# SYSTEM FOR EFFICIENT PROCESSING OF TRANSACTION REQUESTS RELATED TO AN ACCOUNT IN A DATABASE

**Inventor:** Chunmao Zhu

## BACKGROUND

### Field

**[0001]** This disclosure is generally related to databases. More specifically, this disclosure is related to a system for efficiently processing transactions requests to transfer funds to or from an account in a database.

### Related Art

**[0002]** The progress of technology brings both an increasing convenience and risk to financial transactions. An accounting database can include information for multiple accounts, where funds may be transferred to and from an account by locking a corresponding record in the database. For example, a transaction request (e.g., to transfer funds into an account or to transfer funds out of an account) may include two main operations: generating a record of the request; and updating the balance on the account. To preserve the accuracy of the database while performing these operations, the record corresponding to the account may be locked upon receiving a transaction request, and may be automatically unlocked upon completion of the transaction request. The database may receive multiple transactions requests for one account at the same time. However, only one request or "thread" may hold the lock for the account at any given time. Other requests or threads must wait to obtain the lock, and the database can only perform the transaction requests on the account in a sequential manner. Thus, the account may become a "hotspot" account that creates a performance bottleneck for the database.

## SUMMARY

**[0003]** One embodiment provides a system that facilitates efficient processing of requests related to a database. During operation, the system receives, by a server, a request to transfer an

amount in to an account of a user, wherein the account is visible to the user and corresponds to a record in a database. The system selects one of a plurality of shadow accounts that correspond to the account, wherein a shadow account is not visible to the user. The system locks the selected shadow account, which prevents any changes to be made to the corresponding record in the database. The system modifies a balance of the selected shadow account by adding the amount to be transferred in to a current balance of the selected shadow account.

**[0004]** In some embodiments, the database includes a mapping between the account and the plurality of corresponding shadow accounts.

**[0005]** In some embodiments, the system generates a record of the request that indicates the account, the selected shadow account, and the amount.

**[0006]** In some embodiments, the system unlocks the selected shadow account.

**[0007]** In another embodiment, the system receives, by a server, a request to transfer an amount out of an account of a user, wherein the account is visible to the user and corresponds to a record in a database. The system selects one of a plurality of shadow accounts that correspond to the account, wherein a shadow account is not visible to the user. The system locks the selected shadow account, which prevents any changes to be made to the corresponding record in the database. In response to determining that a balance of the selected shadow account is less than the amount to be transferred out, the system does the following: sets a balance of the account to a combined balance of the plurality of shadow accounts; sets a balance of each shadow account to zero; executes the request on the selected shadow account or on the account; and distributes a first amount equal to the balance of the account from the account to the plurality of shadow accounts.

**[0008]** In some embodiments, in executing the request on the selected shadow account or on the account, the system unlocks the selected shadow account, locks the account, and modifies the balance on the account by subtracting the amount from a current balance of the account.

**[0009]** In some embodiments, in executing the request on the selected shadow account or on the account, the system transfers a designated amount from the account to the selected shadow account, wherein the designated amount is of a value greater than or equal to the amount to be transferred out, and modifies the balance on the selected shadow account by subtracting the amount from a current balance of the selected shadow account.

**[0010]** In some embodiments, in response to determining that a balance of the selected shadow account is not less than the amount to be transferred out, the system modifies the balance on the selected shadow account by subtracting the amount from a current balance of the selected shadow account.

**[0011]** In some embodiments, distributing the first amount is based on one or more of: a random distribution of the first amount from the account to the plurality of shadow accounts; an equal distribution of the first amount from the account to the plurality of shadow accounts, wherein an amount distributed into each of the shadow account is based on an average of the first amount divided by the number of shadow accounts; and a predetermined rule for distributing the first amount from the account to the plurality of shadow accounts.

**[0012]** In some embodiments, the database includes one or more of: an account number that corresponds to the account; a balance that indicates a current balance of the account; an account creation time that indicates when the account was created; an account modification time that indicates when the account was last modified; and a flag that indicates if the account is a shadow account.

**[0013]** In another embodiment, the system receives, by a server, a request to transfer an amount out of an account of a user, wherein the account is visible to the user and corresponds to a record in a database. In response to identifying no shadow accounts with a balance greater than or equal to the amount to be transferred, wherein a shadow account is not visible to the user, the system performs the following operations: sets a balance of the account to a combined balance of the plurality of shadow accounts; sets a balance of each shadow account to zero; executes the request on a selected shadow account or on the account; and distributes a first amount equal to the balance of the account from the account to the plurality of shadow accounts.

**[0014]** In some embodiments, in executing the request on a selected shadow account or on the account, the system locks the account and modifies the balance on the account by subtracting the amount from a current balance of the account.

