



(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 I697227 B

(45)公告日：中華民國 109 (2020) 年 06 月 21 日

(21)申請案號：107107225

(22)申請日：中華民國 107 (2018) 年 03 月 05 日

(51)Int. Cl. : H04L9/18 (2006.01)

G06F17/00 (2019.01)

(30)優先權：2017/05/31 中國大陸

201710397591.4

(71)申請人：香港商阿里巴巴集團服務有限公司 (香港地區) ALIBABA GROUP SERVICES
LIMITED (HK)

香港

(72)發明人：王士峰 (CN)

(74)代理人：林志剛

(56)參考文獻：

TW 201627889A

CN 106415530A

US 2017/0083860A1

US 2017/0149819A1

WO 2016/046820A1

審查人員：周官緯

申請專利範圍項數：15 項 圖式數：3 共 38 頁

(54)名稱

資料處理的方法及裝置

(57)摘要

本發明揭示一種資料處理的方法及裝置，該方法中區塊鏈節點在確定獲取到的預處理塊通過共識驗證後，可啟動對下一次待共識的預處理塊進行共識驗證，以透過並行的方式對已通過共識驗證的預處理塊中包含的業務資料進行資料處理。由於區塊鏈節點在確定獲取到的預處理塊通過共識驗證後，採用並行處理方式，啟動對下一次待共識的預處理塊進行共識驗證，且對已通過共識驗證的預處理塊中包含的業務資料進行處理。換句話說，區塊鏈節點實現業務資料在業務共識階段以及業務提交階段的並行處理，既能夠對一部分業務資料執行業務提交階段的資料處理，同時又可執行另一部分業務資料的業務共識階段的共識處理，從而提升系統的業務資料處理效率。

指定代表圖：

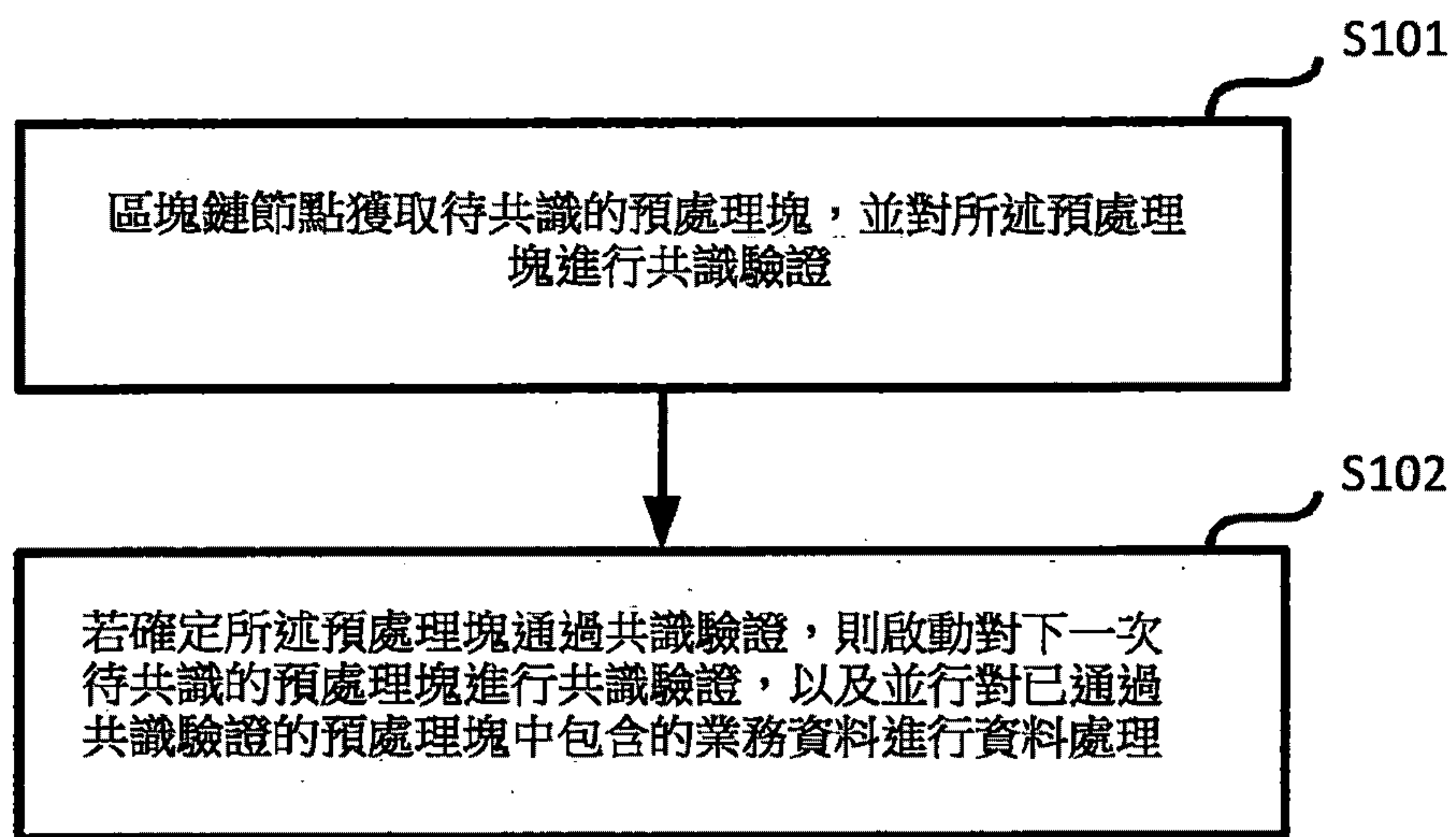


圖 1

【發明摘要】**【中文發明名稱】**

資料處理的方法及裝置

【中文】

本發明揭示一種資料處理的方法及裝置，該方法中區塊鏈節點在確定獲取到的預處理塊通過共識驗證後，可啟動對下一次待共識的預處理塊進行共識驗證，以透過並行的方式對已通過共識驗證的預處理塊中包含的業務資料進行資料處理。由於區塊鏈節點在確定獲取到的預處理塊通過共識驗證後，採用並行處理方式，啟動對下一次待共識的預處理塊進行共識驗證，且對已通過共識驗證的預處理塊中包含的業務資料進行處理。換句話說，區塊鏈節點實現業務資料在業務共識階段以及業務提交階段的並行處理，既能夠對一部分業務資料執行業務提交階段的資料處理，同時又可執行另一部分業務資料的業務共識階段的共識處理，從而提升系統的業務資料處理效率。

第 107107225 號

民國 108 年 12 月 31 日修正

【指定代表圖】第(1)圖。

【代表圖之符號簡單說明】無

【特徵化學式】無

【發明說明書】

【中文發明名稱】

資料處理的方法及裝置

【技術領域】

本發明係有關電腦技術領域，尤其有關一種資料處理的方法及裝置。

【先前技術】

隨著電腦技術的不斷發展，區塊鏈技術的應用範圍得到了進一步的拓展，目前，多種業務模式由於區塊鏈技術的引入而變得更加地有效和安全，從而為用戶提供了更為有效的業務服務。

在實際應用中，涉及區塊鏈技術的業務在業務執行過程中大致可分為三個過程：

一、業務受理階段。在這一階段中，區塊鏈節點可接收用戶透過終端或用戶端發送的待處理的業務資料(也可以稱之為交易資料)，並在通過對該業務資料的驗證後，儲存該業務資料。當然，區塊鏈節點在此階段也可接收由其他區塊鏈節點透過廣播的方式發送的待處理的業務資料，按照上述記載的方式，儲存該業務資料。

二、業務共識階段。在這一階段中，若該區塊鏈節點作為發起共識的主節點，可從儲存的業務資料中撈取一部分業務資料打包成預處理塊，並廣播給其他區塊鏈節點，

以對該預處理塊進行共識驗證。共識網路中的其他區塊鏈節點在接收到該預處理塊後，可根據已儲存的業務資料，對該預處理塊中包含的業務資料進行共識驗證。當然，若該區塊鏈節點不是主節點，則也可接收主節點廣播的預處理塊，並透過自身記憶體中儲存的業務請求，對該預處理塊中包含的業務請求進行共識驗證。

