



(19) 中華民國智慧財產局

(12) 發明說明書公告本

(11) 證書號數：TW I501171 B

(45) 公告日：中華民國 104 (2015) 年 09 月 21 日

(21) 申請案號：099106637

(22) 申請日：中華民國 99 (2010) 年 03 月 08 日

(51) Int. Cl. : G06Q30/02 (2012.01)

(71) 申請人：阿里巴巴集團控股有限公司 (開曼群島) ALIBABA GROUP HOLDING LIMITED
(KY)

香港

(72) 發明人：于新林 (CN)；樓方鑫 (CN)

(74) 代理人：林志剛

(56) 參考文獻：

TW 200741574A

US 5940813A

US 2003/0101136A1

US 2005/0267825A1

US 2007/0156556A1

US 2007/0192254A1

審查人員：曾耀德

申請專利範圍項數：10 項 圖式數：4 共 29 頁

(54) 名稱

帳戶開發處理方法及帳戶開發處理系統

(57) 摘要

一種帳戶開發處理方法，用於帳戶處理中心處理帳戶操作請求，包括：(1)將帳戶分成高開發帳戶和普通帳戶；(2)當接收到某一帳戶處理操作請求時，先判斷該帳戶是否是高開發帳戶，若是高開發帳戶則進行步驟(3)，否則進行步驟(4)；(3)若帳戶被鎖定，則將當前操作進入等待進程佇列等待，直到執行當前操作；(4)進一步判斷帳戶有沒有被鎖定，若帳戶已被鎖定，則直接報錯後返回，否則執行當前操作。通過這種方式可以對高開發帳戶提供更優質的服務，而對於普通帳戶，一旦某天普通帳戶上發生了大量的開發，通過報錯可以有效的控制開發量，防止這類帳戶對系統造成危害。

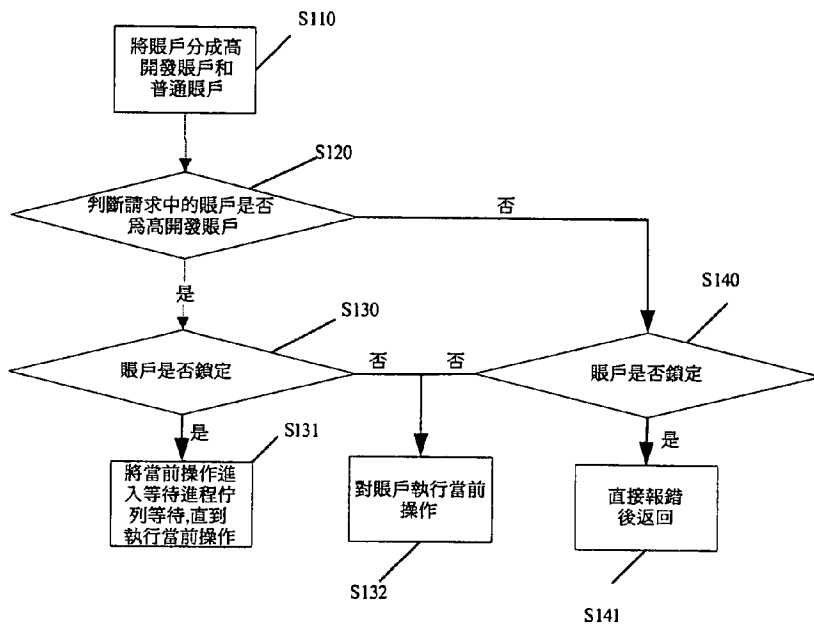


圖3

861443

發明專利說明書

(本申請書格式、順序，請勿任意更動，※記號部分請勿填寫)

※申請案號：99106637

※申請日：99年03月08日

※IPC分類：G06Q 30/02 (2012.01)

一、發明名稱：(中文／英文)

帳戶開發處理方法及帳戶開發處理系統

二、中文發明摘要：

一種帳戶開發處理方法，用於帳戶處理中心處理帳戶操作請求，包括：(1)將帳戶分成高開發帳戶和普通帳戶；(2)當接收到某一帳戶處理操作請求時，先判斷該帳戶是否是高開發帳戶，若是高開發帳戶則進行步驟(3)，否則進行步驟(4)；(3)若帳戶被鎖定，則將當前操作進入等待進程佇列等待，直到執行當前操作；(4)進一步判斷帳戶有沒有被鎖定，若帳戶已被鎖定，則直接報錯後返回，否則執行當前操作。通過這種方式可以對高開發帳戶提供更優質的服務，而對於普通帳戶，一旦某天普通帳戶上發生了大量的開發，通過報錯可以有效的控制開發量，防止這類帳戶對系統造成危害。

三、英文發明摘要：

四、指定代表圖：

- (一) 本案指定代表圖為：第(3)圖。
- (二) 本代表圖之元件符號簡單說明：無

五、本案若有化學式時，請揭示最能顯示發明特徵的化學式：無

六、發明說明：

【發明所屬之技術領域】

本發明係關於網路領域，尤其關於帳戶開發處理方法及帳戶開發處理系統。

【先前技術】

請參閱圖 1，其為現有帳戶處理系統的原理示意圖。該帳戶處理系統包括帳戶處理中心 11 和若干用戶端 12。帳戶處理中心 11 可以通過專線連接用戶端 12，也可以通過網路（網際網路、內部網）等連接至用戶端 12。用戶端 12 可以是多個。用戶端 12 也可以是獨立終端，也可以為若干終端組成的區域網路。帳戶處理中心 11 至少包括伺服器 21 和資料庫 22。資料庫 22 用於儲存帳戶資訊及各種處理資訊。伺服器 21 用於對帳戶進行各種操作和管理。