**[0015]** In some embodiments, in executing the request on a selected shadow account or on the account, the system selects one of the plurality of shadow accounts that correspond to the account. The system transfers a designated amount from the account to the selected shadow account, wherein the designated amount is of a value greater than or equal to the amount to be transferred out. The system modifies the balance on the selected shadow account by subtracting the amount from a current balance of the selected shadow account.

**[0016]** In some embodiments, the system generates a record of the request that indicates one or more of the account, the selected shadow account, and the amount.

**[0017]** In some embodiments, in response to identifying one or more shadow accounts with a balance greater than or equal to the amount to be transferred out, wherein a shadow account is not visible to the user, the system performs the following operations: selects one of the identified shadow accounts; locks the selected shadow account, which prevents any changes to be made to the corresponding record in the database; and modifies the balance on the selected

shadow account by subtracting the amount from a current balance of the selected shadow account.

## BRIEF DESCRIPTION OF THE FIGURES

**[0018]** FIG. 1 illustrates an exemplary computing system that facilitates efficient processing of requests related to an account in a database, in accordance with an embodiment of the present application.

**[0019]** FIG. 2 presents a flowchart illustrating a method by a server for efficiently processing requests related to an account in a database, in accordance with an embodiment of the present application.

**[0020]** FIG. 3 presents a flowchart illustrating a method by a server for efficiently processing a request to transfer funds into an account in a database, in accordance with an embodiment of the present application.

**[0021]** FIG. 4A presents a flowchart illustrating a method by a server for efficiently processing a request to transfer funds out of an account in a database, in accordance with an embodiment of the present application.

**[0022]** FIG. 4B presents a flowchart illustrating a method by a server for efficiently processing a request to transfer funds out of an account in a database, in accordance with an embodiment of the present application.

**[0023]** FIG. 5A presents a flowchart illustrating a method by a server for efficiently processing a request to transfer funds out of an account in a database, including traversal of the corresponding shadow accounts, in accordance with an embodiment of the present application.

**[0024]** FIG. 5B presents a flowchart illustrating a method by a server for efficiently processing a request to transfer funds out of an account in a database, including traversal of the corresponding shadow accounts, in accordance with an embodiment of the present application.

**[0025]** FIG. 6 illustrates an exemplary apparatus that facilitates efficient processing of requests related to an account in a database, in accordance with an embodiment of the present application.

**[0026]** FIG. 7 illustrates an exemplary computer system that facilitates efficient processing of requests related to an account in a database, in accordance with an embodiment of the present application.

**[0027]** In the figures, like reference numerals refer to the same figure elements.

## DETAILED DESCRIPTION

**[0028]** 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. Thus, the present invention is not limited to the embodiments shown, but is to be accorded the widest scope consistent with the principles and features disclosed herein.

### Overview

**[0029]** Embodiments of the present invention solve the problem of a performance bottleneck in a database created by a user account that receives multiple concurrent transaction requests, where each request can only be completed sequentially by obtaining and releasing the same single lock for the account. The database can hold records corresponding to multiple user accounts, and a user can perform a transaction request on an account via a web, mobile, or other computing interface or device. A transaction request can include a request to transfer funds into a user account or out of a user account. Processing such a request accurately requires locking and unlocking an account (e.g., locking a row corresponding to the user account in the database) so that each transaction request may be performed atomically on the account during the time that the request or "thread" holds the lock.

**[0030]** As both the number of accounts and the number of transaction requests on an account increase, the database may receive an increased number of concurrent transaction requests on a given account. However, only one of the multiple concurrent requests may hold the lock for the account at any given time. As a result, the high volume of locking and unlocking operations executed on the account may lead to a "hotspot" in the database. The hotspot may create a performance bottleneck and lead to decreased efficiency for the database. The present system addresses these issues by creating a plurality of shadow accounts for each account, where a shadow account is not visible to the user. The system determines an optimal number of shadow accounts for a given account and distributes the balance of the account to the shadow accounts. The system can subsequently process a transaction request for the account by selecting one of the corresponding shadow accounts, and locking only the selected shadow account. This allows the system to continue processing transaction requests related to the same account, as only the one selected shadow account is locked, while the account itself remains unlocked.

### Exemplary Network and Environment

**[0031]** FIG. 1 illustrates an exemplary computing system that facilitates efficient processing of requests related to an account in a database, in accordance with an embodiment of the present application. In this example, system 100 can include computing devices 130, 132, and 134, which are associated with users 120, 122, and 124, respectively. Computing devices 130-134 can include, for example, a tablet, a mobile phone, an electronic reader, a laptop computer, a desktop computer, or any other computing device. Computing devices 130-134 can communicate with servers 142 and 144 via a network 140. Server 142 can communicate with a storage device 142.1, and server 144 can communicate with a storage device 144.1. In some embodiments, storage devices 142.1 and 144.1 reside, respectively, on servers 142 and 144. Storage device 142.1 can store a table 190, which can include multiple entries, each of which is related to an account. For example, table 190 can include an entry for an account that includes the following: an account name 190.1 that indicates a user-defined or system-defined name for the account; an account number 190.2 that indicates the number for the account; an account balance 190.3 that indicates the current balance for the account; an account creation date 190.4 that indicates the time and date that the account is created; an account modification time 190.5 that indicates the most recent time that the account is modified; a shadow flag 190.6 that indicates whether the account is a shadow account; and a list of corresponding shadow accounts 190.7 that indicates plurality of shadow accounts that are not visible to a user, where each shadow account is identified in list 190.7 by an account number or name for the shadow account and a creation date for the shadow account. Note that the entry described in table 190 can also pertain to a shadow account, in which case shadow flag 190.6 is set to indicate that the account is a shadow account, and list of corresponding shadow accounts 190.7 is set to null.

