



(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 I884989 B

(45)公告日：中華民國 114 (2025) 年 06 月 01 日

(21)申請案號：109133854

(22)申請日：中華民國 109 (2020) 年 09 月 29 日

(51)Int. Cl. : **G05B19/048 (2006.01)****G05B19/4155(2006.01)****B25J9/16 (2006.01)****G06F9/44 (2018.01)**

(30)優先權：2020/07/09 美國

16/924,910

2020/07/10 美國

16/925,544

2020/09/22 世界智慧財產權組織

PCT/US20/51940

(71)申請人：美商尤帕斯公司 (美國) UIPATH, INC. (US)

美國

(72)發明人：丁恩斯 丹尼爾 DINES, DANIEL (US)

(74)代理人：閻啓泰；林景郁

(56)參考文獻：

TW 201608395A

CN 105934723B

CN 107682368A

US 2020/0169933A1

US 2020/0180155A1

審查人員：施孝欣

申請專利範圍項數：20 項 圖式數：15 共 72 頁

(54)名稱

用於經由在第二通信期中運行的機器人程序自動化機器人使在第一通信期中運行的程序自動化的電腦實施方法

(57)摘要

記載了經由在第二通信期中運行的(一個或多個)機器人程序自動化(RPA)機器人來自動化在第一通信期中運行的程序。在一些方面中，表單被顯示在使用者通信期中，但針對第一通信期中的應用程式來獲取資料和/或與資料互動的一個或多個有人值守的 RPA 機器人在一個或多個其他通信期中運行。以這種方式，當(一個或多個)RPA 機器人正在運行時，(一個或多個)RPA 機器人的操作可以不阻止使用者使用其他應用程式或應用實例，但由(一個或多個)RPA 機器人進行或促進的資料修改可以在第一通信期視窗中對使用者為可見。

Automation of a process running in a first session via robotic process automation (RPA) robot(s) running in a second session is disclosed. In some aspects, a form is displayed in a user session, but one or more attended RPA robots that retrieve and/or interact with data for an application in the first session run in one or more other sessions. In this manner, the operation of the RPA robot(s) may not prevent the user from using other applications or instances when the RPA robot(s) are running, but the data modifications made or facilitated by the RPA robot(s) may be visible to the user in the first session window.

指定代表圖：

符號簡單說明：
1500:流程圖

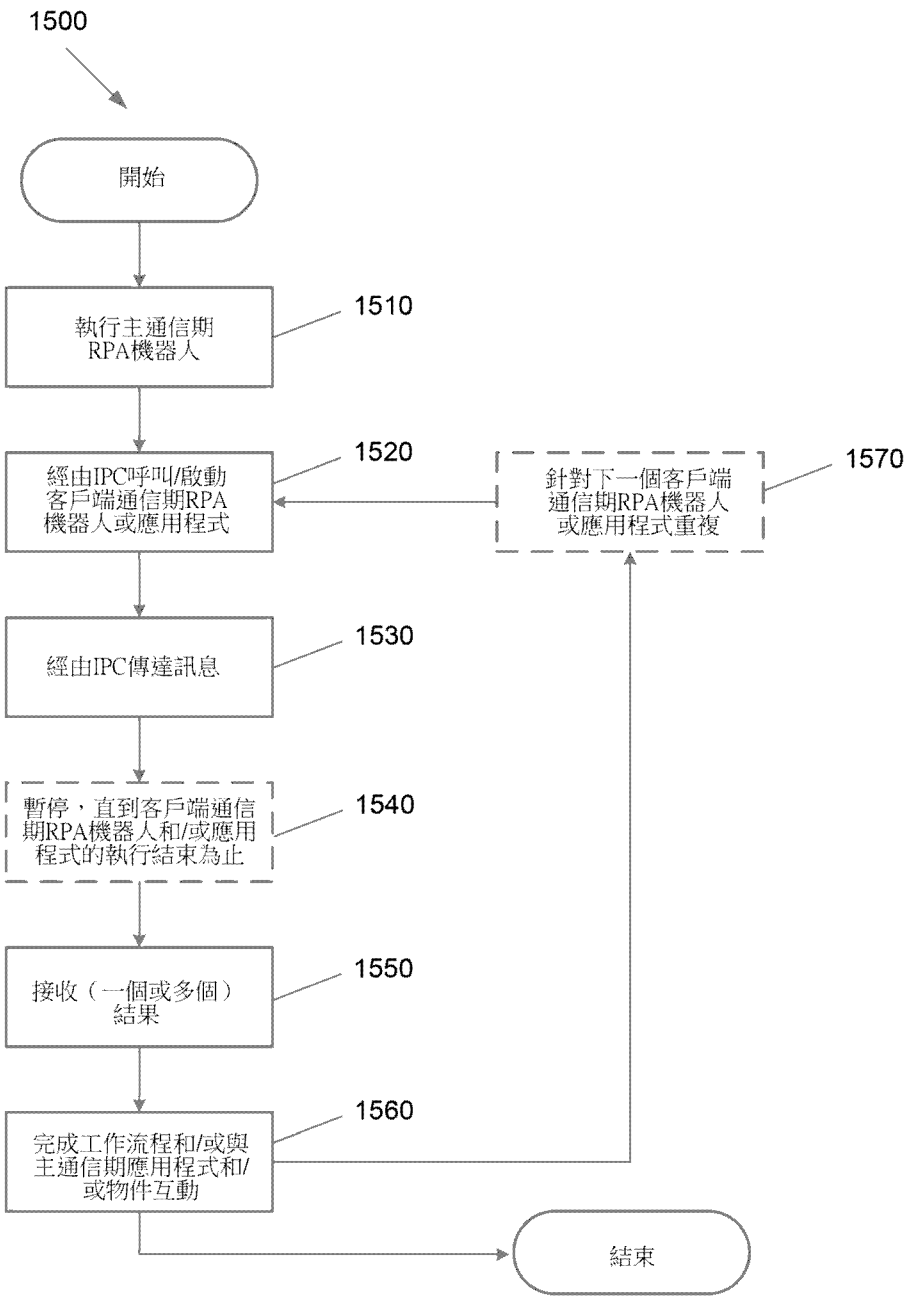


圖15



I884989

【發明摘要】

【中文發明名稱】 用於經由在第二通信期中運行的機器人程序自動化機器人使在第一通信期中運行的程序自動化的電腦實施方法

【英文發明名稱】 COMPUTER-IMPLEMENTED METHODS FOR
AUTOMATION OF A PROCESS RUNNING IN A FIRST
SESSION VIA A ROBOTIC PROCESS AUTOMATION
ROBOT RUNNING IN A SECOND SESSION

【中文】

記載了經由在第二通信期中運行的(一個或多個)機器人程序自動化(RPA)機器人來自動化在第一通信期中運行的程序。在一些方面中,表單被顯示在使用者通信期中,但針對第一通信期中的應用程式來獲取資料和/或與資料互動的一個或多個有人值守的RPA機器人在一個或多個其他通信期中運行。以這種方式,當(一個或多個)RPA機器人正在運行時,(一個或多個)RPA機器人的操作可以不阻止使用者使用其他應用程式或應用實例,但由(一個或多個)RPA機器人進行或促進的資料修改可以在第一通信期視窗中對使用者為可見。

【英文】

Automation of a process running in a first session via robotic process automation (RPA) robot(s) running in a second session is disclosed. In some aspects, a form is displayed in a user session, but one or more attended RPA robots that retrieve and/or interact with data for an application in the first session run in one or more other sessions. In this manner, the operation of the RPA robot(s) may not prevent the user from using other applications or instances when the RPA robot(s) are running, but the

data modifications made or facilitated by the RPA robot(s) may be visible to the user in the first session window.

【指定代表圖】 圖15

【代表圖之符號簡單說明】

1500: 流程圖

【特徵化學式】

無

【發明說明書】

【中文發明名稱】 用於經由在第二通信期中運行的機器人程序自動化機器人使在第一通信期中運行的程序自動化的電腦實施方法

【英文發明名稱】 COMPUTER-IMPLEMENTED METHODS FOR
AUTOMATION OF A PROCESS RUNNING IN A FIRST
SESSION VIA A ROBOTIC PROCESS AUTOMATION
ROBOT RUNNING IN A SECOND SESSION

【技術領域】

【0001】 本申請大體上涉及機器人程序自動化（RPA），且更具體地涉及經由在第二通信期（session）中運行的（一個或多個）RPA機器人使在第一通信期中運行的程序自動化。

相關申請案的交互參照

【0002】 本申請案主張於2020年7月9日提交的美國非臨時專利申請案第16/924,910號的優先權及作為其繼續申請案的、於2020年7月10日提交的美國非臨時專利申請案第16/925,544號的權益。這些早前提交申請案的主題以其整體內容通過引用併入於此。

【先前技術】

【0003】 例如，有人值守的自動化RPA機器人通常在與用戶所在的相同通信期中在由該使用者操作的計算系統上運行。例如，在用戶的命令下，RPA機器人可以與用戶一起工作完成某些任務。然而，在有人值守的自動化場景中，RPA機器人可以“接管”使用者的計算系統。在機器人與計算系統互動時，使用者可能希望執行其他活動，但用戶被阻止執行其他活動。換句話說，機器人以與使用者

相同的方式通過使用者介面（UI）來控制應用程式（例如模擬滑鼠點擊和鍵盤輸入）。

【0004】 存在創建作業系統或在其上運行的應用程式的完整或部分副本的各種技術。模擬器已經提出了數十年，且可以為開發者提供用於測試和調試應用程式的能力。例如，模擬器可以為開發者提供用於測試和調試行動應用程式的能力，該行動應用程式使用不支援直接運行開發工具的作業系統。Android®和iOS®都提供了可從開發機器運行以測試和調試Android®或iOS®應用程式的模擬器，這是由於開發者工具無法在那些行移動作業系統上本機運行。

【0005】 模擬器允許開發者在其本地機器上託管視窗，該本地機器讓開發者測試和調試難以或無法對開發機器執行的應用程式的行為。例如，為了測試和調試對這些事件做出回應的應用程式行為，模擬器允許開發者點擊按鈕以使模擬器旋轉，該按鈕向在模擬器內部運行的應用程式告知設備已旋轉。另一常見示例是多觸控。許多開發者機器不支援觸控，因此模擬器可以讓開發者測試和調試應用程式如何對多個觸控點做出回應。Android®和iOS®模擬器還提供模擬能力。此外，Microsoft®為其通用Windows平臺（UWP）應用程式提供模擬器。

【0006】 虛擬機器在機器上託管第二作業系統，並且可以通過視窗被打開並且被監測。如此運行完全不同的作業系統並且與主機共用硬體。“訪客”機器必須安裝其自身的應用程式副本，並且不能與用戶機器共用公共資源或檔案。

【0007】 Docker®容器在概念上為虛擬機器的混合形式。需要執行的所有應用程式都被封裝於不可變封包中，該不可變封包在主機作業系統上直接被執行。封包不是另一作業系統的完整副本，但該封包不預設共用或無法存取主機上的任何應用程式或資源。因此，從用戶體驗的角度看，Docker®容器感覺上類似於虛擬機器，但從技術上講，這些容器並未在完全獨立的作業系統上執行。

【0008】 然而，提供作業系統（OS）級別虛擬化的常規模擬器、仿真器、

虛擬機器（VM）和混合VM（例如Docker®容器）不能解決在與使用者相同的計算系統上操作的有人值守的自動化機器人出現的問題。因此，使用者實質上變成其自身的計算系統的旁觀者，其監視機器人工作但無法與在機器上的需要使用者互動的其他應用程式互動。因此，一種改進的方法可以是有益的。

【發明內容】

【0009】 本發明的某些實施例可以提供對本領域中尚未經當前RPA技術完全標識、理解或解決的問題和需求的解決方案。例如，本發明的一些實施例涉及經由在第二通信期中運行的（一個或多個）RPA機器人使在第一通信期中運行的程序自動化。

【0010】 在實施例中，一種電腦程式被實施在非暫態電腦可讀取媒體上。該電腦程式被配置為使至少一個處理器：在第一通信期中執行程序間通信（IPC）促進器和RPA驅動程式；以及在第二通信期中執行RPA機器人。該電腦程式還被配置為使至少一個處理器：由IPC促進器經由IPC從RPA機器人接收一個或多個訊息；以及由IPC促進器基於從RPA機器人接收到的一個或多個訊息，來控制RPA驅動程式以與在第一通信期中運行的應用程式或應用程式物件進行互動。

【0011】 在另一實施例中，一種電腦實施方法包括：由在第二通信期中運行的RPA機器人經由IPC向在第一通信期中運行的IPC促進器發送一個或多個訊息；以及由IPC促進器經由IPC從RPA機器人接收一個或多個訊息。該電腦實施方法還包括：由IPC促進器基於從RPA機器人接收到的一個或多個訊息，來控制在第一通信期中運行的RPA驅動程式以與也在第一通信期中運行的應用程式或應用程式物件進行互動。

【0012】 在又一實施例中，一種電腦實施方法包括：由在第一通信期中運行的IPC促進器經由IPC從在第二通信期中運行的RPA機器人接收一個或多個訊

息。該電腦實施方法還包括：由IPC促進器基於從RPA機器人接收到的一個或多個訊息，來控制在第一通信期中運行的RPA驅動程式以與也在第一通信期中運行的應用程式或應用程式物件進行互動。

【圖式簡單說明】

【0013】 為了容易地理解本發明的某些實施例的優點，將通過參考在附圖中圖示的特定實施例來呈現上文簡要描述的本發明的更具體的描述。雖然應理解，這些圖僅描繪了本發明的典型實施例，並且因此不應該被認為是對本發明的範圍的限制，但是本發明將通過附圖的使用以附加的特異性和細節來描述和闡釋，在附圖中：

【0014】 [圖1]是圖示了根據本發明的實施例的機器人程序自動化（RPA）系統的架構圖。

【0015】 [圖2]是圖示了根據本發明的實施例的部署的RPA系統的架構圖。

【0016】 [圖3]是圖示了根據本發明的實施例的設計器、活動與驅動程式之間的關係的架構圖。

【0017】 [圖4]是圖示了根據本發明的實施例的RPA系統的架構圖。

【0018】 [圖5]是圖示了根據本發明的實施例的被配置為促進用於RPA機器人的通信期間（inter-session）自動化的計算系統的架構圖。

【0019】 [圖6A-G]圖示了根據本發明的實施例的使用在機器人通信期中運行的RPA機器人、IPC促進器以及RPA驅動程式在用戶通信期中完成表單的示例。

【0020】 [圖7A-G]圖示了根據本發明的實施例的使用在機器人通信期中運行的RPA機器人經由直接變數修改在使用者通信期中完成表單的示例。

【0021】 [圖8]是圖示了根據本發明的實施例的經由在機器人通信期中運

行的RPA機器人、IPC促進器以及RPA驅動程式使在用戶通信期中運行的程序自動化的程序的流程圖。

【0022】 [圖9]是圖示了根據本發明的實施例的經由直接變數修改經由在機器人通信期中運行的RPA機器人使在用戶通信期中運行的程序自動化的程序的流程圖。

【0023】 [圖10]圖示了根據本發明的實施例的與主通信期應用程式互動的多個用戶端通信期RPA機器人的示例。

【0024】 [圖11]是圖示了根據本發明的實施例的用於在用戶端通信期中運行有人值守的自動化RPA機器人的程序的流程圖。

【0025】 [圖12]是圖示了根據本發明的實施例的在主通信期機器人M1、用戶端通信期應用程式A1與一對用戶端通信期機器人C1和C2之間的多機器人集體工作流程的執行的流程圖。

【0026】 [圖13]是圖示了根據本發明的實施例的並行地執行主通信期RPA機器人和用戶端通信期RPA機器人的流程圖。

【0027】 [圖14]是圖示了根據本發明的實施例的基於針對主通信期應用程式的觸發來執行用戶端通信期RPA機器人的流程圖。

【0028】 [圖15]是圖示了根據本發明的實施例的用於執行通信期間自動化的程序的流程圖。除非另有指示，否則類似的元件符號在整個附圖中始終指示對應特徵。

【0029】 除非另有指示，否則類似的元件符號在整個附圖中始終指示對應特徵。

【實施方式】

【0030】 一些實施例涉及經由在第二通信期中運行的（一個或多個）RPA

機器人使在第一通信期中運行的程序自動化。例如，表單（例如Web頁面、電子郵件應用程式（例如Outlook[®]）、試算表應用程式（例如Excel[®]）、客戶關係管理（CRM）系統應用程式（例如Salesforce[®]）、企業資源管理（ERM）系統、供應鏈管理系統、定制電腦應用程式、行動應用程式、具有可填寫欄位的另一應用程式等）可以被顯示於第一通信期中。在一些實施例中，第一通信期可以是用戶通信期（在本文中也被稱作主通信期或父通信期）。在第一通信期中獲取應用程式的資料和/或與應用程式的資料互動的一個或多個RPA機器人可以在一個或多個其他通信期（在本文中也被稱作機器人通信期、第二通信期或輔助通信期）中運行。在某些實施例中，機器人通信期可以是第一通信期的子通信期。

【0031】 在一些實施例中，在用戶端通信期中運行的RPA機器人經由程序間通信（IPC）向在主通信期中運行的RPA驅動程式發送訊息。基於來自RPA機器人的（一個或多個）IPC通信，RPA驅動程式可以實現與計算系統進行各種作業系統（OS）和/或應用程式設計介面（API）級別互動。例如，驅動程式可以在不脫離本發明的範圍的情況下移動滑鼠、點擊按鈕、與選單互動、在欄位中輸入文字、打開或關閉視窗、移動視窗和/或調整視窗大小或執行任何其他合適的互動。

【0032】 為了實現該功能性，驅動程式可以包括IPC促進器或受IPC促進器控制或由IPC促進器調用。在一些實施例中，IPC促進器可以是單獨的應用程式或程序，或可以是驅動程式的子程序。IPC促進器可以在用戶端通信期中監聽來自RPA機器人的IPC訊息（例如監聽觸發）、向RPA機器人發送通信和從RPA機器人接收通信、監測RPA機器人執行狀態、其組合等。IPC促進器還可以基於來自RPA機器人的（一個或多個）IPC通信來使驅動程式在父通信期中實現期望的功能性。在某些實施例中，IPC促進器可以配置有針對給定RPA機器人的腳本。在一些實施例中，IPC促進器可以被定制設計用於要在父通信期中執行的功能

