



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

(21) International Application Number:
PCT/US2014/048392

(22) International Filing Date:
28 July 2014 (28.07.2014)

(25) Filing Language: English

(26) Publication Language: English

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

(72) Inventors: **BEN ZEEV, Lior**; Shabazi 26, 56100 Yehud (IL). **WAISMAN, Vadim**; Shabazi 19, 56100 Israel (IL).

(74) Agents: **POSSETT, Ramya** et al.; Hewlett-Packard Company, Intellectual Property Administration, 3404 East Harmony Road Mail Stop 35, Fort Collins, Colorado 80528 (US).

(81) Designated States (unless otherwise indicated, for every kind of national protection available): AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BN, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT,

HN, HR, HU, ID, IL, IN, IR, IS, JP, KE, KG, KN, KP, KR, KZ, LA, LC, LK, LR, LS, LT, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PA, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SA, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.

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

Declarations under Rule 4.17:

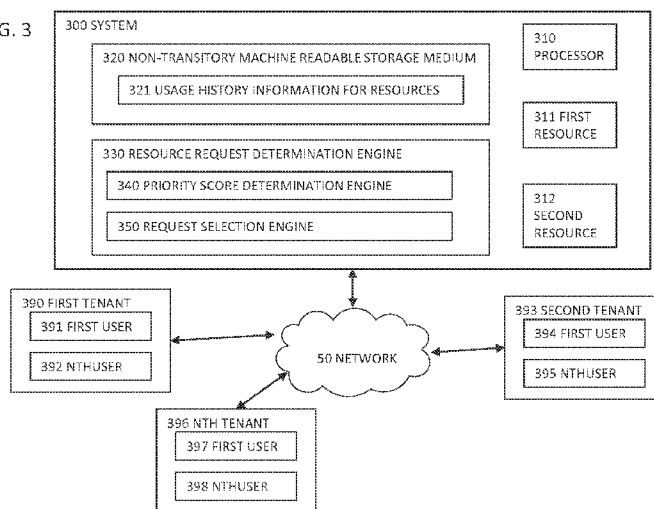
- as to the identity of the inventor (Rule 4.17(i))
- as to applicant's entitlement to apply for and be granted a patent (Rule 4.17(ii))

Published:

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

(54) Title: ACCESSING RESOURCES ACROSS MULTIPLE TENANTS

FIG. 3



(57) Abstract: Examples relate to accessing resources across multiple tenants. One example enables determination, from a list of requests, a request to perform which accesses a resource, responsive to a determination that the resource on a server is available for use for an application. The request to be performed may be determined, for example, by determining a first priority score for a first tenant associated with a first request based on usage of the resource by the first tenant, wherein the first tenant comprises a first set of users, determining a second priority score for a tenant associated with a second request based on usage of the resource by the second tenant, wherein the second tenant comprises a second set of users, and, responsive to the first priority score being lower than the second priority score, selecting the first request as the request to be performed.

ACCESSING RESOURCES ACROSS MULTIPLE TENANTS

BACKGROUND

[0001] Computing systems may allow multiple users to access resources via a single server. These computing systems may receive requests from the multiple users to access a resource via a server. For example, for an application executing on the server, a computing system may receive numerous requests to perform functionality via the application from multiple users, where each request may require access to a resource via the server.

BRIEF DESCRIPTION OF THE DRAWINGS

[0002] The following detailed description references the drawings, wherein:

[0003] FIG. 1 is a block diagram of an example system for accessing resources across multiple tenants;

[0004] FIG. 2 is a flowchart of an example system for accessing resources across multiple tenants;

[0005] FIG. 3 is a flowchart of an example system for accessing resources across multiple tenants;

[0006] FIG. 4 is a flowchart of an example method for accessing resources across multiple tenants;

[0007] FIG. 4A is a flowchart of an example method for determining a priority score for each tenant;

[0008] FIG. 4B is a flowchart of an example method for determining a priority score for each tenant.

DETAILED DESCRIPTION

[0009] The following detailed description refers to the accompanying drawings. Wherever possible, the same reference numbers are used in the drawings and the following description to refer to the same or similar parts. While several examples are described in this document, modifications, adaptations, and other implementations are possible. Accordingly, the following detailed description does not limit the disclosed

examples. Instead, the proper scope of the disclosed examples may be defined by the appended claims.

[0010] A computing system may provide access to a set of resources from a server executing an application used by multiple tenants. The computing system may manage performance of requests received by multiple tenants to ensure that a single tenant does not monopolize a resource or cause a long wait time for another tenant attempting to access the same resource. For example, responsive to a resource not being immediately available to perform a request received from a tenant, the computing system may maintain a list of pending requests for the resource.

[0011] The computing system may determine a request to perform from the list responsive to the resource becoming available. The system may determine the request to perform based on the tenants associated with the requests in the list. For example, the computing system may determine a request to perform based on usage of the resource by the tenants with requests in the list. The computing system may perform a request associated with a tenant calculated to have the least usage of the resource for the past predetermined number of time intervals. In some examples, the computing system may perform a request associated with a user of the tenant with the least usage of the resource. For example, the computing system may select a request to perform associated with a user calculated to have the least usage of the resource from a set of users associated with the tenant calculated to have the least usage of the resource.

[0012] As also noted below, each tenant may comprise an entity that includes a set of users (often multiple users). Each tenant (and each user of a tenant) may access the application via a respective tenant computing system (and user computing device). As such, discussion of functionality performed by a tenant (and/or user) may comprise functionality performed by a respective tenant computing system (and/or user computing device).

[0013] Referring now to the drawings, FIG. 1 is a block diagram of an example system 100 for accessing resources across multiple tenants. In the example depicted in FIG. 1, system 100 includes a non-transitory machine-readable storage medium 120 and a processor 110.