以帳戶處理中心 11 為第三方支付平臺為例，當用戶端 12 向帳戶處理中心 11 發送指令，該指令是指將 A 帳戶中的資金向若干其他帳戶（以帳戶 B1、B2、B3...Bn）支付金額分別為 S1、S2、S3...Sn，現有的支付流程為：

首先，帳戶處理中心 11 接收到用戶端 12 發送的支付操作；

接著，帳戶處理中心 11 先處理一次支付操作：

伺服器 21 鎖住 A 帳戶，判斷 A 帳戶中的金額是否超過 S1 金額，若帳戶中的金額大於等於 S1 金額，則將 A

帳戶的餘額扣除 S1 金額後，釋放 A 帳戶的鎖，再鎖住 B1 帳戶，在 B1 帳戶中增加 S1 金額，操作完畢後釋放 B1 帳戶的鎖。

如此，帳戶處理中心 11 需要進行 n 次上述支付操作。

也就是說，爲了避免帳戶開發處理而有可能影響帳戶內金額，需要保證在同一時刻只允許對帳戶進行一次操作，通常是通過對帳戶進行加鎖來實現的。即，每一個對帳戶的操作，先判斷能否獲得這個帳戶的鎖，若能，則說明對同一帳戶沒有其他操作正在處理，當前能對該帳戶進行本操作。若不能，說明對同一帳戶有其他操作正在處理，需要等待。當等待的操作有多個時，就形成了等待進程佇列。這個等待進程佇列中每一個操作都需要對帳戶進行加鎖和釋放鎖的操作，並且伺服器只能串列處理該些操作，佔用比較多的資源。特別是，當這樣的帳戶（即該帳戶有等待進程佇列）有多個或者帳戶的等待進程佇列中需要處理的操作個數龐大時，就造成整個系統的處理效率低下，而且很容易造成系統的癱瘓。假設資料庫的最大連接數是 100 個。也就是說，資料庫能承受最大開發訪問資料量爲 100 個。當等待進程佇列中有 100 個操作時，整個系統就不能接受新的帳戶操作請求，由此造成後續新帳戶操作的效率低，甚至容易造成整個系統癱瘓。

然而，由於存在一些突發事件而導致某個或某些帳戶的操作異常，比較常見的操作異常爲對帳戶的操作暴增，使得帳戶的等待進程佇列中需要處理的操作個數暴增，由

此造成整個系統的處理效率降低。

這些突發事件有些是由於系統的不穩定或技術原因造成的，有些是由於人爲造成的，比如，網上交易過程中商戶都有本商戶的信用等級，信用等級是購買者或其他用戶對商戶服務及其出售商品的滿意度評價的體現，網上交易過程中無法真實感受到商品，因此信用等級對商戶來說非常重要。以得到更高信用等級爲直接目的，而進行的一些虛假交易，我們稱之爲信用炒作。當一些商戶組成信用炒作團體時，這些商戶相關的帳戶就可能出現頻繁的操作，由於出現某時刻發生在這些帳戶上的操作特別大，使得系統需要消耗大量的資源來處理該些操作，由此也使其他帳戶的正常操作受到影響。

也就是說，現有的帳戶處理中心的帳戶處理操作存在以下的技術缺陷：

現有的支付流程需要對 A 帳戶進行頻繁操作，每一次操作都需要鎖住 A 帳戶，並在處理完畢後釋放 A 帳戶。不僅造成伺服器 21 的處理效率低，更爲重要的是造成伺服器 21 的處理能力急劇下降，甚至不能接收其他用戶端發出的處理請求，嚴重的造成整個伺服器 21 的癱瘓。

特別是，當接收到這類對帳戶進行批次處理請求個數較多，或者這類對帳戶進行批次處理請求中需要處理帳戶特別多時，對整個伺服器 21 造成的運行壓力非常大，容易造成伺服器 21 請求處理的佇列堵塞，容易造成請求處理的失敗率增加。同時，也影響其他用戶的體驗。

【發明內容】

本發明的第一目的在於提供一種帳戶開發處理方法，以解決現有技術中由於突發事件引起同一帳戶上平行作業增加而導致整個系統的處理效率低下的技術問題。

本發明的第二目的在於提供一種帳戶開發處理系統，以解決現有技術中由於突發事件引起同一帳戶上平行作業增加而導致整個系統的處理效率低下的技術問題。

一種帳戶開發處理方法，用於帳戶處理中心處理帳戶操作，包括：

(1) 將帳戶分成高開發帳戶和普通帳戶；

(2) 當接收到某一帳戶處理操作請求時，先判斷該帳戶是否是高開發帳戶，若是高開發帳戶則進行步驟(3)，否則進行步驟(4)；

(3) 若帳戶被鎖定，則將當前操作進入等待進程佇列等待，直到執行當前操作；

(4) 進一步判斷帳戶有沒有被鎖定，若帳戶已被鎖定，則直接報錯後返回，否則執行當前操作。

步驟(4)還包括，當帳戶已被鎖定時，延時預先設定的時間後再次判斷當前操作是否被執行，若否，則直接報錯後返回。

較優地，步驟(1)進一步包括：A1：根據高開發帳戶所占總帳戶數的百分比來設定高開發帳戶的數量Q；A2：統計預先設定的週期內各帳戶對應的本帳戶進行操作的運算元，並按照運算元依次進行排序；A3：取運算元高