性。

【0033】 在一些實施例中，用戶端通信期中的RPA機器人可以直接修改儲存在計算系統上的記憶體中的變數。在父通信期中運行的存取這些變數的應用程式然後可以在顯示刷新或應用程式以其他方式重新載入其UI時更新所顯示的值。例如，可以由在用戶端通信期中運行的RPA機器人直接修改Excel®試算表的變數，這也會改變在父通信期中顯示的值。

【0034】 在一些實施例中，當RPA機器人正在運行時，（一個或多個）RPA機器人的操作可以不阻止使用者使用其他應用程式或應用實例，但當應用程式顯示在第一使用者通信期視窗中刷新時，由（一個或多個）RPA機器人進行的資料修改可以對使用者為可見。然而，在某些實施例中，（一個或多個）RPA機器人可以與不具有使用者介面的應用程式互動。

【0035】 在一些實施例中，程序可以操作如下。使用者可以點擊按鈕或以其他方式使RPA機器人在使用者通信期（例如在使用者開啟其計算系統時啟動的父通信期或主通信期）的子通信期中啟動。在某些實施例中，例如，機器人通信期可能已經在運行中或可以由用戶從機器人工具列（tray）單獨啟動。用戶可以使機器人在子通信期中執行其工作流程，然後機器人可以通過與在父通信期中運行的IPC促進器進行通信來與主通信期中的一個或多個應用程式互動。IPC促進器然後可以使RPA驅動程式在父通信期中實施期望的功能性。

【0036】 在一些實施例中，如果尚未啟動工作流程所屬的至少一個應用程式，那麼在機器人通信期中運行的RPA機器人可以暫停操作並且提供訊息，並且IPC促進器可以啟動該應用程式，以便機器人可以繼續操作。一旦機器人開始執行其工作流程，當經由IPC促進器和RPA驅動程式進行此類改變時，應用程式中的所顯示的資料的改變可以在主通信期視窗中可見。例如，資料可以在驅動程式基於由RPA機器人在用戶端通信期中提供的資料（例如經由將資料保存到文字檔

或資料庫中的IPC等)來填寫資料時而顯現。

【0037】 作為非限制性示例，在一些實施例中，可以在Salesforce®中創建快捷方式，該快捷方式使RPA機器人在用戶端通信期中啟動，或在用戶端通信期中與已運行的RPA機器人互動。當使用者點擊快捷方式時，RPA機器人如在其RPA工作流程中所設計的一般開始執行RPA程序。基於其執行的結果(例如從資料庫中獲取資訊、調用人工智慧(AI)/機器學習(ML)模型並且獲得結果、從多個舊系統中獲得資訊等)，RPA機器人然後與IPC促進器互動，該IPC促進器然後在父通信期中與運行的Salesforce®應用實例互動。例如，這可以導致，在RPA驅動程式執行互動時，正在運行的Salesforce®應用實例中的一些所顯示的資訊發生改變並且對使用者為可見。

【0038】 在某些實施例中，在用戶端通信期中運行的RPA機器人可以使父通信期中的應用程式經由IPC促進器打開。IPC促進器然後可以經由RPA驅動程式存取與應用程式相關聯的資料並且進行添加、改變和/或刪除。以這種方式，即使RPA機器人並未在父通信期中運行，RPA機器人仍然能夠以看起來類似於在父通信期本身中運行的RPA機器人的方式在父通信期應用程式中引起互動，但在功能上有所不同。

【0039】 在一些實施例中，用戶可以使機器人在用戶端通信期中執行存取網站並收集一些資訊的工作流程。例如，RPA機器人然後可以將該資訊提供給IPC促進器，該IPA促進器然後使RPA驅動程式在試算表中輸入所收集的資訊，該試算表在主通信期中為可見。在某些實施例中，應用程式(諸如Salesforce®)在主通信期中打開。用戶然後在讀取當前用戶端ID的用戶端通信期中運行自動化，進入網站(例如HubSpot®)，並且收集與同網站的用戶端互動相關的資訊。然而，在不脫離本發明的範圍的情況下，有人值守的RPA機器人可以經由RPA促進器與任何合適的(一個或多個)應用程式互動和/或可以從任何合適的來源(例如資

料庫、另一應用程式等)獲得資料。

【0040】 一些實施例的應用包括但不限於提供OS級虛擬化的模擬器、模擬器、VM和混合VM(例如Docker®容器)。一些實施例將一個或多個機器人通信期創建並且託管為視窗形式，該視窗包括由有人值守的自動化程序控制的應用程式的UI。在某些實施例中，僅示出了(一個或多個)機器人與之互動的應用程式的介面。在一些實施例中，根本沒有示出用戶端通信期窗口。如本文中所使用的，在不脫離本發明的範圍的情況下，“視窗”可以應用於代表在主UI內所示出的UI的視窗、計算系統的第二顯示器的第二螢幕、虛擬桌面、隔離環境(即，視窗(被稱為“主機”)等，其繪製在環境內部啟動的所有應用程式的UI(被稱為“子”)並且在主機通信期的文本中運行它們。在用戶與第一通信期(例如父通信期)互動時，運行多個通信期允許(一個或多個)機器人在其(一個或多個)通信期中操作。因此，在機器人與其資料互動時，使用者可能能夠與各種應用程式互動。

【0041】 在一些實施例中，在不脫離本發明的範圍的情況下，可以創建和使用用於任何數量的機器人的任何期望數量的通信期。例如，用戶可以在第一通信期中操作，第一機器人可以在第二通信期中操作，第二機器人可以在第三通信期中操作等。在某些實施例中，多個機器人可以在單個通信期中操作，潛在地輪流經由RPA促進器以與一個或多個共用應用程式互動。

【0042】 例如，可以經由Windows®終端服務子通信期實施用於創建通信期的功能性，該Windows®終端服務子通信期可以在無需使用者登出的情況下將通信期創建回用戶自身的機器中。新創建的通信期顯現為子視窗，且包含並啟動存在於用戶通信期中的應用程式。換句話說，在用戶與機器人之間的分離發生在UI級別。例如，如果檔案被刪除，那麼這會在計算系統上運行的所有通信期中發生。

【0043】 某些實施例可以被部署用於機器人程序自動化(RPA)。圖1是圖

示了根據本發明的實施例的RPA系統100的架構圖。RPA系統100包括允許開發者設計和實現工作流程的設計器110。設計器110可以提供用於應用程式整合以及自動化第三方應用程式、管理資訊技術（IT）任務和事業IT程序的解決方案。設計器110可以支援自動化專案的開發，該自動化專案是事業程序的圖形表示。簡而言之，設計器110支援工作流程和機器人的開發和部署。

【0044】 自動化專案通過向開發者提供對在工作流程中開發的一組定制步驟（在本文中被定義為“活動”）之間的執行順序和關係的控制來實現基於規則的程序的自動化。設計器110的實施例的一個商業示例是UiPath Studio™。每個活動可以包括動作，諸如點擊按鈕、讀取檔案、寫入日誌面板等。在一些實施例中，工作流程可以被嵌套或嵌入。

【0045】 一些類型的工作流程可以包括但不限於序列、流程圖、FSM和/或全域例外處理常式。序列可能特別適合於線性程序，從而能夠在不使工作流程混亂的情況下從一個活動流向另一活動。流程圖可能特別適合於更複雜的事業邏輯，從而實現通過多個分支邏輯運算子以更為不同方式進行的決策整合以及活動連接。FSM可能特別適合於較大工作流程。FSM可以在其執行中使用有限數量個狀態，這些狀態由條件（即，過渡）或活動觸發。全域例外處理常式可能特別適合於在遇到執行錯誤時確定工作流程行為且特別適合於調試程序。

【0046】 一旦工作流程在設計器110中被開發，事業程序的執行就由指揮器（conductor）120編排，該指揮器編排執行設計器110中所開發的工作流程的一個或多個機器人130。指揮器120的實施例的一個商業示例是UiPath Orchestrator™。指揮器120支援對環境中的資源的創建、監測以及部署的管理。利用第三方解決方案和應用程式，指揮器120可以充當整合點、或充當聚合點中的一者。

【0047】 指揮器120可以管理一隊機器人130，從而從集中點連接並執行機

器人130。可以管理的機器人130的類型包括但不限於：有人值守的機器人132、無人值守的機器人134、開發機器人（類似於無人值守的機器人134，但用於開發和測試目的）以及非生產型機器人（類似於有人值守的機器人132，但用於開發和測試目的）。有人值守的機器人132可由用戶事件觸發或被安排為自動發生，並且與人類一起在相同計算系統上操作。有人值守的機器人132可以與指揮器120一起用於集中程序部署和記錄媒體。有人值守的機器人132可以幫助人類用戶完成各種任務，並且可以由用戶事件觸發。在一些實施例中，無法從這種類型的機器人上的指揮器120開啟程序和/或這些程序無法在鎖定螢幕下運行。在某些實施例中，有人值守的機器人132只可以從機器人工具列或根據命令提示而被開啟。在一些實施例中，有人值守的機器人132應該在人類監督下運行。

【0048】 無人值守的機器人134在虛擬環境中以無人值守的方式運行並且可以使許多程序自動化。無人值守的機器人134可以負責遠端執行、監測、排程並且為工作隊列提供支援。在一些實施例中，可以從設計器110運行針對所有機器人類型的調試。有人值守的機器人和無人值守的機器人兩者都可以使各種系統和應用程式自動化，該系統和應用程式包括但不限於大型主機、Web應用程式、VM、企業應用程式（例如由SAP®、SalesForce®、Oracle®等生產的那些企業應用程式）以及計算系統應用程式（例如桌上型電腦和膝上型電腦應用程式、行動設備應用程式、可穿戴電腦應用程式等）。

【0049】 指揮器120可以具有各種能力，包括但不限於供應、部署、版本化、配置、排隊、監測、日誌記錄和/或提供互連性。供應可以包括在機器人130與指揮器120（例如Web應用程式）之間創建和維護連接。開發可以包括確保將封包（package）版本正確遞送至所指派的機器人130以供執行。在一些實施例中，版本化可以包括某一程序或配置的唯一應用實例的管理。配置可以包括維護和遞送機器人環境和程序配置。排隊可以包括提供對佇列和佇列項的管理。監測可

以包括追蹤機器人標識資料和維護使用者許可權。日誌記錄可以包括將日誌儲存和編入索引至資料庫（例如SQL資料庫）和/或另一儲存機制（例如ElasticSearch[®]，其提供用於儲存和快速查詢較大資料集的能力）。指揮器120可以通過充當第三方解決方案和/或應用程式的通信的集中點來提供互連性。

【0050】 機器人130是運行內置於設計器110中的工作流程的執行代理器。（一個或多個）機器人130的一些實施例的一個商業示例是UiPath Robots™。在一些實施例中，機器人130預設安裝Microsoft Windows[®]服務控制管理器（SCM）管理的服務。結果，此類機器人130可以在本地系統帳號下打開互動式Windows[®]通信期，並且具有Windows[®]服務的許可權。

【0051】 在一些實施例中，機器人130可以在使用者模式下被安裝。針對此類機器人130，這意謂其具有與已經安裝了給定機器人130的用戶相同的許可權。此特徵也可以供高密度（HD）機器人使用，該高密度機器人確保了每台機器以其最大潛力的充分利用。在一些實施例中，任何類型的機器人130可以在HD環境中被配置。

【0052】 在一些實施例中，機器人130被劃分成分別專用於特定自動化任務的若干組件。在一些實施例中，機器人元件包括但不限於SCM管理的機器人服務、使用者模式機器人服務、執行器、代理器以及命令列。SCM管理的機器人服務管理和監測Windows[®]通信期，並且充當指揮器120與執行主機（即，機器人130在其上被執行的計算系統）之間的代理伺服器（proxy）。這些服務受機器人130信任並且管理機器人130的憑證。控制台應用程式在本地系統下由SCM啟動。

【0053】 在一些實施例中，使用者模式機器人服務管理和監測Windows[®]通信期，並且充當在指揮器120與執行主機之間的代理伺服器。使用者模式機器人服務可以受機器人130信任並且管理機器人130的憑證。在未安裝SCM管理的機器人服務的情況下，Windows[®]應用程式可以被自動啟動。

【0054】 執行器可以在Windows®通信期下運行給定作業(即,執行器可以執行工作流程。執行器可以知道每個監測器的每英寸點數(DPI)設置。代理器可以是Windows® Presentation Foundation (WPF)應用程式,其在系統工具列視窗中顯示可用的作業。代理器可以是服務的用戶端。代理器可以請求開啟或停止作業以及改變設置。命令列是服務的用戶端。命令列是可以請求開啟作業並等待其輸出的控制台應用程式。

【0055】 如上文所闡釋一般使機器人130的元件分開說明開發者、支援使用者以及計算系統更容易地運行、標識和追蹤每個元件正在執行的內容。特殊行為可以以這種方式按元件來配置,諸如針對執行器和服務設置不同的防火牆規則。在一些實施例中,執行器可以始終知道每個監測器的DPI設置。結果,不論創建工作流程的計算系統的配置如何,工作流程都可以在任何DPI處被執行。在一些實施例中,來自設計器110的項目也可以獨立於流覽縮放級別。在一些實施例中,針對DPI不知道的或有意地標記為不知道的應用程式,DPI可以被禁用。

【0056】 圖2是圖示了根據本發明的實施例的部署的RPA系統200的架構圖。在一些實施例中,RPA系統200可以是圖1的RPA系統100或可以是該RPA系統的一部分。應注意,在不脫離本發明的範圍的情況下,用戶端側、伺服器側或兩者都可以包括任何期望數量的計算系統。在用戶端側,機器人應用程式210包括執行器212、代理器214和設計器216。然而,在一些實施例中,設計器216可以不在計算系統210上運行。執行器212正在運程序。如圖2中所示,若干事業專案可以同時運行。在該實施例中,代理器214(例如Windows®服務)是所有執行器212的單個聯繫點。該實施例中的所有訊息被日誌記錄到指揮器230中,該指揮器經由資料庫伺服器240、索引子伺服器250或兩者進一步處理這些訊息。如上文相對於圖1所討論的,執行器212可以是機器人元件。

【0057】 在一些實施例中,機器人表示機器名稱與用戶名之間的關聯。機

器人可以同時管理多個執行器。在支援同時運行的多個互動式通信期的計算系統（例如Windows®伺服器2012）上，多個機器人可以同時運行，每個機器人在使用唯一用戶名的單獨Windows®通信期中。上文將此稱為HD機器人。

【0058】 代理器214還負責發送機器人的狀態（例如週期性地發送指示機器人仍在運作的“心跳”訊息）並且下載要被執行的封包的所需版本。在一些實施例中，代理器214與指揮器230之間的通信始終由代理器214來起始。在通知場景中，代理器214可以打開WebSocket通道，稍後由指揮器230使用該WebSocket通道來向機器人發送命令（例如開啟、停止等）。

【0059】 在伺服器側，包括了呈現層（Web應用程式232、開放資料協定（OData）代表性狀態轉移（REST）的應用程式設計介面（API）端點234以及通知和監測API 236）、服務層（API實施/事業邏輯238）以及持久層（資料庫伺服器240和索引子伺服器250）。指揮器230包括Web應用程式232、OData REST API端點234、通知和監測API 236以及API實施/事業邏輯238。在一些實施例中，用戶在指揮器230的介面中（例如經由流覽器220）執行的大多數動作是通過調用各種API來執行的。在不脫離本發明的範圍的情況下，此類動作可以包括但不限於在機器人上開啟作業、在佇列中添加/移除資料、排程作業以便以無人值守的方式運行等。Web應用程式232是伺服器平臺的視覺層。在該實施例中，Web應用程式232使用超文字標記語言（HTML）和JavaScript（JS）。然而，可以在不脫離本發明的範圍的情況下使用任何期望標記語言、指令碼語言或任何其他格式。在該實施例中，用戶經由流覽器220與來自Web應用程式232的Web頁面互動，以便執行各種動作來控制指揮器230。例如，用戶可以創建機器人群組、向機器人指派封包、分析每個機器人和/或每個程序的日誌、開啟和停止機器人等。

【0060】 除了Web應用程式232之外，指揮器230還包括暴露OData REST的API端點234的服務層。然而，在不脫離本發明的範圍的情況下可以包括其他端

點。REST API由Web應用程式232和代理器214兩者消耗。在該實施例中，代理器214是用戶端電腦上的一個或多個機器人的監督器。

【0061】 在該實施例中，REST API覆蓋配置、日誌記錄、監測以及排隊功能性。在一些實施例中，配置端點可以用於定義和配置應用程式的使用者、許可權、機器人、資產、發佈以及環境。日誌記錄REST端點可以用於日誌記錄不同資訊，諸如（例如）錯誤、由機器人發送的顯式訊息以及其他環境特定資訊。如果在指揮器230中使用了開啟作業命令，那麼可以由機器人使用部署REST端點來查詢應該被執行的封包版本。排隊REST端點可以負責佇列和佇列項管理，諸如將資料添加至佇列、從佇列中獲得異動、設置異動的狀態等。

【0062】 監測REST端點可以監測Web應用程式232和代理器214。通知和監測API 236可以用於註冊代理器214、向代理器214遞送配置設置並且用於從伺服器 and 代理器214發送/接收通知的REST端點。在一些實施例中，通知和監測API 236還可以使用WebSocket通信。

【0063】 在該實施例中，持久層包括一對伺服器——資料庫伺服器240（例如SQL伺服器）和索引子伺服器250。在該實施例中，資料庫伺服器240儲存機器人、機器人群組、相關聯的程序、用戶、角色、排程等的配置。在一些實施例中，該資訊通過Web應用程式232來管理。資料庫伺服器240可以管理佇列和佇列項。在一些實施例中，資料庫伺服器240可以儲存由機器人（除了索引子伺服器250之外或代替索引子伺服器250）日誌記錄的訊息。

【0064】 在一些實施例中可選的索引子伺服器250儲存由機器人日誌記錄的資訊並且將該資訊編入索引。在某些實施例中，索引子伺服器250可以通過配置設置來禁用。在一些實施例中，索引子伺服器250使用ElasticSearch[®]，其為開源專案全文檢索搜尋引擎。由機器人日誌記錄的訊息（例如使用比如日誌訊息或寫入行的活動）可以通過（一個或多個）日誌記錄REST端點被發送到索引子伺