三、業務提交階段。在這一階段中，在該區塊鏈節點確定業務共識階段處理的預處理塊通過共識驗證後，可將該預處理塊中包含的業務資料儲存在區塊鏈中。不僅如此，該區塊鏈節點可將這些業務資料儲存在指定的資料庫中，並從自身的儲存空間中釋放該預處理塊中所包含的業務資料。

在現有技術中，對於同一個部分的業務資料，區塊鏈節點通常需要在業務共識階段完成的基礎上，才能進入業務提交階段，而只有在該業務提交階段完成後，區塊鏈節點才可以對下一次待共識的預處理塊進行共識驗證。

然而，由於現有技術中，對於業務資料處理過程中所經歷的業務共識階段和業務提交階段是串列執行的，區塊鏈節點只有在完成一次業務資料處理的業務提交階段後，方能啟動下一次業務資料處理的業務共識階段，這樣就勢必增加了兩次業務資料處理的時間間隔，進而降低了整個系統的業務處理效率。

【發明內容】

本發明實施例提供一種資料處理的方法，用以解決現有技術的區塊鏈技術中存在的業務處理效率低的問題。

本發明實施例提供了一種資料處理的方法，包括：

區塊鏈節點獲取待共識的預處理塊，並對所述預處理塊進行共識驗證；

若確定所述預處理塊通過共識驗證，則啟動對下一次待共識的預處理塊進行共識驗證，以及並行對已通過共識驗證的預處理塊中包含的業務資料進行資料處理。

本發明實施例提供一種資料處理的裝置，用以解決現有技術業務共識效率較低的問題。

本發明實施例提供了一種資料處理的裝置，包括：

獲取模組，獲取待共識的預處理塊，並對所述預處理塊進行共識驗證；

處理模組，若確定所述預處理塊通過共識驗證，則啟動對下一次待共識的預處理塊進行共識驗證，以及並行對已通過共識驗證的預處理塊中包含的業務資料進行資料處理。

本發明實施例提供一種資料處理的裝置，用以解決現有技術業務共識效率較低的問題。

本發明實施例提供了一種資料處理的裝置，包括：記憶體和至少一個處理器，其中：所述記憶體儲存有程式，並被配置成由所述至少一個處理器執行以下步驟：

獲取待共識的預處理塊，並對所述預處理塊進行共識驗證；

若確定所述預處理塊通過共識驗證，則啟動對下一次待共識的預處理塊進行共識驗證，以及並行對已通過共識驗證的預處理塊中包含的業務資料進行資料處理。

本發明實施例採用的上述至少一個技術方案能夠達到以下有益效果：

在本發明實施例中，由於區塊鏈節點在確定獲取到的預處理塊通過共識驗證後，採用並行處理方式，啟動對下一次待共識的預處理塊進行共識驗證，並且對已通過共識驗證的預處理塊中包含的業務資料進行資料處理。換句話說，區塊鏈節點實現了業務資料在業務共識階段以及業務提交階段的並行處理，既能夠對一部分業務資料執行業務提交階段的資料處理，同時又可執行另一部分業務資料的業務共識階段的共識處理，這樣能夠縮短兩次業務共識階段的共識處理的時間間隔，從而有效提升了系統的業務資料處理效率。

【圖式簡單說明】

此處所說明的附圖用來提供對本發明的進一步理解，構成本發明的一部分，本發明的示意性實施例及其說明用來解釋本發明，並不構成對本發明的不當限定。在附圖中：

圖1為本發明實施例提供的資料處理過程的示意圖；

圖2為本發明實施例所提供的區塊鏈節點進行資料處理的示意圖；

圖3為本發明實施例提供的一種資料處理的裝置示意圖。

【實施方式】

為了使本技術領域的人員更好地理解本發明中的技術方案，下面將結合本發明實施例中的附圖，對本發明實施例中的技術方案進行清楚、完整地描述，顯然，所描述的實施例僅僅是本發明的一部分實施例，而不是全部的實施例。基於本發明中的實施例，本領域普通技術人員在沒有作出創造性勞動前提下所獲得的所有其他實施例，都應當屬於本發明保護的範圍。

圖1為本發明實施例提供的資料處理過程的示意圖，具體包括以下步驟：

S101：區塊鏈節點獲取待共識的預處理塊，並對所述預處理塊進行共識驗證。

在本發明實施例中，區塊鏈節點在業務共識階段中，可獲取在一次共識所要的預處理塊(這裏將目前獲取的預處理塊稱之為本次共識所獲取的預處理塊)。其中，該預處理塊可以是由該區塊鏈節點根據自身保存的業務資料而產生的，也可以是從其他區塊鏈節點中獲取到的。

S102：若確定所述預處理塊通過共識驗證，則啟動對下一次待共識的預處理塊進行共識驗證，以及並行對已通過共識驗證的預處理塊中包含的業務資料進行資料處理。

上述區塊鏈節點在確定出本次共識中待共識的預處理

塊通過共識驗證後，可採用並行處理的方式，對已通過共識驗證的預處理塊進行業務提交階段的資料處理，這樣，能夠有效實現在對下一個預處理塊進行業務共識階段的共識處理時，還可以同步對已通過共識驗證的預處理塊進行業務提交階段的資料處理。

由此可見，在本發明實施例中，在業務資料的處理過程中，業務共識節點的業務共識階段的共識處理與業務提交階段的資料處理是同步並行處理的。即假設待共識的預處理塊為至少兩個，那麼透過本發明提供的技術方案，能夠實現在對已通過共識驗證的預處理塊進行業務提交階段的資料處理時，同步可以並行地對尚未通過共識驗證的預處理塊進行業務共識階段的共識處理。

需要說明的是，對於本次共識中待共識的預處理塊，在確定透過對本次共識中待共識的預處理塊的共識處理時，啟動對下一次共識的待共識的預處理塊的共識處理，同時獲取針對本次共識驗證通過的預處理塊的處理參數，以便於用來實現業務資料處理過程中業務提交階段的資料處理的處理器(後續可以稱之為預設的處理器)基於這些處理參數對本次共識驗證通過的預處理塊進行資料處理。

例如：一種情況：在啟動對下一次共識的待共識的預處理塊的共識處理時，預設的處理器基於產生的處理參數對本次共識驗證通過的預處理塊進行資料處理。這裏可以理解為對下一次共識的待共識的預處理塊的共識處理與對本次共識驗證通過的預處理塊進行資料處理是同時並行處

理的，有效縮短了本次共識的待共識的預處理塊的共識處理與下一次共識的待共識的預處理塊的共識處理之間的時間間隔；

另一種情況：在啟動對下一次共識的待共識的預處理塊的共識處理時，預設的處理器在對之前共識驗證通過的預處理塊進行資料處理，那麼本次共識驗證通過的預處理塊將被儲存至預設的佇列中進行等待，預設的處理器按照某種規則(例如：按照共識完成時間順序等)依次對該佇列中儲存的、通過共識驗證的預處理塊進行資料處理，這裏可以理解為本次共識的待共識的預處理塊的共識處理與該預處理塊的資料處理是透過非同步方式完成的。

例如，假設有 A、B、C 三個預處理塊，這三個預處理塊按照字母的排列順序依次發送至區塊鏈節點中進行共識驗證。區塊鏈節點確定預處理塊 A 通過共識驗證後，可透過預設的處理器對預處理塊 A 進行資料處理。與此同時，區塊鏈節點可對預處理塊 B 進行共識驗證，而當確定預處理塊 B 通過共識驗證後，發現還未完成對預處理塊 A 的資料處理時，則可將通過共識驗證的預處理塊 B 儲存在預設的佇列中進行等待，並繼續對預處理塊 C 進行共識驗證。當確定完成對預處理塊 A 的資料處理時，則可從預設的佇列中提取出預處理塊 B，繼而透過預設的處理器，對預處理塊 B 進行資料處理。