的前 Q 位帳戶做爲高開發帳戶。並且，步驟（1）最好還包括，將所有高開發帳戶的帳戶號保存在記憶體中。

步驟（2）還包括通過將帳戶號和預先保存在記憶體中的所有高開發帳戶的帳戶號進行對比，如有相同，則該帳戶爲高開發帳戶，否則，該帳戶爲普通帳戶。

執行當前操作進一步包括：鎖住帳戶；執行對帳戶的操作；執行完畢後，釋放該帳戶對應的鎖。

一種帳戶開發處理系統，包括帳戶處理中心，該帳戶處理中心進一步包括：

高開發帳戶庫：用於儲存所有高開發帳戶資訊；

帳戶性質判斷單元：用於接收到某一帳戶處理操作請求時，根據高開發帳戶庫去判斷該帳戶是否是高開發帳戶；

高開發帳戶處理單元：用於進一步判斷帳戶有沒有被鎖定，若帳戶被鎖定，則將當前操作進入等待進程佇列等待，直到執行當前操作，否則直接執行當前操作；

普通帳戶處理單元：用於進一步判斷帳戶有沒有被鎖定，若帳戶已被鎖定，則直接報錯後返回，否則執行當前操作。

所述高開發帳戶處理單元進一步包括：

高開發帳戶鎖定判斷子單元，用於判斷該高開發帳戶當前是否被鎖；

執行操作子單元，用於鎖住帳戶，再執行對帳戶的操作，後釋放該帳戶對應的鎖。

所述普通帳戶處理單元進一步包括：

普通帳戶鎖定判斷子單元，用於判斷該帳戶當前是否被鎖；

執行操作子單元，用於鎖住帳戶，再執行對帳戶的操作，後釋放該帳戶對應的鎖；

鎖定處理子單元，用於該帳戶被鎖後接收到對該帳戶的操作，採用直接報錯後返回的處理。

一種帳戶開發處理方法，用於帳戶處理中心處理用戶端發送的帳戶操作請求，包括：

(1) 帳戶處理中心將帳戶分成若干種類型帳戶，每一種類型設置對應的帳戶被鎖定後的處理操作；

(2) 當接收到用戶端發送的某一帳戶處理操作請求時，先確定該帳戶所屬的類型；

(3) 帳戶處理中心進一步判斷帳戶有沒有被鎖定，若帳戶被鎖定，就按照該帳戶所屬類型對應的帳戶被鎖定後的處理操作流程進行操作，否則執行當前操作。

步驟(1)中每一種類型設置對應的帳戶被鎖定後的處理操作進一步包括：

A1：帳戶被鎖定後，判斷 M 的次數是否為零，若是，則直接報錯後返回，若否，過等待時間 T 後判斷帳戶是否被鎖定，如果被鎖定，則 M 的次數減一，再進行步驟 A1，如果未被鎖定，則執行當前操作，

其中，M 為每一種類型設定的對應的等待次數，M 的範圍可以為 0 至無限大。

與現有技術相比，本發明具有以下優點：

對於一個普通的帳戶，在帳戶上發生的業務是有規律的，在一段時間內有個固定的模式。以某個商務平臺為例，一個普通的買家一天的交易量是有限的，每天不會發生太多的交易，也就是這個帳戶上的交易開發性不是很大，如果某天在這個帳戶上發生了大量的交易，很可能就是一些非法業務。而對於大的賣家，則在其上就可能有大量的開發，會有很多買家買這個賣家的商品，從而產生大的開發交易。根據以上特點，發明人把帳戶分為兩類，高開發帳戶和普通帳戶。對於高開發帳戶，當帳戶餘額發生變化，在鎖定帳戶時如果發現帳戶已經被鎖，那麼一直等待而不報錯返回。而對於普通帳戶發生餘額變動，在鎖定帳戶時如果發現帳戶被鎖，直接報錯返回，通過這種方式我們可以對高開發帳戶提供更優質的服務，而對於普通客戶，這樣一旦某天普通帳戶上發生了大量的開發，通過報錯可以有有效的控制開發量，防止這類帳戶對系統造成危害。

另外，本發明還可以對普通帳戶進行操作請求時，若該帳戶已被鎖定时，延時預先設定的時間後再次判斷當前操作是否被執行，若否，則直接報錯後返回。這種延時處理，能夠降低普通帳戶可能存在的開發處理的報錯率。

【實施方式】

以下結合附圖，具體說明本發明。

請參閱圖 1 和圖 2，其為本發明的一種帳戶開發處理

系統的原理結構示意圖。它包括帳戶處理中心 11 和用戶端 12。用戶端 12 提出對帳戶處理中心 11 上的某個帳戶或某些帳戶的操作請求，並接收帳戶處理中心 11 返回的處理結果。

該帳戶處理中心 11 進一步包括伺服器 31 和資料庫 32。資料庫 32 用於儲存帳戶資訊及各種處理資訊。伺服器 31 用於對帳戶進行各種操作和管理。在本發明中，資料庫 32 除了包括用於儲存帳戶資訊的帳戶儲存單元 321、用來儲存帳戶處理操作記錄資訊的帳戶操作記錄儲存單元 322 之外，還包括：

高開發帳戶庫 323：用於儲存所有高開發帳戶資訊。高開發帳戶資訊主要是指能唯一識別高開發帳戶的帳戶號。本發明可以按照各種規則來設定本中心的高開發帳戶，比如，可以將重點帳戶設定為高開發帳戶。也可以將帳戶一定時間段內進行操作的運算元從高到低排序，取運算元高的若干帳戶作為高開發帳戶。

伺服器 31 進一步包括帳戶性質判斷單元 311、高開發帳戶處理單元 312 和普通帳戶處理單元 313。

帳戶性質判斷單元 311：用於接收到某一帳戶處理操作請求時，根據高開發帳戶庫去判斷該帳戶是否是高開發帳戶，如果是，就觸發高開發帳戶處理單元 312，如果不是，則觸發普通帳戶處理單元 313。