服器250，在索引伺服器250中這些訊息被編入索引以供將來使用。

【0065】 圖3是圖示了根據本發明的實施例的設計器310、活動320、330與驅動程式340之間的關係300的架構圖。根據上文，開發者使用設計器310來開發由機器人執行的工作流程。工作流程可以包括用戶定義的活動320和UI自動化活動330。一些實施例能夠標識圖像中的非文字視覺分量，在本文中稱作電腦視覺（CV）。與此類分量相關的一些CV活動可以包括但不限於點擊、鍵入、取得文字、懸停、元素存在、刷新範圍、高亮等。在一些實施例中，點擊使用例如CV、光學字元辨識（OCR）、模糊文字匹配以及多定錨（multi-anchor）來標識元素，並且點擊該元素。鍵入可以使用上述內容來標識元素和該元素中的類型。取得文字可以標識特定文字的位置並且使用OCR來掃描該位置。懸停可以標識元素並且懸停於元素上方。元素存在可以使用上文所描述的技術來檢查在螢幕上是否存在元素。在一些實施例中，可以存在在設計器310中可以被實施的數百個甚至數千個活動。然而，在不脫離本發明的範圍的情況下，任何數量和/或類型的活動都可以是可用的。

【0066】 UI自動化活動330是利用較低階代碼編寫的特殊較低階活動（例如CV活動）的子集並且有助於與螢幕的互動。UI自動化活動330經由允許機器人與期望軟體互動的驅動程式340來支援這些互動。例如，驅動程式340可以包括OS驅動程式342、流覽器驅動程式344、VM驅動程式346、企業應用程式驅動程式348等。

【0067】 驅動程式340可以與OS進行低階互動，進而尋找掛機、監測鍵入等。驅動程式可以有助於與Chrome[®]，IE[®]，Citrix[®]，SAP[®]等的整合。例如，“點擊”活動經由驅動程式340在這些不同應用程式中扮演相同角色。

【0068】 圖4是圖示了根據本發明的實施例的RPA系統400的架構圖。在一些實施例中，RPA系統400可以是或包括圖1和/或圖2的RPA系統100和/或200。

RPA系統400包括運行機器人的多個用戶端的計算系統410。計算系統410能夠經由在其上運行的Web應用程式與指揮器計算系統420進行通信。指揮器計算系統420又能夠與資料庫伺服器430和可選的索引子伺服器440進行通信。

【0069】 相對於圖1和圖3，應注意，雖然在這些實施例中使用了Web應用程式，但可以在不脫離本發明的範圍的情況下使用任何合適的用戶端/伺服器軟體。例如，指揮器可以運行伺服器側應用程式，該伺服器側應用程式與用戶端計算系統上的非基於網路的用戶端軟體應用程式進行通信。

【0070】 圖5是圖示了根據本發明的實施例的被配置為針對RPA機器人促進通信期間（inter-session）自動化的計算系統500的架構圖。在一些實施例中，計算系統500可以是本文中所描繪和/或描述的計算系統中的一者或多者。計算系統500包括匯流排505或用於傳達資訊的其他通信機制、以及被耦合至匯流排505以處理資訊的（一個或多個）處理器510。（一個或多個）處理器510可以是任何類型的通用或專用處理器，包括中央處理單元（CPU）、專用積體電路（ASIC）、現場可程式設計閘陣列（FPGA）、圖形處理單元（GPU）、其多個實例和/或其任何組合。（一個或多個）處理器510還可以具有多個處理核心，並且至少一些核心可以被配置為執行特定功能。在一些實施例中，可以使用多並行處理。在某些實施例中，（一個或多個）處理器510中的至少一者可以是包括模仿生物神經的處理元件的神經形態電路。在一些實施例中，神經形態電路可以不需要馮·諾依曼（Von Neumann）計算架構的典型組件。

【0071】 計算系統500還包括用於儲存將由（一個或多個）處理器510執行的資訊和指令的記憶體515。記憶體515可以由隨機存取記憶體（RAM）、唯讀記憶體（ROM）、閃速記憶體、快取記憶體、靜態儲存裝置（諸如磁片或光碟）或任何其他類型的非暫態電腦可讀取媒體或其組合的任何組合所組成。非暫態電腦可讀取媒體可以是由（一個或多個）處理器510可以存取的任何可用媒體，並

且可以包括揮發性媒體、非揮發性媒體或兩者。媒體也可以是可移除的、不可移除的或兩者。

【0072】 附加地，計算系統500包括通信設備520（諸如收發器）以經由無線和/或有線連接提供對通信網路的存取。在一些實施例中，在不脫離本發明的範圍的情況下，通信設備520可以被配置為使用分頻多存取（FDMA）、單載波 FDMA（SC-FDMA）、分時多存取（TDMA）、分碼多存取（CDMA）、正交分頻多工（OFDM）、正交分頻多存取（OFDMA）、全球行動系統（GSM）通信、通用封包無線電服務（GPRS）、全球行動電信系統（UMTS）、CDMA2000、寬頻 CDMA（W-CDMA）、高速下行封包存取（HSDPA）、高速上行封包存取（HSUPA）、高速封包存取（HSPA）、長期演進（LTE）、LTE高階（LTE-A）、802.11x、Wi-Fi、Zigbee、超寬頻（UWB）、802.16x、802.15、家用節點B（HnB）、藍牙、射頻標識（RFID）、紅外資料協會（IrDA）、近場通信（NFC）、第五代通訊（5G）、新無線電（NR）、其任何組合和/或任何其他現有或將來實施通信標準和/或協議。在一些實施例中，在不脫離本發明的範圍的情況下，通信設備520可以包括一個或多個天線，這些天線是單數的、陣列的、相控的、切換的、波束成形的、波束導向的、其組合和/或任何其他天線配置。

【0073】 （一個或多個）處理器510進一步經由匯流排505被耦合到顯示器525，諸如電漿顯示器、液晶顯示器（LCD）、發光二極體（LED）顯示器、場發射顯示器（FED）、有機發光二極體（OLED）顯示器、可撓性OLED顯示器、可撓性基板顯示器、投影顯示器、4K顯示器、高解析顯示器、Retina[®]顯示器、面內切換（IPS）顯示器或用於向使用者顯示資訊的任何其他合適的顯示器。顯示器525可以被配置為使用電阻性、電容性、表面聲波（SAW）電容性、紅外、光學成像、色散信號技術、聲學脈衝識別、受抑（frustrated）全內反射等的觸控（觸覺）顯示器、三維（3D）觸控顯示器、多輸入觸控顯示器、多觸控顯示器等。可

以在不脫離本發明的範圍的情況下使用任何合適的顯示裝置和觸覺I/O。

【0074】 鍵盤530和游標控制設備535（諸如電腦滑鼠、觸控板等）進一步被耦合到匯流排505以使得使用者能夠與計算系統500介接。然而，在某些實施例中，可以不存在實體鍵盤和滑鼠，並且使用者可以僅通過顯示器525和/或觸控板（未示出）與設備互動。輸入裝置的任何類型和組合可以被用作設計選擇的問題。在某些實施例中，不存在實體輸入裝置和/或顯示器。例如，用戶可以經由與之通信的另一計算系統與計算系統500遠端地互動，或計算系統500可以自主地操作。

【0075】 記憶體515儲存在被（一個或多個）處理器510執行時提供功能性的軟體模組。模組包括計算系統500的作業系統540。模組還包括被配置為執行本文中所描述的程序或其衍生物的全部或部分的通信期間自動化模組545。計算系統500可以包括一個或多個附加功能性模組550，其包括附加功能性。

【0076】 本領域的技術人員應瞭解，在不脫離本發明的範圍的情況下，“系統”可以被實施為伺服器、嵌入式計算系統、個人電腦、控制台、個人數位助理（PDA）、手機、平板計算設備、量子計算系統或任何其他合適的計算設備或設備的組合。將上文所描述的功能呈現為由“系統”執行並不旨在以任何方式限制本發明的範圍，而是旨在提供本發明的許多實施例的一個示例。實際上，本文中所記載的方法、系統和裝置可以按照與包括雲計算系統的計算技術一致的當地語系化和分散式的形式來實現。

【0077】 應注意，本說明書中所描述的一些系統特徵已經作為模組而被呈現，以便更具體地強調其實現獨立性。例如，模組可以被實現為包括定制的超大規模積體（VLSI）電路或閘陣列、現成半導體（諸如邏輯晶片、電晶體或其他離散元件）的硬體電路。模組也可以被實現在可程式設計硬體設備中，諸如現場可程式設計閘陣列、可程式設計陣列邏輯、可程式設計邏輯裝置、圖形處理單元等。

【0078】 模組也可以至少部分地被實現在軟體中以便由各種類型的處理器執行。例如，可執行代碼的所標識的單元可以包括電腦指令的一個或多個物理區塊或邏輯區塊，這些電腦指令可以例如被組織成物件、程序或功能。雖然如此，但所標識的模組的可執行檔不需要在實體上位於一處，而是可以包括儲存在不同位置中的全異指令，這些指令在以邏輯方式被結合在一起時包括該模組並且實現了該模組的既定目的。此外，模組可以被儲存在電腦可讀取媒體上，在不脫離本發明的範圍的情況下，該電腦可讀取媒體可以是例如硬碟、快閃記憶體裝置、RAM、磁帶和/或用於儲存資料的任何其他此類非暫態電腦可讀取媒體。

【0079】 實際上，可執行代碼的模組可以是單個指令或許多指令，並且甚至可以分佈在若干不同的程式碼片段上、分佈在不同程式之間以及跨多個記憶體裝置。類似地，運算資料在本文中可以被標識和圖示為在模組內，並且該運算資料可以被實施為任何合適的形式並且被組織在任何合適類型的資料結構內。運算資料可以被收集為單個資料集，或可以分佈於包括在不同儲存裝置上的不同位置上，並且可以至少部分地僅作為系統或網路上的電子信號存在。

【0080】 通常，在有人值守的自動化中，當在相同通信期中運行的RPA機器人完成其任務時，人類用戶等待。然而，一些實施例創建一個或多個機器人通信期，以在其中託管並運行RPA機器人。與現有RPA系統不同在於，當（一個或多個）機器人正在（一個或多個）機器人通信期中運行自動化時，用戶可以從與他們的計算系統互動的能力中受益。在一些實施例中，用戶仍然監測機器人正在做什麼並且通過用於（一個或多個）機器人通信期的（一個或多個）主機自動化視窗來與機器人互動。在一些實施例中，用戶和（一個或多個）機器人通信期可能正在遠端機器上運行，該遠端機器由使用者的計算系統控制。

【0081】 然而，在一些實施例中，RPA機器人不會經由IPC促進器以與用戶正在使用的應用程式互動。RPA機器人而是可以經由RPA促進器和RPA驅動程

式以與應用程式或其他程序互動，該應用程式或其他程序對用戶為不可見或以其他方式未被使用者使用。這樣的機器人通常不會直接增加與應用程式的用戶互動並且可以被視為是無人值守的機器人。

【0082】 在某些實施例中，RPA機器人可以正在使用者的計算系統上運行並且通過遠端執行時間（remote runtime）（例如經由UiPath Remote Runtime™）來驅動遠端計算系統。UiPath Remote Runtime™是促進在遠端應用程式或桌面（諸如Citrix虛擬App和Desktops™）與專用UiPath®擴展（例如針對Citrix®的UiPath®擴展或針對Windows®遠端桌面的UiPath®擴展）之間的通信的元件。UiPath Remote Runtime™收集與遠端應用程式的目標UI元素相關的資訊並且將該資訊發送給對應擴展，以使得在UI Explorer™中本機生成選擇器。

【0083】 圖6A-G圖示了根據本發明的實施例的使用在機器人通信期中運行的RPA機器人、IPC促進器以及RPA驅動程式在用戶通信期中完成表單的示例。在圖6A中，示出了使用者通信期視窗600，其中使用者能夠與UI中的應用程式互動並且當前沒有任何機器人正在執行。機器人工具列圖示610在使用者通信期視窗600的右下部分為可見。RPA驅動程式640包括IPC促進器642作為該實施例中的子程序。然而，在某些實施例中，在不脫離本發明的範圍的情況下，IPC促進器642可以是與RPA驅動程式640進行通信的單獨應用程式或程序。在此類實施例中，IPC促進器642可以經由IPC、經由API呼叫（call）等以與RPA驅動程式640進行通信。實際上，在某些實施例中，可以不使用IPC。

【0084】 在圖6B中，用戶啟動Web流覽器並且訪查發票創建Web頁面620。在圖6C中，用戶上拉機器人工具列612（例如通過點擊機器人工具列圖示610），並且選擇用戶端通信期機器人選項614，以在其計算系統上執行。如在圖6D中所示，在選擇了要執行的機器人之後，用於機器人通信期的機器人通信期視窗630顯示為螢幕上的子視窗。RPA將在機器人通信期中操作。在該實施例中，用於用

戶端通信期機器人的視窗632在機器人通信期視窗630內自動啟動，並且包括按鈕634，用於運行表單填寫器工作流程，以在使用者通信期中針對Web頁面620獲取資料。

【0085】 在一些實施例中，可以不顯示機器人通信期視窗，並且機器人可以在不對使用者為可見的情況下自動啟動、操作和關閉。在某些實施例中，機器人可以在完成其工作流程之後關閉其通信期。在一些實施例中，並非從機器人工具列啟動，而是可以發起機器人通信期，並且機器人可以在不使用機器人工具列612的情況下啟動和操作（例如由於用戶在主通信期的應用程式中點擊了按鈕）。

【0086】 轉向圖6E，在使用者點擊了按鈕634之後，機器人開始獲取用於在Web頁面620中填寫表單欄位的資料。在機器人獲取了資料的一部分時，向IPC促進器642發送該資料，然後該IPC促進器642使RPA驅動程式640將該資料輸入Web流覽器620中。例如，RPA驅動程式640可以將滑鼠移動到給定欄位上，並且使來自資料的文字輸入到該欄位中。在圖6E中，RPA驅動程式640已經完成了發票編號欄位，並且現在已經將滑鼠移動至發票數量欄位，點擊該發票數量欄位，並且輸入數字1。在圖6E中，插入符號出現在數數位1之後。在某些實施例中，RPA促進器642可以在使RPA驅動程式640填寫Web頁面620的欄位之前從機器人632接收完整的資料集。

【0087】 根據上文，使用IPC協定來完成在機器人通信期中運行的機器人與在用戶通信期中運行的IPC促進器642之間的通信。IPC協定可以經由網路、管道、元件物件模型（COM）、遠端進程呼叫（RPC）、通訊端等來促進通信。合適的通信期創建機制和IPC協定也可以用於其他支援的作業系統。機器人也可以經由IPC協定將狀態通知（例如指示了機器人正在啟動、運行、暫停等）、所獲取的資料、錯誤訊息、命令或其他通信發送回IPC促進器642。

【0088】 當Web頁面620未被另一視窗覆蓋或最小化時，來自由機器人所

獲取的資料的相關聯文字在其由RPA驅動程式640輸入時對用戶為可見。在一些實施例中，當RPA驅動程式640針對表單完成資料互動時，使用者可以與其他應用程式互動並且繼續產生作用或甚至與Web頁面620本身互動，儘管如果RPA驅動程式640針對其中使用者輸入有資訊的同一欄位的資料進行修改，則使用者的輸入可能會被重寫。

【0089】 轉向圖6F，在機器人完成了其工作流程之後，在視窗632中顯示訊息。如在後臺中可看到，機器人已經完成了針對表單欄位的資料獲取，並且表單欄位的新輸入在由RPA驅動程式640輸入之後在Web頁面620中為可見。使用者然後可以關閉機器人通信期視窗630，機器人可以自動關閉機器人通信期視窗630（以及潛在地機器人通信期本身），或機器人通信期視窗630可以保持打開狀態。用戶然後可以提交完成的表單。參見圖6G。

【0090】 圖7A-G圖示了根據本發明的實施例的使用在機器人通信期中運行的RPA機器人經由直接變數修改以在使用者通信期中完成表單的示例。在圖7A中，示出了使用者通信期視窗700，其中使用者能夠與UI中的應用程式互動並且當前沒有任何機器人正在執行。機器人工具列圖示710在使用者通信期視窗700的右下部分為可見。

【0091】 在圖7B中，用戶啟動Web流覽器並且訪查發票創建Web頁面720。在圖7C中，用戶上拉機器人工具列712（例如通過點擊機器人工具列圖示710），並且選擇用戶端通信期機器人選項714，以在其計算系統上執行。如在圖7D中所示，在選擇了要執行的機器人之後，用於機器人通信期的機器人通信期視窗730顯現為螢幕上的子視窗。RPA將在機器人通信期中操作。在該實施例中，用於用戶端通信期機器人的視窗732在機器人通信期視窗730內自動啟動，並且包括用於在使用者通信期中運行Web頁面720的表單填寫器工作流程的按鈕734。

【0092】 在一些實施例中，可以不顯示機器人通信期視窗，並且機器人可

以在不對使用者為可見的情況下自動啟動、操作和關閉。在某些實施例中，機器人可以在完成其工作流程之後關閉其通信期。在一些實施例中，並非從機器人工具列啟動，而是可以起始機器人通信期，並且機器人可以在不使用機器人工具列 712 的情況下啟動和操作（例如由於用戶在主通信期的應用程式中點擊了按鈕）。

【0093】 轉向圖7E，在使用者點擊了按鈕734之後，機器人通過存取針對Web頁面720的表單欄位的變數並且與其互動來開始填寫Web頁面720中的表單欄位。例如，在不脫離本發明的範圍的情況下，機器人可以改變針對儲存在RAM、永久性本機儲存區裝置、資料庫或任何其他儲存類型和/或位置中的欄位的記憶體值（memory value）。在某些實施例中，可以將變數儲存為記憶體的一部分，該變數被針對與Web表單720相關聯的Web應用程式而被分配，並且由與Web表單720相關聯的該Web應用程式所使用。

【0094】 當Web頁面720被刷新時，由機器人改變的相應記憶體值使Web頁面720的相應欄位被更新。當Web頁面720未被另一視窗覆蓋或最小化時，經由資料修改而由機器人改變的相關聯文字對使用者為可見。與圖6A-G中的示例不同，其中RPA驅動程式640操縱滑鼠並經由API級別和/或本機機制輸入文字，資料在沒有這種互動的情況下出現在Web頁面720的表單欄位中。在一些實施例中，當機器人完成針對表單的資料互動時，使用者可以與其他應用程式互動並且繼續產生作用或甚至與Web頁面720本身互動，儘管如果機器人針對其中使用者輸入有資訊的同一欄位的資料進行修改，則使用者的輸入可能會被重寫。