[0014] Referring now to the drawings, FIG. 1 is a block diagram of an example system 100 for accessing resources across multiple tenants. System 100 may comprise a cloud server, a mainframe, notebook, desktop, tablet, workstation, mobile device, and/or any other device suitable for executing the functionality described below. In the embodiment of FIG. 1, system 100 includes a non-transitory machine-readable storage medium 120 and a processor 110.

[0015] Processor 110 may be one or more central processing units (CPUs), microprocessors, and/or other hardware devices suitable for retrieval and execution of instructions stored in machine-readable storage medium 120. Processor 110 may fetch, decode, and execute program instructions 121, 122, 123, and/or other instructions to enable accessing resources across multiple tenants, as described below. As an alternative or in addition to retrieving and executing instructions, processor 110 may include one or more electronic circuits comprising a number of electronic components for performing the functionality of one or more of instructions 121, 122, 123, and/or other instructions.

[0016] In one example, the program instructions 121, 122, 123, and/or other instructions can be part of an installation package that can be executed by processor 110 to implement the functionality described herein. In this case, memory 120 may be a portable medium such as a CD, DVD, or flash drive or a memory maintained by a computing device from which the installation package can be downloaded and installed. In another example, the program instructions may be part of an application or applications already installed on system 100.

[0017] Machine-readable storage medium 120 may be any hardware storage device for maintaining data accessible to system 100. For example, machine-readable storage medium 120 may include one or more hard disk drives, solid state drives, tape drives, and/or any other storage devices. The storage devices may be located in system 100 and/or in another device in communication with system 100. For example, machine-readable storage medium 120 may be any electronic, magnetic, optical, or other physical storage device that stores executable instructions. Thus, machine-readable storage medium 120 may be, for example, Random Access Memory (RAM), an Electrically-Erasable Programmable Read-Only Memory (EEPROM), a storage drive, an optical

disc, and the like. As described in detail below, machine-readable storage medium 120 may be encoded with executable instructions for accessing resources across multiple tenants. As detailed below, storage medium 120 may maintain and/or store the data and information described herein.

[0018] Request determination instructions 121, when executed by processor 110, may determine, from a list of requests, a request to perform. The request determination instructions 121, when executed by processor 110, may determine the request to perform responsive to a resource on a server becoming available for use for an application. The application may be, for example, any application executing on a server in the system 100. The application may be executed on the server and may access other resources in the system via the server. In some examples, the application may be a cloud-based application, with an instance of the application being executed on the server in the system 100. The resources accessed by the application may comprise, for example, a database connection pool, a thread pool, a NoSQL connection pool, a file system, another computing device, and/or other resource that may be used by the application.

[0019] In some examples, the request determination instructions 121, when executed by processor 110, may receive a request to access the application from a tenant. A tenant may comprise, for example, an entity that accesses the application. A tenant may comprise a set of users, such that each user of the tenant may independently access the application. A set of independent accesses of the tenant may be performed based on a respective set of requests associated with the tenant. The respective set of requests associated with the tenant may be received from multiple users of the tenant, such that each request is associated with a user from the set of users of the tenant.

[0020] A request may comprise, for example, information related to one or more of: a request identifier, a tenant identifier of a tenant associated with the request, a user identifier of a user of the tenant associated with the request, an identification of a set of resources to be accessed by the request, data to be processed during performance of the request, a set of instructions associated with the request, an application priority associated with the request, and/or other information associated with the request.

[0021] Responsive to receiving a request to access the application from a tenant, the request determination instructions 121, when executed by processor 110, may determine

whether a resource to be accessed by the request is available to perform the request. In examples in which multiple resources are to be accessed by the request, the request determination instructions 121, when executed by processor 110, may determine whether each resource is available in the order in which the resource would be accessed by the request.

[0022] Responsive to determining that the resource (or set of resources) to be accessed by the request is (are) available, the request determination instructions 121, when executed by processor 110, may perform the request.

[0023] Responsive to determining that the resource to be accessed by the request is not available, the request determination instructions 121, when executed by processor 110, may place the request in a list of requests associated with that resource. For example, the request determination instructions 121, when executed by processor 110, may place the request in the list by appending information related to the request at the end of the list of requests. The information related to the request may comprise, for example, information related to the tenant associated with the request, information related to a user associated with the request, a request identifier, an application priority associated with the request, and/or other information related to the request. The list of requests may comprise, for example, a linked list, array, and/or other structure stored in a memory (e.g., the non-transitory storage readable medium 120) that stores the information related to each request in the list.

[0024] In some examples, each resource accessible by the application executing on the server may have associated therewith a list of requests. In some example, the request determination instructions 121, when executed by processor 110, may create a list for a resource responsive to the resource being unavailable to perform a request and having no other pending requests. The request determination instructions 121, when executed by processor 110, may maintain the list of requests for each resource. As such, responsive to a second resource accessible via the server becoming available for use for the application, the request determination instructions 121, when executed by processor 110, may determine a request to perform using the second resource from a list of resources for the second resource.

[0025] In some examples, the request determination instructions 121 may comprise priority score determination instructions 122, request selection instructions 123, and/or other instructions for determining a request to perform for the application. In some examples, the request determination instructions 121, when executed by processor 110, may determine priority scores for each request in the list of requests associated with a resource and may select a request to perform responsive to the resource becoming available based on the determined priority scores. In some examples, priority score determination instructions 122, when executed by processor 110, may determine, for each tenant associated with a request in the list of requests, a respective priority score based on usage of the resource by the tenant. In some examples, the request selection instructions 123, when executed by processor 110, may select a request from a tenant with a lowest priority score as the request to be performed.

[0026] Priority score determination instructions 122, when executed by processor 110, may determine, for each tenant associated with a request in the list of requests, a respective priority score based on usage of the resource by the tenant. For example, the priority score determination instructions 122, when executed by processor 110, may manage information related to usage of the resource for each tenant. For example, the priority score determination instructions 122, when executed by processor 110, may manage information related to usage of the resource by a tenant for a plurality of predetermined time intervals. In some examples, the priority score determination instructions 122, when executed by processor 110, may manage information related to usage of the resource for a predetermined number of prior time intervals in a first in, first out manner.

