.g., size, type, and access pattern), and allocated overall VIOPS.

[0037] FIG. 2A illustrates an exemplary environment 200 for facilitating measurement of the performance of a storage device, in accordance with an embodiment of the present application. Environment 200 can include: an application 202, where incoming requests from application 202 are marked with a “high” priority; an application 204, where incoming requests from application 204 are marked with a “medium” priority; and an application 206, where incoming requests from application 206 are marked with a “medium” priority. Each of applications 202, 204, and 206 may also be associated with a previously allocated portion of the overall VIOPS for the given storage device.

[0038] During operation, an application can send an I/O request, which is received and dispatched by a scheduler component based on the priority of the I/O request, a previously allocated amount of VIOPS for the requesting application, and other factors. For example, application 204 can send an I/O request 205, which is received by priority-aware I/O scheduler 210. Based on the “medium” priority of I/O request 205, as well as any previously allocated amount of VIOPS to application 204, scheduler 210 can schedule and dispatch I/O request 205 (via a communication dispatch I/O request 211) to a device driver 220. Device driver 220 can send data 221 (associated with I/O request 205) to a storage device 240.

[0039] Thus, priority-aware I/O scheduler 210 can take into account the priority of an individual I/O request, and can also control the VIOPS usage across various applications based on configured settings, parameters, or rules for VIOPS usage. These factors for VIOPS usage may be pre-configured by a system, a system administrator, or other user. The factors for VIOPS usage may also result in the dynamic allocation of VIOPS usage across various applications, e.g., during

operation of the storage system or the storage device, or as part of processing an I/O request.

[0040] FIG. 2B illustrates an exemplary environment 250 for facilitating measurement of the performance of a storage device, in accordance with an embodiment of the present application. Environment 250 is similar to environment 200, but includes communications which occur if the priority-aware I/O scheduler is not enabled. Environment 250 can include: a file I/O job 252 marked with a “high” priority; a file I/O job 254 marked with a “medium” priority; and a file I/O job 256 marked with a “low” priority. Each of file I/O jobs 252, 254, and 256 may also be associated with a previously allocated portion of the overall VIOPS for a given storage device.

[0041] During operation, file I/O job 252 may be sent as an I/O request 253 to a user-space file system 260. If a priority-aware I/O scheduler is enabled, I/O request 253 can be received by a priority-aware I/O scheduler 270 (via, e.g., an enabled scheduler 264 communication). Based on the “high” priority of I/O request 253, as well as any previously allocated amount of VIOPS to file I/O job 252 (or to an application or a user associated with file I/O job 252), priority-aware I/O scheduler 270 can schedule and dispatch I/O request 253 (via a communication dispatch I/O request 271) to a device driver (SPDK) 280. Device driver 280 can send data 281 (associated with I/O request 253) to a storage device (NVMe SSD) 280.

[0042] If the priority-aware I/O scheduler is not enabled (e.g., is disabled, does not exist, or is not recognized), user-space file system 260 can send I/O request 253 directly to a device driver (SPDK) 280, via a disabled scheduler 262 communication. I/O requests which are handled this way may be queued using a first in first out (FIFO) policy.

**Measuring the Performance of a Storage Device Based on VIOPS; and
Determining the Cost and Cost Factor of Individual I/O Requests**

[0043] FIG. 3A presents a flowchart 300 illustrating a method for facilitating measurement of the performance of a storage device, in accordance with an embodiment of the present application. During operation, the system determines a normalized cost for an I/O request, wherein the normalized cost is independent of an access pattern and a type of the I/O request, wherein the normalized cost is indicated by a first number of virtual I/O operations consumed by the I/O request, and wherein a virtual I/O operation is used as a logical unit of cost associated with physical I/O operations (operation 302).

[0044] The system identifies a performance metric for the storage device by calculating a second number of virtual I/O operations per second (VIOPS) which can be executed by the storage device, thereby facilitating an enhanced measurement of the performance of the storage device (operation 304). The system allocates incoming I/O requests to the storage device based on the performance metric, e.g., to satisfy a Quality of Service requirement or to achieve an optimal or enhanced load balancing (operation 306).

[0045] The normalized cost in operation 302 can be further based on a “cost factor” for the I/O request. For example, the system can calculate the “VIO cost” of a given I/O request based on the *cost_factor* for the specific I/O request:

$$VIO\_cost = cost\_factor * (request\_size / base\_size). \quad \text{Equation (1)}$$

The *request_size* is the size of the given I/O request, and the *base_size* is a predetermined size for a base unit, such as 4 KB. Assume that the *cost_factor* for a read I/O request is “1,” and that the *cost_factor* for a write I/O request is represented by the variable *k*. The system can perform a specific procedure to select a value of *k* (i.e., a cost factor value) which provides the maximum VIOPS performance on a specific storage device.

[0046] FIG. 3B presents a flowchart 320 illustrating a method for determining the cost factor for an I/O request, in accordance with an embodiment of the present application. During operation, the system selects a set of test cost factor values, to determine a cost factor for an I/O request (operation 322) (e.g., a set of values for k). The system executes varied workloads on a storage device based on varying I/O request sizes and a varying ratio of read I/O requests to write I/O requests (operation 324). For example, a varying ratio of read to write requests can include: 100% read requests to 0% write requests (“100R”); 80% read requests to 20% write requests (“80R”); 50% read requests to 50% write requests (“50R”); and 30% read requests to 70% write requests (“30R”). The system measures the performance of the storage device based on a number of virtual I/O operations per second executed on the storage device for a respective workload (operation 326). If there are remaining workloads to be executed (decision 328), the operation continues as described above for operation 324 (e.g., executing the workload). If there are no remaining workloads to be executed (decision 328), the system determines the cost factor value which projects a most constant virtual I/O operations per second across all the workloads (operation 330). The system sets the cost factor for the I/O request to the determined cost factor value (operation 332). That is, the system identifies a k value which demonstrates or projects the most constant VIOPS, as shown below in relation to FIG.4

[0047] FIG. 4 presents a diagram 400 illustrating exemplary test results of measuring VIOPS over varying ratios of read to write requests on a given storage device, based on various test cost factor values, in accordance with an embodiment of the present application. The x -axis indicates the varying ratios of read to write requests. That is, “100R” indicates 100% read requests to 0% write requests, “90R” indicates 90% read requests to 10% write requests, etc. The y -

axis indicates the measured virtual I/O operations per second consumed by the storage device. Note that the value of the cost factor which provides the most constant VIOPS for the given storage device is when $k = 13$. Note that the results in diagram 400 can reflect a specific I/O request size, with varying ratios of read to write requests.

Exemplary Method for Scheduling I/O Requests Based on Priority and Allocated VIOPS; Exemplary Psuedocode for Scheduling I/O Requests

[0048] FIG. 5 presents a flowchart illustrating a method for scheduling I/O requests based on priority and allocated VIOPS, in accordance with an embodiment of the present application. During operation, the system determines or calculates a total number of virtual I/O operations per second which can be executed by the storage device (operation 502), i.e., the number of VIOPS which can be served by the storage device. The system assigns portions of the total number of VIOPS to one or more applications (operation 504). Assigning the portions of the VIOPS can be based on, e.g.: a priority associated with a respective application; a user associated with the respective application; a predetermined time period which, when exceeded, causes a refresh or reassigning of the portions of the total VIOPS.

[0049] Subsequently, the system receives an I/O request from an application (operation 506). The system schedules or dispatches the received I/O request based on a respective portion of the total number of VIOPS assigned to the application (operation 508). Scheduling the I/O request can be further based on one or more of: a priority associated with the I/O request; a priority associated with the respective application; and a user associated with the respective application. An exemplary priority-aware I/O scheduler is described above in relation to FIGs. 2A and 2B.

[0050] FIG. 6 presents exemplary pseudocode 600 for scheduling or dispatching an I/O request based on priority and allocated VIOPS, in accordance with an embodiment of the present application.