高開發帳戶處理單元 312：用於進一步判斷帳戶有沒有被鎖定，若帳戶被鎖定，則將當前操作進入等待進程佇

列等待，直到執行當前操作，否則直接執行當前操作。

高開發帳戶處理單元 312 進一步包括：

高開發帳戶鎖定判斷子單元 3121，用於判斷該高開發帳戶當前是否被鎖；

等待進程佇列管理子單元 3122，用於管理等待進程佇列的操作，管理等待進程佇列包括對這些操作的進程如何進行處理的，比如按照先來先處理的原則，或者採用優先順序處理的原則等。有些系統自帶有這個管理子單元 3122，此時發明人就無需自設定該管理子單元。

執行操作子單元 3123，用於鎖住帳戶，再執行對帳戶的操作，後釋放該帳戶對應的鎖。

普通帳戶處理單元 313：用於進一步判斷帳戶有沒有被鎖定，若帳戶已被鎖定，則直接報錯後返回，否則執行當前操作。普通帳戶處理單元 313 進一步包括：

普通帳戶鎖定判斷子單元 3131，用於判斷該帳戶當前是否被鎖；

執行操作子單元 3132，用於鎖住帳戶，再執行對帳戶的操作，後釋放該帳戶對應的鎖；

鎖定處理子單元 3133，用於該帳戶被鎖後接收到對該帳戶的操作，採用直接報錯後返回的處理。

普通帳戶處理單元 313 還包括延時子單元 3134，用於當帳戶已被鎖定时，延時預先設定的時間後再次判斷當前操作是否被執行，若否，則直接報錯後返回。

上述公開的是單元僅僅是指邏輯上的單元，通常是用

軟體來實現的，但本發明並不排除用硬體來實現。

請參閱圖 3，其為一種帳戶開發處理方法的流程圖。它用於帳戶處理中心處理帳戶操作，包括：

S110：將帳戶分成高開發帳戶和普通帳戶。在本實例中，現有的帳戶處理中心 11 已存在非常多的帳戶，由此額外設置一高開發帳戶庫 323，用以保存高開發帳戶資訊。判斷某一帳戶是高開發帳戶還是普通帳戶，只需要查找高開發帳戶庫 323 即可獲知。這種方式，無需要對現有的帳戶資訊進行修改，減少對原有系統穩定性的影響。還有一種方式是，對帳戶設定一性質屬性項，每一帳戶在性質屬性項中保存本帳戶的屬性：高開發帳戶還是普通帳戶。這種方式也是可以的，但是需要對現有帳戶進行修改，比較費時間。

將帳戶分為高開發帳戶和普通帳戶，可以通過以下步驟完成：

A1：根據高開發帳戶所占總帳戶數的百分比來設定高開發帳戶的數量 Q ；

A2：統計預先設定的週期內（如一周內）各帳戶進行操作的運算元，並按照運算元依次進行排序；

A3：取運算元高的前 Q 位帳戶做為高開發帳戶，並將高開發帳戶資訊保存在高開發帳戶庫 323 中。

並且，定週期根據運算元來更新高開發帳戶庫 323 中的高開發帳戶資訊。通過這種方式，能將本來運算元比較多的帳戶找出來。

爲了提高後續處理的效率，可以將高開發帳戶庫 323 中的所有高開發帳戶的帳戶號保存在伺服器 31 的記憶體中。

S120：當接收到某一帳戶處理操作請求時，先判斷該帳戶是否是高開發帳戶，若是高開發帳戶則進行步驟 S130，否則進行步驟 S140。

本發明是通過以下步驟判斷該帳戶是否是高開發帳戶的：

將帳戶號和預先保存在記憶體或高開發帳戶庫 323 中的所有高開發帳戶的帳戶號進行對比，如相同，則該帳戶爲高開發帳戶，否則，該帳戶爲普通帳戶。

S130：判斷帳戶是否被鎖定，若是，則將當前操作進入等待進程佇列等待，直到執行當前操作（S131），若否，則對帳戶執行當前操作（S132）。

S140：進一步判斷帳戶有沒有被鎖定，若帳戶已被鎖定，則直接報錯後返回（S141），否則執行當前操作（S132）。

S130 的步驟可以通過 oracle 語句來實現 “
`select * from account where accountNo='xxxxx' for update`
”

該語句表示：。對 accountNo 爲 xxxxx 的帳戶進行操作，該命令隱含 “`select * from account where accountNo='xxxxx' for update wait`”

即，若該帳戶被鎖後，一直等待，直到能執行該操作

S140 的步驟也可以通過 JAVA 語句來實現。

```
select * from account where accountNo='yyyyy' for u
pdate nowait
```

這條語句表示：如果對帳戶進行操作時發現該帳戶已經被鎖，則報錯。上述的僅是一種實現的範例，並非局限於此，任何實現本思想的實現方式，都應在本發明的保護範圍內。

執行當前操作進一步包括：鎖住帳戶；執行對帳戶的操作；執行完畢後，釋放該帳戶對應的鎖。

請參閱圖 4，其為另一種帳戶開發處理方法的流程圖。

本實例中的 S240 的實施步驟與圖 3 的 S140 的實施步驟不同，其他步驟與圖 3 中的步驟大致相同。

S110：將帳戶分成高開發帳戶和普通帳戶。

S120：當接收到某一帳戶處理操作請求時，先判斷該帳戶是否是高開發帳戶，若是高開發帳戶則進行步驟 S130，否則進行步驟 S240。

S130：判斷帳戶是否被鎖定，若是，則將當前操作進入等待進程佇列等待，直到執行當前操作（S131），若否，則對帳戶執行當前操作（S132）。

S240：進一步判斷帳戶有沒有被鎖定，若帳戶已被鎖定，則等待預先設定的時間（S241），進一步判斷當前操作是否被執行（S242），若否，直接報錯後返回（S243）