[0027] Information related to usage of the resource for an individual time interval may comprise, for example, a number of requests from the tenant for the resource, a percentage of processing capability of the resource used by the tenant, an amount of time of the time interval during which processing capability was used by the resource, and/or other metric to measure usage of the resource by the tenant during the time interval.

[0028] In some examples, the priority score determination instructions 122, when executed by processor 110, may manage information related to usage of the resource by

each user in a set of users of a tenant in a manner similar to the management of information related to each tenant.

[0029] The priority score determination instructions 122, when executed by processor 110, may determine, for each time interval for which information related to usage of the resource by the first tenant is stored in the storage medium 120, a respective amount of usage of the request by the first tenant. The priority score determination instructions 122, when executed by processor 110, may calculate a priority score for the tenant based on the respective amounts of usage. For example, the priority score determination instructions 122, when executed by processor 110, may use a formula to calculate the priority score for a tenant:

$$P[T]=T[X1]*N + T[X2]*(N-1)+...+T[Xn]*1.$$

[0030] In the example formula, $P[t]$ may comprise the priority score for the tenant, while N may comprise the number of predetermined time intervals over which priority score is calculated. $T[Xn]$ may comprise, for example, a metric representing usage of the resource by the tenant, where $X1$ may be the newest time interval and XN may be the oldest time interval.

[0031] In some examples, the priority score determination instructions 122, when executed by processor 110, may also consider a number of users in the first set of users while calculating the priority score. For example, a tenant with a larger number of users may have a priority score indicating an increased amount of access to the resource for that tenant as compared to a tenant with a smaller number of users.

[0032] In some examples, the priority score determination instructions 122, when executed by processor 110, may also consider an application priority of the request when calculating the priority score. The application priority of the request may indicate an importance of the request to the application. The application priority associated with the request may be determined from information related to the request, based on a number of resources the request may access, based on information provided by the application, and/or by other manners.

[0033] In some examples, the priority score determination instructions 122, when executed by processor 110, may also consider a user priority associated with a user associated with the request. For example, the user priority may be determined from

information related to the request, based on a role of the user as part of the tenant, based on a role of the user in the application, and/or in other manners.

[0034] In some examples, the priority score determination instructions 122, when executed by processor 110, may consider one or more additional factors (e.g., application priority, number of users in the tenant, user priority, and/or other factors) when calculating the priority score for a tenant. For example, the priority score determination instructions 122, when executed by processor 110, may use the following example formula to calculate the priority score for a tenant:

$$P[T] = (T[X1]*N + T[X2]*(N-1) + \dots + T[Xn]*1) * (MP+1-RP) * ((MT-TS)/100).$$

[0035] In the example formula, $P[t]$ may comprise the priority score for the tenant, while N may comprise the number of predetermined time intervals over which priority score is calculated. $T[Xn]$ may comprise, for example, a metric representing usage of the resource by the tenant, where $X1$ may be the newest time interval and XN may be the oldest time interval. RP may comprise an application priority of the request, while MP may comprise a maximum application priority for any request of the application. TS may comprise tenant size of the tenant, while AS may comprise average tenant size.

[0036] In some examples, the priority score determination instructions 122, when executed by processor 110, may only determine priority score for tenants associated with requests in the list of requests for a resource. In these examples, the request selection instructions 123, when executed by processor 110, may select a request associated with a tenant with the lowest priority score. For example, responsive to the list of requests comprising only a single request for the tenant with the lowest priority score, the request selection instructions 123, when executed by processor 110, may select that request to perform. In other examples, responsive to the list of requests comprising multiple requests for that tenant, the request selection instructions 123, when executed by processor 110, may randomly select a request from the multiple requests, may select the newest request, may select the oldest request, and/or may otherwise select a request to perform from the multiple requests associated with that tenant.

[0037] In some examples, the priority score determination instructions 122, when executed by processor 110, may determine a user priority score for users associated with requests of a tenant with the lowest priority score. The request selection instructions 123,

when executed by processor 110, may select a request associated with a user with the lowest priority score from a set of requests associated with the tenant with the lowest priority score.

[0038] For example, responsive to the list of requests comprising only a single request for the tenant with the lowest priority score, the request selection instructions 123, when executed by processor 110, may select that request to perform. In some examples, responsive to the list of requests comprising multiple requests for that tenant, the request selection instructions 123, when executed by processor 110, may select a request randomly, may select the newest request, may select the oldest request, may select a request based on a calculated user priority score for each user associated with a request of the multiple requests, and/or may otherwise select a request to perform.

[0039] In some examples, the request selection instructions 123, when executed by processor 110, may select a request associated with a user with the lowest priority score from a set of requests associated with the tenant with the lowest priority score. The priority score determination instructions 122, when executed by processor 110, may calculate a user priority score associated with each user associated with a request in a manner similar to calculating a priority score for a tenant associated with requests.

[0040] As mentioned above, information related to usage of the resource by each user of the tenant may be managed. For example, the priority score determination instructions 122, when executed by processor 110, may determine usage of the resource by each user for predetermined time intervals. In some examples, the priority score determination instructions 122, when executed by processor 110, may calculate a user priority score using one or more of the formulas and/or considering one or more additional factors as set forth above for calculating priority score for a tenant, using the metrics associated with usage by the user instead of usage by a tenant.

[0041] The priority score determination instructions 122, when executed by processor 110, may determine, for each tenant associated with a request in the list of requests, a respective priority score based on usage of the resource by the tenant in various manners, and is not limited to the examples described herein.

[0042] FIG. 2 is a block diagram of an example system 200 for accessing resources across multiple tenants. As with system 100, system 200 may comprise a cloud server, a

mainframe, notebook, desktop, tablet, workstation, mobile device, and/or any other device suitable for executing the functionality described below. As with processor 110 of FIG. 1, processor 210 may be one or more CPUs, microprocessors, and/or other hardware devices suitable for retrieval and execution of instructions.