5 **Exemplary Method for Looking Up VIOPS Capability Based on Target Latency; Exemplary Data Structure**

[0051] FIG. 7A presents a flowchart 700 illustrating a method for looking up VIOPS capability in a given storage device based on a specific target latency value, in accordance with an embodiment of the present application. During
10 operation, the system generates a data structure which maps a range of latency values to a number of virtual I/O operations per second that can be executed by the storage device (operation 702). The system obtains the data structure in response to one or more of: initiating a startup or a configuration procedure; and requesting, by the storage device or by a user or application associated with the
15 storage device, the data structure (operation 704). The system determines a specific read latency or a specific write latency based on a size of a specific I/O request (operation 706). The system performs a lookup in the data structure, based on the specific read latency or the specific write latency, to obtain a number of virtual I/O operations per second mapped to a second range of latency values
20 which include the specific read latency or the specific write latency (operation 708).

[0052] FIG. 7B illustrates an exemplary data structure 750 mapping target latency range values to VIOPS capability in a given storage device, in accordance with an embodiment of the present application. Data structure 750 can include
25 entries which include a latency range 752 and a maximum VIOPS executable by the storage device 754. For example, an entry 760 can include a latency range of “101 microseconds – 300 microseconds” and a maximum VIOPS value of

“300,000.” Similarly, an entry 762 can include a latency range of “301 microseconds – 500 microseconds” and a maximum VIOPS value of “200,000.”

Exemplary Computer System and Apparatus

5 [0053] FIG. 8 illustrates an exemplary computer system 800 that facilitates measurement of the performance of a storage device, in accordance with an embodiment of the present application. Computer system 800 includes a processor 802, a volatile memory 804, and a storage device 806. Computer system 800 may be a computing device or a storage device. Volatile memory 804
10 can include memory (e.g., RAM) that serves as a managed memory, and can be used to store one or more memory pools. Computer system 800 can be coupled to a display device 810, a keyboard 812, and a pointing device 814. Storage device 806 can store an operating system 816, a content-processing system 818, and data 832.

15 [0054] Content-processing system 818 can include instructions, which when executed by computer system 800, can cause computer system 800 to perform methods and/or processes described in this disclosure. For example, content-processing system 818 can include instructions for receiving and transmitting data packets, including a request to write or read data, an I/O request,
20 data to be encoded and stored, or a block or a page of data.

 [0055] Content-processing system 818 can further include instructions for determining a normalized cost for an I/O request, wherein the normalized cost is independent of an access pattern and a type of the I/O request, wherein the normalized cost is indicated by a first number of virtual I/O operations consumed
25 by the I/O request, and wherein a virtual I/O operation is used as a logical unit of cost associated with physical I/O operations (cost-determining module 822). Content-processing system 818 can include instructions for identifying a

performance metric for the storage device by calculating a second number of virtual I/O operations per second which can be executed by the storage device, thereby facilitating an enhanced measurement of the performance of the storage device (VIOPS-calculating module 824). Content-processing system 818 can include instructions for allocating incoming I/O requests to the storage device based on the performance metric to satisfy a Quality of Service requirement (request-scheduling module 830).

[0056] Content-processing system 818 can also include instructions for determining a cost factor for the I/O request (cost-determining module 822).

10 Content-processing system 818 can include instructions for generating a data structure which maps a range of latency values to a fourth number of virtual I/O operations per second that can be executed by the storage device (data structure-managing module 826). Content-processing system 818 can include instructions for determining a specific read latency or a specific write latency based on a size

15 of a specific I/O request (data structure-managing module 826). Content-processing system 818 can additionally include instructions for performing a lookup in the data structure, based on the specific read latency or the specific write latency, to obtain a fifth number of virtual I/O operations per second mapped to a second range of latency values which include the specific read

20 latency or the specific write latency (data structure-managing module 826).

[0057] Content-processing system 818 can further include instructions for assigning portions of the second number of virtual I/O operations per second to one or more applications (VIOPS-allocating module 828). Content-processing system 818 can include instructions for receiving a second I/O request from an

25 application (communication module 820). Content-processing system 818 can include instructions for scheduling the second I/O request based on a respective

portion of the second number of virtual I/O operations per second assigned to the application (request-scheduling module 830).

- [0058] Data 832 can include any data that is required as input or that is generated as output by the methods and/or processes described in this disclosure.
- 5 Specifically, data 832 can store at least: data to be stored, written, loaded, moved, retrieved, accessed, deleted, or copied; a cost; a normalized cost; a request; an I/O request; a physical I/O operation; an access pattern of an I/O request; a type or size of an I/O request; a virtual I/O; a number of virtual I/Os; a logical unit of cost associated with physical I/O operations; a number of virtual I/O per second
- 10 (VIOPS); a performance metric; the measurement of the performance of a storage device based on the VIOPS; a cost factor; a set of test cost factor values; a varied workload; varying I/O request sizes; a varying ratio of read I/O request to write I/O requests; a projection of VIOPS across a workload; a predetermined size for a base unit; a first ratio of an I/O request size to the predetermined base unit size; a
- 15 data structure; a data structure entry; a latency value; a range of latency values; an entry mapping a range of latency values to a number of VIOPS that can be executed by a storage device; a startup or a configuration procedure; a request for the data structure; a read latency; a write latency; an assigned portion of a number of VIOPS; an application; an application-allocated VIOPS portion; a priority or a
- 20 user associated with an I/O request or an application; a predetermined time period; and a new or updated re-assigned or re-allocated portion of the number of VIOPS.

- [0059] FIG. 9 illustrates an exemplary apparatus 900 that facilitates measurement of the performance of a storage device, in accordance with an embodiment of the present application. Apparatus 900 can comprise a plurality of
- 25 units or apparatuses which may communicate with one another via a wired, wireless, quantum light, or electrical communication channel. Apparatus 900 may be realized using one or more integrated circuits, and may include fewer or

more units or apparatuses than those shown in FIG. 9. Further, apparatus 900 may be integrated in a computer system, or realized as a separate device which is capable of communicating with other computer systems and/or devices.

Specifically, apparatus 900 can comprise units 902-912 which perform functions or operations similar to modules 820-830 of computer system 800 of FIG. 8, including: a communication unit 902; a cost-determining unit 904; a VIOPS-calculating unit 906; a data structure-managing unit 908; a VIOPS-allocating unit 910; and a request-scheduling unit 912.

[0060] The data structures and code described in this detailed description are typically stored on a computer-readable storage medium, which may be any device or medium that can store code and/or data for use by a computer system. The computer-readable storage medium includes, but is not limited to, volatile memory, non-volatile memory, magnetic and optical storage devices such as disk drives, magnetic tape, CDs (compact discs), DVDs (digital versatile discs or digital video discs), or other media capable of storing computer-readable media now known or later developed.

[0061] The methods and processes described in the detailed description section can be embodied as code and/or data, which can be stored in a computer-readable storage medium as described above. When a computer system reads and executes the code and/or data stored on the computer-readable storage medium, the computer system performs the methods and processes embodied as data structures and code and stored within the computer-readable storage medium.

[0062] Furthermore, the methods and processes described above can be included in hardware modules. For example, the hardware modules can include, but are not limited to, application-specific integrated circuit (ASIC) chips, field-programmable gate arrays (FPGAs), and other programmable-logic devices now known or later developed. When the hardware modules are activated, the

hardware modules perform the methods and processes included within the hardware modules.

[0063] The foregoing embodiments described herein have been presented for purposes of illustration and description only. They are not intended to be
5 exhaustive or to limit the embodiments described herein to the forms disclosed. Accordingly, many modifications and variations will be apparent to practitioners skilled in the art. Additionally, the above disclosure is not intended to limit the embodiments described herein. The scope of the embodiments described herein is defined by the appended claims.