**[0032]** Thus, system 100 depicts a system that facilitates efficient processing of requests to transfer funds into and out of an account in a database. The system provides a method to distribute transaction requests relating to a single user account by creating a set of corresponding shadow accounts for each user account, and performing the functionality described herein.

## Establishing a System to Efficiently Process Requests on an Account

**[0033]** FIG. 2 presents a flowchart illustrating a method by a server for efficiently processing requests related to an account in a database, in accordance with an embodiment of the present application. During operation, the system creates, by a server, an account with an account balance of a value greater than zero (operation 202). The system determines an optimal number ("N") of shadow accounts for the account (operation 204). A shadow account is a "secondary" account that corresponds to an account, and is not visible to a user. A shadow account may contain the same structure as an account, as described above in relation to table 190

of FIG. 1. The greater the number of shadow accounts, the better the reduction of "hotspot" accounts in a database. However, increasing the number of shadow accounts may result in performance issues due to additional space and resource consumption. Thus, the system can determine the optimal number N of shadow accounts corresponding to an account based on a balanced consideration and analysis of hotspot reduction and resource consumption. For example, the system can identify a hotspot based on the amount of time spent to complete a transaction request on an account, or based on a large number of failure messages associated with an account (e.g., a "Failure to acquire database lock" message). These characteristics may be logged in a system log. The system can analyze the logs over a certain period of time, gradually increase the number of shadow accounts, and continue analyzing the system logs to determine the optimal number N of shadow accounts. The analysis can include detecting that no related error report is generated in the system log, or that the amount of time spent to complete a transaction request on the account falls within a predetermined acceptable range.

**[0034]** Upon determining the optimal number N of shadow accounts, the system can create N shadow accounts corresponding to the account (operation 206). The system can create a mapping relationship between the account and the N shadow accounts (operation 208) (e.g., list of corresponding shadow accounts 190.7 as depicted and described in relation to table 190 of FIG. 1). The system can subsequently distribute the value of the account balance into the N shadow accounts (operation 210). This distribution can be performed based on several methods. A first method is a random distribution, where the account balance is randomly divided into N parts, and allocated to the N shadow accounts. A second method is average distribution, where the account balance ("M") is divided equally into N parts, and the average amount of M divided by N is allocated to the N shadow accounts. A third method can be based on one or more predetermined rules which determine the amount to be allocated to each of the N shadow accounts. The system sets the value of the account balance to zero (operation 212), and can subsequently receive a transaction request (operation 214). A transaction request can include a request to transfer funds into an account, or a request to transfer funds out of an account. Processing a transaction request is described below in relation to FIGs. 3, 4A-4B, and 5A-5B.

**[0035]** In addition, upon creating the N shadow accounts and distributing the balance of the account to the N shadow accounts, the system may receive a query for the balance of the account. In order to determine the balance of the account, the system can combine or aggregate the balance from each of the N shadow accounts (without transferring any amounts).

## Efficiently Processing a Request to Transfer Funds Into an Account

**[0036]** FIG. 3 presents a flowchart illustrating a method by a server for efficiently processing a request to transfer funds into an account in a database, in accordance with an embodiment of the present application. During operation, the system receives, by a server, a transaction request to transfer an amount into an account ("transfer-in transaction request") (operation 302). The system selects a shadow account from N shadow accounts corresponding to the account (operation 304). The system may select the shadow account randomly or based on a predetermined rule. For example, the system can order the N shadow accounts based on the value of their balance, and select the shadow account with the smallest balance. The system locks the selected shadow account (operation 306) and executes the transfer-in transaction request on the selected shadow account (operation 308). The system modifies the balance on the selected shadow account by adding the amount to be transferred in to a current balance of the selected shadow account (operation 310). Upon completion of the modification of the balance, the system can automatically unlock the selected shadow account (operation 312). The system then generates a record of the executed transaction based on the transfer-in transaction request and the selected shadow account (operation 314). For example, the generated record can indicate the account, the selected shadow account, the type of transaction request, the name or number of the shadow account, the time of completion of the transfer-in transaction request, and the amount involved in the transaction request.