[0043] As detailed below, system 200 may include a series of engines 220-240 for accessing resources across multiple tenants. Each of the engines may generally represent any combination of hardware and programming. For example, the programming for the engines may be processor executable instructions stored on a non-transitory machine-readable storage medium and the hardware for the engines may include at least one processor of the system 200 to execute those instructions. In addition or as an alternative, each engine may include one or more hardware devices including electronic circuitry for implementing the functionality described below.

[0044] Resource request determination engine 220 may determine, responsive to a resource on a server becoming available for use for an application, a request to perform from a list of requests, wherein the request comprises access to the resource. In some examples, the resource request determination engine 220 may determine the request to perform in a manner the same as or similar to that of the resource request determination instructions 122 of system 100. Further details regarding an example implementation of resource request determination engine 220 are provided above in connection with resource request determination instructions 121 of FIG. 1.

[0045] In some examples, resource request determination engine may comprise priority score determination engine 230, request selection engine 240, and/or other engines to determine a request to perform.

[0046] Priority score determination engine 230 may determine a respective priority score for each tenant associated with a request in the list based on usage of the resource by the tenant. In some examples, the priority score determination engine 230 may determine the priority scores for tenants in a manner the same as or similar to that of the system 100. Further details regarding an example implementation of priority score determination engine 230 are provided above in connection with priority score determination instructions 122 of FIG. 1.

[0047] Request selection engine 240 may select a request to perform the list of requests, wherein the selected request is associated with a tenant with the lowest priority score. In some examples, the request selection engine 240 may select the request to perform in a manner the same as or similar to that of the system 100. Further details regarding an example implementation of request selection engine 240 are provided above in connection with request selection instructions 123 of FIG. 1.

[0048] FIG. 3 is a block diagram of an example system for accessing resources across multiple tenants. As illustrated in FIG. 3 and described below, system 300 may comprise a processor 310, a non-transitory machine readable storage medium 320, a series of engines 330-350 for accessing resources across multiple tenants, a first resource 311, a second resource 312, and/or other components.

[0049] The system 300 may communicate with tenants via tenant computing systems 390, 393, 396, and/or other tenant computing systems. Each tenant system may comprise multiple user computing devices. For example, tenant computing system 390 for a first tenant may comprise computing devices 391 and 392 for a first user and second user, respectively. Similarly, tenant computing system 393 for a second tenant may comprise computing devices 394 and 395 for a first user and second user, respectively. Tenant computing system 396 for a third tenant may comprise computing devices 397 and 398 for a first user and second user, respectively. The number of tenant computing systems and user computing devices in each tenant system are not limited to the examples described herein.

[0050] As with processor 110 of FIG. 1, processor 310 may be one or more CPUs, microprocessors, and/or other hardware devices suitable for retrieval and execution of instructions. As with machine-readable storage medium 120, non-transitory machine-readable storage medium 320 may be any hardware storage device for maintaining data accessible to system 100. As with engines 220-240, engines 330-350 may generally represent any combination of hardware and programming.

[0051] Non-transitory machine-readable storage medium 320 may store usage history information 321 for resources. The usage history information 321 may comprise, for example, information related to usage of each resource accessible via the server for the application. The information related to usage of an individual resource may comprise, for

example, usage information by each tenant for a predetermined number of time intervals. The usage information for a tenant may be the same as or similar to the usage information described above.

[0052] Resource request determination engine 330 may perform functionality the same as or similar to that of the resource request determination engine 220 of system 200. Further details regarding an example implementation of resource request determination engine 330 are provided above in connection with resource request determination engine 220 of system 200.

[0053] In some examples, resource request determination engine 330 may comprise priority score determination engine 340, request selection engine 350, and/or other engines to determine a request to perform.

[0054] Priority score determination engine 340 may perform functionality the same as or similar to that of the priority score determination engine 230 of FIG. 2. Further details regarding an example implementation of priority score determination engine 340 are provided above in connection with priority score determination engine 230 of FIG. 2.

[0055] Request selection engine 350 may perform functionality the same as or similar to that of the request selection engine 240 of FIG. 2. Further details regarding an example implementation of request selection engine 350 are provided above in connection with request selection engine 240 of FIG. 2.

[0056] The resources 311, 312, and/or other resources of system 300 may comprise, for example, resources similar to the resources described with regard to FIG. 1. The resources 311, 312, and/or other resources may generally comprise a combination of hardware and software for storing information and/or providing functionality for the application.

[0057] As shown in FIG. 3, each tenant computing system (and user computing device) may communicate with system 100 to access resources of system 100 for the application.

[0058] FIG. 4 is a flowchart of an example method for execution by a computing device for accessing resources across multiple tenants.

[0059] Although execution of the methods described below are with reference to system 100 of FIG. 1 system 200 of FIG. 2, and/or system 300 of FIG. 3, other suitable devices for execution of this method will be apparent to those of skill in the art. The

method described in FIG. 4 and other figures may be implemented in the form of executable instructions stored on a machine-readable storage medium, such as storage medium 120, by one or more engines described herein, and/or in the form of electronic circuitry.

[0060] In an operation 400, a request to perform may be determined from a list of requests responsive to a resource on a server becoming available for use via an application, wherein performing the request comprises accessing the resource. For example, the system 100 (and/or the resource request determination instructions 121, the resource request determination engine 220, or other resource of the system 100) may determine the request to perform. The system 100 may determine the request to perform a manner similar or the same as that described above in relation to the execution of the resource request determination instructions 121, the resource request determination engine 220, and/or other resource of the system 100.

[0061] In some examples, operation 400 may comprise operations 410, 420, and/or other operations to determine a request to perform.