【0095】 轉向圖7F，在機器人完成了其工作流程之後，在視窗732中顯示訊息。如在後臺可看到，機器人已經完成了針對與表單欄位相關聯的相應記憶體變數（memory variable）的資料修改，並且針對表單欄位的新輸入在網路720刷新之後在網路720中為可見。使用者然後可以關閉機器人通信期視窗730，機器人可以自動關閉機器人通信期視窗730（以及潛在地機器人通信期本身），或機器人

通信期視窗730可以保持打開狀態。用戶然後可以提交完成的表單。參見圖7G。

【0096】 在一些實施例中，機器人通信期可以是子通信期，並且子通信期可以經由作業系統的子通信期API被創建。在一些實施例中，可以在不脫離本發明的範圍的情況下使用Windows®終端服務子通信期或由作業系統提供的另一子通信期API來創建第二通信期。被配置為啟動（一個或多個）機器人的機器人工具列應用程式（例如UiPath®機器人代理桌面）或另一應用程式然後可以在作業系統中使用具有適當參數的創建程序API，以在該子通信期中開始機器人程序。然後，機器人工具列應用程式或其他合適的應用程式可以使用合適的協定（例如構建於具名管道上的協定）與機器人程序通信。

【0097】 圖8是圖示了根據本發明的實施例的經由在機器人通信期中運行的RPA機器人、IPC促進器以及RPA驅動程式使在用戶通信期中運行的程序自動化的程序800的流程圖。在810處，該程序開始於啟動使用者通信期視窗。例如，這可以是與在使用者計算系統上運行的作業系統相關聯的主視窗。然後在820處，啟動機器人通信期視窗作為使用者通信期視窗的子視窗。在一些實施例中，例如，可以回應於機器人被起始或以其他方式啟動來啟動機器人通信期視窗。然後在830處，在機器人通信期中起始機器人。

【0098】 機器人在840處被執行並且經由IPC與IPC促進器通信。IPC促進器可以經由IPC從機器人接收狀態通知、所獲取的資料、錯誤訊息、命令或其他通信。IPC促進器還可以向機器人發送訊息（諸如命令、狀態更新、來自IPC促進器或RPA驅動程式的錯誤訊息等）。

【0099】 在850處，使用從機器人獲取的資料，RPA促進器控制或以其他方式使RPA驅動程式使用來自機器人的所獲取的資料和/或其他通信來執行應用程式互動。例如，可以使RPA驅動程式移動滑鼠、點擊欄位、輸入文字、點擊按鈕、導覽選單、打開或關閉應用程式或視窗等。在860處，在一些實施例中，在

機器人完成其執行之後，可以結束機器人通信期並且可以自動關閉機器人通信期視窗。

【0100】 圖9是圖示了根據本發明的實施例的經由直接變數修改以經由在機器人通信期中運行的RPA機器人使在用戶通信期中運行的程序自動化的程序900的流程圖。在910處，該程序開始於啟動使用者通信期視窗。例如，這可以是與在使用者計算系統上運行的作業系統相關聯的主視窗。然後在920處，啟動機器人通信期視窗作為使用者通信期視窗的子視窗。在一些實施例中，例如，可以回應於機器人被起始或以其他方式啟動來啟動機器人通信期視窗。然後在930處，機器人在機器人通信期中被起始，並且在940處，機器人與關聯於（一個或多個）應用程式（例如Web頁面、試算表、ERP應用程式、銷售應用程式等）的資料互動，該（一個或多個）應用程式在來自機器人通信期的用戶通信期中運行。例如，機器人可以在計算系統記憶體中創建、改變或刪除所儲存的資訊，以使改變發生在共用儲存資料中，該共用儲存資料在使用者通信期中由（一個或多個）應用程式存取和使用。在950處，在一些實施例中，在機器人完成其執行之後，可以結束機器人通信期並且可以自動關閉機器人通信期視窗。

【0101】 在一些實施例中，可以在用戶端通信期中隔離多個運行的自動化程序（例如UI自動化程序）。例如，這可以允許經由IPC從主通信期中按需求呼叫這些程序。以這種方式，可以按需求從主通信期應用程式、主通信期RPA機器人、另一用戶端通信期RPA機器人等中呼叫多個RPA機器人程序。在用戶端通信期中運行的RPA機器人也可能能夠從主通信期中收集資料。在某些實施例中，在用戶端通信期中運行的RPA機器人可以在執行其自動化或自動化的一部分之前等待來自主通信期的某個觸發。

【0102】 圖10是圖示了根據本發明的實施例的與主通信期應用程式互動的多個用戶端通信期RPA機器人的示例的架構圖1000。在圖10中，主通信期1010

和用戶端通信期1020正在運行。用戶端通信期1020包括用戶端通信期RPA機器人1 1030和用戶端通信期RPA機器人2 1032。

【0103】 用戶端通信期RPA機器人1030、1032與單獨的IPC促進器應用程式1040或嵌入式RPA促進器程序或作為驅動程式1050的一部分的子常式進行通信。在一些實施例中，例如，驅動程式1050可以是圖3的驅動程式340。IPC促進器1052使驅動程式1050以由用戶端通信期RPA機器人1030或1032的相應工作流程邏輯所限定的方式來與主通信期應用程式和/或相關聯的應用程式物件1060（例如針對Microsoft Windows®應用程式的元件物件模型（COM）物件）互動。例如，可以使驅動程式1050在主通信期應用程式1060中移動滑鼠、填寫表單資料、點擊一個或多個按鈕、與選單互動、其組合等。在某些實施例中，IPC促進器應用程式1040可以與主通信期應用程式和/或相關聯的應用程式物件1060直接通信。

【0104】 在某些實施例中，在用戶端通信期中運行的RPA機器人可以是有人值守的自動化機器人。例如，在用戶端通信期中運行的RPA機器人可以在主通信期中要求來自用戶的輸入。這可以經由IPC而與主通信期機器人、驅動程式等一起完成。

【0105】 圖11是圖示了根據本發明的實施例的用於在用戶端通信期中運行有人值守的自動化RPA機器人的程序的流程圖1100。在該實施例中，在1120處，用戶利用主通信期應用程式1110，該主通信期應用程式使用戶端通信期RPA機器人程序經由IPC開始。在一些實施例中，主通信期應用程式1110可以是允許主通信期中的用戶與用戶端通信期中的有人值守的RPA機器人互動的RPA助理應用程式。例如，RPA助理應用程式的儀錶板可以示出機器人所支援的當前RPA程序，並且用戶可以選擇要執行的程序。在某些實施例中，在用戶端通信期RPA機器人正在運行的同時，RPA助理應用程式可以顯示用戶端通信期工作流程的步驟和/

或執行狀態。

【0106】 在用戶端通信期RPA機器人在1120處開始之後，在1130處，用戶端通信期RPA機器人運行其邏輯的第一部分。在1130處執行了邏輯的第一部分之後，在1140處，用戶端通信期RPA機器人要求用戶輸入。這可以經由在主通信期中運行的應用程式中的訊息（經由與主通信期應用程式互動的IPC促進器和驅動程式而顯現）、來自在主通信期中運行的RPA機器人的訊息等而發生。然後，用戶提供所請求的輸入，此輸入經由IPC被發送到用戶端通信期RPA機器人，並且在1150處使用此輸入來執行邏輯的第二部分。在用戶端通信期RPA機器人程序在1160處結束之後，主通信期應用程式1110可以接收RPA機器人程序已經結束的通知。

【0107】 在某些實施例中，自動化的部分可以在主通信期中運行，而自動化的其他部分可以在用戶端通信期中運行。例如，主通信期應用程式或RPA機器人可以在主通信期中執行某些操作，然後在主通信期應用程式或機器人到達其執行中的某個點、接收觸發、邏輯條件被滿足等的時候使在用戶端通信期中運行的RPA機器人執行自動化的另一部分。用戶端通信期RPA機器人可以按順序或與主通信期應用程式或機器人並行地執行其自動化的部分。用戶端通信期RPA機器人然後可以向主通信期應用程式或機器人提供指示、所請求的資料、執行結果等。

【0108】 圖12是圖示了根據本發明的實施例的在主通信期機器人M1、用戶端通信期應用程式A1與一對用戶端通信期機器人C1和C2之間的多機器人集體工作流程的執行的流程圖1200。A1、C1和/或C2可以在同一用戶端通信期中或不同用戶端通信期中。M1開始執行其工作流程並且到達呼叫C1完成工作流程的活動。在一些實施例中，這可以由“調用工作流程”活動來執行，該“調用（invoke）工作流程”活動具有被設置為在用戶端通信期中執行RPA程序的旗標。

M1使**C1**經由**IPC**執行其工作流程，並且呼叫活動等待。在此期間，在一些實施例中，**M1**可以執行其他任務。

【0109】 在**C1**完成其工作流程之後，**C1**經由**IPC**通知**M1**，並且**M1**恢復執行，直到其到達呼叫**A1**以開始執行或以完成任務的活動為止，這是在用戶端通信期中的程序。**M1**使**A1**經由**IPC**執行和/或執行任務，且呼叫活動（例如“調用程序”活動）等待。在一些實施例中，調用程序活動可以具有被設置為在用戶端通信期中執行**A1**的旗標。在一些實施例中，再一次，**M1**可以在等待**A1**的同時執行其他任務。

【0110】 在**A1**完成其程序或任務之後，**A1**經由**IPC**通知**M1**，並且**M1**恢復執行，直到其到達呼叫**C2**以完成工作流程的活動為止。**M1**使**C2**經由**IPC**執行其工作流程，並且呼叫活動等待。在一些實施例中，再一次，**M1**可以在等待**C2**的同時執行其他任務。

【0111】 在**C2**完成其工作流程之後，經由**IPC**通知**M1**，並且**M1**恢復執行，直到其到達再次呼叫**C1**以完成工作流程的活動為止。這可以是先前執行的同一工作流程活動，或可以是不同的工作流程或一組活動。**M1**使**C1**經由**IPC**執行其工作流程，並且呼叫活動等待。在一些實施例中，再一次，**M1**可以在等待**C1**的同時執行其他任務。在**C1**完成其工作流程之後，經由**IPC**通知**M1**恢復執行，直到完成**M1**的工作流程為止。在某些實施例中，**C1**可以在最初被調用之後保持運行，並且當再次被**M1**呼叫時仍然正在運行。

【0112】 根據上文，在一些實施例中，**M1**可以包括一個或多個“調用工作流程”活動和/或“調用程序”活動，該“調用工作流程”活動和/或“調用程序”活動可以調用（一個或多個）**RPA**機器人（例如**C1**和/或**C2**）和/或用戶端通信期中的其他程序（例如**A1**）。這樣的活動可以開始相應的（一個或多個）**RPA**機器人和/或（一個或多個）程序、與已經在運行的（一個或多個）**RPA**機器人和/或（一個或

多個) 程序通信等。可以在RPA開發者針對M1開發工作流程時創建這些活動。在某些實施例中，(一個或多個) RPA機器人可以經由RPA促進器或其他合適的功能性將RPA機器人和/或程序從用戶端通信期調用到主通信期中。

【0113】 在執行期間，M1可以經由IPC與A1、C1和C2通信，反之亦然。例如，M1可以向A1、C1和C2發送評論和請求，並且M1可以從A1、C1和C2接收狀態訊息、結果和錯誤訊息等。以這種方式，M1可以充當控制用戶端通信期中的用戶端通信期RPA機器人和/或其他程序(例如應用程式)的主RPA機器人。在某些實施例中，M1、A1、C1和/或C2可以並行地操作。

【0114】 在一些實施例中，M1可以執行所有與在主通信期中運行的應用程式的互動，這可以消除對IPC促進器的需要。因此，C1和C2可以執行各種資料獲取和處理任務，且M1可以執行UI互動任務。在某些實施例中，可以並行地執行M1、C1和/或C2的至少一些工作流程活動。在圖13的流程圖1300中示出了一個此示例，其中基於M1的邏輯來執行C1，並且此執行並行地發生。可以在C1的節段(section)期間而在M1與C1之間發送通信。然而，在某些實施例中，C1可以獨立完成其執行，並且可以在M1結束之後完成其工作流程的執行。

【0115】 在一些實施例中，所有RPA機器人活動在用戶端通信期中運行。可以由機器人執行器從主通信期中呼叫RPA機器人，例如，諸如UiPath®機器人工具列、IPC促進器、另一應用程式等。RPA機器人的執行結果和其他通信可以經由IPC被提供給機器人執行器或其他應用程式。

【0116】 在某些實施例中，標準化通信格式可以用於用戶端通信期與主通信期之間的IPC通信。例如，主通信期和用戶端通信期程序可以交換XAML資料。這有助於確保通信是相應程序可以識別的格式。

【0117】 在一些實施例中，出於同步目的，可以在同一通信期中運行的RPA程序之間使用IPC。例如，多個RPA程序可以在後臺中與主通信期中的前臺

程序並行地運行。IPC可以提供同步機制，以在後臺程序與前臺或主程序之間交換資訊。

【0118】 在某些實施例中，將在主RPA工作流程程序中載入驅動程式（例如圖3的驅動程式340）。子RPA程序與主RPA工作流程程序之間的通信可以經由IPC發生。這允許主RPA工作流程程序利用驅動程式以基於IPC資訊來為用戶端通信期機器人實施操作。

【0119】 在一些實施例中，用戶端通信期RPA機器人可以與在主通信期中打開的應用程式一起工作。例如，用戶端通信期RPA機器人可以與在主通信期中打開的Excel[®]試算表一起工作。例如，這可以經由IPC而在用戶端通信期RPA機器人與RPA促進器之間發生。RPA促進器可以基於由用戶端通信期RPA機器人提供的資訊（例如將資料輸入試算表中、修改表格值等）來使驅動程式實施與Excel[®]試算表的（一個或多個）互動。在某些實施例中，活動可以被包括在RPA工作流程中，該RPA工作流程自動捕獲兩個通信期之間的應用程式資料（例如MicrosoftOffice[®]活動、Web流覽器活動等）。以這種方式，用戶端通信期RPA機器人實質上可以與在主通信期中運行的應用程式一起工作，就如同他們在RPA機器人自身的通信期中運行一樣。實際上，在一些實施例中，RPA開發者可能不知道其正在開發的工作流程或其一部分將在用戶端通信期中被執行。

【0120】 在某些實施例中，例如，當使用設計器應用程式來設計RPA機器人時，RPA開發者可以在“調用工作流程”活動或“調用程序”活動上設置用戶端通信期旗標。調用工作流程活動調用RPA工作流程。調用程序活動執行可用於本地機器的程序。設置此旗標可以使RPA工作流程或程序在用戶端通信期而非主通信期中被調用。

【0121】 在一些實施例中，用戶端通信期RPA機器人的執行可以是基於來自主通信期應用程式和/或RPA機器人的觸發。圖14是圖示了根據本發明的實施

例的基於針對主通信期應用程式的觸發來執行用戶端通信期RPA機器人的流程圖1400。主通信期應用程式A1（在某些實施例中其可以是RPA機器人）開始執行並且等待觸發。在接收到觸發之後，A1經由IPC開始執行用戶端通信期RPA機器人C1。可以在C1的節段期間而在A1與C1之間發送通信。然後，A1等待另一觸發，且C1執行其工作流程。

【0122】 圖15是圖示了根據本發明的實施例的用於執行通信期間自動化的程序的流程圖1500。在1510處，該程序開始於在計算系統的主通信期中執行RPA機器人。在1520處，由主通信期RPA機器人經由IPC在用戶端通信期中呼叫或啟動RPA機器人或應用程式。在1530處，經由IPC在用戶端通信期RPA機器人或應用程式與主通信期RPA機器人之間傳達訊息。在一些實施例中，在1540處，主通信期RPA機器人暫停其執行，直到用戶端通信期RPA機器人或應用程式的執行結束為止。

【0123】 在1550處，由主通信期RPA機器人經由IPC接收到用戶端通信期RPA機器人或應用程式的執行的（一個或多個）結果。然後，主通信期RPA機器人在1560處使用（一個或多個）結果，以完成主通信期RPA機器人的工作流程的至少一部分、與在主通信期中運行的應用程式或應用程式物件互動、或兩者。在一些實施例中，在1570處，針對下一個用戶端通信期RPA機器人或用戶端通信期應用程式重複該程序，並且該程序返回到步驟1520。

【0124】 在一些實施例中，在主通信期RPA機器人的工作流程中兩次或更多次地呼叫或啟動用戶端通信期RPA機器人、用戶端通信期應用程式或兩者。在某些實施例中，經由主通信期RPA機器人的工作流程的調用程序活動來呼叫或啟動用戶端通信期應用程式，經由主通信期RPA機器人的工作流程的調用工作流程活動來呼叫或啟動用戶端通信期RPA機器人，或兩者。在一些實施例中，調用程序活動包括指示用戶端通信期應用程式在用戶端通信期中運行的旗標，調用工

作流程活動包括指示用戶端通信期RPA機器人在用戶端通信期中運行的旗標，或兩者。在某些實施例中，用戶端通信期應用程式、用戶端通信期RPA機器人或兩者在提供結果之後繼續運行。在一些實施例中，在主通信期RPA機器人與用戶端通信期RPA機器人和/或用戶端通信期應用程式之間的IPC通信是標準化通信格式。

【0125】 根據本發明的實施例，圖8-15中執行的程序步驟可以由電腦程式執行，對指令編碼，以供（一個或多個）處理器執行圖8-15中所描述的（一個或多個）程序的至少一部分。電腦程式可以被實施在非暫態電腦可讀取媒體上。電腦可讀取媒體可以是但不限於：硬碟、快閃記憶體裝置、RAM、磁帶和/或用於儲存資料的任何其他此類媒體或媒體組合。電腦程式可以包括編碼指令，編碼指令用於控制計算系統的（一個或多個）處理器（例如圖5的計算系統500的（一個或多個）處理器510）以實現圖8-15中所描述的程序步驟的全部或一部分，還可以將編碼指令儲存在電腦可讀取媒體上。

【0126】 電腦程式可以被實施在硬體、軟體或混合實施中。電腦程式可以由彼此可操作地通信並且被設計成傳遞資訊或指令以進行顯示的模組所組成。電腦程式可以被配置為在通用電腦、ASIC或任何其他合適的設備上操作。

【0127】 將容易地理解，如本文中的諸圖中大體上描述和圖示的，本發明的各種實施例的元件可以以多種不同配置來佈置和設計。因此，如附圖中所表示，對本發明的實施例的詳細描述並不旨在限制所要求保護的本發明的範圍，而是僅僅表示本發明的所選擇的實施例。