**Efficiently Processing a Request to Transfer Funds Out of an Account**

**[0037]** FIG. 4A presents a flowchart illustrating a method by a server for efficiently processing a request to transfer funds out of an account in a database, in accordance with an embodiment of the present application. During operation, the system receives, by a server, a transaction request to transfer an amount out of an account ("transfer-out transaction request") (operation 402). The system selects a shadow account from N shadow accounts corresponding to the account (operation 404). As described above, the system may select the shadow account randomly or based on a predetermined rule (e.g., as described below in relation to operations 504-510 of FIG. 5A). The system locks the selected shadow account (operation 406). The system then determines whether the balance of the selected shadow account is less than the amount to be transferred out (decision 408). If it is not, the system executes the transfer-out transaction request on the selected shadow account (operation 410) and modifies the balance on the selected shadow account by subtracting the amount form a current balance of the selected shadow account (operation 412). Upon completion of the modification of the balance, the system can automatically unlock the selected shadow account (operation 414). The system then

generates a record of the executed transaction based on the transfer-out transaction request and the selected shadow account (operation 430).

**[0038]** If the balance of the selected shadow account is less than the amount to be transferred out, the system sets the balance of the account to the aggregated balance from the N shadow accounts (operation 420), and sets the balance of each of the N shadow accounts to zero (operation 422). In other words, the system adds or combines the balance from each of the N shadow accounts, moves the combined balance to the account, and resets the balance on each of the N shadow accounts to zero. At this point, the operation can continue as described at Label A or Label B of FIG. 4B.

**[0039]** FIG. 4B presents a flowchart illustrating a method by a server for efficiently processing a request to transfer funds out of an account in a database, in accordance with an embodiment of the present application. Label A depicts an operation performed on the account, while Label B depicts an operation performed on the selected shadow account. Under Label A, the system can determine whether a negative balance is allowed on the account (decision 452). If a negative balance is not allowed on the account, the system determines whether the aggregated balance on the account is greater than the amount to be transferred out (decision 454). If it is not, the transaction request fails and the system generates a failure alert (operation 456), and the operation ends. If the aggregated balance is greater than the amount to be transferred out, the system can proceed with operation 458. If a negative balance is allowed on the account (decision 452), the system can unlock the selected shadow account and lock the account (operation 458). The system can execute the transfer-out transaction request on the account (operation 460), modify the balance on the account by subtracting the amount to be transferred out from the current balance of the account (operation 462), and unlock the account (operation 464). The operation then continues as described in relation to FIG. 4A, operation 428. In some embodiments, upon setting the balance of all the shadow accounts to zero, the system automatically unlocks the account and the shadow accounts. The system can also maintain the lock on the account and the shadow accounts so that the account does not need to be locked again.

**[0040]** Under Label B, the system can transfer a designated amount from the account to the selected shadow account (operation 472). The designated amount can be of a value greater than or equal to the amount to be transferred out, and can be determined by the system based on a predetermined rule. For example, the designated amount can be an amount that leaves a predetermined margin, amount, or percentage upon completion of the transfer-out request. The system can execute the transfer-out transaction request on the selected shadow account (operation 474), and modify the balance on the selected shadow account by subtracting the amount from a

current balance of the selected shadow account (operation 476). The operation then continues as described in relation to FIG. 4A, operation 428.

**[0041]** Upon completing the operations described above in relation to FIG. 4B, the system distributes the balance of the account from the account to the N shadow accounts (operation 428). The account balance is the remaining account balance after completion of the operations described in FIG. 4B, which includes executing the transfer-out transaction request on the account or the selected shadow account. The remaining account balance can be distributed from the account to the N shadow accounts based on the methods described in relation to FIG. 2 (i.e., based on a random distribution, an average distribution, or a predetermined rule). Finally, the system generates a record of the executed transaction based on the transfer-out transaction request and the selected shadow account (operation 430).

**Efficiently Processing a Request to Transfer Funds Out of an Account by Traversing Shadow Accounts**

**[0042]** FIG. 5A presents a flowchart illustrating a method by a server for efficiently processing a request to transfer funds out of an account in a database, including traversal of the corresponding shadow accounts, in accordance with an embodiment of the present application. During operation, the system receives, by a server, a transaction request to transfer an amount out of an account ("transfer-out transaction request") (operation 502). The system determines whether there are any shadow accounts corresponding to the account with a balance greater than or equal to the amount to be transferred out (decision 504) (e.g., a "sufficiently funded shadow account"). The system can traverse the entire set of N shadow accounts to identify one or more sufficiently funded shadow accounts, or the system can traverse the set of N shadow accounts and stop traversing upon identifying one sufficiently funded shadow account. The system can traverse the set of N shadow accounts corresponding to the account based on any order or characteristic, such as by shadow account number, shadow account creation date, or shadow account modification date. If there are one or more sufficiently funded shadow accounts, the system determines whether there are multiple sufficiently funded shadow accounts (i.e., two or more sufficiently funded shadow accounts) (operation 506). If there are not (e.g., the system identifies only one sufficiently funded shadow account), the system selects that one shadow account (operation 508). If there are multiple shadow accounts (e.g., the system identifies two or more sufficiently funded shadow accounts), the system selects one of the multiple shadow accounts (operation 510). The system can perform this selection either randomly or based on a predetermined rule. For example, the system can select the shadow account with the greatest balance from the identified set of multiple shadow accounts.