，若是，則執行當前操作（S132）。當帳戶未被鎖定時，則也是執行當前操作（S132）。

對帳戶進行鎖住有很多種實現方式，比如，一個常見的方式是，對帳戶設置一個標誌位元，標誌位元為 1，表示這個帳戶是鎖住的，標誌位元為 0，表示這個帳戶未被鎖住。檢測該帳戶是否被鎖住，就是檢測該標誌位元。

本實例中，在接收到對普通帳戶的操作請求時，發現當前該帳戶有正在處理的操作，延時一預設時間再去判斷能否進行處理。能夠降低普通帳戶可能存在的開發處理的報錯率。

本發明實施例將帳戶劃分為高開發帳戶和普通帳戶兩種類型進行說明，當然在具體實施過程中，可以不受本發明實施例的限制將帳戶劃分多種類型，例如：第一類型帳戶、第二類型帳戶、第三類型帳戶……第 N 類型帳戶。具體可以通過以下步驟完成：

A1：根據第一類型帳戶所占總帳戶數的百分比來設定第一類型帳戶的數量 Q_1 ，根據第二類型帳戶所占總帳戶數的百分比來設定第二類型帳戶的數量 Q_2 ，以此類推，可設定第 N 類型帳戶的數量 Q_n ；

A2：統計預先設定的週期內（如一周內）各帳戶進行操作的運算元，並按照運算元由低到高依次進行排序；

A3：取運算元最低的前 Q_1 位帳戶做為第一類型帳戶，取接下來的 Q_2 位帳戶做為第二類型帳戶，以此類推可獲得各種類型的帳戶資訊，並將各種類型的帳戶資訊保存

在資料庫中。

對帳戶劃分成多種類型，還可以通過以下步驟來實現：

B1：預先設定第一類型帳戶、第二類型帳戶、...第 N 類型帳戶各自在總帳戶數中所占的比例；

B2：統計預先設定的週期內（如一周內）各帳戶進行操作的運算元，並按照運算元由低到高依次進行排序成帳戶佇列；

B3：統計當前的總帳戶數 M；

B4：按照各類型帳戶所占的比例和總帳戶數 M，計算出各類型帳戶所占的數量；

B5：每一類型帳戶依次從帳戶佇列中取對應數量的帳戶作為本類型帳戶中的帳戶，並將該些帳戶資訊保存至對應的資料庫中。

本實施例的處理思想為：將帳戶劃分為第一類型帳戶、第二類型帳戶、第三類型帳戶...第 N 類型帳戶，針對不同類型的帳戶設定不同的處理策略。比如，將不同的類型帳戶分成不同的優先順序，優先順序越高，當帳戶被鎖定時，則等待預先設定的時間也就越長。

舉個實例來說，所述處理策略可以為：對於第一類型的帳戶，若帳戶被鎖定時，則直接報錯返回；對於第二類型的帳戶，當帳戶被鎖定時，則等待預先設定的時間 t，判斷該帳戶是否仍被鎖定，若仍被鎖定，則直接報錯返回；對於第三類型的帳戶，當帳戶被鎖定時，則等待預先設

定的時間 t 進行判斷後，發現該帳戶仍被鎖定時，則再等待一個預先設定的時間 t ，若仍被鎖定，則報錯返回，以此類推可針對不同類型的帳戶設定不同的處理策略。

通過上述對帳戶分成若干種類型，為每一種類型的帳戶設定不同的處理策略，即，

一種帳戶開發處理方法，用於帳戶處理中心處理用戶端發送的帳戶操作請求，包括：

(1) 帳戶處理中心將帳戶分成若干種類型帳戶，每一種類型設置對應的帳戶被鎖定後的處理操作；

(2) 當接收到用戶端發送的某一帳戶處理操作請求時，先確定該帳戶所屬的類型；

(3) 帳戶處理中心進一步判斷帳戶有沒有被鎖定，若帳戶被鎖定，就按照該帳戶所屬類型對應的帳戶被鎖定後的處理操作流程進行操作，否則執行當前操作。

步驟 (1) 中每一種類型設置對應的帳戶被鎖定後的處理操作進一步包括：

K1：帳戶被鎖定後，判斷 M 的次數是否為零，若是，則直接報錯後返回，若否，過等待時間 T 後判斷帳戶是否被鎖定，如果被鎖定，則 M 的次數減一，再進行步驟 K1，如果未被鎖定，則執行當前操作，