10

What Is Claimed Is:

- 1 1. A computer-implemented method for facilitating measurement of a
2 performance of a storage device, the method comprising:
3 determining a normalized cost for an I/O request, wherein the normalized
4 cost is independent of an access pattern and a type of the I/O request, wherein the
5 normalized cost is indicated by a first number of virtual I/O operations consumed
6 by the I/O request, and wherein a virtual I/O operation is used as a logical unit of
7 cost associated with physical I/O operations;
8 identifying a performance metric for the storage device by calculating a
9 second number of virtual I/O operations per second which can be executed by the
10 storage device; and
11 allocating incoming I/O requests to the storage device based on the
12 performance metric.
- 1 2. The method of claim 1, wherein determining the normalized cost
2 comprises:
3 determining a cost factor for the I/O request, which involves:
4 selecting a set of test cost factor values;
5 executing varied workloads on the storage device based on varying
6 I/O request sizes and a varying ratio of read I/O requests to write I/O
7 requests;
8 measuring the performance of the storage device based on a third
9 number of virtual I/O operations per second executed on the storage
10 device for each workload;
11 determining a test cost factor value which projects a most constant
12 virtual I/O operations per second performance across all the workloads;

13 and
14 setting the cost factor for the I/O request to the determined test cost
15 factor value.

1 3. The method of claim 2, wherein determining the normalized cost
2 further comprises:
3 determining a first ratio of a size of the I/O request to a predetermined size
4 for a base unit; and
5 scaling the first ratio by the determined cost factor for the I/O request.

1 4. The method of claim 1, further comprising:
2 generating a data structure which maps a range of latency values to a
3 fourth number of virtual I/O operations per second that can be executed by the
4 storage device.

1 5. The method of claim 4, further comprising:
2 obtaining the data structure in response to one or more of:
3 initiating a startup or a configuration procedure; and
4 requesting, by the storage device or by a user or application
5 associated with the storage device, the data structure.

1 6. The method of claim 4, further comprising:
2 determining a specific read latency or a specific write latency based on a
3 size of a specific I/O request; and
4 performing a lookup in the data structure, based on the specific read
5 latency or the specific write latency, to obtain a fifth number of virtual I/O
6 operations per second mapped to a second range of latency values which include

7 the specific read latency or the specific write latency.

1 7. The method of claim 1, further comprising:
2 assigning portions of the second number of virtual I/O operations per
3 second to one or more applications;
4 receiving a second I/O request from an application; and
5 scheduling the second I/O request based on a respective portion of the
6 second number of virtual I/O operations per second assigned to the application.

1 8. The method of claim 7, wherein assigning the portions of the
2 second number of virtual I/O operations per second is based on one or more of:
3 a priority associated with a respective application;
4 a user associated with the respective application; and
5 a predetermined time period which, when exceeded, causes a refresh or
6 reassigning of the portions of the second number of virtual I/O operations per
7 second to the one or more application.

1 9. The method of claim 7, wherein scheduling the second I/O request
2 is further based on:
3 a priority associated with the second I/O request;
4 a priority associated with the respective application; and
5 a user associated with the respective application.

1 10. A computer system for facilitating measurement of a performance
2 of a storage device, the system comprising:
3 a processor; and
4 a memory coupled to the processor and storing instructions, which when

5 executed by the processor cause the processor to perform a method, the method
6 comprising:
7 determining a normalized cost for an I/O request, wherein the normalized
8 cost is independent of an access pattern and a type of the I/O request, wherein the
9 normalized cost is indicated by a first number of virtual I/O operations consumed
10 by the I/O request, and wherein a virtual I/O operation is used as a logical unit of
11 cost associated with physical I/O operations;
12 identifying a performance metric for the storage device by calculating a
13 second number of virtual I/O operations per second which can be executed by the
14 storage device; and
15 allocating incoming I/O requests to the storage device based on the
16 performance metric.

1 11. The computer system of claim 10, wherein determining the
2 normalized cost comprises:
3 determining a cost factor for the I/O request, which involves:
4 selecting a set of test cost factor values;
5 executing varied workloads on the storage device based on varying
6 I/O request sizes and a varying ratio of read I/O requests to write I/O
7 requests;
8 measuring the performance of the storage device based on a third
9 number of virtual I/O operations per second executed on the storage
10 device for each workload;
11 determining a test cost factor value which projects a most constant
12 virtual I/O operations per second performance across all the workloads;
13 and
14 setting the cost factor for the I/O request to the determined test cost

15 factor value.

1 12. The computer system of claim 11, wherein determining the
2 normalized cost further comprises:
3 determining a first ratio of a size of the I/O request to a predetermined size
4 for a base unit; and
5 scaling the first ratio by the determined cost factor for the I/O request.

1 13. The computer system of claim 10, wherein the method further
2 comprises:
3 generating a data structure which maps a range of latency values to a
4 fourth number of virtual I/O operations per second that can be executed by the
5 storage device.

1 14. The computer system of claim 13, wherein the method further
2 comprises:
3 obtaining the data structure in response to one or more of:
4 initiating a startup or a configuration procedure; and
5 requesting, by the storage device or by a user or application
6 associated with the storage device, the data structure.

1 15. The computer system of claim 13, wherein the method further
2 comprises:
3 determining a specific read latency or a specific write latency based on a
4 size of a specific I/O request; and
5 performing a lookup in the data structure, based on the specific read
6 latency or the specific write latency, to obtain a fifth number of virtual I/O

7 operations per second mapped to a second range of latency values which include
8 the specific read latency or the specific write latency.

1 16. The computer system of claim 9, wherein the method further
2 comprises:
3 assigning portions of the second number of virtual I/O operations per
4 second to one or more applications;
5 receiving a second I/O request from an application; and
6 scheduling the second I/O request based on a respective portion of the
7 second number of virtual I/O operations per second assigned to the application.

1 17. The computer system of claim 16, wherein assigning the portions
2 of the second number of virtual I/O operations per second is based on one or more
3 of:
4 a priority associated with a respective application;
5 a user associated with the respective application; and
6 a predetermined time period which, when exceeded, causes a refresh or
7 reassigning of the portions of the second number of virtual I/O operations per
8 second to the one or more application.

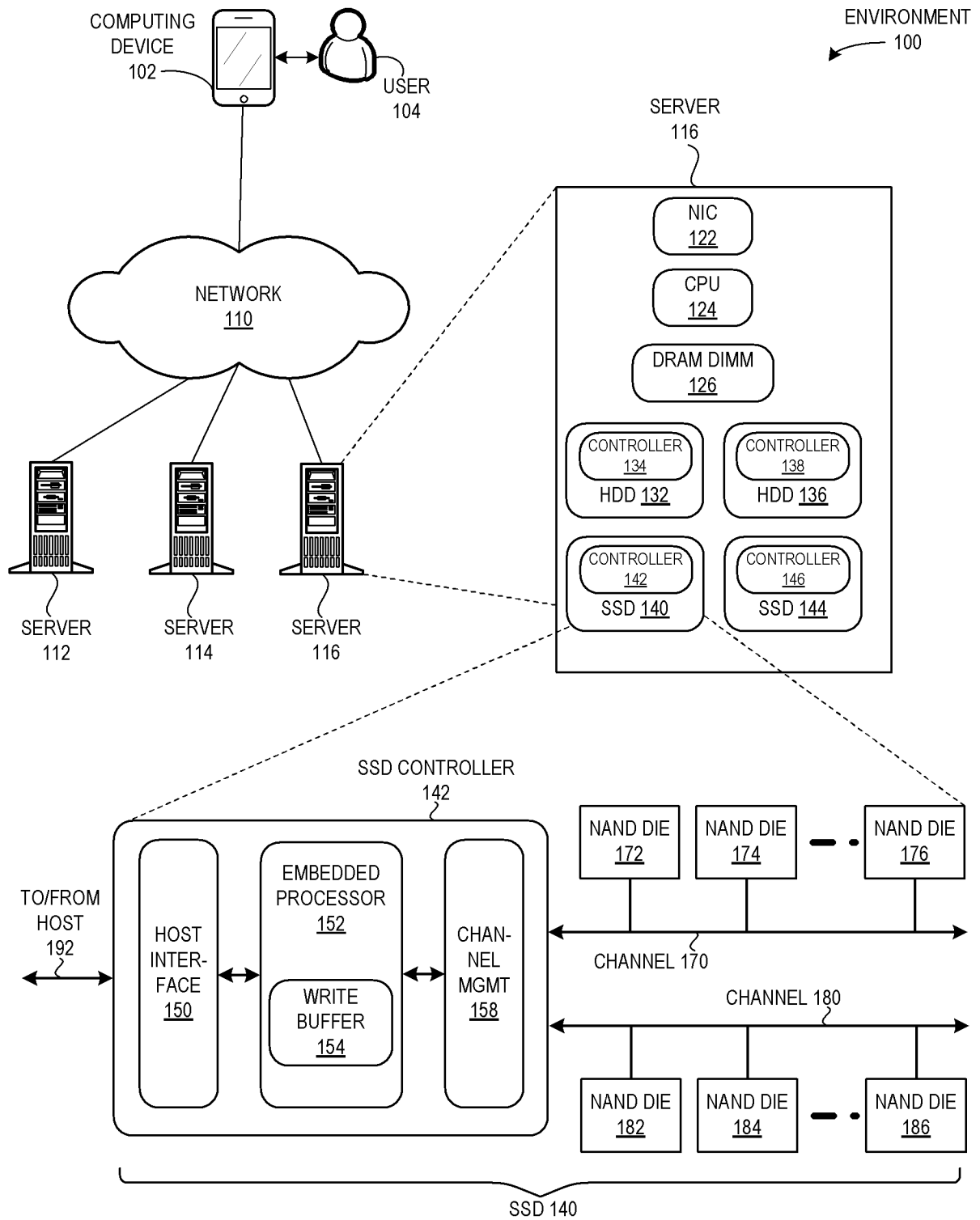
1 18. The computer system of claim 16, wherein scheduling the second
2 I/O request is further based on:
3 a priority associated with the second I/O request;
4 a priority associated with the respective application; and
5 a user associated with the respective application.