**[0043]** Upon selecting the shadow account, the system locks the selected shadow account (operation 512), executes the transfer-out transaction request on the selected shadow account (operation 514), and modifies the balance on the selected shadow account by subtracting the amount from a current balance of the selected shadow account (operation 516). The system then generates a record of the executed transaction based on the transfer-out transaction request and the selected shadow account (operation 530).

**[0044]** If there is no shadow account with a balance greater than or equal to the amount to be transferred out (decision 504), the system sets the balance of the account to the aggregated balance from the N shadow accounts (operation 520), and sets the balance of each of the N shadow accounts to zero (operation 522). This is similar to operations 420 and 422 described in relation to FIG. 4A. At this point, the operation can continue as described at Label C or Label D of FIG. 5B.

**[0045]** FIG. 5B presents a flowchart illustrating a method by a server for efficiently processing a request to transfer funds out of an account in a database, including traversal of a plurality of shadow accounts, in accordance with an embodiment of the present application. Label C depicts an operation performed on the account, while Label D depicts an operation performed on the selected shadow account. Under Label C, the system can lock the account (operation 552). The system can execute the transfer-out transaction request on the account (operation 554) and modify the balance on the account by subtracting the amount to be transferred out from the current balance of the account (operation 556). Note that while operations and functionality pertaining to a negative account balance are not depicted in FIG. 5B, the operations of FIG. 5B can include that functionality (e.g., as described in relation to operations 452, 454, and 456 of FIG. 4B).

**[0046]** Under Label D, the system can select and lock a shadow account (operation 571), which is similar to operations 404 and 406 described in relation to FIG. 4A. The system can transfer a designated amount from the account to the selected shadow account (operation 572). The designated amount can be of a value greater than or equal to the amount to be transferred out, and can be determined by the system based on a predetermined rule. The system can execute the transfer-out transaction request on the selected shadow account (operation 574) and modify the balance on the selected shadow account by subtracting the amount from a current balance of the selected shadow account (operation 567). The operation then continues as described in relation to FIG. 5A, at operation 528.

**[0047]** Upon completing the operations described above in relation to FIG. 5B, the system distributes the balance of the account (which is the remaining account balance after completion of the operations described in FIG. 5B, which includes executing the transfer-out

transaction request) from the account to the N shadow accounts (operation 528). The remaining account balance can be distributed from the account to the N shadow accounts based on the methods described in relation to FIG. 2. Finally, the system generates a record of the executed transaction based on the transfer-out transaction request and the selected shadow account (operation 530).

## Exemplary Apparatus and Computer System

**[0048]** FIG. 6 illustrates an exemplary apparatus that facilitates efficient processing of requests related to an account in a database, in accordance with an embodiment of the present application. Apparatus 600 can comprise a plurality of modules which may communicate with one another via a wired or wireless communication channel. Apparatus 600 may be realized using one or more integrated circuits, and may include fewer or more modules than those shown in FIG. 6. Further, apparatus 600 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 600 can comprise a communication module 602, a shadow account selecting module 604, a shadow account processing module 606, a balance modifying module 608, a balance distributing module 610, and a record generating module 612.

**[0049]** In some embodiments, communication module 602 can send and/or receive data packets to/from other network nodes across a computer network, such as a content centric network, where a data packet can correspond to a request to transfer an amount in to or out of an account of a user. Shadow account selecting module 604 can select one of a plurality of shadow accounts that correspond to the account. Shadow account processing module 606 can lock and unlock the selected shadow account or the account. Balance modifying module 608 can modify the balance of a selected shadow account by adding the amount to be transferred in to or by subtracting the amount to be transferred out from a current balance of the selected shadow account. Record generating module 612 can generate a record that indicates the account, the selected shadow account, and the amount transferred in or out.

**[0050]** In response to determining that a balance of a selected shadow account is less than the amount to be transferred out, or in response to identifying no shadow accounts with a balance greater than or equal to the amount to be transferred out (shadow account processing module 606), balance modifying module 608 can also set a balance of the account to a combined balance of the shadow accounts, and can further set the balance of each shadow account to zero. Balance distributing module 610 can distribute a first amount equal to the balance of the account to the shadow accounts.

**[0051]** Balance modifying module 610 can further transfer a designated amount from the account to a selected shadow account, where the designated amount is of a value greater than or equal to an amount specified in the request. In response to determining that a balance of a selected shadow account is not less than the amount to be transferred out (shadow account processing module 606), balance modifying module 608 can modify the balance on the selected shadow account by subtracting the amount from a current balance of the selected shadow account.