其中， M 為每一種類型設定的對應的等待次數， M 的範圍可以為 0 至無限大。

以上公開的僅為本發明的幾個具體實施例，但本發明並非局限於此，任何本領域的技術人員能思之的變化，都

應落在本發明的保護範圍內。

【圖式簡單說明】

圖 1 為現有帳戶處理系統的原理示意圖；

圖 2 為本發明的一種帳戶處理中心的原理結構示意圖

；

圖 3 為帳戶處理方法的一種流程示意圖；

圖 4 為帳戶處理方法的另一種流程示意圖。

【主要元件符號說明】

11：帳戶處理中心

12：用戶端

21：伺服器

22：資料庫

31：伺服器

32：資料庫

311：帳戶性質判斷單元

312：高開發帳戶處理單元

313：普通帳戶處理單元

321：帳戶儲存單元

322：帳戶操作記錄儲存單元

323：高開發帳戶庫

3121：高開發帳戶鎖定判斷子單元

3122：等待進程佇列管理子單元

3123：執行操作子單元

3131：普通帳戶鎖定判斷子單元

3132：執行操作子單元

3133：鎖定處理子單元

3134：延時子單元

七、申請專利範圍：

1. 一種帳戶開發處理方法，其特徵在於，該方法包括：

伺服器對帳戶進行各種操作和管理；

將預先設定的週期內各帳戶進行操作的運算元依次進行排序，將帳戶分成高開發帳戶和普通帳戶；

將該等高開發帳戶的帳戶號保存在記憶體中；

當接收到用戶端發送的某一帳戶處理操作請求時，先判斷該帳戶是否是高開發帳戶，該判斷步驟包括：

透過將帳戶號和預先保存在記憶體中的所有高開發帳戶的帳戶號進行對比；以及

如有相同，則該帳戶為高開發帳戶，否則，該帳戶為普通帳戶；

若是高開發帳戶則進行步驟（A），否則進行步驟（B）；

（A）若帳戶被鎖定，帳戶處理中心將當前操作進入等待進程佇列等待，直到執行當前操作；

（B）帳戶處理中心進一步判斷帳戶有沒有被鎖定，若帳戶已被鎖定，則直接報錯後返回，否則執行當前操作，該執行當前操作步驟包括：

鎖住帳戶；以及

執行對帳戶的操作。

2. 如申請專利範圍第 1 項之方法，其中，步驟（B）還包括，

當帳戶已被鎖定時，延時預先設定的時間後再次判斷當前操作是否被執行，若否，則直接報錯後返回。

3.如申請專利範圍第 1 項之方法，進一步包括：

根據高開發帳戶所占總帳戶數的百分比來設定高開發帳戶的數量 Q ；以及

取運算元高的前 Q 位帳戶做為高開發帳戶。

4.如申請專利範圍第 1 項之方法，其中，執行當前操作進一步包括：

執行完畢後，釋放該帳戶對應的鎖。

5.一種帳戶開發處理系統，包括：

帳戶處理中心，包括：

高開發帳戶庫：用於儲存所有按照各種規則來設定的高開發帳戶資訊；

帳戶性質判斷單元：用於接收到某一帳戶處理操作請求時，根據高開發帳戶庫去判斷該帳戶是否是高開發帳戶，該判斷步驟包括：

透過將帳戶號和預先保存在記憶體中的所有高開發帳戶的帳戶號進行對比；以及

如有相同，則該帳戶為高開發帳戶，否則，該帳戶為普通帳戶；

高開發帳戶處理單元：用於進一步判斷帳戶有沒有被鎖定，若帳戶被鎖定，則將當前操作進入等待進程佇列等待，直到執行當前操作，否則直接執行當前操作；

普通帳戶處理單元：用於進一步判斷帳戶有沒有被鎖

定，若帳戶已被鎖定，則直接報錯後返回，否則執行當前操作，該執行當前操作步驟包括：

鎖住帳戶；以及

執行對帳戶的操作。

6.如申請專利範圍第 5 項之系統，其中，該高開發帳戶處理單元進一步包括：

高開發帳戶鎖定判斷子單元，用於判斷該高開發帳戶當前是否被鎖；

執行操作子單元，用於鎖住帳戶，再執行對帳戶的操作，後釋放該帳戶對應的鎖。

7.如申請專利範圍第 5 項之系統，其中，該普通帳戶處理單元進一步包括：

普通帳戶鎖定判斷子單元，用於判斷該帳戶當前是否被鎖；

執行操作子單元，用於鎖住帳戶，再執行對帳戶的操作，後釋放該帳戶對應的鎖；

鎖定處理子單元，用於該帳戶被鎖後接收到對該帳戶的操作，採用直接報錯後返回的處理。

8.如申請專利範圍第 7 項之系統，其中，還包括延時子單元，用於當帳戶已被鎖定时，延時預先設定的時間後再次判斷當前操作是否被執行，若否，則直接報錯後返回。

9.一種帳戶開發處理方法，其特徵在於，該方法包括：

伺服器對帳戶進行各種操作和管理；

將帳戶分成若干種不同類型帳戶，該等不同類型帳戶包含高開發帳戶；

將該等高開發帳戶的帳戶號保存在記憶體中；

每一種類型設置對應的帳戶被鎖定後的處理操作；

當接收到用戶端發送的某一帳戶處理操作請求時，先確定該帳戶所屬的類型，該判斷步驟包括：

透過將帳戶號和預先保存在記憶體中的所有高開發帳戶的帳戶號進行對比；以及

如有相同，則該帳戶為高開發帳戶，否則，該帳戶為普通帳戶；

帳戶處理中心進一步判斷帳戶有沒有被鎖定，若帳戶被鎖定，就按照該帳戶所屬類型對應的帳戶被鎖定後的處理操作流程進行操作，否則執行當前操作，該執行當前操作步驟包括：

鎖住帳戶；以及

執行對帳戶的操作。

10.如申請專利範圍第 9 項之方法，其中，將帳戶分成若干種不同類型帳戶的步驟中每一種類型設置對應的帳戶被鎖定後的處理操作進一步包括：

A1：帳戶被鎖定後，判斷 M 的次數是否為零，若是，則直接報錯後返回，若否，過等待時間 T 後判斷帳戶是否被鎖定，如果被鎖定，則 M 的次數減一，再進行步驟 A1，如果未被鎖定，則執行該當前操作步驟，