所以，對於每一個預處理塊來說，該預處理塊的共識處理和資料處理這兩個階段是透過非同步的方式完成的。

而對於不同的預處理塊來說，一個預處理塊的共識處理和另一個已通過共識驗證的預處理塊的資料處理可以透過同步的方式來進行處理。

區塊鏈節點在確定出本次共識中待共識的預處理塊通過共識驗證後，在本發明實施例中包括但不限於執行以下兩大類操作：

1、第一類操作：確定與本次共識驗證通過的預處理塊對應的處理參數。該處理參數中包含有對所述預處理塊中的業務資料進行處理的參數，以便於預設的處理器可以按照該處理參數對所述預處理塊進行處理，以完成業務提交階段的相關操作。

下面針對第一類操作進行詳細說明。

這裏的處理參數可以包含但不限於：儲存參數、釋放參數、刪除參數、上鏈參數等，而上述只是簡單的舉例說明了處理參數中所包含的一些參數，在實際應用中，該處理參數還可包含有其他的參數，而具體包含有哪些參數則可由區塊鏈節點在業務提交階段所執行的具體操作來決定。

例如，釋放參數用以指示對通過共識驗證的預處理塊執行將該預處理塊從儲存空間中進行釋放的操作；

儲存參數用以指示將通過共識驗證的預處理塊中包含的業務資料儲存至指定位置。針對不同的預處理塊確定的儲存參數不同。在該儲存參數中包含儲存位置。

刪除參數用以指示將通過共識驗證的預處理塊在業務

共識階段產生的訊息(如PBFT共識中的預準備訊息、準備訊息、確認訊息)進行刪除，以降低儲存壓力。

上鏈參數用以指示按照預處理塊中包含的上一個區塊的頭Hash，將該預處理塊以區塊的形式上鏈至上一個區塊所在的區塊鏈中。

較優地，在本發明實施例中，區塊鏈節點在對相鄰下一次待共識的預處理塊進行共識驗證的過程中，還可透過預設的處理器實現對通過本次共識驗證的預處理塊中包含的業務資料進行處理的平行作業。

假設現有技術中區塊鏈節點執行一次業務資料處理過程中的業務共識階段以及業務提交階段，可以理解成是透過同一個線程來完成這兩個階段的。區塊鏈節點需要透過該線程先完成本次業務資料處理過程中的業務共識階段，而後再利用該線程執行本次業務資料處理過程中的業務提交階段。顯然，現有技術中區塊鏈節點是透過串列的方式來執行一次業務資料處理過程中所涉及的業務共識階段以及業務提交階段，因此，增加了相鄰兩次業務資料處理之間的時間間隔，降低了業務資料處理效率。

為了有效的解決上述問題，在本發明實施例中，區塊鏈節點中預先設置一個處理器(該處理器可以採用非同步處理方式操作，這裏不做具體限定)，該處理器可用來執行業務提交階段所涉及的操作。換句話說，在本發明實施例中，區塊鏈節點可以採用兩個線程分別實現業務資料處理過程中的業務共識階段的共識處理和業務提交階段的資

料處理，亦即，由一個線程用以對待共識的預處理塊進行共識處理，另一個線程用以對共識驗證通過的預處理塊進行資料處理，這樣對於同一個預處理塊，其共識處理與其資料處理透過非同步方式完成。

這樣一來，區塊鏈節點利用該處理器執行業務提交階段所涉及的操作的過程中，區塊鏈節點可以不受影響的啟動相鄰下一次待共識的預處理塊的共識任務，就開始下一次共識的執行，從而極大的縮短了相鄰兩次共識之間的時間間隔，進而提高了共識效率。

在本發明實施例中，本次共識所確定的處理參數均可透過上一次共識所確定的處理參數得出。

以儲存參數為例來進行說明。區塊鏈節點在確定本次共識中待共識的預處理塊通過共識驗證後，可根據本次共識已通過共識驗證的預處理塊的儲存參數，確定出相鄰下一次待共識的預處理塊的儲存參數，並儲存。

例如，假設通過本次共識驗證的預處理塊所對應的儲存參數(也可以理解為本次共識所對應的儲存參數)中規定了需要將通過本次共識驗證的預處理塊中包含的業務資料儲存在關係型資料庫A中的a表上。則區塊鏈節點可以依照該關係型資料庫A中各表的字母順序，確定出通過相鄰下一次共識驗證的預處理塊所對應的儲存參數為：將通過此次共識驗證(亦即，上面提到的相鄰下一次共識驗證)的預處理塊中包含的業務資料儲存在關係型資料庫A的b表中。

區塊鏈節點可將確定出的下一次共識的儲存參數進行

儲存，這樣使得預設的處理器在確定下一次共識的預處理塊通過共識驗證時，可根據得到的儲存參數，確定出需要將通過此次通過共識驗證的預處理塊中包含的業務資料儲存在哪一個資料庫中的哪一個表中。

較優地，在本發明實施例中，區塊鏈節點在確定出本次共識中待共識的預處理塊通過共識驗證後，也可根據該預處理塊以及該預處理塊的儲存參數，確定出相鄰下一次共識的預處理塊的儲存參數並儲存。

具體地，若儲存參數中包含的儲存位置是以儲存指標形式存在的，亦即，儲存指標所指向的位置即為預處理塊中業務資料的儲存位置，區塊鏈節點在確定出本次共識中待共識的預處理塊通過共識驗證後，可以以目前儲存指標所在位置為起點，按照本次共識的預處理塊的大小，移動儲存指標的位置，並將移動後的儲存指標的位置確定為下一次共識所對應的儲存參數，亦即，相鄰下一次待共識的預處理塊的儲存參數。

例如，假設每次共識的處理參數中包含有一個以儲存指標basepointer的形式而出現的儲存參數，該儲存指標(亦即，儲存參數)指向業務資料的具體儲存位置。該儲存指標basepointer的初始值可以設為0，每經過一次共識後，區塊鏈節點可根據經過本次共識的預處理塊的大小以及本次共識所對應的處理參數中包含的儲存指標，確定出相鄰下一次共識所對應的處理參數中，儲存指標的具體值。在第一次共識中，區塊鏈節點確定出通過此次共識驗證的預

處理塊的大小為1024位元組，則區塊鏈節點可根據確定出的預處理塊的大小，以及儲存指標 `basepointer` 的初始值0，確定出第二次共識所對應的處理參數中包含的儲存指標 `basepointer` 為1024，並儲存。相應地，在第二次共識中，區塊鏈節點可透過處理器，將通過第二次共識驗證的預處理塊中包含的業務資料儲存在儲存指標 `basepointer=1024` 所對應的儲存位置中。

在第二次共識中，區塊鏈節點確定出通過此次共識驗證的預處理塊的大小為10位元組，則區塊鏈節點可根據確定出的通過此次共識驗證的預處理塊的大小，以及上一次共識(亦即，第一次共識)對應的處理參數中所包含的儲存指標 `basepointer=1024`，確定出第三次共識對應的處理參數中包含的儲存指標 `basepointer` 為1034並儲存。相應地，在第三次共識中，區塊鏈節點可透過處理器，將通過第三次共識驗證的預處理塊中包含的業務資料儲存在儲存指標 `basepointer=1034` 所對應的儲存位置中，後續的共識則可以此類推。

需要說明的是，在本發明實施例中，除了可以透過本次共識的預處理塊的大小，確定下一次共識對應的處理參數中包含的儲存參數外，還可透過預處理塊的其他資訊來進行確定，具體採用哪一種資訊來進行確定則可由區塊鏈節點的運維人員來決定，在此就不一一舉例說明了。

需要說明的是，對於處理參數中所包含的不同參數來說，區塊鏈節點在業務提交階段所執行的操作，決定了這

些參數是否每經過一次共識就需要發生相應的變更。如，對於上述提到的儲存參數來說，由於每次共識所涉及的預處理塊中包含的業務資料不可能儲存在相同的儲存位置中，所以，每經過一次共識後，儲存參數則需要發生相應的變更。而對於上述提到的上鏈參數來說，無論是哪次共識所涉及的預處理塊，預處理塊一旦通過了共識網路的共識驗證，則均需要將預處理塊以區塊的形式儲存在區塊鏈中。換句話說，無論是哪個預處理塊，該預處理塊一旦通過共識驗證，則處理器均需要對該預處理塊執行上鏈操作。所以，每經過一次共識後，該上鏈參數則不需要發生相應的變更，亦即，對於每次共識來說，上鏈參數可以是一致的。