[0062] In an operation 410, a respective priority score for each tenant associated with a request in the list may be determined based on usage of the resource by each tenant. For example, the system 100 (and/or the priority score determination instructions 122, priority score determination engine 230, or other resource of the system 100) may determine the priority scores. The system 100 may determine the priority scores in a manner similar or the same as that described above in relation to the execution of the priority score determination instructions 122, priority score determination engine 230 of the system 100.

[0063] In some examples, a priority score for a tenant may be determined in various manners. FIG. 4A is a flowchart of an example method for execution by a computing device for determining a priority score for each tenant.

[0064] In an operation 411, a first amount of usage of the resource by a tenant for a first predetermined time interval may be determined. For example, the system 100 (and/or the priority score determination instructions 122, priority score determination engine 230, or other resource of the system 100) may determine the first amount of usage. The system 100 may determine the first amount of usage in a manner similar or the same

as that described above in relation to the execution of the priority score determination instructions 122, priority score determination engine 230 of the system 100.

[0065] In an operation 412, a second amount of usage of the resource by a tenant for a first predetermined time interval may be determined. For example, the system 100 (and/or the priority score determination instructions 122, priority score determination engine 230, or other resource of the system 100) may determine the second amount of usage. The system 100 may determine the second amount of usage in a manner similar or the same as that described above in relation to the execution of the priority score determination instructions 122, priority score determination engine 230 of the system 100.

[0066] In an operation 413, a priority score for a tenant may be calculated based on the first amount of usage and the second amount of usage. For example, the system 100 (and/or the priority score determination instructions 122, priority score determination engine 230, or other resource of the system 100) may calculate the priority score. The system 100 may calculate the priority score in a manner similar or the same as that described above in relation to the execution of the priority score determination instructions 122, priority score determination engine 230 of the system 100.

[0067] FIG. 4B is a flowchart of another example method for execution by a computing device for determining a priority score for each tenant.

[0068] In an operation 411, a user request related to a user of a tenant may be determined as the request to perform responsive to a priority score of the tenant being the lowest determined priority score. For example, the system 100 (and/or the priority score determination instructions 122, priority score determination engine 230, or other resource of the system 100) may determine the user request. The system 100 may determine the user request in a manner similar or the same as that described above in relation to the execution of the priority score determination instructions 122, priority score determination engine 230 of the system 100.

[0069] Returning to FIG. 4, in an operation 420, a request to be performed may be selected from the list of requests associated with a first tenant with a lowest priority score. For example, the system 100 (and/or the request selection instructions 123, the request selection engine 240, or other resource of the system 100) may select the request. The system 100 may select the request in a manner similar or the same as that described

above in relation to the execution of the request selection instructions 123, the request selection engine 240, or other resource of the system 100.

[0070] The foregoing disclosure describes a number of example embodiments for accessing resources across multiple tenants. The disclosed examples may include systems, devices, computer-readable storage media, and methods for accessing resources across multiple tenants. For purposes of explanation, certain examples are described with reference to the components illustrated in FIGS. 1-4B. The functionality of the illustrated components may overlap, however, and may be present in a fewer or greater number of elements and components. Further, all or part of the functionality of illustrated elements may co-exist or be distributed among several geographically dispersed locations. Moreover, the disclosed examples may be implemented in various environments and are not limited to the illustrated examples.

[0071] Further, the sequence of operations described in connection with FIGS. 1-4B are examples and are not intended to be limiting. Additional or fewer operations or combinations of operations may be used or may vary without departing from the scope of the disclosed examples. Furthermore, implementations consistent with the disclosed examples need not perform the sequence of operations in any particular order. Thus, the present disclosure merely sets forth possible examples of implementations, and many variations and modifications may be made to the described examples. All such modifications and variations are intended to be included within the scope of this disclosure and protected by the following claims.

CLAIMSWe claim:

1. A non-transitory machine-readable storage medium comprising instructions for accessing resources across multiple tenants, the instructions executable by a processor of a computing device to:

responsive to a resource on a server becoming available for use via an application, determine, from a list of requests, a request to perform which accesses the resource,

wherein determining the request from the list of requests comprises:

determining a first priority score for a first tenant associated with a first request based on usage of the resource by the first tenant, the first tenant comprising a first set of users;

determining a second priority score for a tenant associated with a second request based on usage of the resource by the second tenant, the second tenant comprising a second set of users;

responsive to the first priority score being lower than the second priority score, selecting the first request as the request to be performed.

2. The storage medium of claim 1, wherein the instructions to select the first priority score comprise instructions to:

manage, for a plurality of predetermined time intervals, information related to usage of the resource by a plurality of tenants, the plurality of tenants including the first tenant and the second tenant;

determine, for a first predetermined time interval for which information related to usage of the resource by the first tenant is stored in the storage medium, a first amount of usage of the request by the first tenant;

determine, for a second predetermined time interval for which information related to usage of the resource by the first tenant is stored in the storage medium, a second amount of usage of the request by the first tenant; and

calculate the first priority score based on the first amount of usage and the second amount of usage.

3. The storage medium of claim 2, wherein the instructions to determine the first priority score comprise instructions to:

calculate the first priority score further based on a number of users in the first set of users.

4. The storage medium of claim 3, wherein the instructions to determine the first priority score comprise instructions to:

calculate the first priority score further based on a first user priority score determined for a first user of the first set of users of the first tenant associated with the request.

5. The storage medium of claim 4, wherein the instructions to determine the first request comprise instructions to:

manage, for a plurality of predetermined time intervals, information related to usage of the resource by the first set of users of the first tenant; and

responsive to the first priority score being lower than the second priority score, determine a user request from the first set of users of the first tenant as the request to be performed, wherein determining the user request comprises:

determining, for a first predetermined time interval for which information related to usage of the resource by a first user of the first set of users is stored in the storage medium, a first user amount of usage of the request by the first user;

determining, for a second predetermined time interval for which information related to usage of the resource by the first user is stored in the storage medium, a second user amount of usage of the request by the first user;

calculating the first user priority score based on the first user amount of usage and the second user amount of usage;

calculating a second user priority score based on usage of the resource by a second user of the first set of users; and

responsive to the calculated first user priority score being less than the second user priority score, selecting a first user request associated with the first user as the user request, wherein the first user request comprises the first request.