其中， M 為每一種類型設定的對應的等待次數， M 的範圍可以為 0 至無限大。

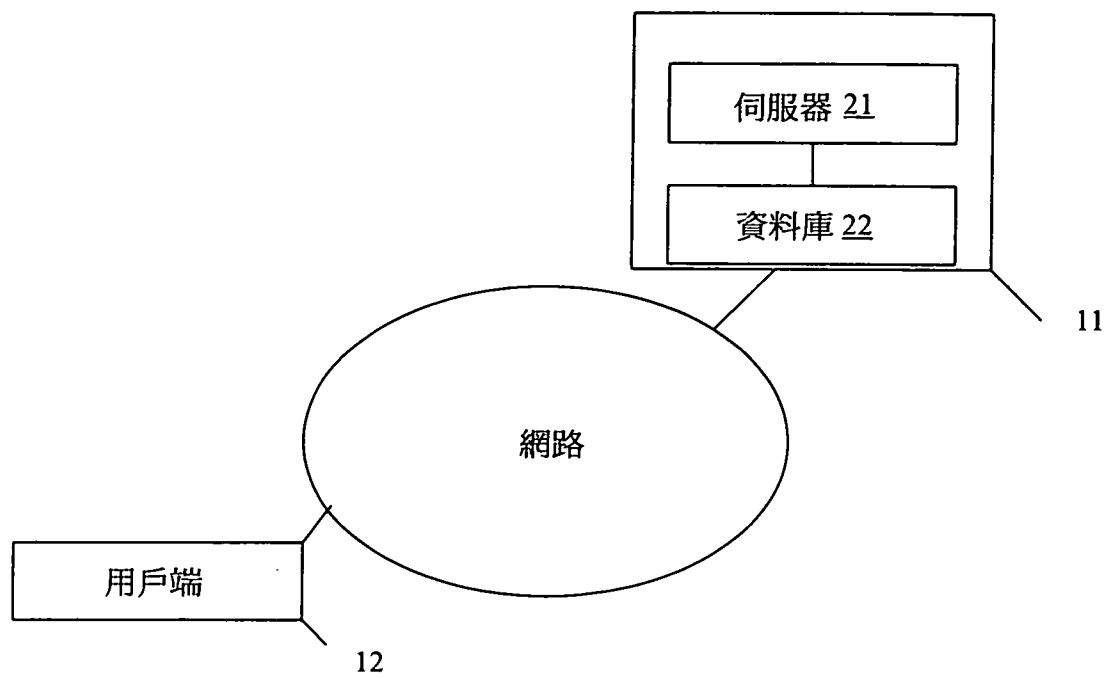


圖 1

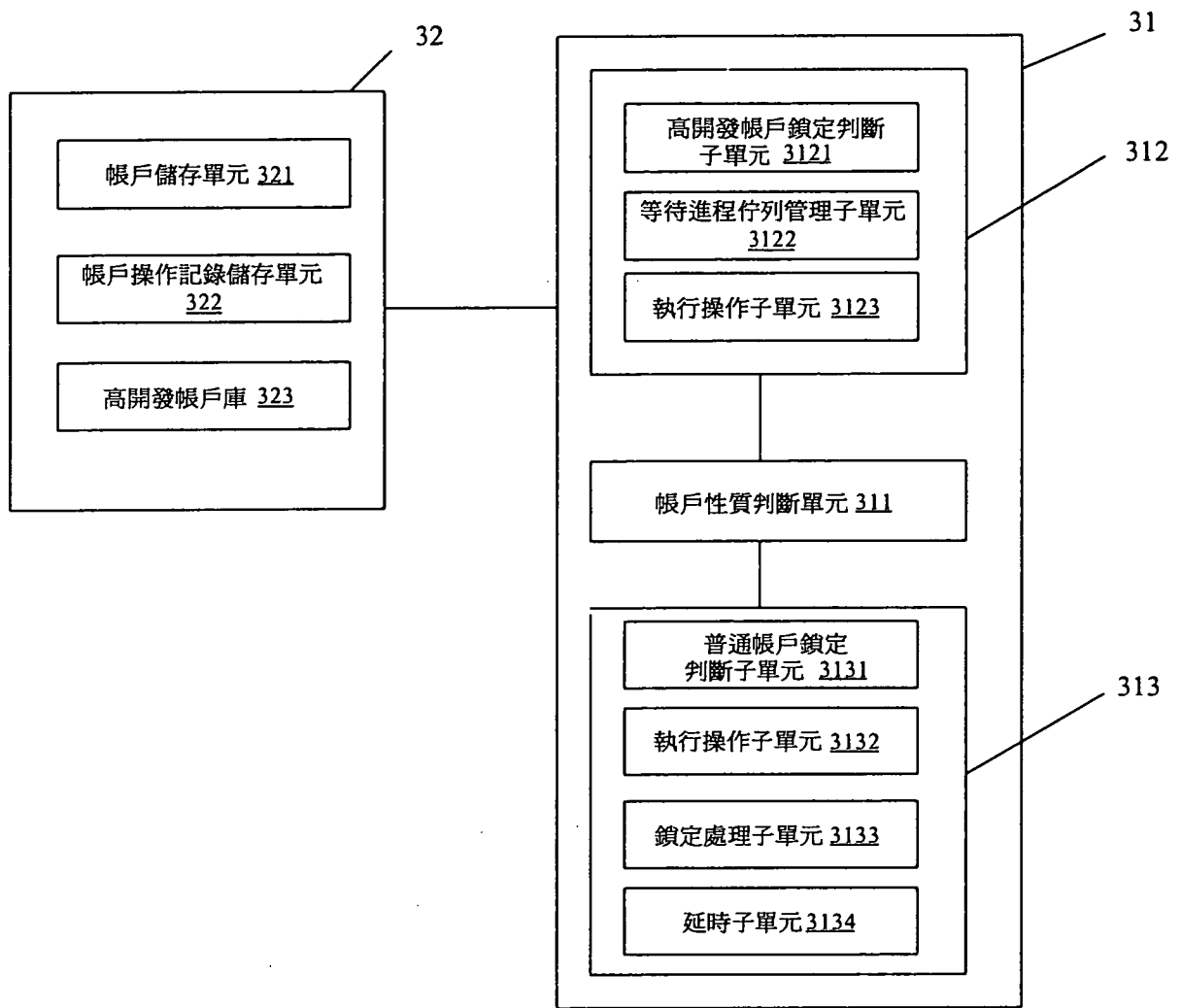