較優地，在本發明實施例中，區塊鏈節點可將得到的下一次共識的處理參數儲存在一個預設的佇列中。例如，將相鄰下一次業務共識的預處理塊的儲存參數儲存在該預設的佇列中。

這樣一來，處理器可從該預設的佇列中獲取到處理參數（亦即，獲取到已通過共識驗證的預處理塊的處理參數），進而根據該處理參數所包含的儲存參數，將與該儲存參數對應的通過共識驗證的預處理塊中包含的業務資料進行儲存。

其中，這裏提到的預設的佇列可以是先進先出佇列（First Input First Output，FIFO），也可以是其他類型的佇列，這裏不做具體限定。處理器可獲取到儲存在該FIFO佇

列中的處理參數，並根據該處理參數中所包含的儲存參數，從儲存的已通過共識驗證的預處理塊中確定出與該儲存參數對應的預處理塊，進而根據該儲存參數，將該預處理塊中包含的業務資料進行儲存。

具體地，處理器可從上述FIFO佇列中獲取到儲存參數，而後，處理器可進一步的從確定出與該儲存參數對應的待處理的預處理塊。例如：區塊鏈節點在對本次共識的待共識的預處理塊的共識驗證通過時，產生該預處理塊的儲存參數，並確定該預處理塊與該儲存參數之間的對應關係，那麼處理器即可根據該對應關係，確定出與該儲存參數對應的待處理的預處理塊。再例如：區塊鏈節點在對本次共識的待共識的預處理塊的共識驗證通過時，產生該預處理塊的儲存參數，確定產生該儲存參數的第一時間，並確定本次共識的待共識的預處理塊的共識驗證通過的第二時間，建立第一時間和第二時間之間的對應關係，那麼處理器可以根據儲存參數的產生時間，查找與該產生時間之間滿足設定條件的共識處理結束時間對應的預處理塊，查找出的預處理塊可以確定為與該儲存參數對應的待處理的預處理塊。再例如：處理器從FIFO佇列中獲取位於佇列出口最前端的儲存參數，那麼處理器從區塊鏈節點的儲存空間中查找儲存位置排在最靠前的已通過共識驗證的預處理塊，將該預處理塊確定為與該儲存參數對應的預處理塊。

處理器在確定出與該儲存參數對應的待處理的預處理塊後，可根據該儲存參數，將該待處理的預處理塊中包含

的業務資料儲存在該儲存參數所規定的儲存位置中。

需要說明的是，在本發明實施例中，區塊鏈節點除了可採用上述說明的FIFO佇列來儲存各處理參數外，還可透過其他的佇列來進行儲存，如雙端佇列等，在此就不一一舉例說明了。

處理器除了可根據獲取的處理參數中包含的儲存參數，對該儲存參數對應的待處理的預處理塊中包含的業務資料進行儲存外，還可根據該處理參數中包含的其他參數，對該待處理的預處理塊執行其他操作。

例如，處理器可以根據該處理參數中的釋放參數，將該待處理的預處理塊中包含的業務資料從儲存空間中釋放。再例如，處理器可根據獲取到的處理參數中包含的刪除參數，將本次共識中業務共識階段所產生的諸如預準備訊息、準備訊息、確認訊息等進行刪除，以節省區塊鏈節點的儲存空間。當然，處理器還可根據該處理參數中的其他參數來執行其他的操作，在此就不一一舉例說明了。

2、第二類操作：根據本次共識所對應的共識參數，更新下一次共識所對應的共識參數。換句話說，區塊鏈節點可確定出已通過本次共識的預處理塊所對應的共識參數，並根據確定出的該共識參數，得到相鄰下一次待共識的預處理塊對應的共識參數。

需要說明的是，在本發明實施例中，區塊鏈節點在確定本次共識需要進行共識的預處理塊通過共識驗證後，除了需要根據本次共識所對應的處理參數，來得到下一次共

識所對應的處理參數並儲存外，還可根據本次共識所對應的共識參數，更新下一次共識所對應的共識參數。換句話說，區塊鏈節點可確定出已通過本次共識驗證的預處理塊所對應的共識參數，並根據確定出的該共識參數，得到相鄰下一次待共識的預處理塊對應的共識參數。

其中，這裏提到的共識參數可以理解為一次共識所對應的屬性資訊。例如，以 PBFT 共識舉例說明，在 PBFT 共識過程中，一次共識通常對應有一個視圖編號 v ，該視圖編號 v 用以唯一標識這一次共識。在一次共識中，無論共識網路中的哪一個區塊鏈節點作為主節點，其產生的預處理塊中所包含的上一個區塊的頭 Hash 通常都是目前位於區塊鏈上最後一個區塊的頭 Hash。對於這裏提到的視圖編號 v 以及上一個區塊的頭 Hash，則可以稱之為本次共識所對應的共識參數。

當然，共識參數除了可包含有上述說明的視圖編號 v 以及上一個區塊的頭 Hash 外，還可包含有其他的資訊，而對於不同共識方式來說，共識參數中所包含的內容也存在有一定的差異，具體就不進行詳細說明了。

區塊鏈節點在確定本次共識的預處理塊通過共識驗證後，可進一步的確定出本次共識所對應的共識參數，進而透過更新該共識參數的方式，得到下一次共識所對應的共識參數，亦即，得到相鄰下一次待共識的預處理塊對應的共識參數。

還以 PBFT 共識方式為例，假設在本次共識所對應的

共識參數中包含有視圖編號 v 以及上一個區塊的頭 Hash，其中，視圖編號 v 為 16、上一個區塊的頭 Hash 為 0929d9sldom23oix239xed。區塊鏈節點在確定出本次共識所涉及的預處理塊通過共識驗證後，可將視圖編號 16 更新為 17，並根據預處理塊中包含的該預處理塊的頭 Hash：679xx9a9a8dfa23389xx34，將下一次共識所基於的區塊的頭 Hash 更新為 679xx9a9a8dfa23389xx34。這樣一來，下一次共識所對應的共識參數應為：視圖編號 v ：17、上一個區塊的頭 Hash：679xx9a9a8dfa23389xx34。

由於區塊鏈節點能夠根據本次共識所對應的共識參數，得到下一次共識所對應的共識參數，這樣一來，該區塊鏈節點可以根據得到的下一次共識所對應的共識參數，啟動對相鄰下一次待共識的預處理塊進行共識驗證。其中，這裏提到的共識參數可以被儲存在記憶體中，也可以被儲存在區塊鏈節點所對應的資料庫中，當然還可以總體變數的方式存在。

需要說明的是，在本發明實施例中，該共識參數也可以與處理參數對應儲存在上述預設的佇列中。這樣一來，用於共識處理的處理器可以從該預設的佇列中獲取到共識參數，並根據共識參數，啟動一次新的共識處理；用於資料處理的處理器從該預設的佇列中獲取到處理參數，進而根據獲取到的處理參數，啟動對該處理參數對應的預處理塊中包含的業務資料進行資料處理。

例如，假設處理器從 FIFO 佇列中獲取到一個處理參數

以及與之對應的共識參數，該共識參數中包含有視圖編號 v ，這樣一來，該處理器可以從區塊鏈節點自身的儲存空間中，確定出與該視圖編號 v 相對應的待處理的預處理塊，進而根據獲取到處理參數，對該待處理的預處理塊進行處理。

當然，在本發明實施例中，處理參數也可不被儲存在上述提到的預設的佇列中，如，可將處理參數儲存在區塊鏈節點的自身記憶體中，也可將該處理參數儲存在區塊鏈節點所對應的資料庫中，抑或是儲存在區塊鏈節點的其他位置中，在此就不一一舉例說明了。

從上述方法中可以看出，由於區塊鏈節點在確定獲取到的預處理塊通過共識驗證後，採用並行處理方式，啟動對相鄰下一次待共識的預處理塊進行共識驗證，並且對已通過共識驗證的預處理塊中包含的業務資料進行處理。換句話說，區塊鏈節點實現了業務資料在業務共識階段以及業務提交階段的並行處理，既能夠對一部分業務資料執行業務提交階段的資料處理，同時又可執行另一部分業務資料的業務共識階段的共識處理，這樣能夠縮短相鄰兩次業務共識階段的共識處理的時間間隔，從而有效提升了系統的業務資料處理效率。