**[0052]** In response to identifying one or more shadow accounts with a balance greater than or equal to the amount to be transferred out (shadow account processing module 724), various modules of apparatus 600 can perform the following functionality: shadow account selecting module 604 can select one of the identified shadow accounts; shadow account processing module 606 can lock the selected shadow account; and balance modifying module 608 can modify the balance on the selected shadow account by subtracting the amount from a current balance of the selected shadow account.

**[0053]** FIG. 7 illustrates an exemplary computer system that facilitates efficient processing of requests related to an account in a database, in accordance with an embodiment of the present application. Computer system 702 includes a processor 704, a memory 706, and a storage device 708. Memory 706 can include a volatile memory (e.g., RAM) that serves as a managed memory, and can be used to store one or more memory pools. Furthermore, computer system 702 can be coupled to a display device 710, a keyboard 712, and a pointing device 714. Storage device 708 can store an operating system 716, a content-processing system 718, and data 730.

**[0054]** Content-processing system 718 can include instructions, which when executed by computer system 702, can cause computer system 702 to perform methods and/or processes described in this disclosure. Specifically, content-processing system 718 may include instructions for sending and/or receiving data packets to/from other network nodes across a computer network, such as a content centric network, where a data packet can correspond to a request to transfer an amount in to or out of an account of a user. Content-processing system 718 can include instructions for selecting one of a plurality of shadow accounts that correspond to the account (shadow account selecting module 722). Content-processing system 718 can also include instructions for locking and unlocking the selected shadow account or the account (shadow account processing module 724). Content-processing system 718 can include instructions for modifying the balance of a selected shadow account by adding the amount to be transferred in to or by subtracting the amount to be transferred out from a current balance of the selected shadow account (balance modifying module 726). Content-processing system 718 can

further include instructions for generating a record that indicates the account, the selected shadow account, and the amount transferred in or out (record generating module 730).

**[0055]** Content-processing system 718 can additionally include instructions for, in response to determining that a balance of a selected shadow account is less than the amount to be transferred out (shadow account processing module 724), or in response to identifying no shadow accounts with a balance greater than or equal to the amount to be transferred out (shadow account processing module 724), setting a balance of the account to a combined balance of the shadow accounts, and setting the balance of each shadow account to zero (balance modifying module 726). Content-processing system 718 can include instructions for distributing a first amount equal to the balance of the account to the shadow accounts (balance distributing module 728).

**[0056]** Content-processing system 718 can also include instructions for transferring a designated amount from the account to a selected shadow account, where the designated amount is of a value greater than or equal to an amount specified in the request (balance modifying module 726). Content-processing system 718 can include instructions for, in response to determining that a balance of a selected shadow account is not less than the amount to be transferred out (shadow managing processing module 724), modifying the balance on the selected shadow account by subtracting the amount from a current balance of the selected shadow account (balance modifying module 726).

**[0057]** Content-processing system 718 can further include instructions for, in response to identifying one or more shadow accounts with a balance greater than or equal to the amount to be transferred out (shadow account processing module 724): selecting one of the identified shadow accounts (shadow account selecting module 722); locking the selected shadow account (shadow account processing module 724); and modifying the balance on the selected shadow account by subtracting the amount from a current balance of the selected shadow account (balance modifying module 728).

**[0058]** Data 732 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. Specifically, data 732 can store at least: a packet or message that corresponds to a request to transfer an amount in to or out of an account; an amount to transfer in or out; a designated amount which is greater than or equal to the amount; an account of a user; a plurality of shadow accounts that correspond to the account and are not visible to the user; a record of a request that indicates the account, a selected shadow account and/or the amount; a balance for the account; a balance for each shadow account; an account creation date; an account modification date; a shadow flag that indicates whether an account is a shadow account or not; a method for distributing funds from the account to the

shadow accounts, where the method can be based on one or more of a random distribution, an average distribution, or another predetermined rule for distribution;

**[0059]** 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.

**[0060]** 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.

**[0061]** 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.

**[0062]** The foregoing descriptions of embodiments of the present invention have been presented for purposes of illustration and description only. They are not intended to be exhaustive or to limit the present invention 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 present invention. The scope of the present invention is defined by the appended claims.

**What Is Claimed Is:**

1. A computer system for efficiently processing requests related to a database, the system comprising:

a processor; and

a memory coupled to the processor and storing instructions, which when executed by the processor cause the processor to perform a method, the method comprising:

creating, by a server, a plurality of shadow accounts that correspond to an account of a user;

distributing, by the server, from the account of the user, a first amount equal to a balance of the account to the plurality of shadow accounts;

receiving, by the server, a request to transfer a second amount in to the account of the user, wherein the account is visible to the user and corresponds to a record in a database;

selecting one of the plurality of shadow accounts that correspond to the account, wherein a shadow account is not visible to the user;

locking the selected shadow account, which prevents any changes to be made to the corresponding record in the database; and

modifying a balance of the selected shadow account by adding the second amount to be transferred in to a current balance of the selected shadow account.