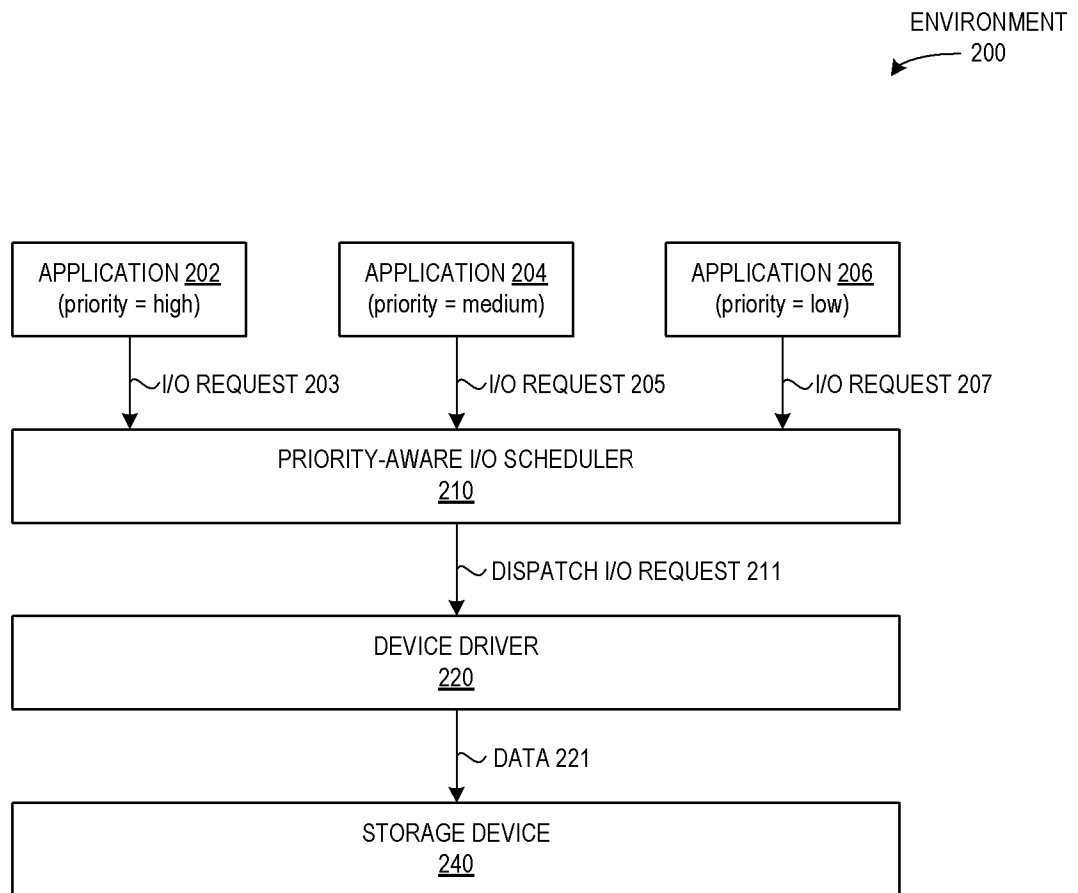
1 19. An apparatus for facilitating measurement of a performance of a

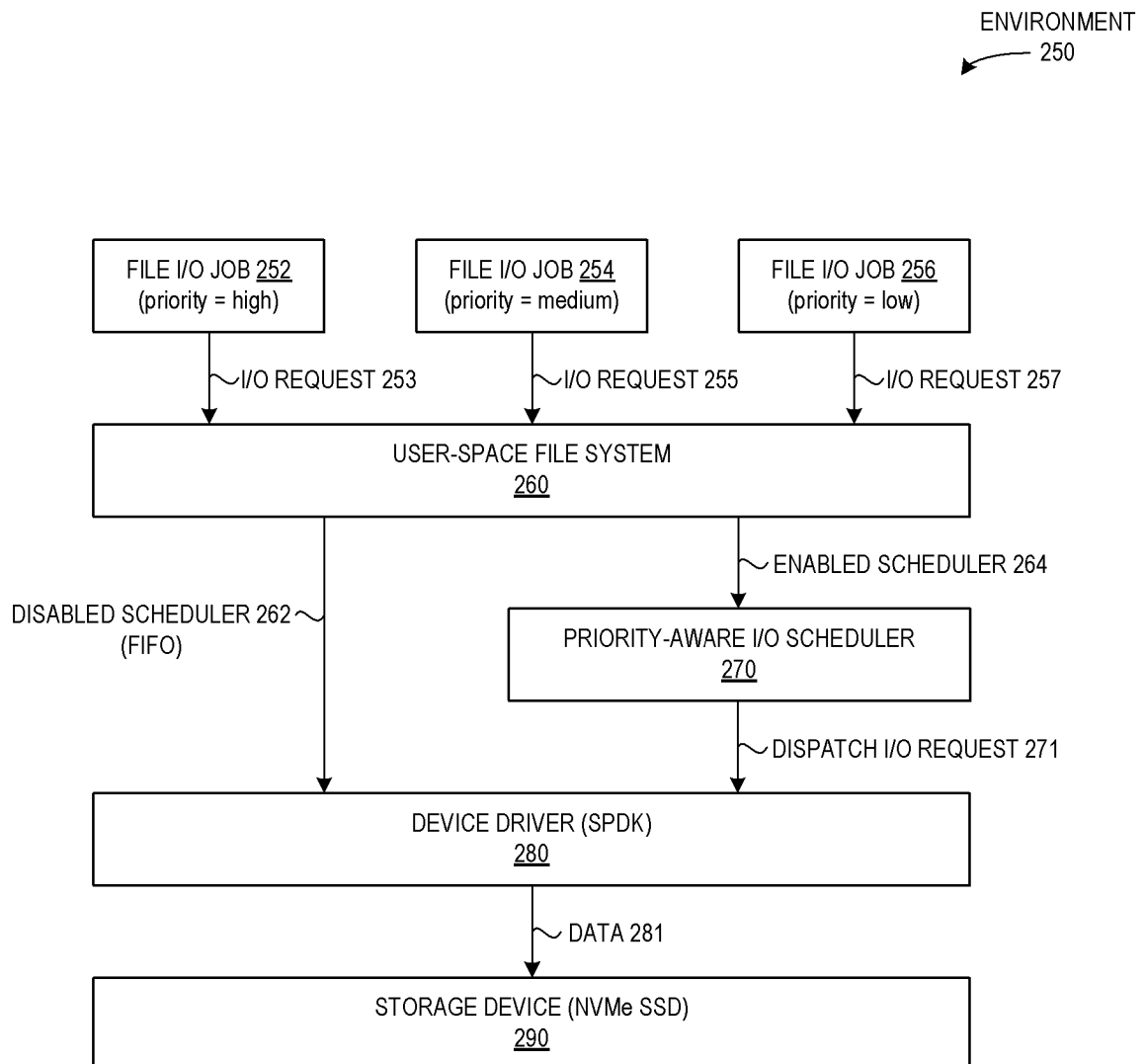
2 storage device, the device comprising:
3 a cost-determining module configured to determine a normalized cost for
4 an I/O request, wherein the normalized cost is independent of an access pattern
5 and a type of the I/O request, wherein the normalized cost is indicated by a first
6 number of virtual I/O operations consumed by the I/O request, and wherein a
7 virtual I/O operation is used as a logical unit of cost associated with physical I/O
8 operations;
9 a virtual I/O operation per second (VIOPS) calculating module configured
10 to identify a performance metric for the storage device by calculating a second
11 number of virtual I/O operations per second which can be executed by the storage
12 device; and
13 a request-scheduling module configured to allocate incoming I/O requests
14 to the storage device based on the performance metric.