為了進一步的說明本發明所提到的資料處理方法，下面將詳細簡明的說明該資料處理方法所涉及的各過程，如圖2所示。

圖2為本發明實施例所提供的區塊鏈節點進行資料處

理的示意圖。

在業務受理階段中，用戶可透過終端中安裝的用戶端向區塊鏈節點發送業務資料，而區塊鏈節點可將接收到的業務資料進行驗證，並將驗證通過的業務資料儲存在自身對應的儲存空間中來進行保存。

在業務共識階段中，區塊鏈節點可獲取到在本次共識中待共識的預處理塊，其中，若該區塊鏈節點作為發起此次共識的主節點時，可從自身的儲存空間中撈取一部分業務資料，並將這部分業務資料打包成預處理塊，此時該區塊鏈節點將獲取到需要在本次共識中進行共識的預處理塊。與此同時，該區塊鏈節點需要將該預處理塊透過廣播的方式發送至共識網路中的其他區塊鏈節點中，以使其他的區塊鏈節點對該預處理塊進行共識驗證。

若該區塊鏈節點並非發起此次共識的主節點，則可從發起此次共識的主節點獲取到需要在本次業務共識中進行共識的預處理塊，繼而對該預處理塊進行共識驗證。

區塊鏈節點在確定該預處理塊通過共識驗證後，可根據本次共識所對應的共識參數(亦即，該預處理塊所對應的共識參數)，更新得到下一次共識所對應的共識參數(亦即，相鄰下一次待共識的預處理塊對應的共識參數)，以備下一次共識所使用。不僅如此，區塊鏈節點還可根據該預處理塊以及本次共識所對應的處理參數(亦即，該已通過本次共識驗證的預處理塊對應的處理參數)，得到下一次共識所對應的處理參數(亦即，相鄰下一次待共識的預

處理塊所對應的處理參數)，並將得到的下一次共識所對應的處理參數儲存在FIFO佇列中。

區塊鏈節點在得到相鄰下一次共識對應的共識參數以及處理參數的過程中，可啟動執行相鄰下一次共識的業務共識階段，亦即，啟動對相鄰下一次待共識的預處理塊進行共識驗證。而區塊鏈節點在啟動執行相鄰下一次共識的業務共識階段中，可以透過處理器，實現對本次共識中的業務提交階段執行平行作業。

換句話說，區塊鏈節點將業務提交階段所涉及的操作交給了處理器來完成，而自身則可進行相鄰的下一次共識驗證，從而實現了一次共識中，業務共識階段和業務提交階段的並行處理，縮短了兩次相鄰共識之間的時間間隔，進而提高了共識效率。

處理器可從FIFO佇列中獲取到本次共識所對應的處理參數，進而根據該處理參數中包含的上鏈參數，將通過本次共識的預處理塊以區塊的形式儲存在區塊鏈中；處理器可根據該處理參數中包含的是否參數以及儲存參數，將該預處理塊中包含的業務資料從區塊鏈節點自身的儲存空間中進行釋放，並將釋放出的業務資料按照該儲存參數中的規定，儲存在相應的儲存位置中；處理器可根據該處理屬性中包含的刪除參數，將業務共識階段所產生的諸如PBFT共識的預準備訊息、準備訊息、確認訊息等進行刪除，以節省區塊鏈節點的儲存空間。

以上為本發明實施例提供的資料處理方法，基於同樣

的思路，本發明實施例還提供資料處理的裝置，如圖3所示。

圖3為本發明實施例提供的一種資料處理的裝置示意圖，具體包括：

獲取模組301，獲取待共識的預處理塊，並對所述預處理塊進行共識驗證；

處理模組302，若確定所述預處理塊通過共識驗證，則啟動對下一次待共識的預處理塊進行共識驗證，以及並行對已通過共識驗證的預處理塊中包含的業務資料進行資料處理。

所述處理模組302，透過預設的處理器並行對所述已通過共識驗證的預處理塊中包含的業務資料進行資料處理。

所述處理模組302，針對所述已通過共識驗證的預處理塊，執行以下操作：呼叫處理器獲取儲存參數，所述儲存參數中包含儲存位置；根據所述儲存參數，確定與所述儲存參數對應的待處理的預處理塊，並將確定的所述待處理的預處理塊中包含的業務資料儲存在所述儲存位置中。

所述獲取模組301，在確定所述預處理塊通過共識驗證後，根據所述預處理塊以及所述預處理塊的儲存參數，確定相鄰下一次共識的預處理塊的儲存參數，並儲存。

所述獲取模組301，將相鄰下一次共識的預處理塊的儲存參數儲存在先進先出FIFO佇列中。

所述處理模組302，呼叫所述處理器從所述FIFO佇列

中獲取儲存參數。

所述獲取模組301，若確定所述預處理塊通過共識驗證，則確定所述預處理塊對應的共識參數，並根據確定的所述預處理塊對應的共識參數，得到相鄰下一次待共識的預處理塊對應的共識參數，所述共識參數用以指示所述區塊鏈節點對待共識的預處理塊進行共識驗證。

所述處理模組302，在得到相鄰下一次待共識的預處理塊對應的共識參數時，基於得到的所述共識參數，啟動對相鄰下一次待共識的預處理塊進行共識驗證。

基於同樣的思路，本發明實施例還提供另一種資料處理的裝置，具體包括：

記憶體和至少一個處理器，其中：所述記憶體儲存有程式，並被配置成由所述至少一個處理器執行以下步驟：

獲取待共識的預處理塊，並對所述預處理塊進行共識驗證；

若確定所述預處理塊通過共識驗證，則啟動對下一次待共識的預處理塊進行共識，以及並行對已通過共識驗證的預處理塊中包含的業務資料進行資料處理。

其中，處理器透過記憶體儲存的程式所執行的具體操作可以參照上述實施例中所記載的內容，在此則不再贅述。

在本發明實施例中，區塊鏈節點在確定獲取到的預處理塊通過共識驗證後，可啟動對下一次待共識的預處理塊進行共識驗證，以透過並行的方式對已通過共識驗證的預

處理塊中包含的業務資料進行資料處理。由於區塊鏈節點在確定獲取到的預處理塊通過共識驗證後，採用並行處理方式，啟動對下一次待共識的預處理塊進行共識驗證，並且對已通過共識驗證的預處理塊中包含的業務資料進行處理。換句話說，區塊鏈節點實現了業務資料在業務共識階段以及業務提交階段的並行處理，既能夠對一部分業務資料執行業務提交階段的資料處理，同時又可執行另一部分業務資料的業務共識階段的共識處理，這樣能夠縮短兩次業務共識階段的共識處理的時間間隔，從而有效提升了系統的業務資料處理效率。

在20世紀90年代，對於一個技術的改進可以很明顯地區分是硬體上的改進(例如，對二極體、電晶體、開關等電路結構的改進)還是軟體上的改進(對於方法流程的改進)。然而，隨著技術的發展，當今的很多方法流程的改進已經可以視為硬體電路結構的直接改進。設計人員幾乎都透過將改進的方法流程編程到硬體電路中來得到相應的硬體電路結構。因此，不能說一個方法流程的改進就不能用硬體實體模組來實現。例如，可編程邏輯裝置(Programmable Logic Device, PLD)(例如，現場可編程閘陣列(Field Programmable Gate Array, FPGA))就是這樣的一種積體電路，其邏輯功能由用戶對裝置編程來確定。由設計人員自行編程來把一個數位系統“集成”在一片PLD上，而不需要請晶片製造廠商來設計和製作專用的積體電路晶片。而且，如今，取代手工地製作積體電路晶片，這