【0128】 貫穿本說明書描述的本發明的特徵、結構或特性可以在一個或多個實施例中以任何合適的方式組合。例如，貫穿本說明書，對“某些實施例”、“一些實施例”或類似語言的參考是指結合實施例所描述的特定特徵、結構或特性被包括在本發明的至少一個實施例中。因此，貫穿本說明書，用語“在某些實施例

中”、“在一些實施例中”、“在其他實施例中”或類似語言的出現並不一定全都是指同一組實施例，並且所描述的特徵、結構或特性可以在一個或多個實施例中以任何合適的方式組合。

【0129】 應注意，貫穿本說明書，對特徵、優點或類似語言的參考並不意謂可以用本發明實施所有特徵和優點都應該在或都在本發明的任何單個實施例中。相反，涉及特徵和優點的語言被理解為是指結合實施例所描述的特定特徵、優點或特性被包括在本發明的至少一個實施例中。因此，貫穿本說明書，對特徵和優點以及類似語言的討論可以但並不一定是指相同實施例。

【0130】 此外，本發明的所描述的特徵、結構和特性可以在一個或多個實施例中以任何合適的方式組合。相關領域的技術人員應認識到，可以在不具有特定實施例的特定特徵或優點中的一者或多者的情況下實踐本發明。在其他情況下，可以不存在於本發明的所有實施例中的附加特徵和優點在某些實施例中可能被識別。

【0131】 本領域的普通技術人員應容易地理解，可以利用不同順序的步驟和/或利用與所記載的配置不同的配置的硬體元件來實踐如上文所描述的本發明。因此，儘管已經基於這些優選實施例描述了本發明，但對於本領域的技術人員而言將明顯的是，某些修改、變型和備選構造將是明顯的，同時仍在本發明的精神和範圍內。因此，為了確定本發明的界限和邊界，應當參考所附申請專利範圍。

【符號說明】

【0132】

100: 機器人程序自動化（RPA）系統

110: 設計器

120: 指揮器

130: 機器人

132: 有人值守的機器人

134: 無人值守的機器人

200: RPA系統

210: 機器人應用程式/計算系統

212: 執行器

214: 代理器

216: 設計器

220: 流覽器

230: 指揮器

232: Web應用程式

234: 應用程式設計介面（API）端點

236: 通知和監測API

238: API實施/事業邏輯

240: 資料庫伺服器

250: 索引子伺服器

300: 關係

310: 設計器

320: 活動

330: UI自動化活動

340: 驅動程式

342: OS驅動程式

344: 流覽器驅動程式

346: VM驅動程式
348: 企業應用程式驅動程式
400: RPA系統
410: 計算系統
420: 指揮器計算系統
430: 資料庫伺服器
440: 索引子伺服器
500: 計算系統
505: 匯流排
510: 處理器
515: 記憶體
520: 通信設備
525: 顯示器
530: 鍵盤
535: 游標控制設備
540: 作業系統
545: 通信期間（inter-session）自動化模組
550: 附加功能性模組
600: 使用者通信期視窗
610: 機器人工具列圖示
612: 機器人工具列
614: 用戶端通信期機器人選項
620: 發票創建Web頁面
630: 機器人通信期視窗

632: 視窗
634: 按鈕
640: RPA驅動程式
642: IPC促進器
700: 使用者通信期視窗
710: 機器人工具列圖示
712: 機器人工具列
714: 用戶端通信期機器人選項
720: 發票創建Web頁面
730: 機器人通信期視窗
732: 視窗
734: 按鈕
800: 程序
900: 程序
1000: 架構圖
1010: 主通信期
1020: 用戶端通信期
1030: 用戶端通信期RPA機器人
1032: 用戶端通信期RPA機器人
1040: IPC促進器應用程式
1050: 驅動程式
1052: IPC促進器
1060: 主通信期應用程式和/或相關聯的應用程式物件
1100: 流程圖

- 1110: 主通信期應用程式
- 1200: 流程圖
- 1300: 流程圖
- 1400: 流程圖
- 1500: 流程圖

【發明申請專利範圍】

【請求項1】一種電腦實施方法，包括：

在計算系統的主通信期中執行機器人程序自動化（RPA）機器人；

由所述主通信期RPA機器人經由程序間通信（IPC）在客戶端通信期中呼叫（call）或啟動RPA機器人；

由所述主通信期RPA機器人經由IPC從所述客戶端通信期RPA機器人接收一個或多個訊息；以及

由所述主通信期RPA機器人使用從所述客戶端通信期RPA機器人接收到的所述一個或多個訊息的結果以：完成所述主通信期RPA機器人的工作流程的至少一部分、與在所述主通信期中運行的應用程式或應用程式物件進行互動、或執行前述兩者。

【請求項2】根據請求項1所述的電腦實施方法，還包括：

由所述主通信期RPA機器人經由IPC在所述客戶端通信期中呼叫或啟動應用程式；

在所述應用程式的任務的執行期間和/或在所述應用程式的任務的執行之後，由所述主通信期RPA機器人經由IPC從所述客戶端通信期應用程式接收一個或多個訊息；以及

由所述主通信期RPA機器人使用從所述客戶端通信期應用程式接收到的所述一個或多個訊息的結果以：完成所述主通信期RPA機器人的所述工作流程的至少一部分、與在所述主通信期中運行的所述應用程式或應用程式物件進行互動、或執行前述兩者。

【請求項3】根據請求項2所述的電腦實施方法，還包括：

暫停所述主通信期RPA機器人的執行，直到所述客戶端通信期應用程式完成執行。

【請求項4】根據請求項2所述的電腦實施方法，其中所述客戶端通信期應用程式在所述主通信期RPA機器人的所述工作流程中被呼叫或被啟動兩次或更多次。

【請求項5】根據請求項2所述的電腦實施方法，還包括：

由所述主通信期RPA機器人經由IPC在所述客戶端通信期中呼叫或啟動附加應用程式；

經由IPC從所述附加客戶端通信期應用程式接收一個或多個訊息；以及

由所述主通信期RPA機器人使用從所述附加客戶端通信期應用程式接收到的所述一個或多個訊息的結果以：完成所述主通信期RPA機器人的所述工作流程的一部分、與在所述主通信期中運行的所述應用程式或應用程式物件進行互動、或執行前述兩者。

【請求項6】根據請求項2所述的電腦實施方法，其中所述客戶端通信期應用程式經由所述主通信期RPA機器人的所述工作流程的調用（invoke）程序活動而被呼叫或被啟動。

【請求項7】根據請求項6所述的電腦實施方法，其中所述呼叫程序活動包括指示所述客戶端通信期應用程式在所述客戶端通信期中運行的旗標。

【請求項8】根據請求項2所述的電腦實施方法，其中所述客戶端通信期應用程式在所述任務被完成之後繼續運行。

【請求項9】根據請求項2所述的電腦實施方法，其中在所述主通信期RPA機器人與所述客戶端通信期應用程式之間的所述IPC通信是按標準化通信格式。

【請求項10】根據請求項1所述的電腦實施方法，還包括：

暫停所述主通信期RPA機器人的執行，直到所述客戶端通信期RPA機器人完成執行。

【請求項11】根據請求項1所述的電腦實施方法，還包括：

由所述主通信期RPA機器人經由IPC在所述客戶端通信期中呼叫或啟動附加RPA機器人；

經由IPC從所述附加客戶端通信期RPA機器人接收一個或多個訊息；以及

由所述主通信期RPA機器人使用從所述附加客戶端通信期RPA機器人接收到的一個或多個訊息的結果以：完成所述主通信期RPA機器人的所述工作流程的一部分、與在所述主通信期中運行的所述應用程式或應用程式物件進行互動、或執行前述兩者。

【請求項12】根據請求項1所述的電腦實施方法，其中所述客戶端通信期RPA機器人在所述主通信期RPA機器人的所述工作流程中被呼叫或被啟動兩次或更多次。

【請求項13】根據請求項1所述的電腦實施方法，其中所述客戶端通信期RPA機器人經由所述主通信期RPA機器人的所述工作流程的調用工作流程活動而被呼叫或被啟動。

【請求項14】根據請求項13所述的電腦實施方法，其中所述調用工作流程活動包括指示所述客戶端通信期RPA機器人在所述客戶端通信期中運行的旗標。

【請求項15】一種電腦實施方法，包括：

在計算系統的主通信期中執行機器人程序自動化（RPA）機器人；

由主通信期RPA機器人經由程序間通信（IPC）在客戶端通信期中呼叫或啟動應用；

由所述主通信期RPA機器人經由IPC從客戶端通信期應用程式接收一個或多個訊息；以及

由所述主通信期RPA機器人使用從所述客戶端通信期應用程式接收到的所述一個或多個訊息的結果以：完成所述主通信期RPA機器人的工作流程的至少一部分、與在所述主通信期中運行的應用程式或應用程式物件進行互動、或執行前

述兩者。

【請求項16】根據請求項15所述的電腦實施方法，還包括：

由所述主通信期RPA機器人經由IPC在所述客戶端通信期中呼叫或啟動附加應用程式；

經由IPC從所述附加客戶端通信期應用程式接收一個或多個訊息；以及

由所述主通信期RPA機器人使用從所述附加客戶端通信期應用程式接收到的所述一個或多個訊息的結果以：完成所述主通信期RPA機器人的所述工作流程的一部分、與在所述主通信期中運行的所述應用程式或應用程式物件進行互動、或執行前述兩者。

【請求項17】根據請求項15所述的電腦實施方法，其中所述客戶端通信期應用程式經由所述主通信期RPA機器人的所述工作流程的調用程序活動而被呼叫或被啟動。

【請求項18】一種電腦實施方法，包括：

在計算系統的主通信期中執行機器人程序自動化（RPA）機器人；

由主通信期RPA機器人經由程序間通信（IPC）在客戶端通信期中呼叫或啟動RPA機器人；

由所述主通信期RPA機器人經由IPC在所述客戶端通信期中呼叫或啟動應用程式；

由所述主通信期RPA機器人經由IPC從所述客戶端通信期RPA機器人和所述客戶端通信期應用程式接收一個或多個訊息；以及

由所述主通信期RPA機器人使用從所述客戶端通信期RPA機器人和所述客戶端通信期應用程式接收到的所述一個或多個訊息的結果以：完成所述主通信期RPA機器人的工作流程的至少一部分、與在所述主通信期中運行的一個或多個應用程式和/或應用程式物件進行互動、或執行前述兩者。

【請求項19】根據請求項18所述的電腦實施方法，還包括：

由所述主通信期RPA機器人經由IPC在所述客戶端通信期中呼叫或啟動附加應用程式、附加RPA機器人、或前述兩者；

經由IPC從所述附加客戶端通信期應用程式、所述附加RPA機器人、或前述兩者接收一個或多個訊息；以及

由所述主通信期RPA機器人使用從所述附加客戶端通信期應用程式、所述附加客戶端通信期RPA機器人、或前述兩者接收到的所述一個或多個訊息的結果以：完成所述主通信期RPA機器人的所述工作流程的一部分、與在所述主通信期中運行的所述一個或多個應用程式和/或應用程式物件中的至少一者互動、或執行前述兩者。

【請求項20】根據請求項18所述的電腦實施方法，其中所述客戶端通信期應用程式經由所述主通信期RPA機器人的所述工作流程的調用程序活動而被呼叫或被啟動，或者所述客戶端通信期RPA機器人經由所述主通信期RPA機器人的所述工作流程的調用工作流程活動而被呼叫或被啟動，或者進行前述二者。