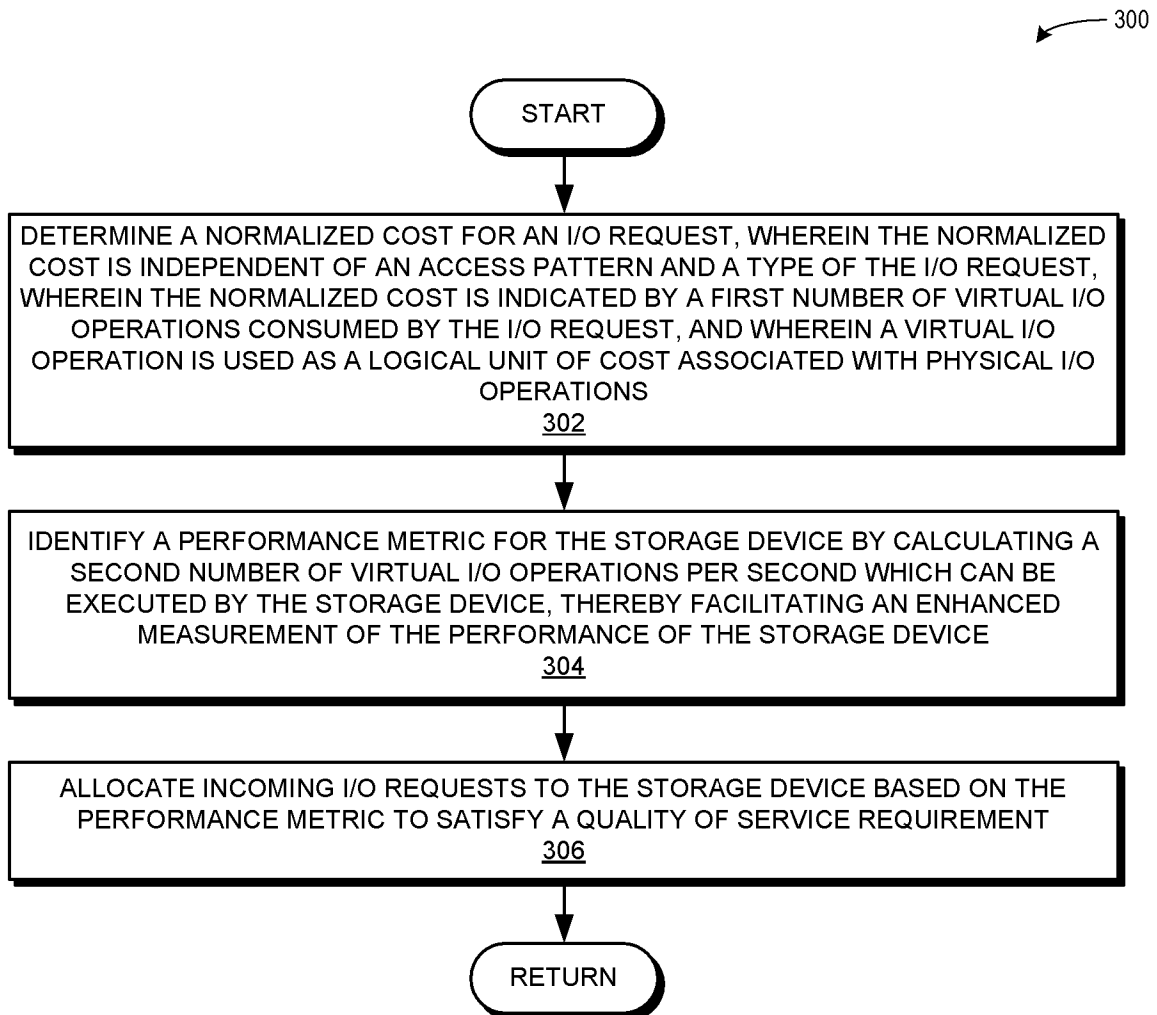
1 20. The apparatus of claim 19, wherein the cost-determining module is
2 further configured to:
3 determine a cost factor for the I/O request, which involves:
4 selecting a set of test cost factor values;
5 executing varied workloads on the storage device based on varying
6 I/O request sizes and a varying ratio of read I/O requests to write I/O
7 requests;
8 measuring the performance of the storage device based on a third
9 number of virtual I/O operations per second executed on the storage
10 device for each workload;
11 determining a test cost factor value which projects a most constant
12 virtual I/O operations per second performance across all the workloads;
13 and

- 14 setting the cost factor for the I/O request to the determined test cost
- 15 factor value;
- 16 determine a first ratio of a size of the I/O request to a predetermined size
- 17 for a base unit; and
- 18 scale the first ratio by the determined cost factor for the I/O request.