2. The computer system of claim 1, wherein the database includes a mapping between the account and the plurality of corresponding shadow accounts.

3. The computer system of claim 1, wherein the method further comprises:

generating a record of the request that indicates the account, the selected shadow account, and the second amount.

4. The computer system of claim 1, wherein the method further comprises:

unlocking the selected shadow account.

5. A computer system for efficiently processing requests related to a database, the system comprising:

a processor; and

a memory coupled to the processor and storing instructions, which when executed by the processor cause the processor to perform a method, the method comprising:

creating, by a server, a plurality of shadow accounts that correspond to an account of a user;

distributing, by the server, from the account of the user, a first amount equal to a balance of the account to the plurality of shadow accounts;

receiving, by the server, a request to transfer a second amount out of the account of the user, wherein the account is visible to the user and corresponds to a record in a database;

selecting one of the plurality of shadow accounts that correspond to the account, wherein a shadow account is not visible to the user;

locking the selected shadow account, which prevents any changes to be made to the corresponding record in the database; and

in response to determining that a balance of the selected shadow account is less than the second amount to be transferred out:

setting a balance of the account to a combined balance of the plurality of shadow accounts;

setting a balance of each shadow account to zero;

executing the request on the selected shadow account or on the account; and

distributing, from the account of the user, a third amount equal to the balance of the account to the plurality of shadow accounts.

6. The computer system of claim 5, wherein the database includes a mapping between the account and the plurality of corresponding shadow accounts.

7. The computer system of claim 5, wherein executing the request on the selected shadow account or on the account further comprises:

unlocking the selected shadow account;

locking the account; and

modifying the balance on the account by subtracting the second amount from a current balance of the account.

8. The computer system of claim 5, wherein executing the request on the selected shadow account or on the account further comprises:

transferring a designated amount from the account to the selected shadow account, wherein the designated amount is of a value greater than or equal to the second amount to be transferred out; and

modifying the balance on the selected shadow account by subtracting the second amount from a current balance of the selected shadow account.

9. The computer system of claim 5, wherein the method further comprises:

generating a record of the request that indicates the account, the selected shadow account, and the second amount.

10. The computer system of claim 5, wherein in response to determining that a balance of the selected shadow account is not less than the second amount to be transferred

out, the method further comprises:

modifying the balance on the selected shadow account by subtracting the second amount from a current balance of the selected shadow account.

11. The computer system of claim 5, wherein distributing the third amount is based on one or more of:

a random distribution of the third amount from the account to the plurality of shadow accounts;

an equal distribution of the third amount from the account to the plurality of shadow accounts, wherein a fourth amount distributed into each of the shadow accounts is based on an average of the third amount divided by the number of shadow accounts; and

a predetermined rule for distributing the third amount from the account to the plurality of shadow accounts.

12. The computer system of claim 5, wherein the database includes one or more of:

an account number that corresponds to the account;

a balance that indicates a current balance of the account;

an account creation time that indicates when the account was created;

an account modification time that indicates when the account was last modified; and

a flag that indicates if the account is a shadow account.

13. A computer system for efficiently processing requests related to a database, the system comprising:

a processor; and

a memory coupled to the processor and storing instructions, which when executed by the processor cause the processor to perform a method, the method comprising:

creating, by a server, a plurality of shadow accounts that correspond to an

account of a user;

distributing, by the server, from the account of the user, a first amount equal to a balance of the account to the plurality of shadow accounts;

receiving, by the server, a request to transfer a second amount out of the account of the user, wherein the account is visible to the user and corresponds to a record in a database; and

in response to identifying none of the plurality of shadow accounts with a balance greater than or equal to the second amount to be transferred, wherein a shadow account is not visible to the user:

setting a balance of the account to a combined balance of the plurality of shadow accounts;

setting a balance of each shadow account to zero;

executing the request on a selected shadow account or on the account; and

distributing, from the account of the user, a third amount equal to the balance of the account to the plurality of shadow accounts.

14. The computer system of claim 13, wherein the database includes a mapping between the account and the plurality of corresponding shadow accounts.

15. The computer system of claim 13, wherein executing the request on a selected shadow account or on the account further comprises:

locking the account; and

modifying the balance on the account by subtracting the second amount from a current balance of the account.

16. The computer system of claim 13, wherein executing the request on a selected

shadow account or on the account further comprises:

selecting one of the plurality of shadow accounts that correspond to the account;

transferring a designated amount from the account to the selected shadow account, wherein the designated amount is of a value greater than or equal to the second amount to be transferred out; and

modifying the balance on the selected shadow account by subtracting the second amount from a current balance of the selected shadow account.

17. The computer system of claim 13, wherein the method further comprises:

generating a record of the request that indicates one or more of the account, the selected shadow account, and the second amount.