圖2

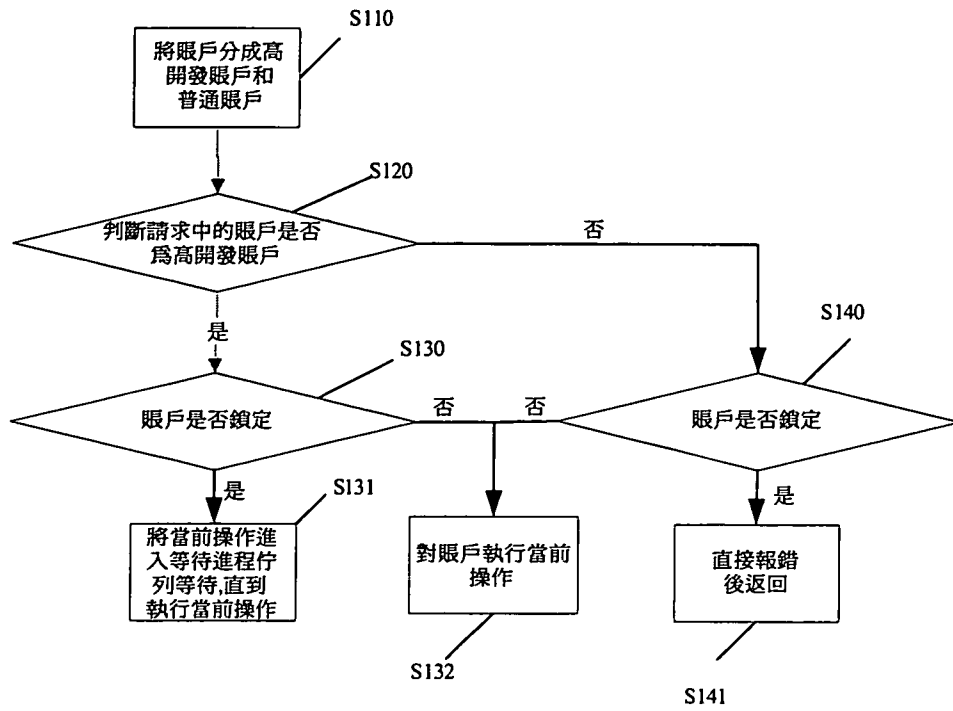


圖3

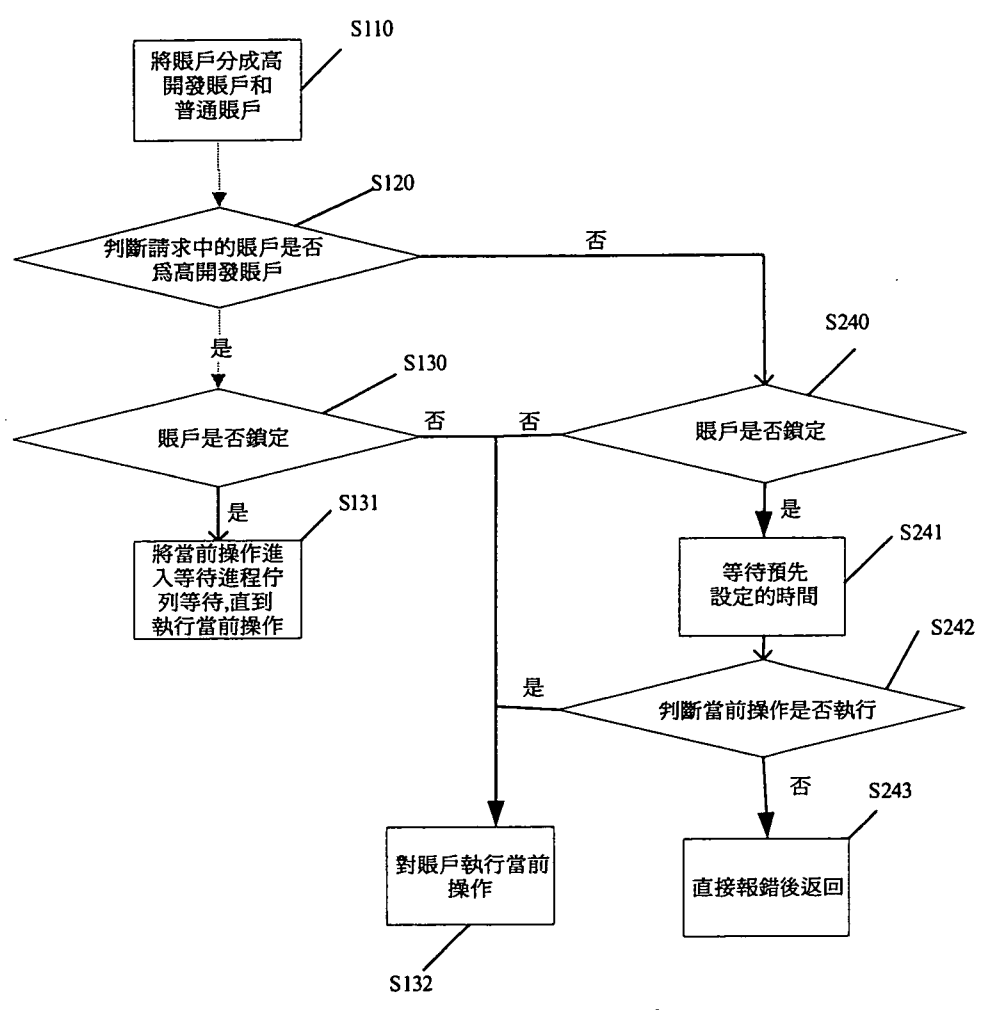


圖4