**FIG. 1**

**FIG. 2A**

**FIG. 2B**

**FIG. 3A**

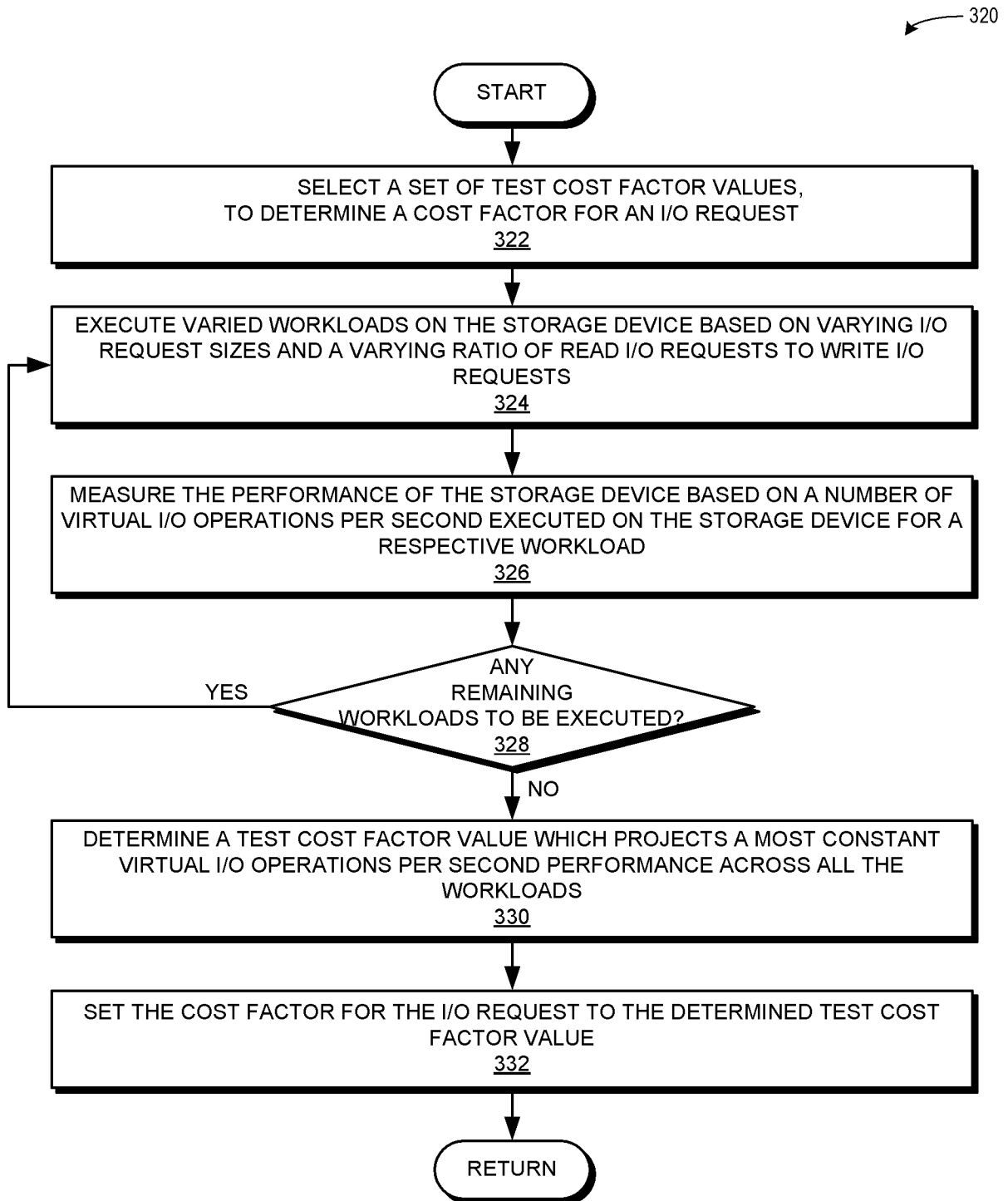


FIG. 3B

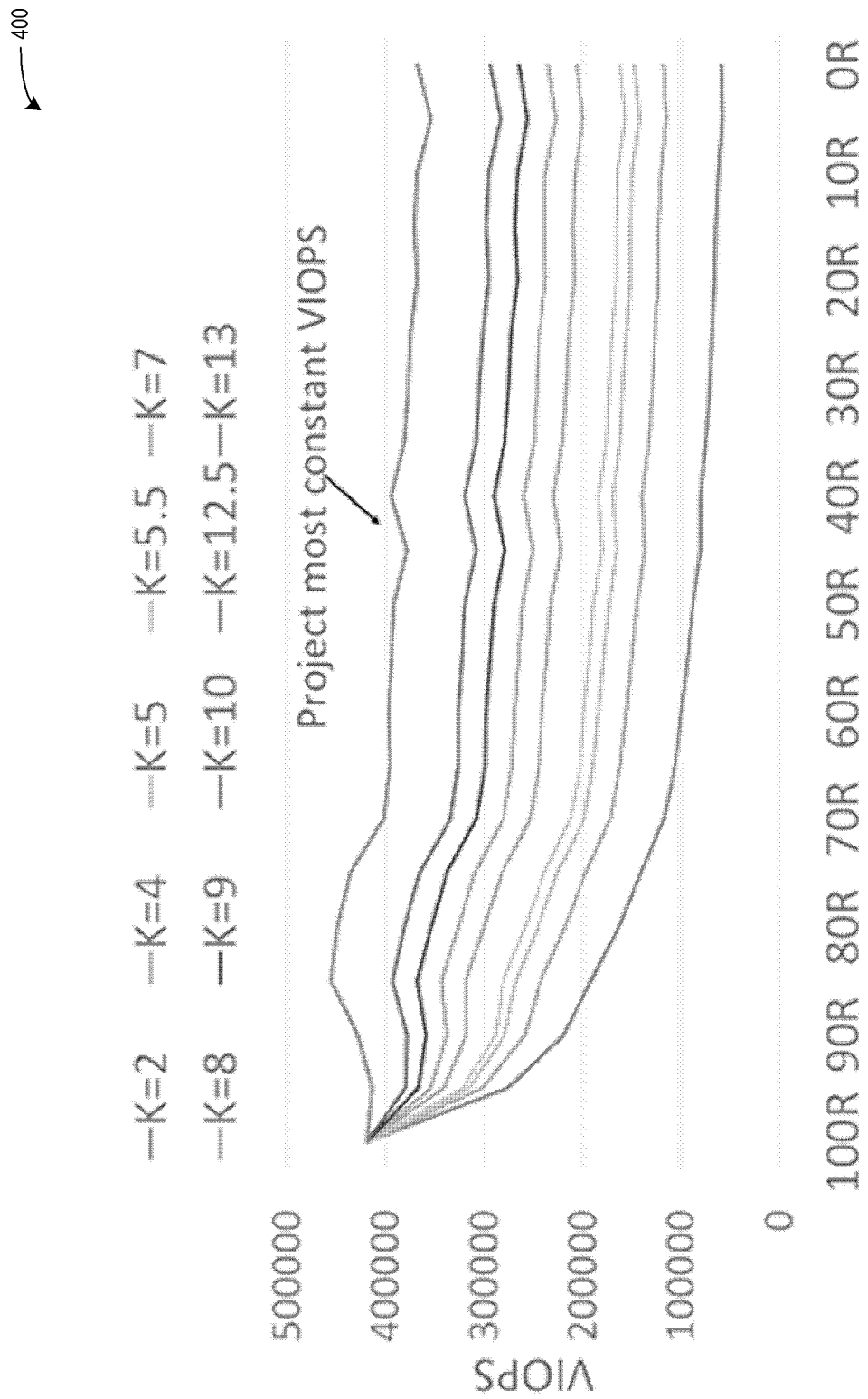
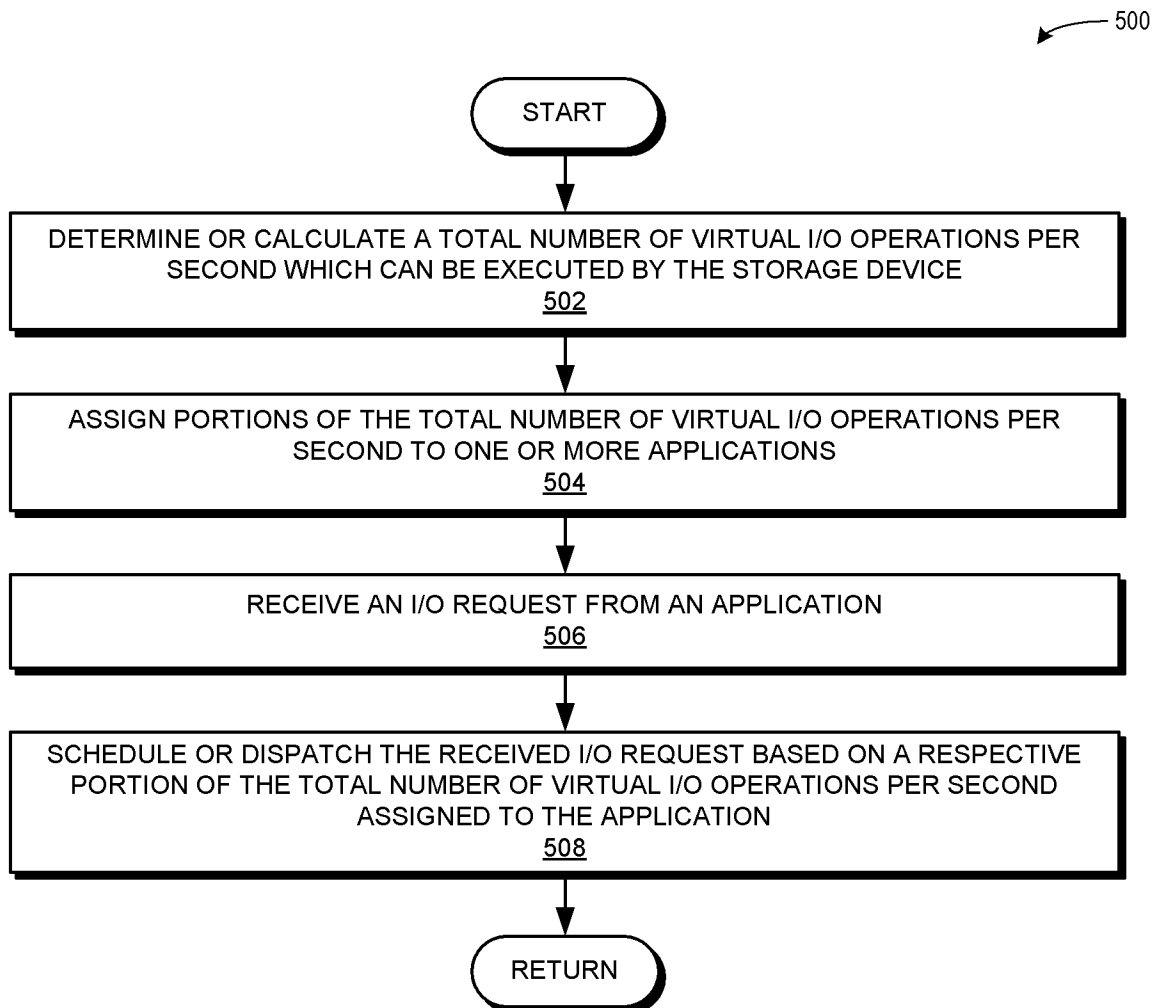