種編程也多半改用“邏輯編譯器(logic compiler)”軟體來實現，它與程式開發撰寫時所用的軟體編譯器相類似，而要編譯之前的原始代碼也得用特定的編程語言來撰寫，此稱之為硬體描述語言(Hardware Description Language，HDL)，而HDL也並非僅有一種，而是有許多種，如ABEL(Advanced Boolean Expression Language)、AHDL(Altera Hardware Description Language)、Confluence、CUPL(Cornell University Programming Language)、HDCal、JHDL(Java Hardware Description Language)、Lava、Lola、MyHDL、PALASM、RHDL(Ruby Hardware Description Language)等，目前最普遍使用的是VHDL(Very-High-Speed Integrated Circuit Hardware Description Language)與Verilog。本領域技術人員也應該清楚，只需要將方法流程用上述幾種硬體描述語言稍作邏輯編程並編程到積體電路中，就可以很容易得到實現該邏輯方法流程的硬體電路。

控制器可以按任何適當的方式來實現，例如，控制器可以採取例如微處理器或處理器以及儲存可由該(微)處理器執行的電腦可讀程式碼(例如，軟體或韌體)的電腦可讀媒體、邏輯閘、開關、特殊應用積體電路(Application Specific Integrated Circuit，ASIC)、可編程邏輯控制器和嵌入式微控制器的形式，控制器的例子包括但不限於以下微控制器：ARC 625D、Atmel AT91SAM、Microchip PIC18F26K20 以及Silicone Labs C8051F320，記憶體控制

器還可以被實現為記憶體的控制邏輯的一部分。本領域技術人員也知道，除了以純電腦可讀程式碼方式來實現控制器以外，完全可以透過將方法步驟進行邏輯編程來使得控制器以邏輯閘、開關、特殊應用積體電路、可編程邏輯控制器和嵌入式微控制器等的形式來實現相同功能。因此這種控制器可以被認為是一種硬體部件，而對其內包括的用來實現各種功能的裝置也可以視為硬體部件內的結構。或者甚至，可以將用來實現各種功能的裝置視為既可以是實現方法的軟體模組又可以是硬體部件內的結構。

上述實施例闡明的系統、裝置、模組或單元，具體可以由電腦晶片或實體實現，或者由具有某種功能的產品來實現。一種典型的實現設備為電腦。具體的，電腦例如可以為個人電腦、膝上型電腦、蜂巢式電話、相機電話、智慧型電話、個人數位助理、媒體播放器、導航設備、電子郵件設備、遊戲控制臺、平板電腦、可穿戴式設備或者這些設備中的任何設備的組合。

為了描述的方便，描述以上裝置時以功能分為各種單元分別描述。當然，在實施本發明時可以把各單元的功能在同一個或多個軟體和/或硬體中實現。

本領域內的技術人員應明白，本發明的實施例可提供為方法、系統、或電腦程式產品。因此，本發明可採用完全硬體實施例、完全軟體實施例、或結合軟體和硬體態樣的實施例的形式。而且，本發明可採用在一個或多個其中包含有電腦可用程式碼的電腦可用儲存媒體(包括但不限

於磁碟記憶體、CD-ROM、光學記憶體等)上實施的電腦程式產品的形式。

本發明是參照根據本發明實施例的方法、設備(系統)、和電腦程式產品的流程圖和/或方塊圖來予以描述的。應理解可由電腦程式指令來實現流程圖和/或方塊圖中的每一個流程和/或方塊、以及流程圖和/或方塊圖中的流程和/或方塊的結合。可提供這些電腦程式指令到通用電腦、專用電腦、嵌入式處理機或其他可編程資料處理設備的處理器以產生一個機器，使得透過電腦或其他可編程資料處理設備的處理器執行的指令而產生用以實現在流程圖中的一個流程或多個流程和/或方塊圖中的一個方塊或多個方塊中指定的功能的裝置。

這些電腦程式指令也可被儲存在能引導電腦或其他可編程資料處理設備以特定方式操作的電腦可讀記憶體中，使得儲存在該電腦可讀記憶體中的指令產生包括指令裝置的製造品，該指令裝置實現在流程圖中的一個流程或多個流程和/或方塊圖中的一個方塊或多個方塊中指定的功能。

這些電腦程式指令也可被裝載到電腦或其他可編程資料處理設備上，使得在電腦或其他可編程設備上執行一系列操作步驟以產生電腦實現的處理，從而在電腦或其他可編程設備上執行的指令提供用以實現在流程圖中的一個流程或多個流程和/或方塊圖中的一個方塊或多個方塊中指定的功能的步驟。

在一個典型的配置中，計算設備包括一個或多個處理器(CPU)、輸入/輸出介面、網路介面和記憶體。

記憶體可能包括電腦可讀媒體中的非永久性記憶體，隨機存取記憶體(RAM)和/或非易失性記憶體等形式，如唯讀記憶體(ROM)或快閃記憶體(flash RAM)。記憶體是電腦可讀媒體的示例。

電腦可讀媒體包括永久性和非永久性、可移動式和非可移動式媒體可以由任何方法或技術來實現資訊儲存。資訊可以是電腦可讀指令、資料結構、程式的模組或其他資料。電腦的儲存媒體的例子包括，但不限於相變記憶體(PRAM)、靜態隨機存取記憶體(SRAM)、動態隨機存取記憶體(DRAM)、其他類型的隨機存取記憶體(RAM)、唯讀記憶體(ROM)、電可擦除可編程唯讀記憶體(EEPROM)、快閃記憶體或其他記憶體技術、唯讀光碟唯讀記憶體(CD-ROM)、數位多功能光碟(DVD)或其他光學儲存、磁盒式磁帶，磁帶磁碟片儲存或其他磁性儲存設備或任何其他非傳輸媒體，可用來儲存可以被計算設備訪問的資訊。按照本文中的界定，電腦可讀媒體不包括暫態性電腦可讀媒體(transitory media)，如調變的資料信號和載波。

還需要說明的是，術語“包括”、“包含”或者其任何其他變體意在涵蓋非排他性的包含，從而使得包括一系列要素的過程、方法、商品或者設備不僅包括那些要素，而且還包括沒有明確列出的其他要素，或者是還包括為這種過程、方法、商品或者設備所固有的要素。在沒有更多限制

的情況下，由語句“包括一個……”限定的要素，並不排除在包括所述要素的過程、方法、商品或者設備中還存在有另外的相同要素。

本領域技術人員應明白，本發明的實施例可提供為方法、系統或電腦程式產品。因此，本發明可採用完全硬體實施例、完全軟體實施例或結合軟體和硬體態樣的實施例的形式。而且，本發明可採用在一個或多個其中包含有電腦可用程式碼的電腦可用儲存媒體(包括但不限於磁碟記憶體、CD-ROM、光學記憶體等)上實施的電腦程式產品的形式。

本發明可以在由電腦執行的電腦可執行指令的一般上下文中描述，例如程式模組。一般地，程式模組包括執行特定任務或實現特定抽象資料類型的常式、程式、物件、元件、資料結構等等。也可以在分散式計算環境中實踐本發明，在這些分散式計算環境中，由透過通信網路而被連接的遠端處理設備來執行任務。在分散式計算環境中，程式模組可以位於包括儲存設備在內的本地和遠端電腦儲存媒體中。

本說明書中的各個實施例均採用漸進的方式來予以描述，各個實施例之間相同相似的部分互相參見即可，每個實施例重點說明的都是與其他實施例的不同之處。尤其，對於系統實施例而言，由於其基本相似於方法實施例，所以描述的比較簡單，相關之處參見方法實施例的部分說明即可。

以上所述僅為本發明的實施例而已，並不用來限制本發明。對於本領域技術人員來說，本發明可以有各種更改和變化。凡在本發明的精神和原理之內所作的任何修改、等同替換、改進等，均應包含在本發明的申請專利範圍的範疇之內。

【符號說明】

301：獲取模組

302：處理模組

【發明申請專利範圍】

【第1項】

一種資料處理的方法，包括：

區塊鏈節點獲取待共識的預處理塊，並對該預處理塊進行共識驗證；

若確定該預處理塊通過共識驗證，則啟動對下一次待共識的預處理塊進行共識驗證，以及並行對已通過共識驗證的預處理塊中包含的業務資料進行資料處理；以及

若確定該預處理塊通過共識驗證，則確定該預處理塊對應的共識參數，並根據確定的該預處理塊對應的共識參數，得到相鄰下一次待共識的預處理塊對應的共識參數，該共識參數用來指示該區塊鏈節點對待共識的預處理塊進行共識驗證。