18. The computer system of claim 13, wherein in response to identifying one or more shadow accounts with a balance greater than or equal to the second amount to be transferred out, wherein a shadow account is not visible to the user:

selecting one of the identified shadow accounts;

locking the selected shadow account, which prevents any changes to be made to the corresponding record in the database; and

modifying the balance on the selected shadow account by subtracting the second amount from a current balance of the selected shadow account.

19. The computer system of claim 13, wherein distributing the third amount is based on one or more of:

a random distribution of the third amount from the account to the plurality of shadow accounts;

an equal distribution of the third amount from the account to the plurality of shadow accounts, wherein a fourth amount distributed into each of the shadow accounts is based on an

average of the third amount divided by the number of shadow accounts; and

a predetermined rule for distributing the third amount from the account to the plurality of shadow accounts.

20. The computer system of claim 13, wherein the database includes one or more of:

an account number that corresponds to the account;

a balance that indicates a current balance of the account;

an account creation time that indicates when the account was created;

an account modification time that indicates when the account was last modified; and

a flag that indicates if the account is a shadow account.

21. A computer-implemented method for efficiently processing requests related to a database, the method comprising:

creating, by a server, a plurality of shadow accounts that correspond to an account of a user;

distributing, by the server, from the account of the user, a first amount equal to a balance of the account to the plurality of shadow accounts;

receiving, by the server, a request to transfer a second amount in to the account of the user, wherein the account is visible to the user and corresponds to a record in a database;

selecting one of the plurality of shadow accounts that correspond to the account, wherein a shadow account is not visible to the user;

locking the selected shadow account, which prevents any changes to be made to the corresponding record in the database; and

modifying a balance of the selected shadow account by adding the second amount to be transferred in to a current balance of the selected shadow account.

22. A computer-implemented method for efficiently processing requests related to a database, the method comprising:

creating, by a server, a plurality of shadow accounts that correspond to an account of a user;

distributing, by the server, from the account of the user, a first amount equal to a balance of the account to the plurality of shadow accounts;

receiving, by the server, a request to transfer a second amount out of the account of the user, wherein the account is visible to the user and corresponds to a record in a database;

selecting one of the plurality of shadow accounts that correspond to the account, wherein a shadow account is not visible to the user;

locking the selected shadow account, which prevents any changes to be made to the corresponding record in the database; and

in response to determining that a balance of the selected shadow account is less than the second amount to be transferred out:

setting a balance of the account to a combined balance of the plurality of shadow accounts;

setting a balance of each shadow account to zero;

executing the request on the selected shadow account or on the account; and

distributing, from the account of the user, a third amount equal to the balance of the account to the plurality of shadow accounts.

23. The method of claim 22, wherein executing the request on the selected shadow account or on the account further comprises:

unlocking the selected shadow account;

locking the account; and

modifying the balance on the account by subtracting the second amount from a current balance of the account.

24. The method of claim 22, wherein executing the request on the selected shadow account or on the account further comprises:

transferring a designated amount from the account to the selected shadow account, wherein the designated amount is of a value greater than or equal to the second amount to be transferred out; and

modifying the balance on the selected shadow account by subtracting the second amount from a current balance of the selected shadow account.

25. A computer-implemented method for efficiently processing requests related to a database, the method comprising:

creating, by a server, a plurality of shadow accounts that correspond to an account of a user;

distributing, by the server, from the account of the user, a first amount equal to a balance of the account to the plurality of shadow accounts;

receiving, by the server, a request to transfer a second amount out of the account of the user, wherein the account is visible to the user and corresponds to a record in a database; and

in response to identifying none of the plurality of shadow accounts with a balance greater than or equal to the second amount to be transferred, wherein a shadow account is not visible to the user:

setting a balance of the account to a combined balance of the plurality of shadow accounts;

setting a balance of each shadow account to zero;

executing the request on a selected shadow account or on the account; and

distributing, from the account, a third amount equal to the balance of the account to the plurality of shadow accounts.

26. The method of claim 25, wherein executing the request on a selected shadow account or on the account further comprises:

locking the account; and

modifying the balance on the account by subtracting the second amount from a current balance of the account.

27. The method of claim 25, wherein executing the request on a selected shadow account or on the account further comprises:

selecting one of the plurality of shadow accounts that correspond to the account;

transferring a designated amount from the account to the selected shadow account, wherein the designated amount is of a value greater than or equal to the second amount to be transferred out; and

modifying the balance on the selected shadow account by subtracting the second amount from a current balance of the selected shadow account.

28. The method of claim 25, wherein in response to identifying one or more shadow accounts with a balance greater than or equal to the second amount to be transferred out, wherein a shadow account is not visible to the user:

selecting one of the identified shadow accounts;

locking the selected shadow account, which prevents any changes to be made to the corresponding record in the database; and

modifying the balance on the selected shadow account by subtracting the second amount from a current balance of the selected shadow account.