FIG. 4

**FIG. 5**

 600

```
Req dequeue()
{
    bool found = false;

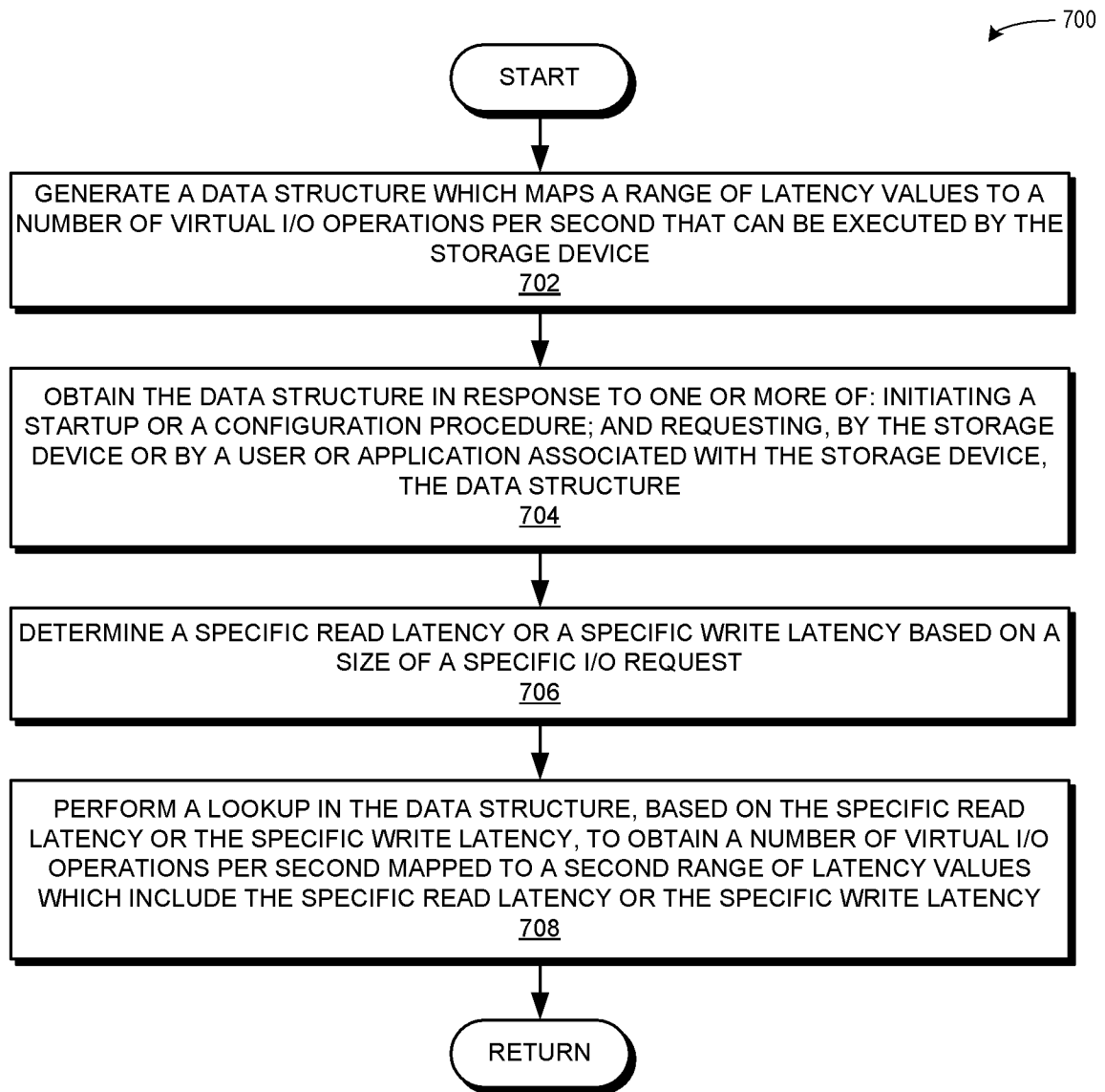
    //anti-starvation policy
    If (req) {
        //priority queue
        req = get_request_from_priority_queue();
        If (req && has_sufficient_io_budget(req)) {
            found = true;
        }
    }

    If (found) {
        //update available viops budget
        update_io_budget(req);

        //update last dispatched IO's timestamp of current req's priority
        update_starvation_timestamp(req);

        //remove request from priority queue
        remove_request_from_priority_queue(req);
        return req;
    } else {
        return NULL;
    }
}
```