6. The storage medium of claim 2, wherein the instructions to determine the first priority score comprise instructions to:

calculate the first priority score based on an application priority of the first request, wherein the application priority of the first request indicates an importance of the first request to the application.

7. The storage medium of claim 1, further comprising instructions executable by the processor to:

receive the first request to access the application from the first tenant;

determine whether the resource to be accessed by the first request is available to perform the first request; and

responsive to the resource not being available to perform the first request, place the first request in the list of requests.

8. The storage medium of claim 7, further comprising instructions executable by the processor to:

perform the first request responsive to determining that the resource to be accessed by the first request is available to perform the first request.

9. A system for accessing resources across multiple tenants, the system comprising:

a resource request determination engine to determine, responsive to a resource on a server becoming available for use via an application, a request to perform from a list of requests, wherein the request comprises access to the resource,

wherein the resource request determination engine comprises:

a priority score determination engine to:

determine a first priority score for a first tenant associated with a first set of requests in the list based on usage of the resource by the first tenant, the first tenant comprising a first set of users; and

determine a second priority score for a second tenant associated with a second set of requests in the list based on usage of the resource by the second tenant, the second tenant comprising a second set of users; and

a request selection engine to:

responsive to the first priority score being lower than the second priority score, select a first request from the first set of requests associated with the first tenant as the request to be performed.

10. The system of claim 9, wherein the resource request determination engine determines, responsive to a second resource on the server becoming available for use via the application, a second request to perform from a second list of requests, wherein the second request comprises access to the second resource.

11. The system of claim 9, wherein the priority score determination engine selects the first request from the first set of requests associated with the first tenant by:

managing, for a plurality of predetermined time intervals, information related to usage of the resource by the first set of users of the first tenant; and

determining a user request from the first set of users of the first tenant as the first request, wherein determining the user request comprises:

determining, for a first predetermined time interval for which information related to usage of the resource by a first user of the first set of users is stored in the storage medium, a first user amount of usage of the request by the first user;

determining, for a second predetermined time interval for which information related to usage of the resource by the first user is stored in the storage medium, a second user amount of usage of the request by the first user;

calculating the first user priority score based on the first user amount of usage and the second user amount of usage;

calculating a second user priority score based on usage of the resource by a second user of the first set of users; and

responsive to the calculated first user priority score being less than the second user priority score, selecting a first user request associated with the first user as the first request.

12. The system of claim 11, wherein the priority score determination engine:

receives the first request to access the application from the first tenant;

determines whether the resource to be accessed by the first request is available to perform the request; and

responsive to the resource not being available to perform the first request, places the first request in the list of requests.

13. A method for accessing resources across multiple tenants, the method comprising:

responsive to a resource on a server becoming available for use via an application, determine a request to perform from a list of requests, wherein performing the request comprises accessing the resource,

wherein determining the request from the list of requests comprises:

determining a respective priority score for each tenant associated with a request in the list based on usage of the resource by each tenant; and

selecting, as the request to perform, a request from the list of requests associated with a first tenant with a lowest priority score.

14. The method of claim 13, wherein determining a respective priority score for the first tenant comprises:

managing, for a plurality of predetermined time intervals, information related to usage of the resource by each tenant;

determining, for a first predetermined time interval for which information related to usage of the resource by the first tenant is stored in the storage medium, a first amount of usage of the request by the first tenant;

determining, for a second predetermined time interval for which information related to usage of the resource by the first tenant is stored in the storage medium, a second amount of usage of the request by the first tenant; and

calculating the respective priority score for the first tenant based on the first amount of usage and the second amount of usage.

15. The method of claim 13, further comprising:

receiving the request to access the application from the first tenant;

determining whether the resource to be accessed by the request is available to perform the request; and

responsive to the resource not being available to perform the request, placing the request in the list of requests.