【第2項】

如申請專利範圍第1項所述的方法，並行對已通過共識驗證的預處理塊中包含的業務資料進行處理，具體包括：

透過預設的處理器並行對該已通過共識驗證的預處理塊中包含的業務資料進行資料處理。

【第3項】

如申請專利範圍第2項所述的方法，透過預設的處理器並行對該已通過共識驗證的預處理塊中包含的業務資料進行資料處理，具體包括：

針對該已通過共識驗證的預處理塊，執行以下操作：

調用處理器獲取儲存參數，該儲存參數中包含儲存位置；以及

根據該儲存參數，確定與該儲存參數對應的待處理的預處理塊，並將確定的該待處理的預處理塊中包含的業務資料儲存在該儲存位置中。

【第4項】

如申請專利範圍第1至3項中任一項所述的方法，該方法還包括：

在確定該預處理塊通過共識驗證後，根據該預處理塊以及該預處理塊的儲存參數，確定相鄰下一次待共識的預處理塊的儲存參數，並儲存。

【第5項】

如申請專利範圍第4項所述的方法，儲存相鄰下一次待共識的預處理塊的儲存參數，具體包括：

將相鄰下一次待共識的預處理塊的儲存參數儲存在先進先出FIFO佇列中。

【第6項】

如申請專利範圍第5項所述的方法，調用處理器獲取儲存參數，具體包括：

調用該處理器從該FIFO佇列中獲取儲存參數。

【第7項】

如申請專利範圍第1項所述的方法，啟動對下一次待共識的預處理塊進行共識驗證，包括：

在得到相鄰下一次待共識的預處理塊對應的共識參數

時，基於得到的該共識參數，啟動對相鄰下一次待共識的預處理塊進行共識驗證。

【第8項】

一種資料處理的裝置，包括：

獲取模組，獲取待共識的預處理塊，並對該預處理塊進行共識驗證；以及

處理模組，若確定該預處理塊通過共識驗證，則啟動對下一次待共識的預處理塊進行共識驗證，以及並行對已通過共識驗證的預處理塊中包含的業務資料進行資料處理，而且若確定該預處理塊通過共識驗證，則確定該預處理塊對應的共識參數，並根據確定的該預處理塊對應的共識參數，得到相鄰下一次待共識的預處理塊對應的共識參數，該共識參數用來指示該區塊鏈節點對待共識的預處理塊進行共識驗證。

【第9項】

如申請專利範圍第8項所述的裝置，該處理模組，透過預設的處理器並行對該已通過共識驗證的預處理塊中包含的業務資料進行資料處理。

【第10項】

如申請專利範圍第9項所述的裝置，該處理模組，針對該已通過共識驗證的預處理塊，執行以下操作：調用處理器獲取儲存參數，該儲存參數中包含儲存位置；根據該儲存參數，確定與該儲存參數對應的待處理的預處理塊，並將確定的該待處理的預處理塊中包含的業務資料儲存在

儲存位置中。

【第11項】

如申請專利範圍第8至10項中任一項所述的裝置，該獲取模組，在確定該預處理塊通過共識驗證後，根據該預處理塊以及該預處理塊的儲存參數，確定相鄰下一次待共識的預處理塊的儲存參數，並儲存。

【第12項】

如申請專利範圍第11項所述的裝置，該獲取模組，將相鄰下一次待共識的預處理塊的儲存參數儲存在先進先出佇列FIFO佇列中。

【第13項】

如申請專利範圍第12項所述的裝置，該處理模組，調用該處理器從該FIFO佇列中獲取儲存參數。

【第14項】

如申請專利範圍第8項所述的裝置，該處理模組，在得到相鄰下一次待共識的預處理塊對應的共識參數時，基於得到的該共識參數，啟動對相鄰下一次待共識的預處理塊進行共識驗證。

【第15項】

一種資料處理的裝置，包括：記憶體和至少一個處理器，其中，該記憶體儲存有程式，並被配置成由該至少一個處理器執行以下步驟：

獲取待共識的預處理塊，並對該預處理塊進行共識驗證；

若確定該預處理塊通過共識驗證，則啟動對下一次待共識的預處理塊進行共識驗證，以及並行對已通過共識驗證的預處理塊中包含的業務資料進行資料處理；以及

若確定該預處理塊通過共識驗證，則確定該預處理塊對應的共識參數，並根據確定的該預處理塊對應的共識參數，得到相鄰下一次待共識的預處理塊對應的共識參數，該共識參數用來指示該區塊鏈節點對待共識的預處理塊進行共識驗證。