【發明圖式】

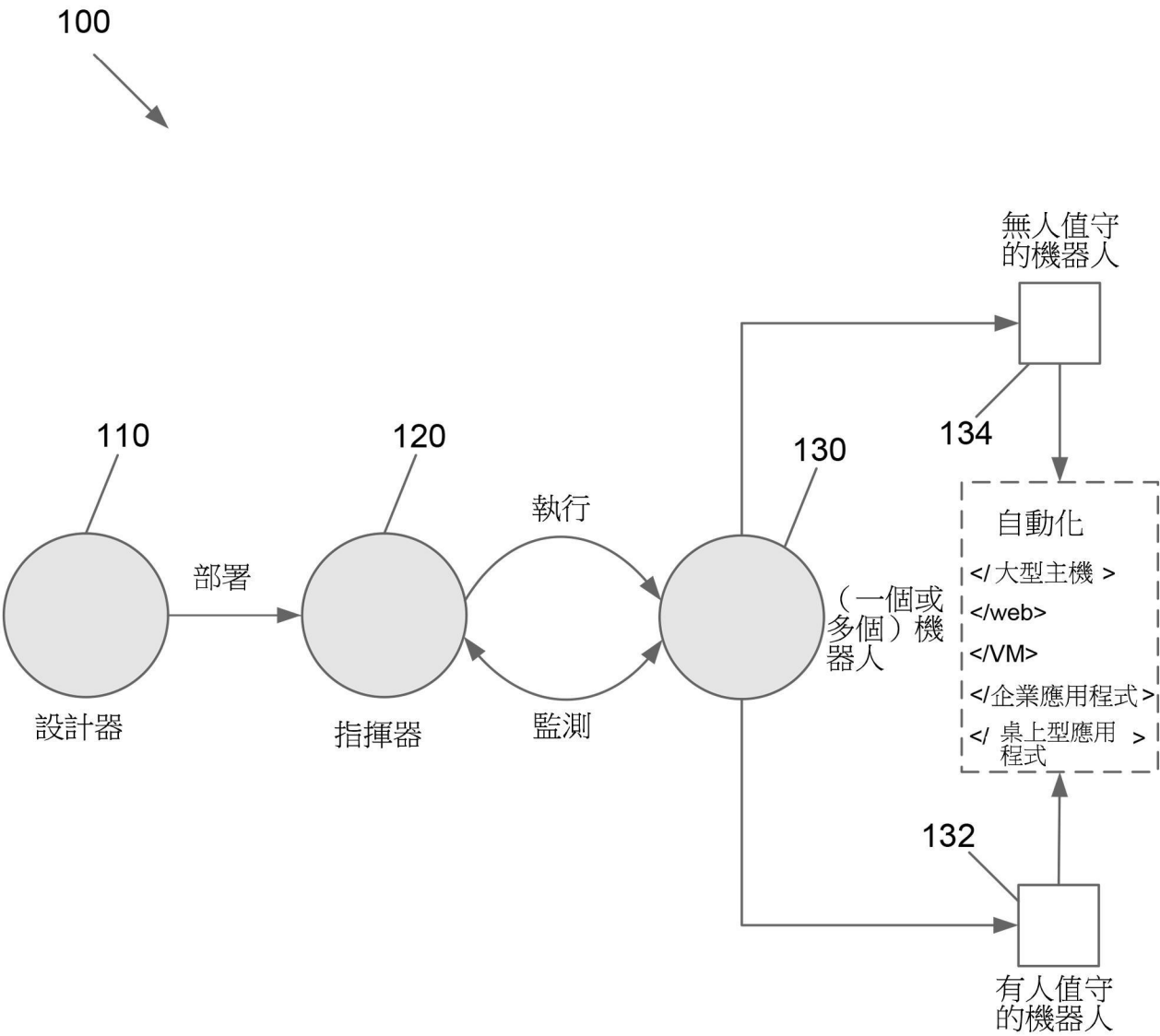


圖1

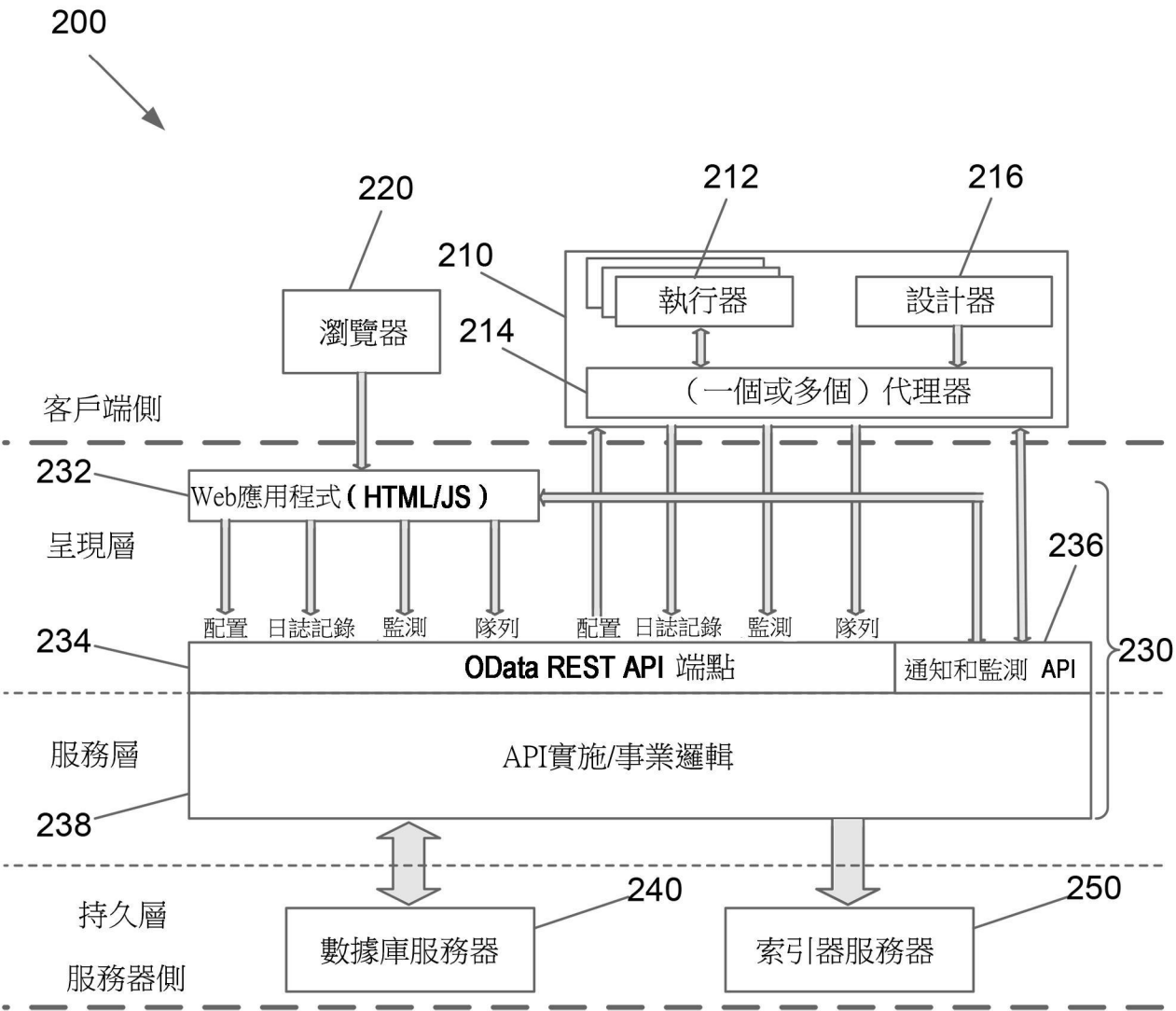


圖2

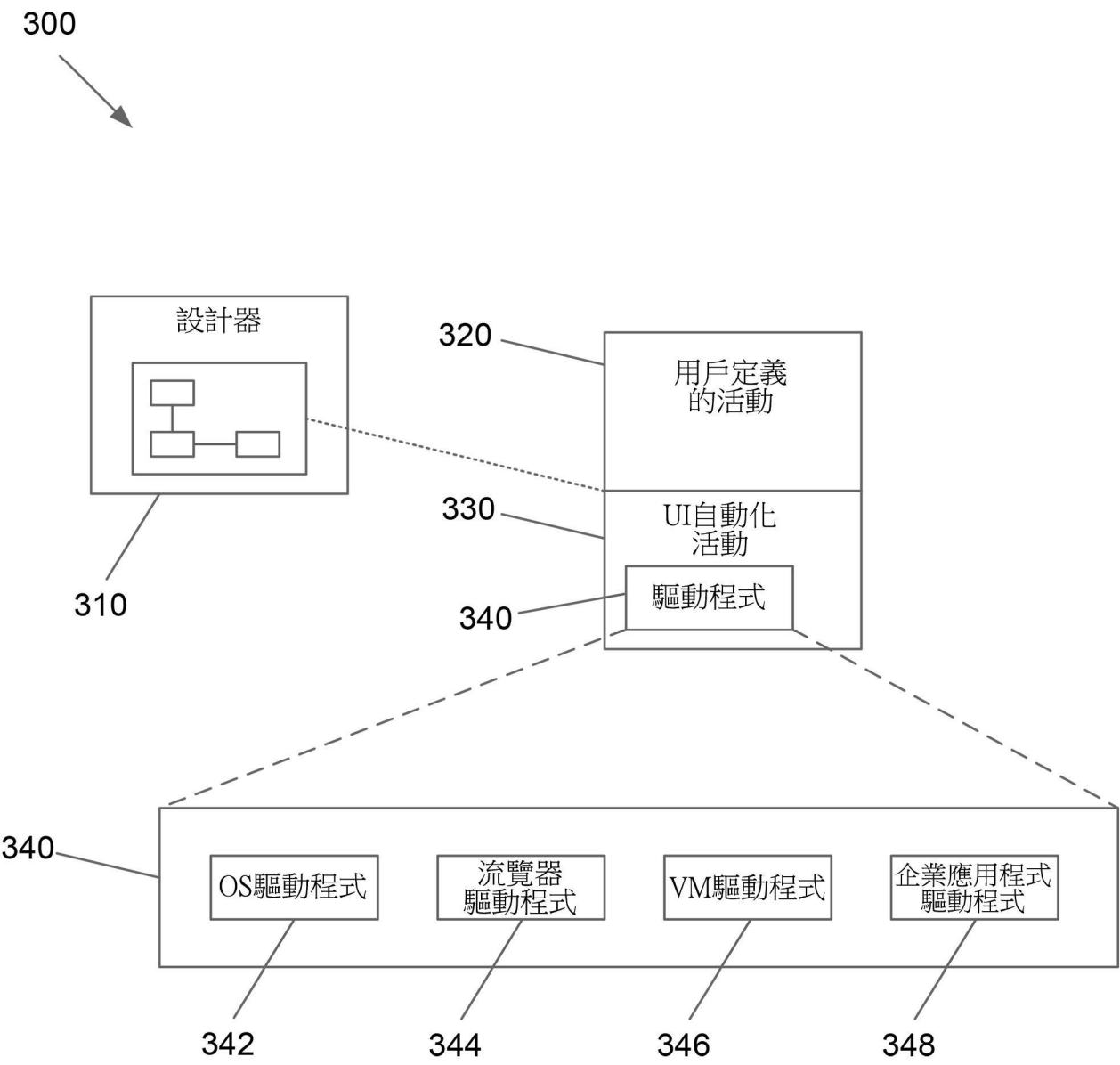


圖3

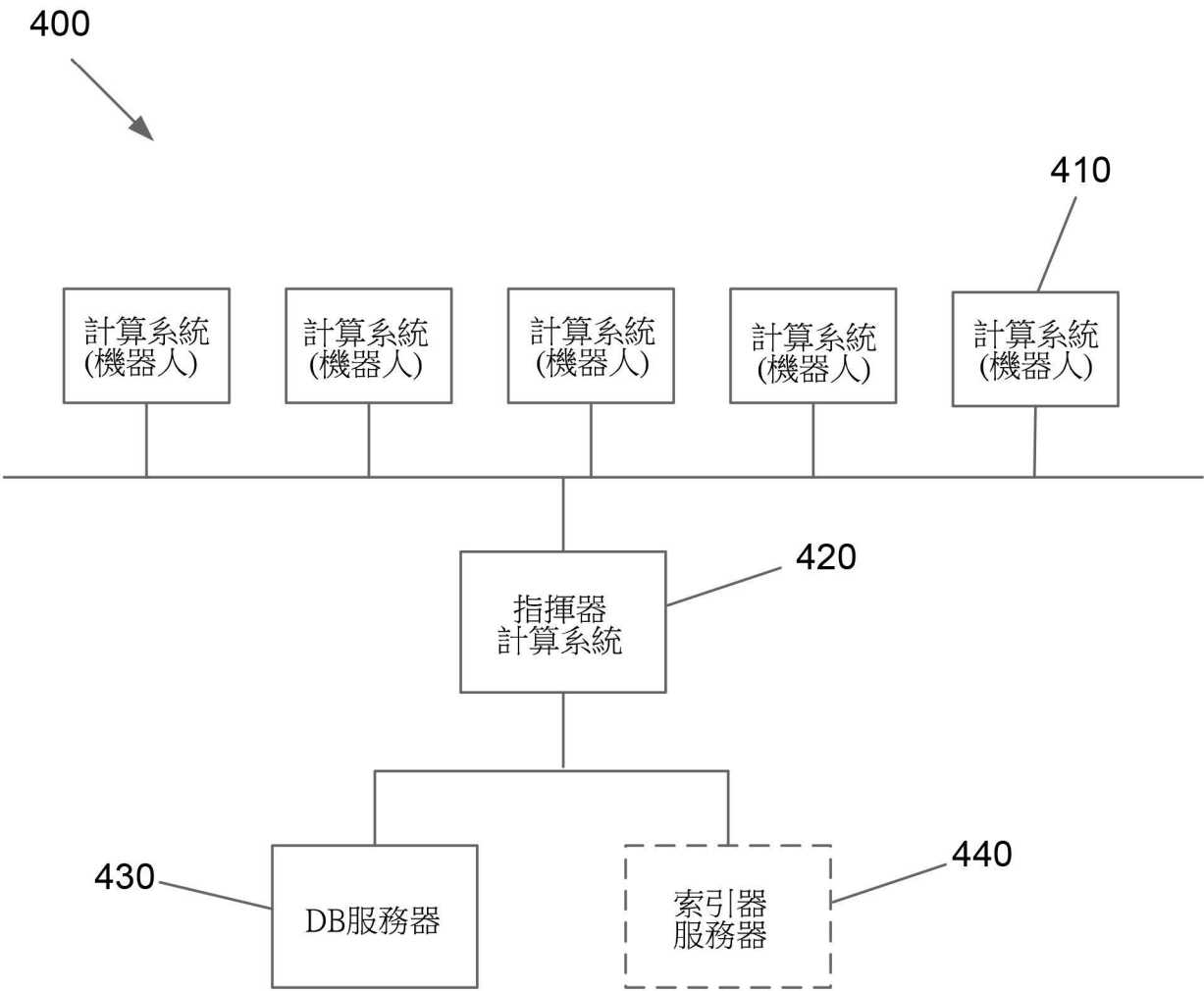


圖4

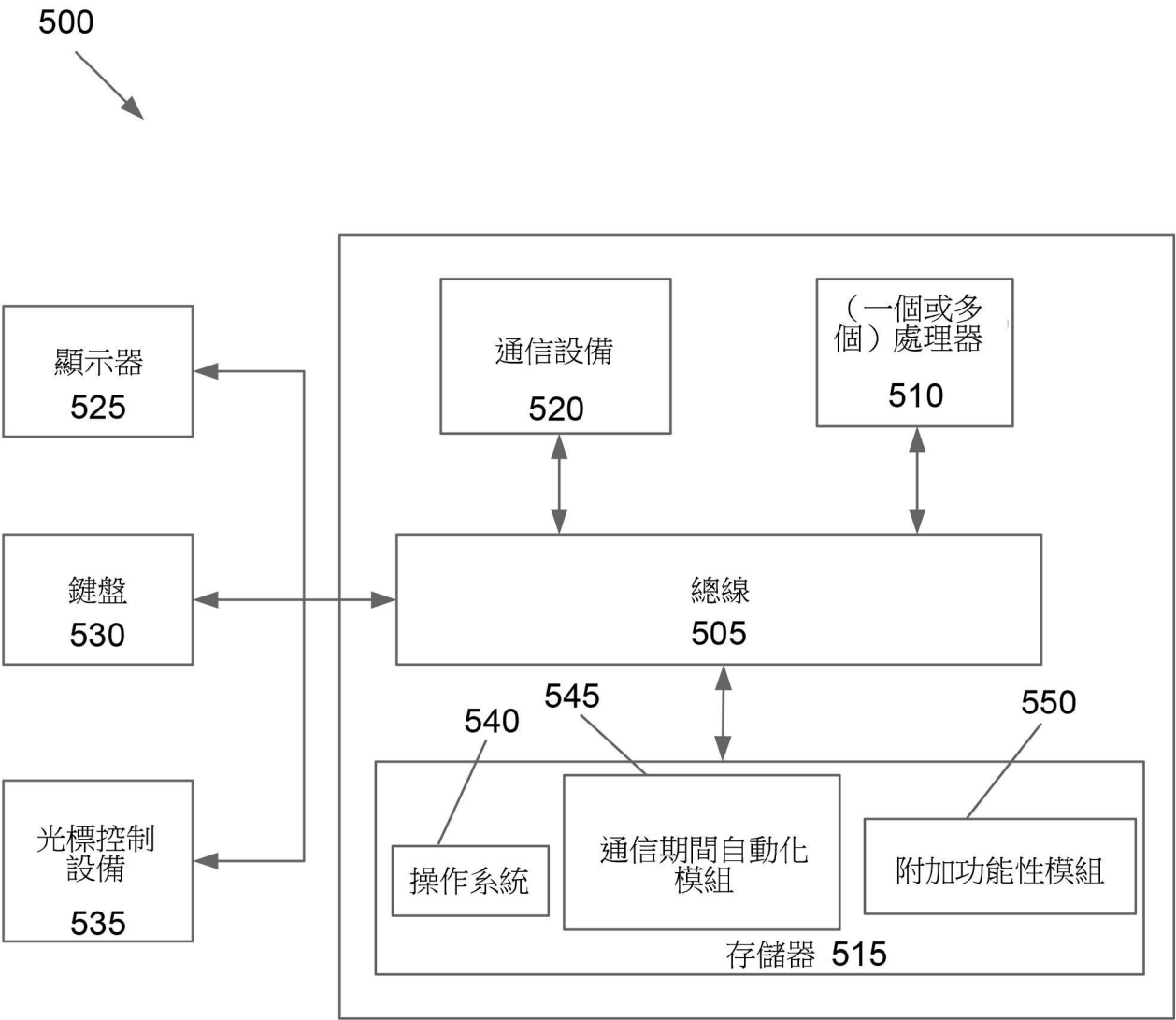


圖5

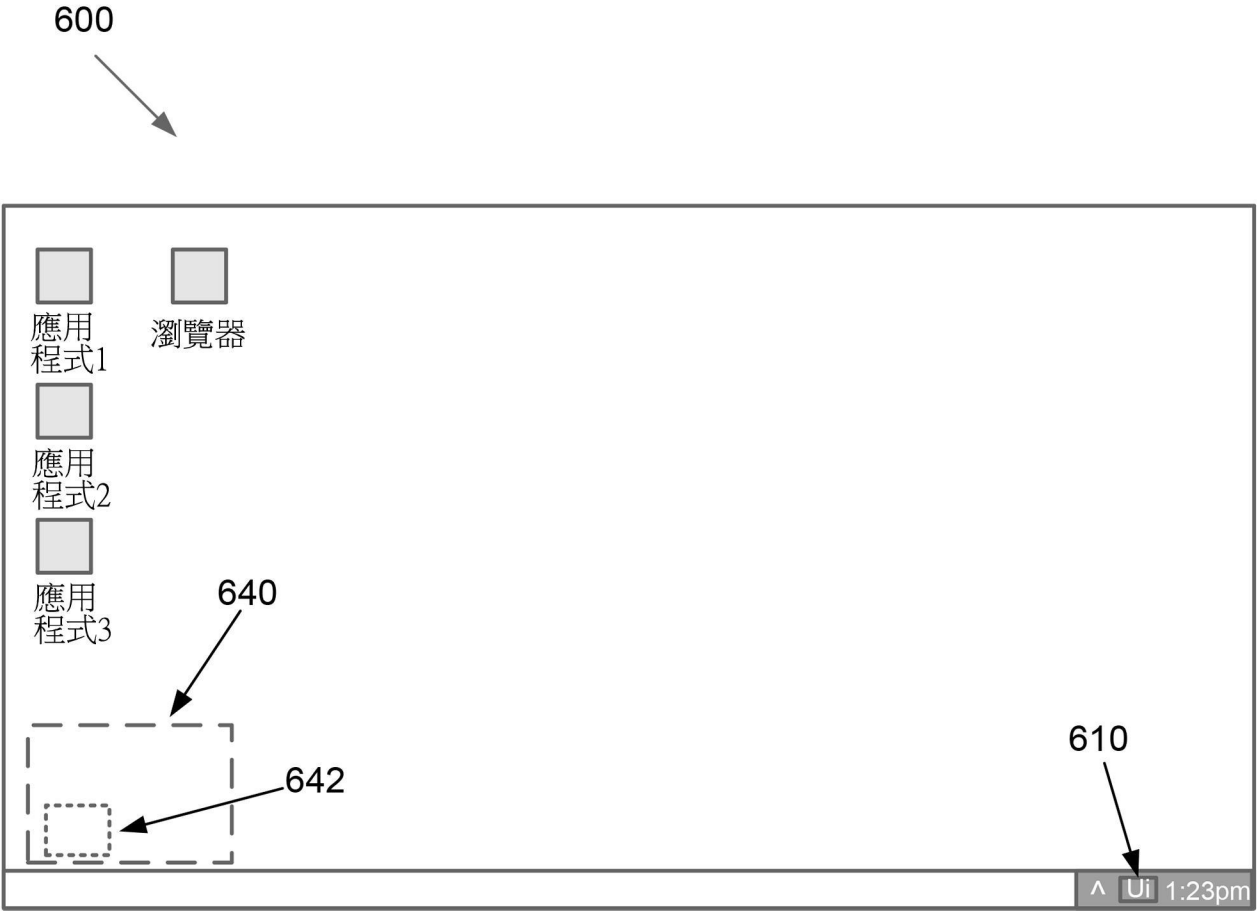