FIG. 1

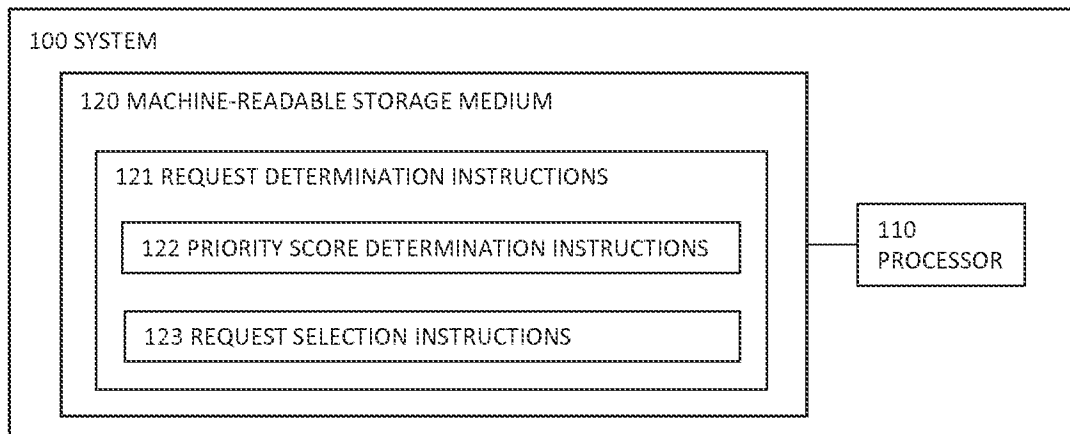


FIG. 2

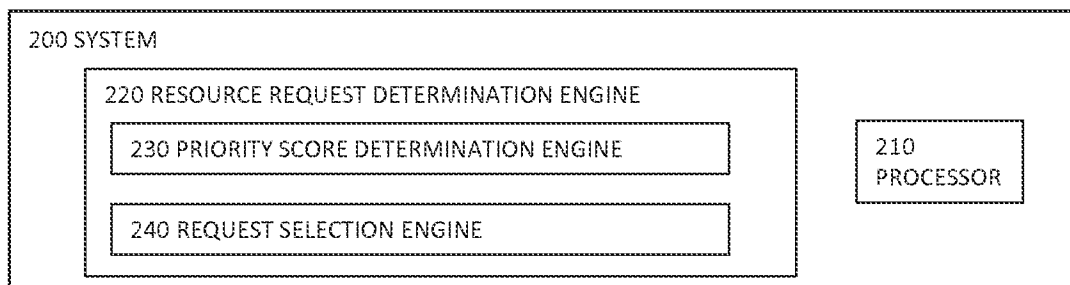


FIG. 3

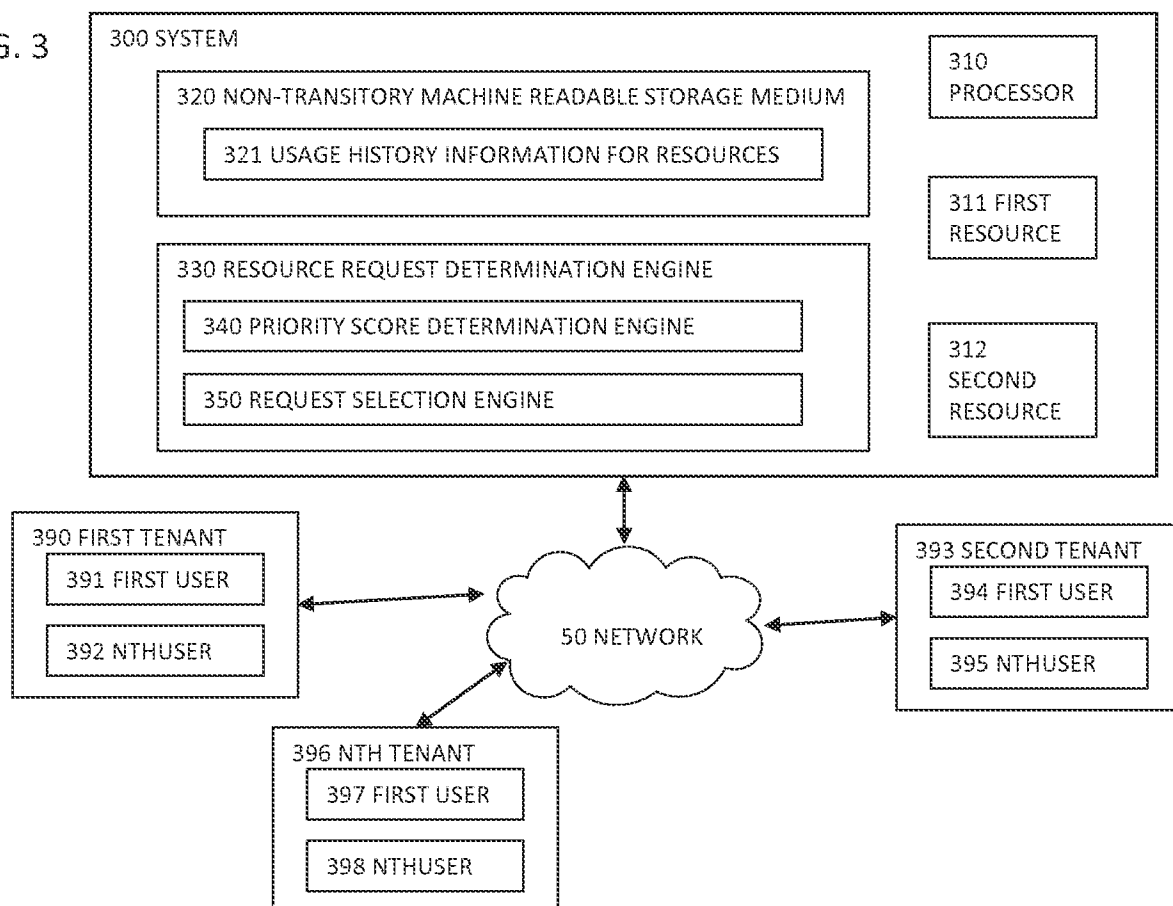


FIG. 4

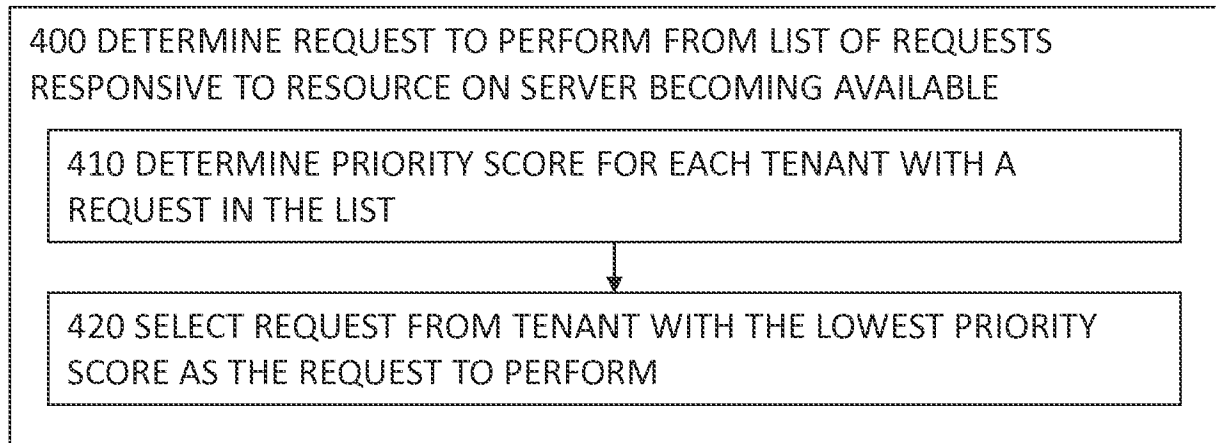


FIG. 4A

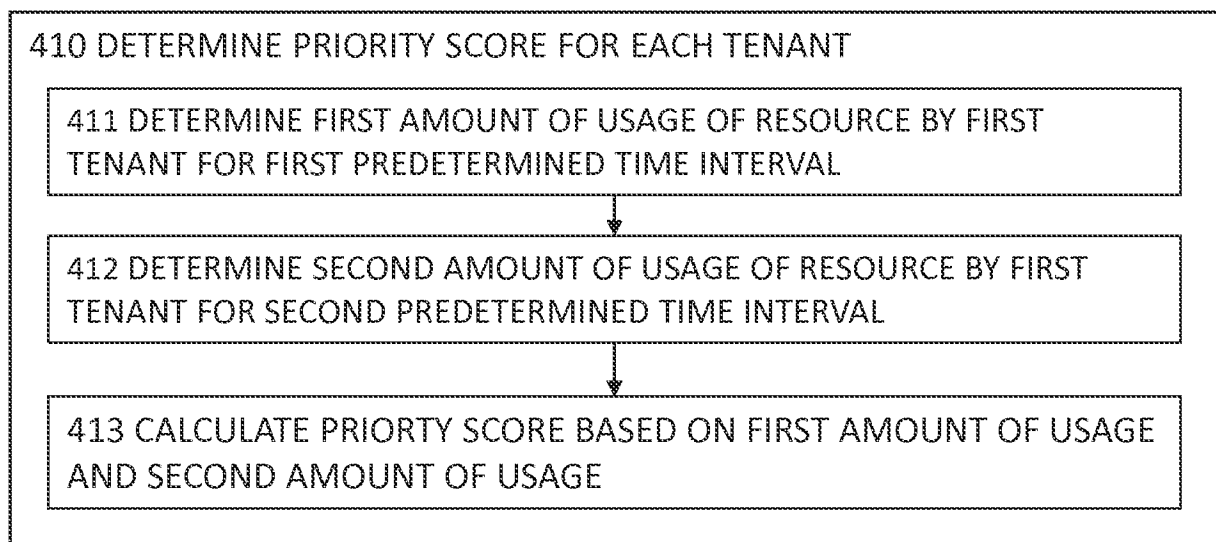
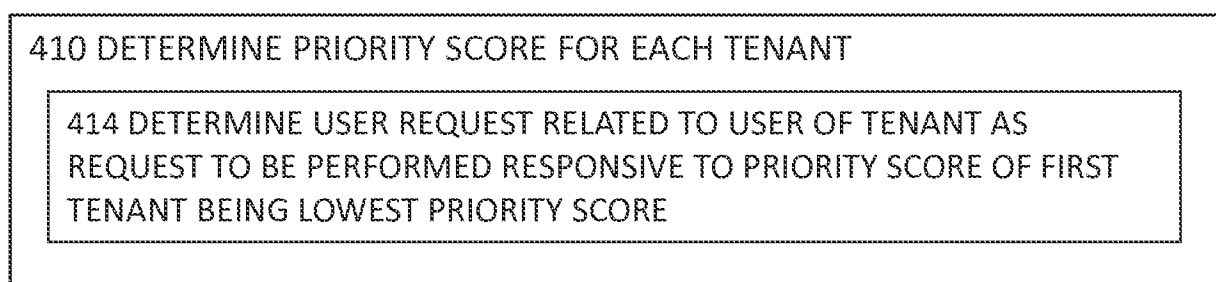


FIG. 4B



INTERNATIONAL SEARCH REPORT

International application No.
PCT/US2014/048392**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 9/50; H04L 29/08; G06Q 30/00; G06F 9/46; G06F 9/455; G06Q 10/00; G06F 21/00

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

Korean utility models and applications for utility models

Japanese utility models and applications for utility models

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

eKOMPASS(KIPO internal) & Keywords: tenant, determine, usage, priority

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	US 2013-0254407 A1 (WILLIAM D. PIJEWSKI et al.) 26 September 2013 See abstract, paragraphs [0005]–[0051], claims 1, 14, 17 and figures 1–2.	1–15
A	US 2010-0049570 A1 (LI XIN HUI et al.) 25 February 2010 See abstract, paragraphs [0007]–[0019], claims 1, 10 and figures 1(d),1(e),2.	1–15
A	US 2010-0077449 A1 (KWOK THOMAS YU-KIU et al.) 25 March 2010 See abstract, paragraphs [0038]–[0040], claim 1 and figure 5.	1–15
A	US 2013-0227560 A1 (MICHAEL P. MCGRATH et al.) 29 August 2013 See abstract, paragraphs [0023]–[0027], claim 1 and figure 1.	1–15



Further documents are listed in the continuation of Box C.



See patent family annex.

* Special categories of cited documents:

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier application or patent but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art

"&" document member of the same patent family

Date of the actual completion of the international search

23 April 2015 (23.04.2015)

Date of mailing of the international search report

24 April 2015 (24.04.2015)

Name and mailing address of the ISA/KR

International Application Division
Korean Intellectual Property Office
189 Cheongsu-ro, Seo-gu, Daejeon Metropolitan City, 302-701,
Republic of Korea

Facsimile No. ++82 42 472 7140

Authorized officer

KWON, Hyun Su

Telephone No. +82-42-481-8686



INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No.

PCT/US2014/048392

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 2013-0254407 A1	26/09/2013	US 8782224 B2	15/07/2014
US 2010-0049570 A1	25/02/2010	CN 101656657 A US 8347306 B2	24/02/2010 01/01/2013
US 2010-0077449 A1	25/03/2010	US 8424059 B2	16/04/2013
US 2013-0227560 A1	29/08/2013	None	