【發明圖式】

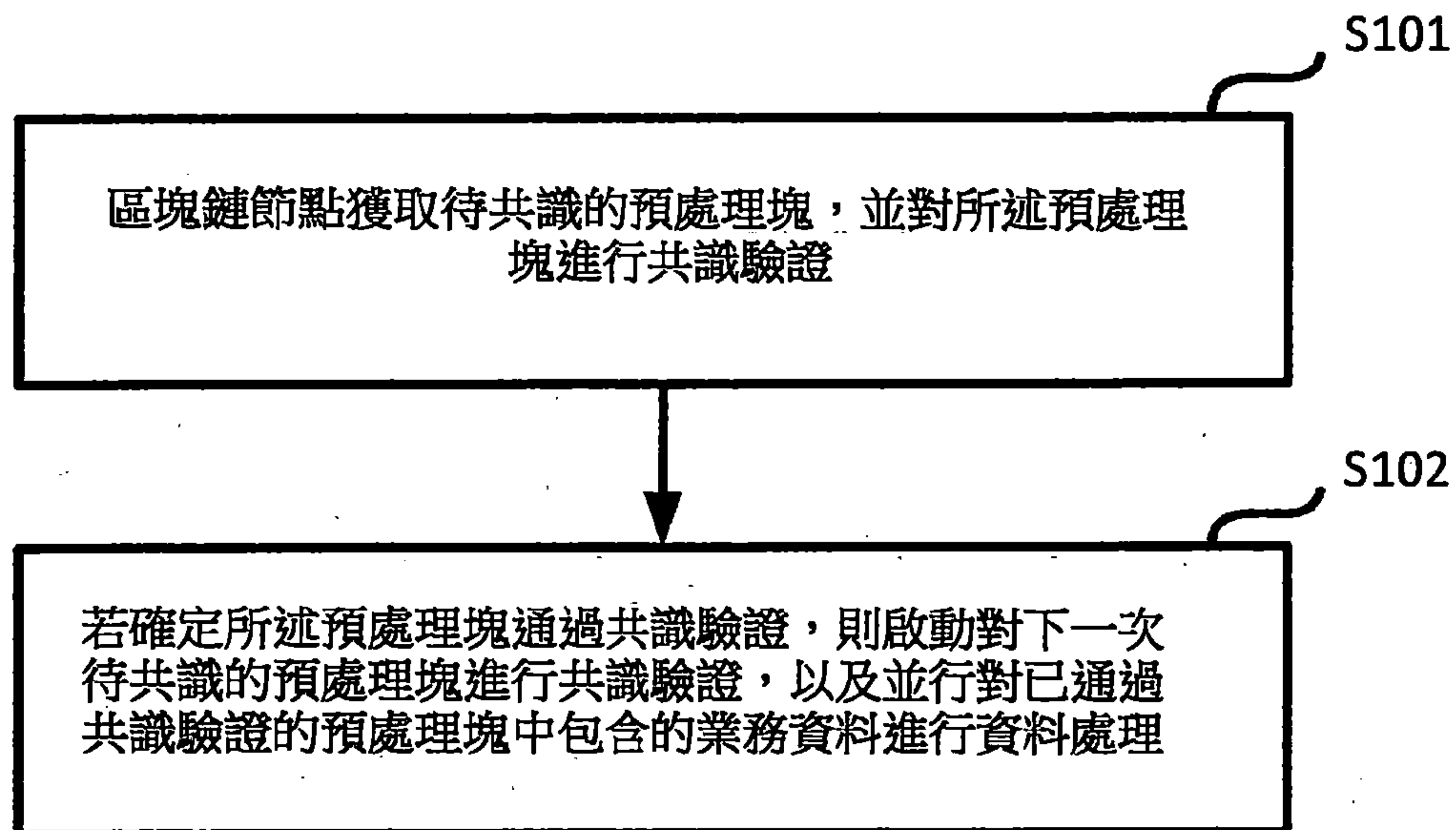


圖 1

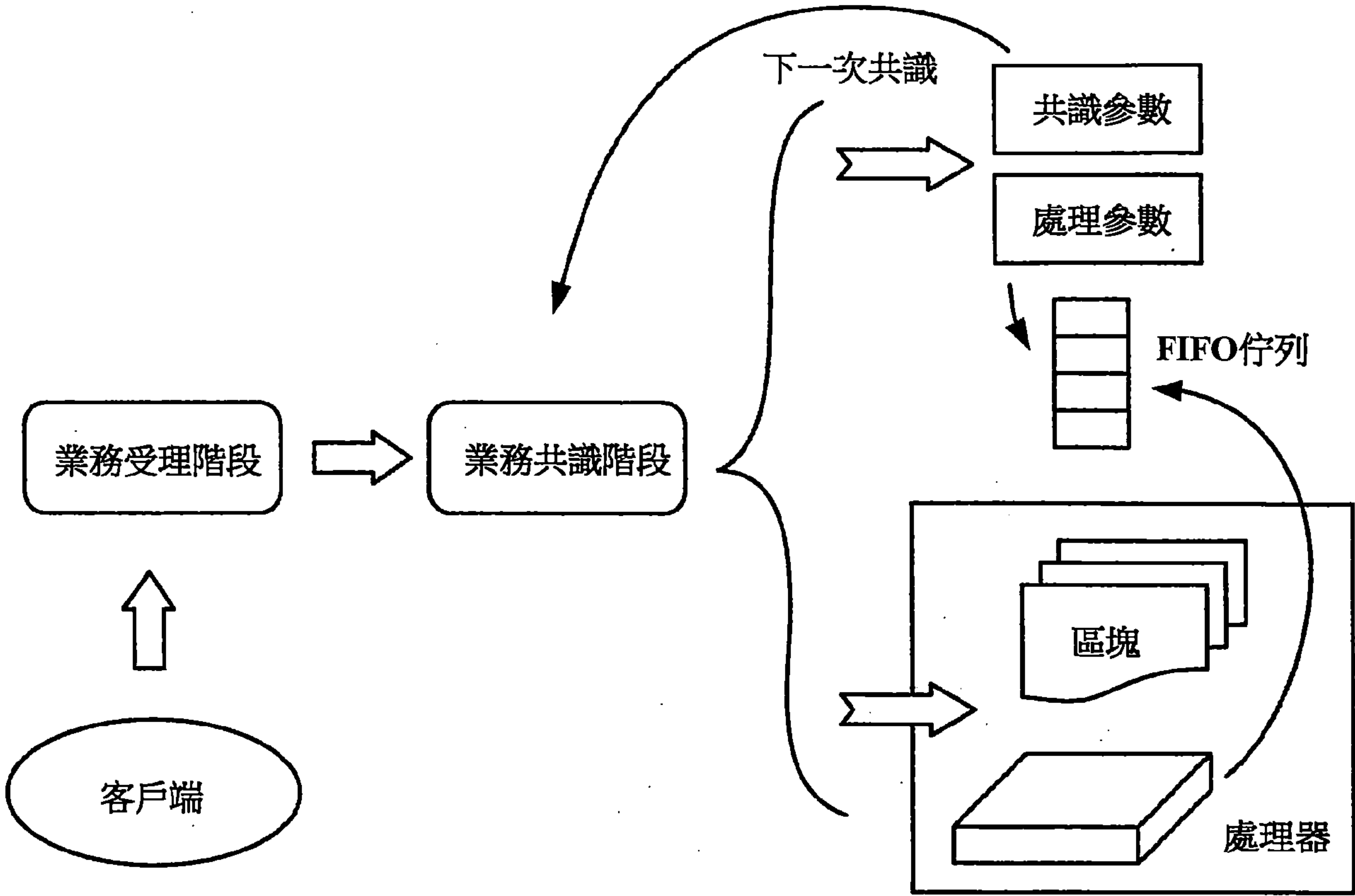


圖 2

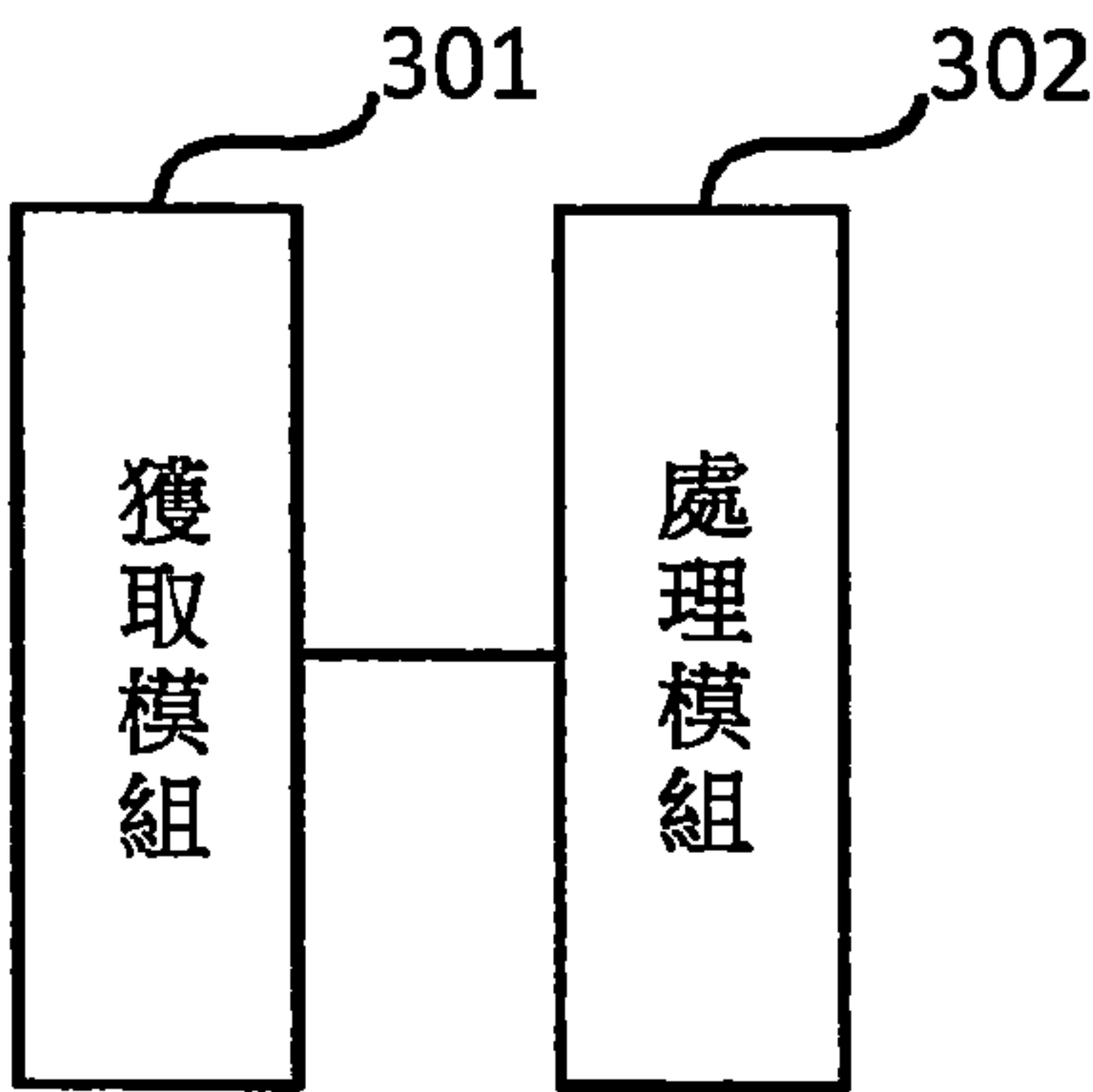


圖 3