圖6A

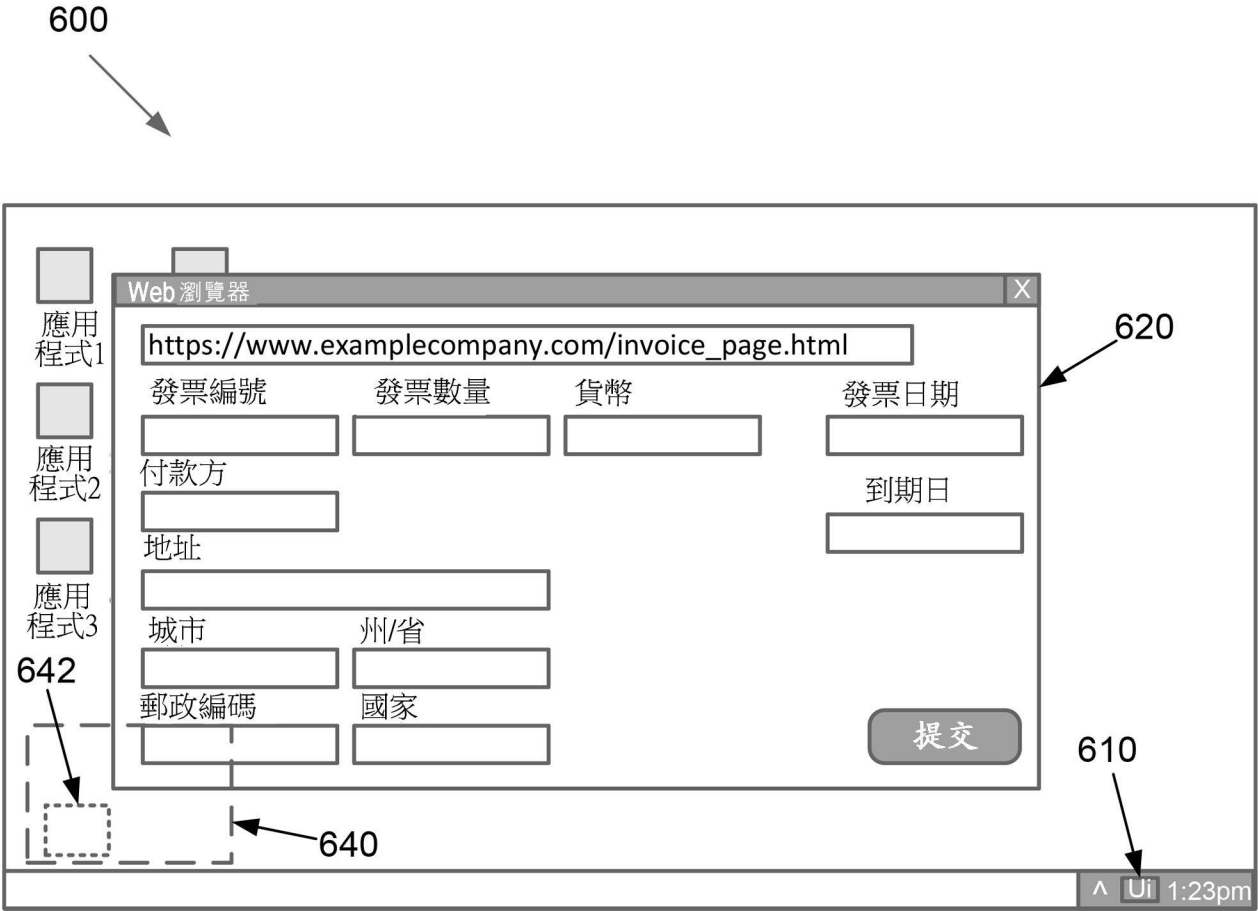


圖6B

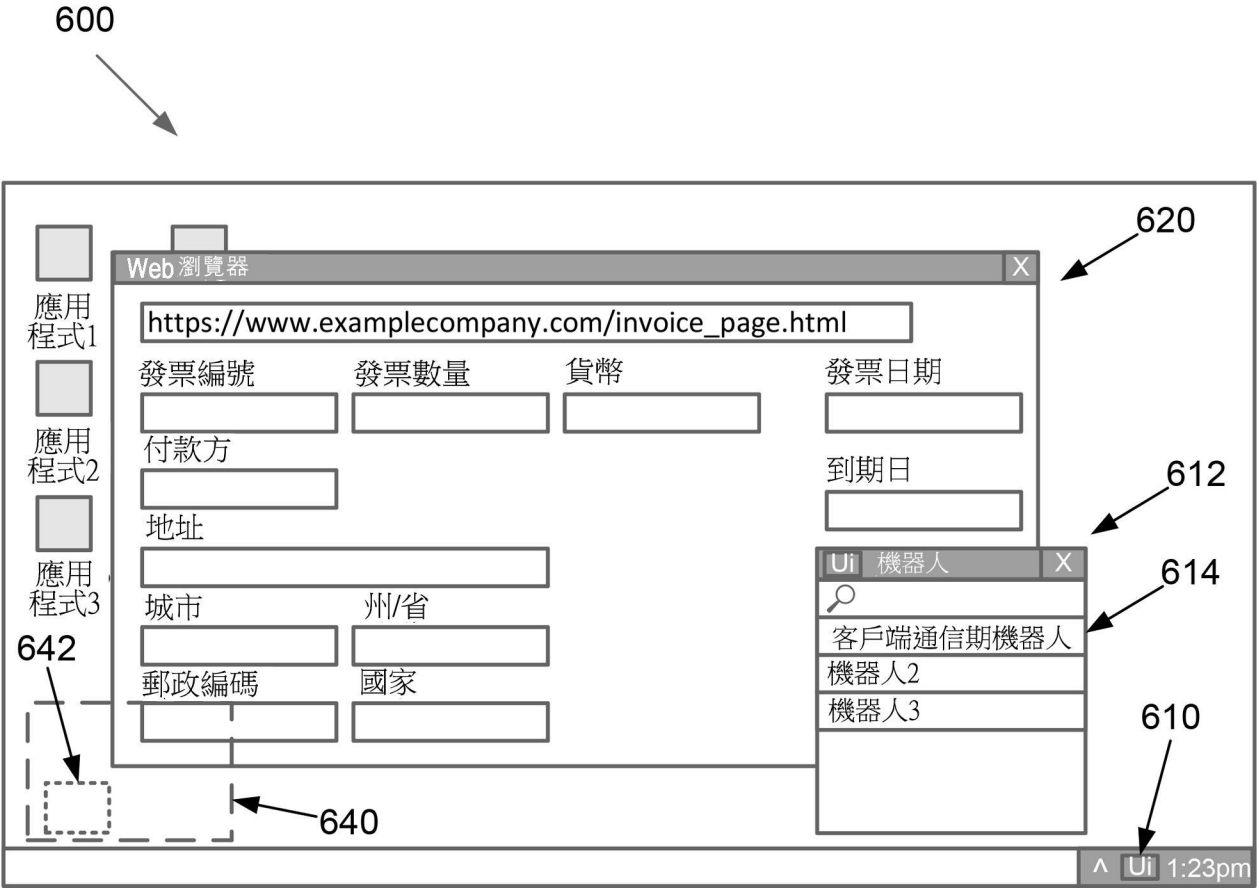


圖6C

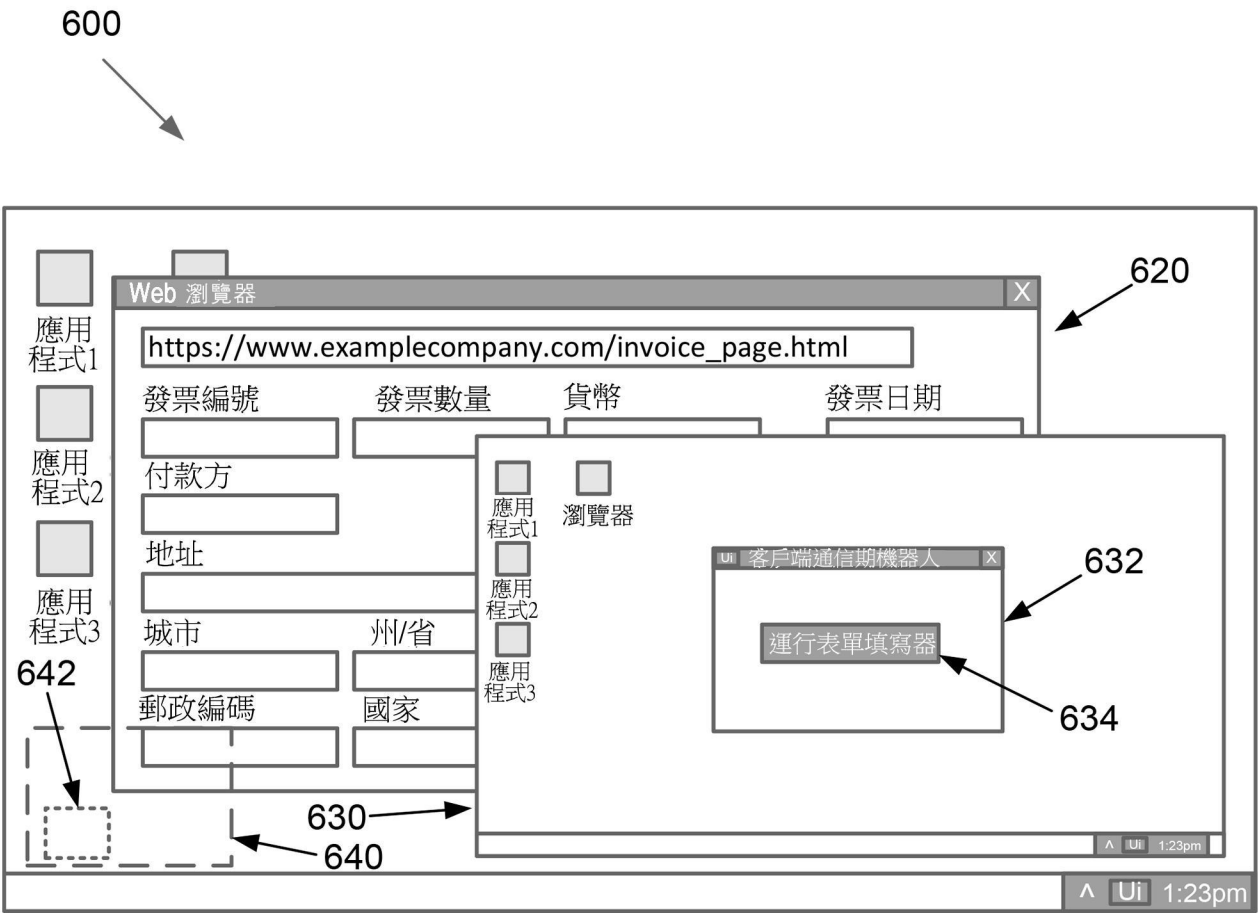


圖6D

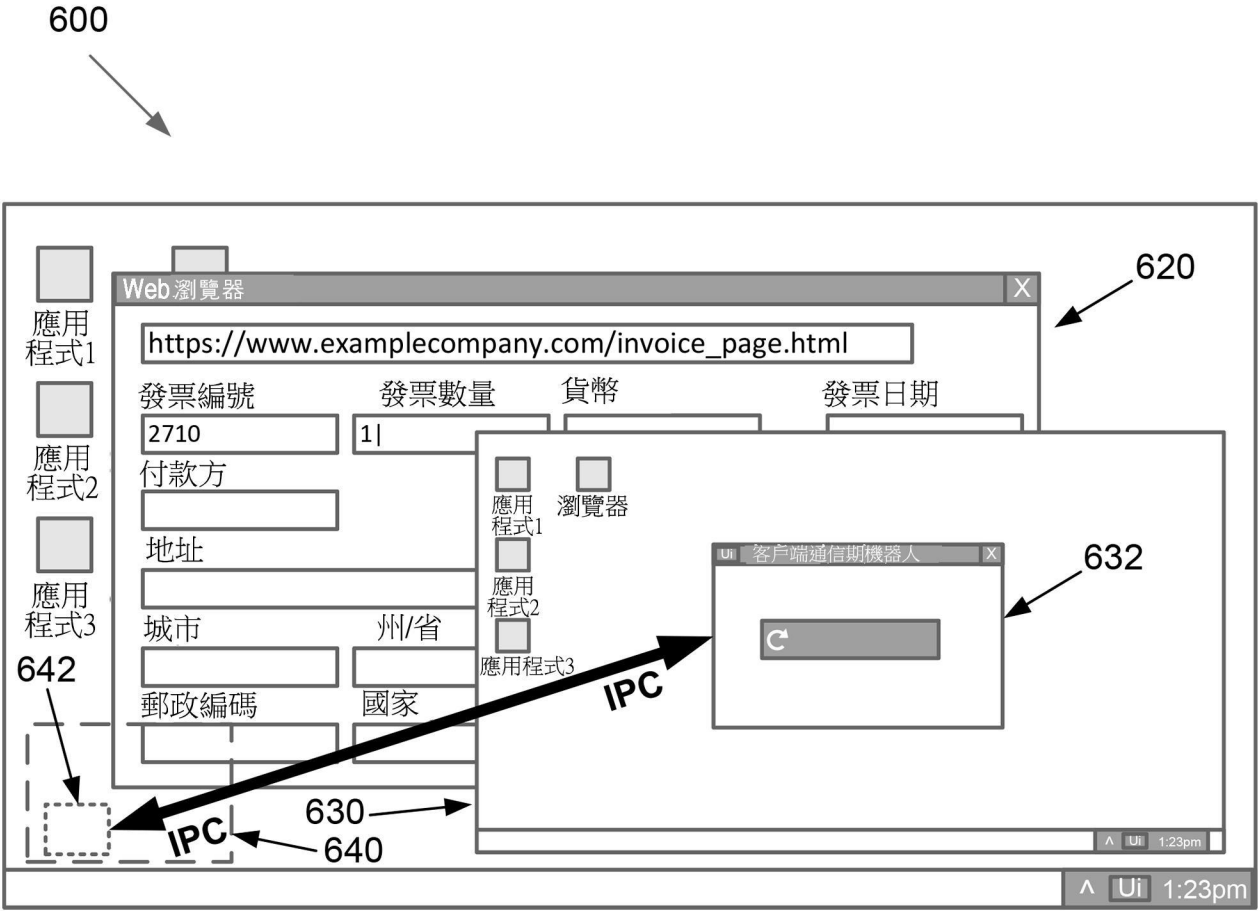


圖6E

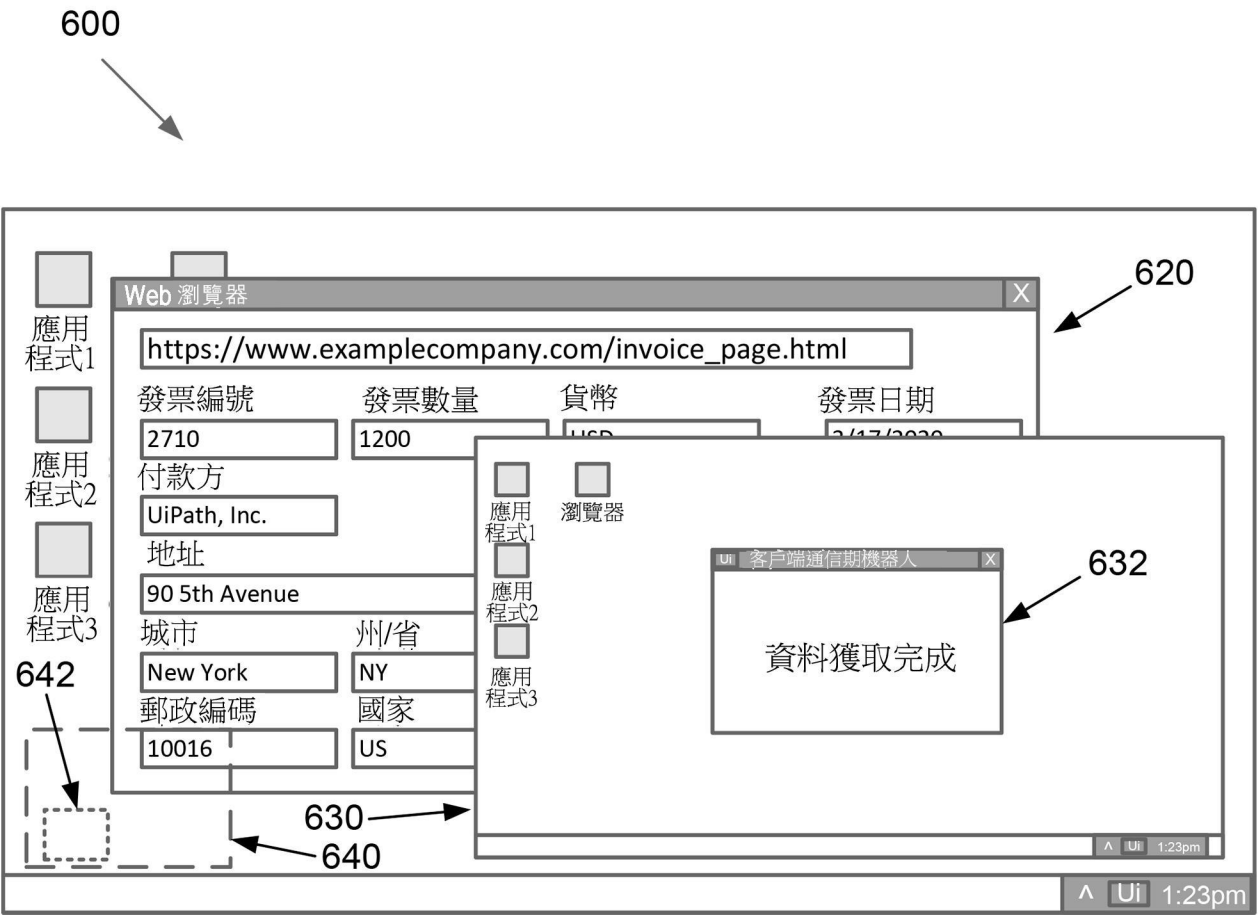


圖6F