FIG. 6

**FIG. 7A**

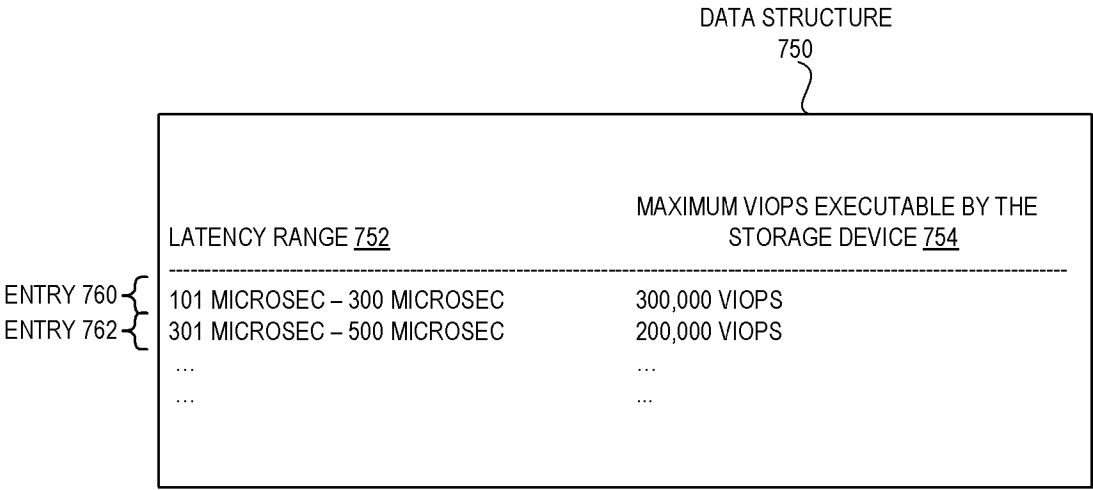


FIG. 7B

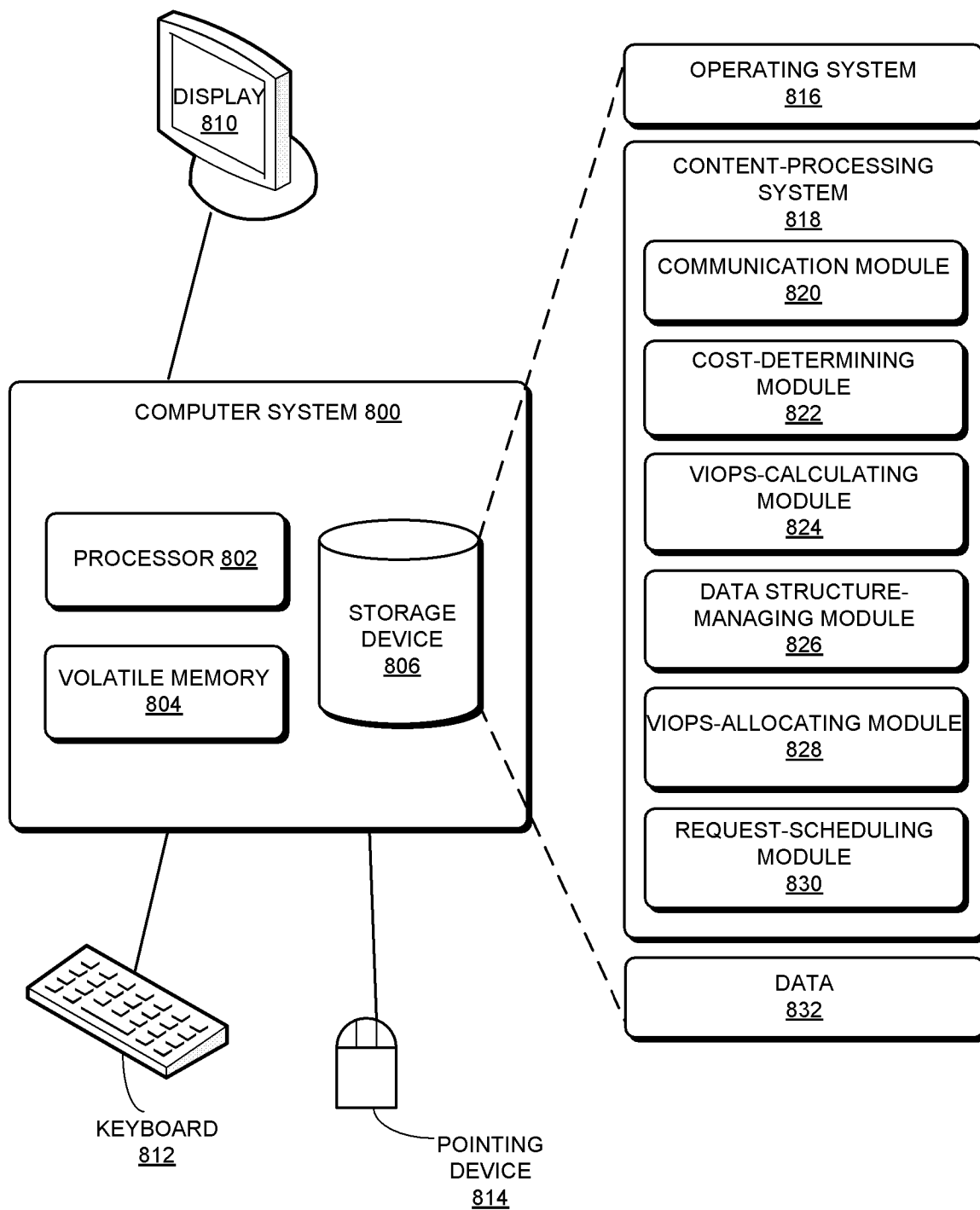
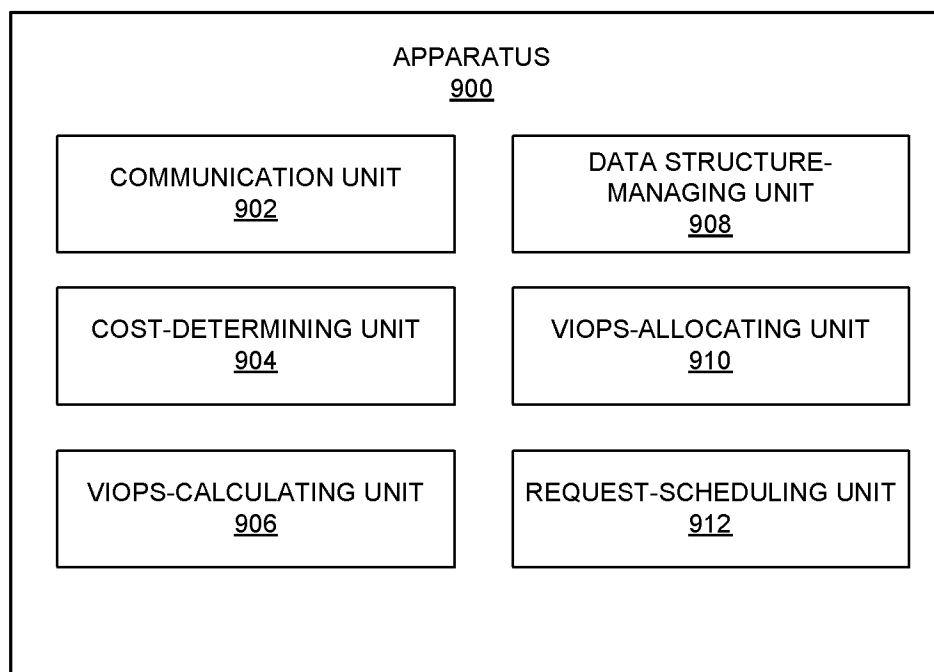


FIG. 8

**FIG. 9**

INTERNATIONAL SEARCH REPORT

International application No.

PCT/CN2018/092632

A. CLASSIFICATION OF SUBJECT MATTER

G06F 9/50(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

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

WPI,EPODOC,CNKI,CNPAT,IEEE: storage, normalized cost, I/O request, virtual I/O operations, logical unit, performance metric, allocate, Quality of Service

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	US 2018067771 A1 (PURE STORAGE, INC.) 08 March 2018 (2018-03-08) claims 1-20, description, paragraphs [0017]-[0120] and figures 1-14	1-20
A	US 8347302 B1 (AMAZON TECHNOLOGIES, INC.) 01 January 2013 (2013-01-01) the whole document	1-20
A	US 2017177270 A1 (HITACHI, LTD.) 22 June 2017 (2017-06-22) the whole document	1-20
A	EP 2966562 A1 (NEXENTA SYSTEMS, INC.) 13 January 2016 (2016-01-13) the whole document	1-20



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

11 January 2019

Date of mailing of the international search report

27 February 2019

Name and mailing address of the ISA/CN

National Intellectual Property Administration, PRC
6, Xitucheng Rd., Jimen Bridge, Haidian District, Beijing
100088
China

Authorized officer

TANG,Guangqiang

Facsimile No. (86-10)62019451

Telephone No. 86- (10) -53961734

INTERNATIONAL SEARCH REPORT
Information on patent family members

International application No.

PCT/CN2018/092632

Patent document cited in search report			Publication date (day/month/year)	Patent family member(s)			Publication date (day/month/year)
US	2018067771	A1	08 March 2018	US	10146585	B2	04 December 2018
				US	2018067775	A1	08 March 2018
				US	2018067772	A1	08 March 2018
				WO	2018048592	A1	15 March 2018
US	8347302	B1	01 January 2013	None			
US	2017177270	A1	22 June 2017	WO	2016038710	A1	17 March 2016
EP	2966562	A1	13 January 2016	US	2016011816	A1	14 January 2016