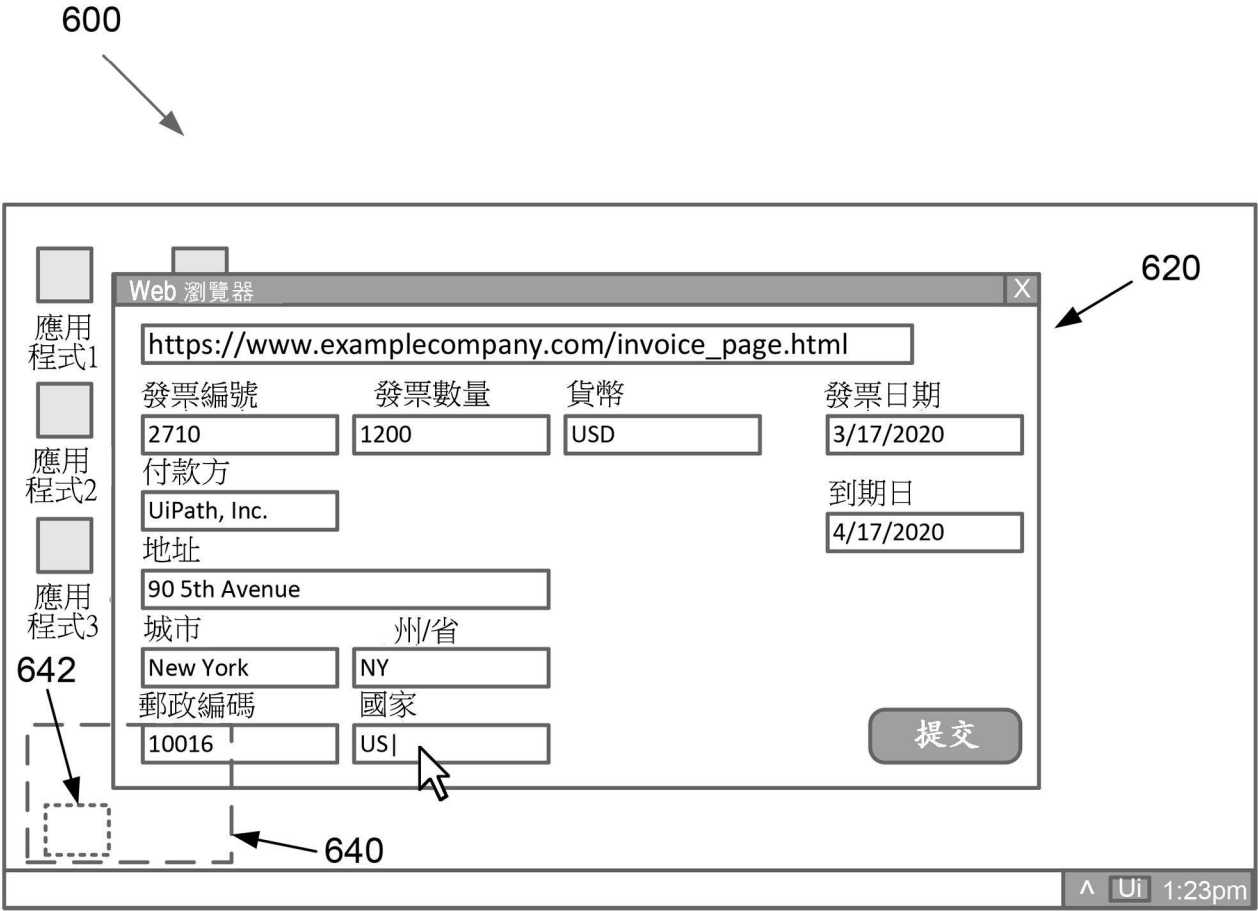


圖6G

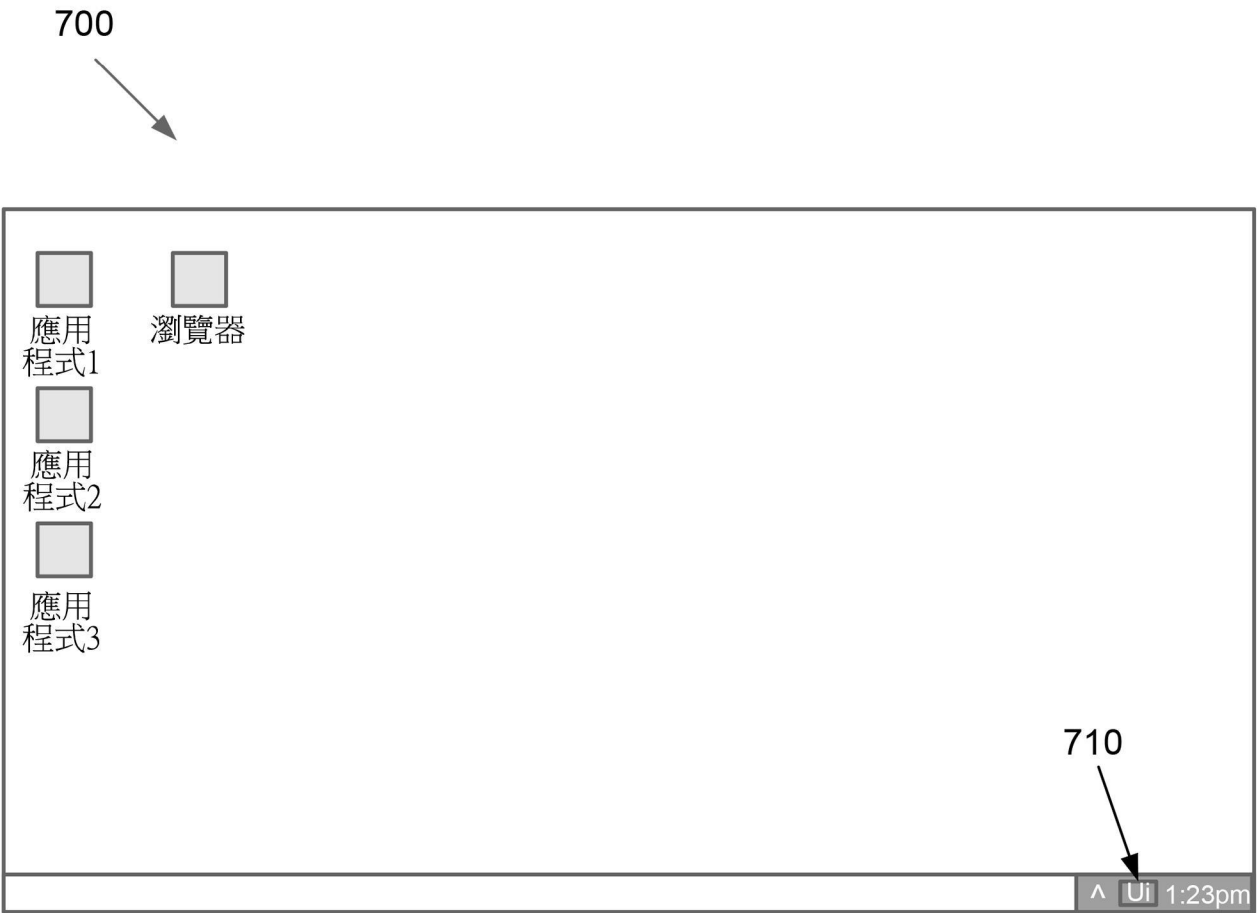


圖7A

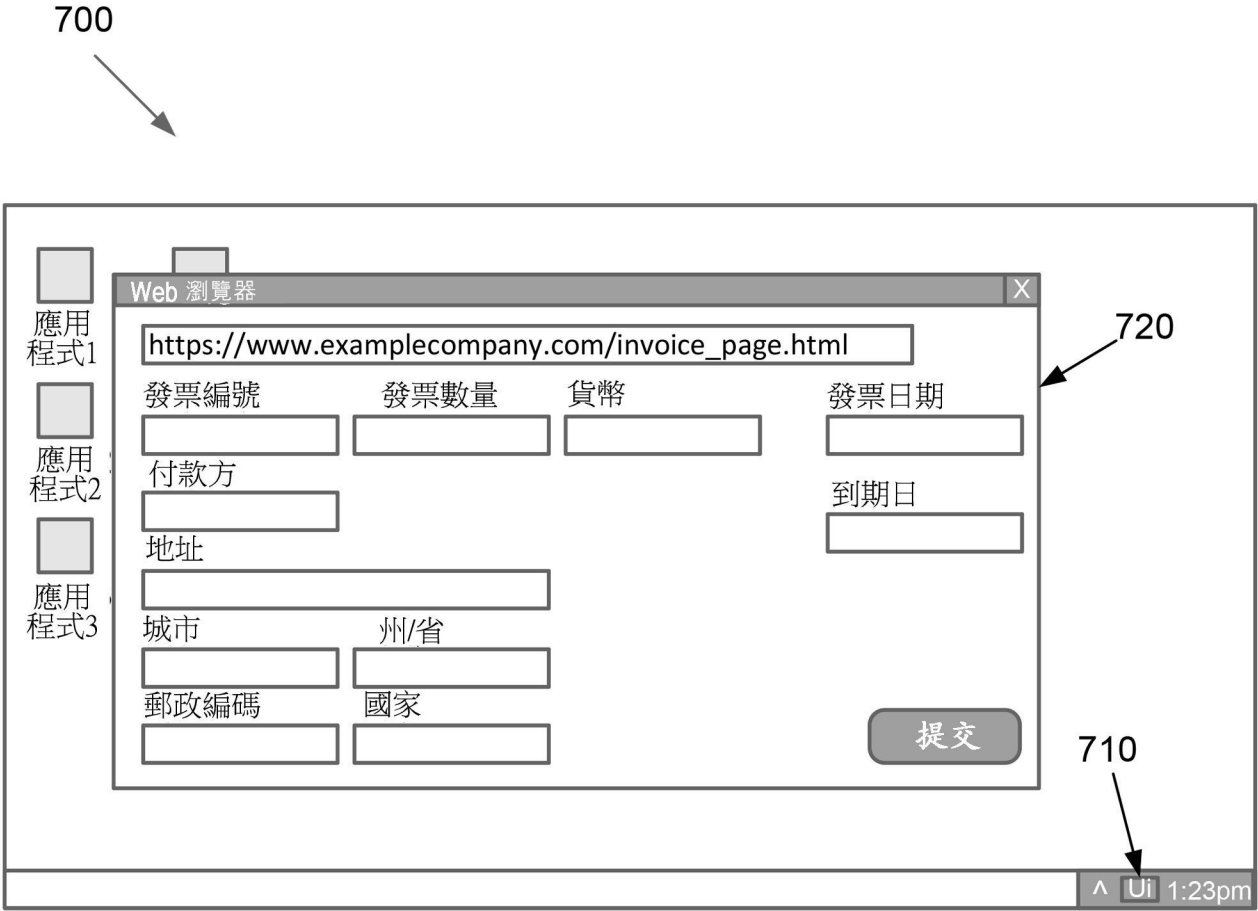


圖7B

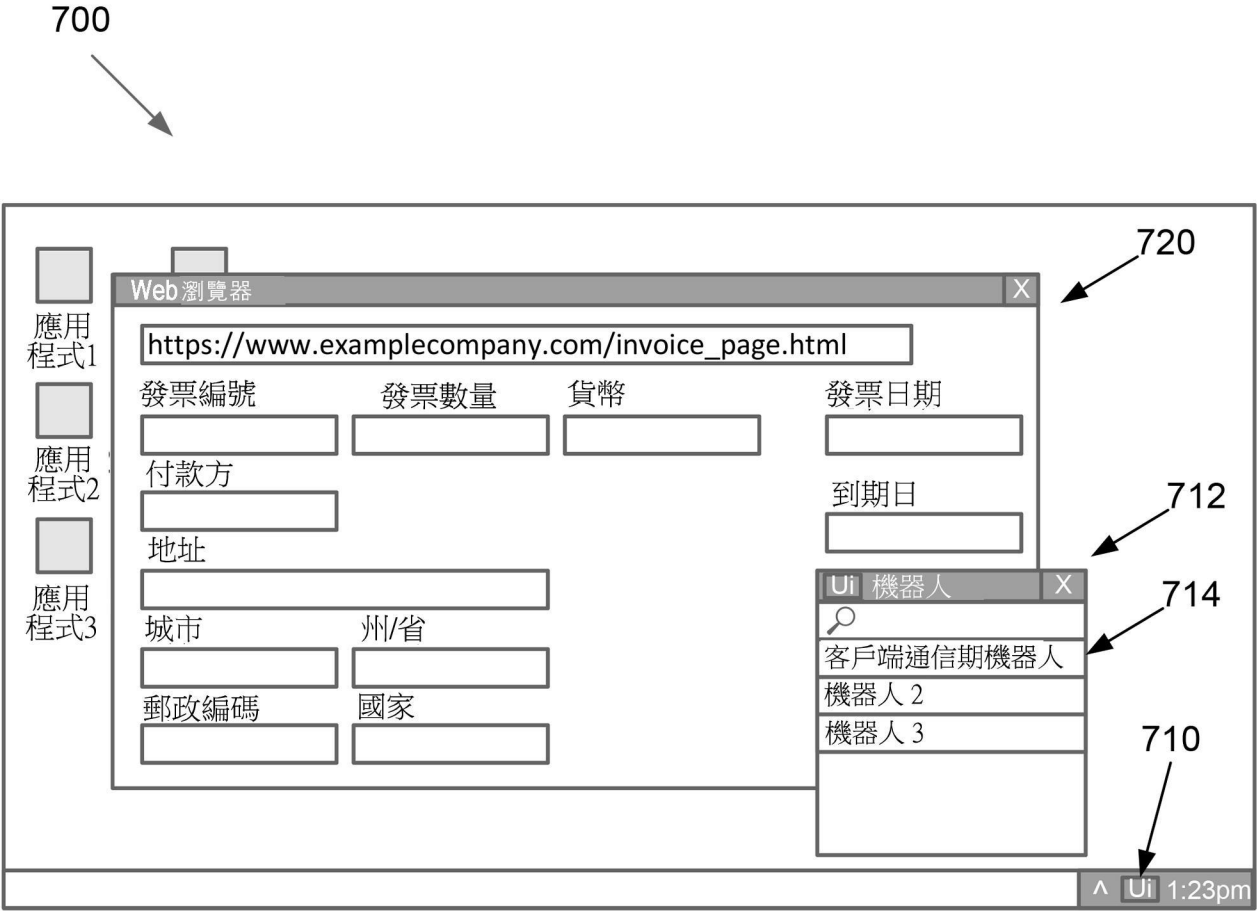


圖7C

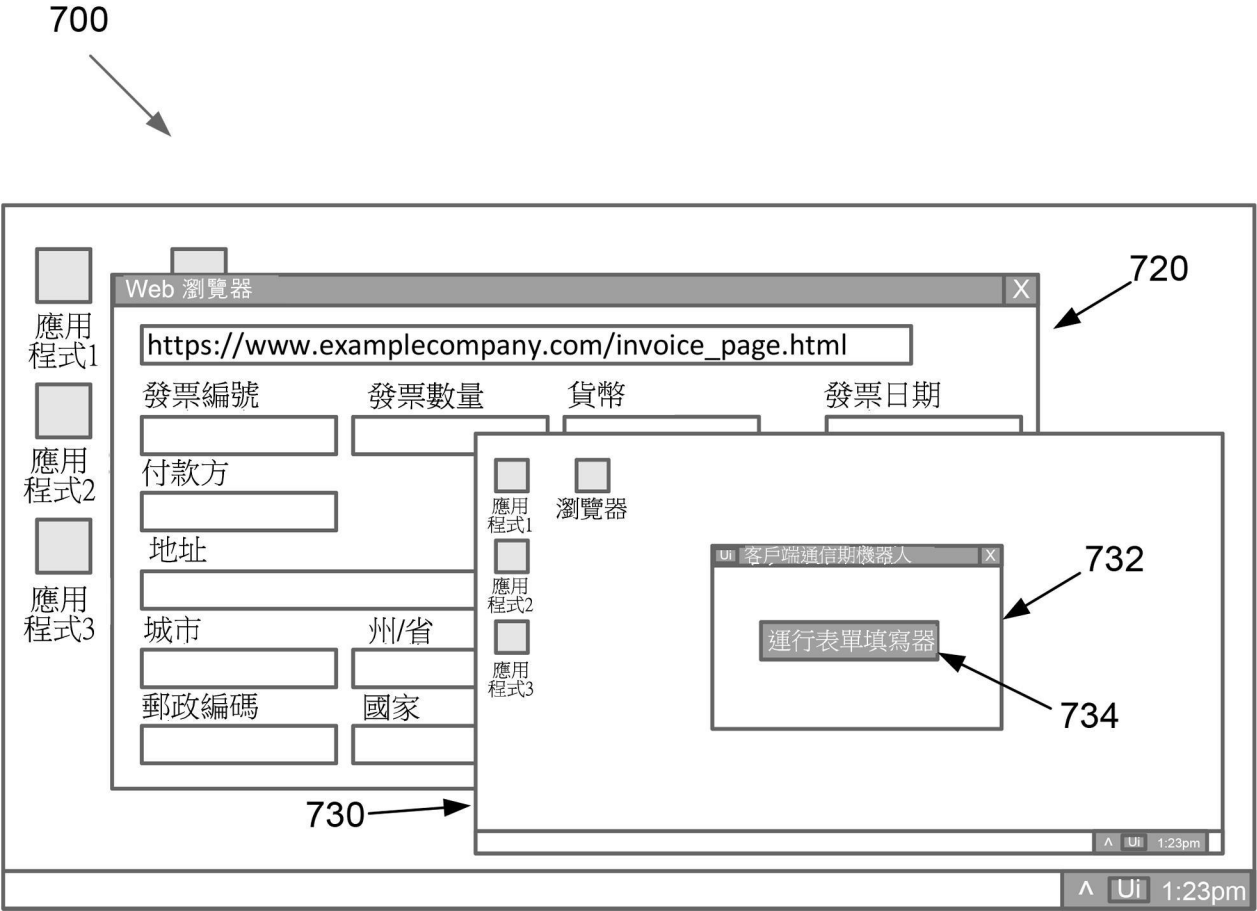


圖7D

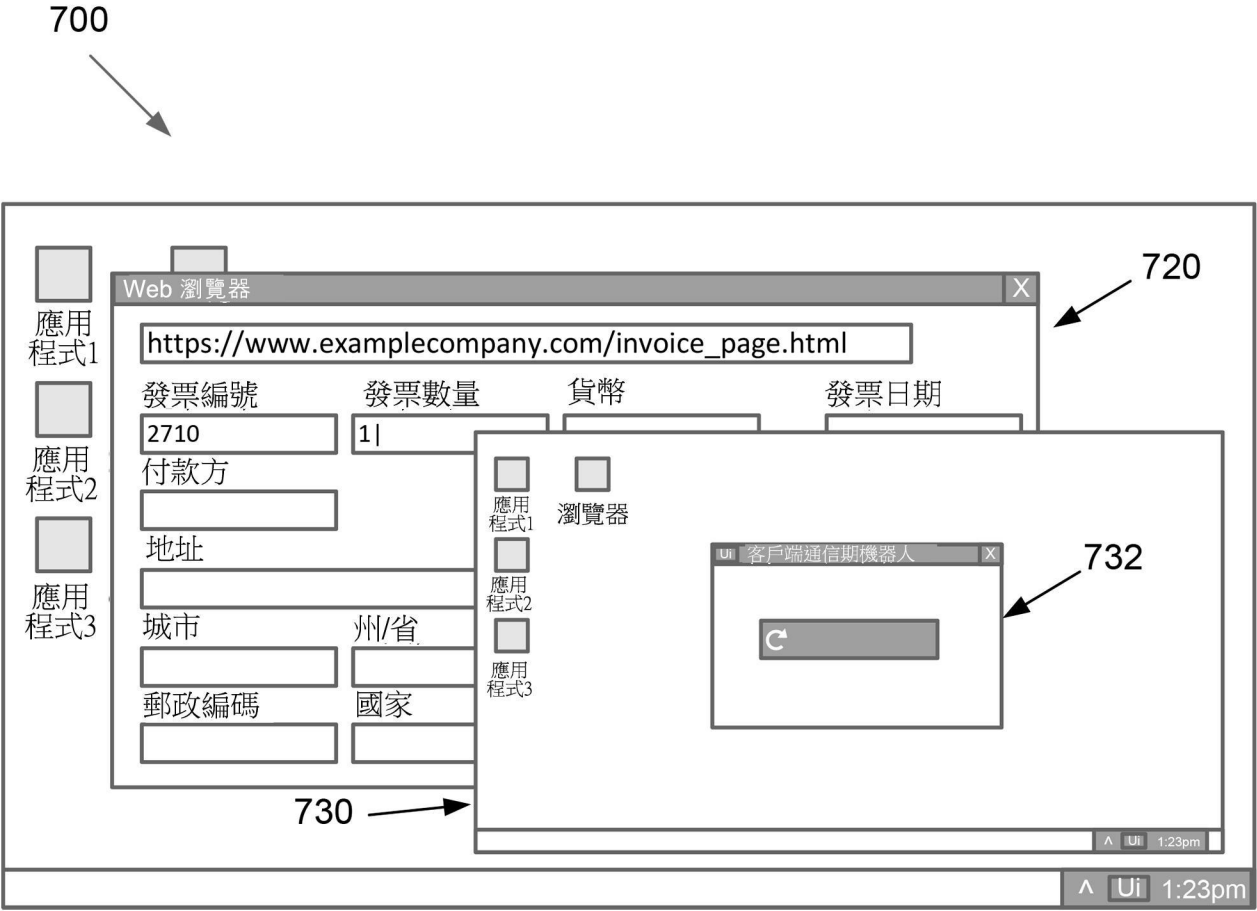


圖7E

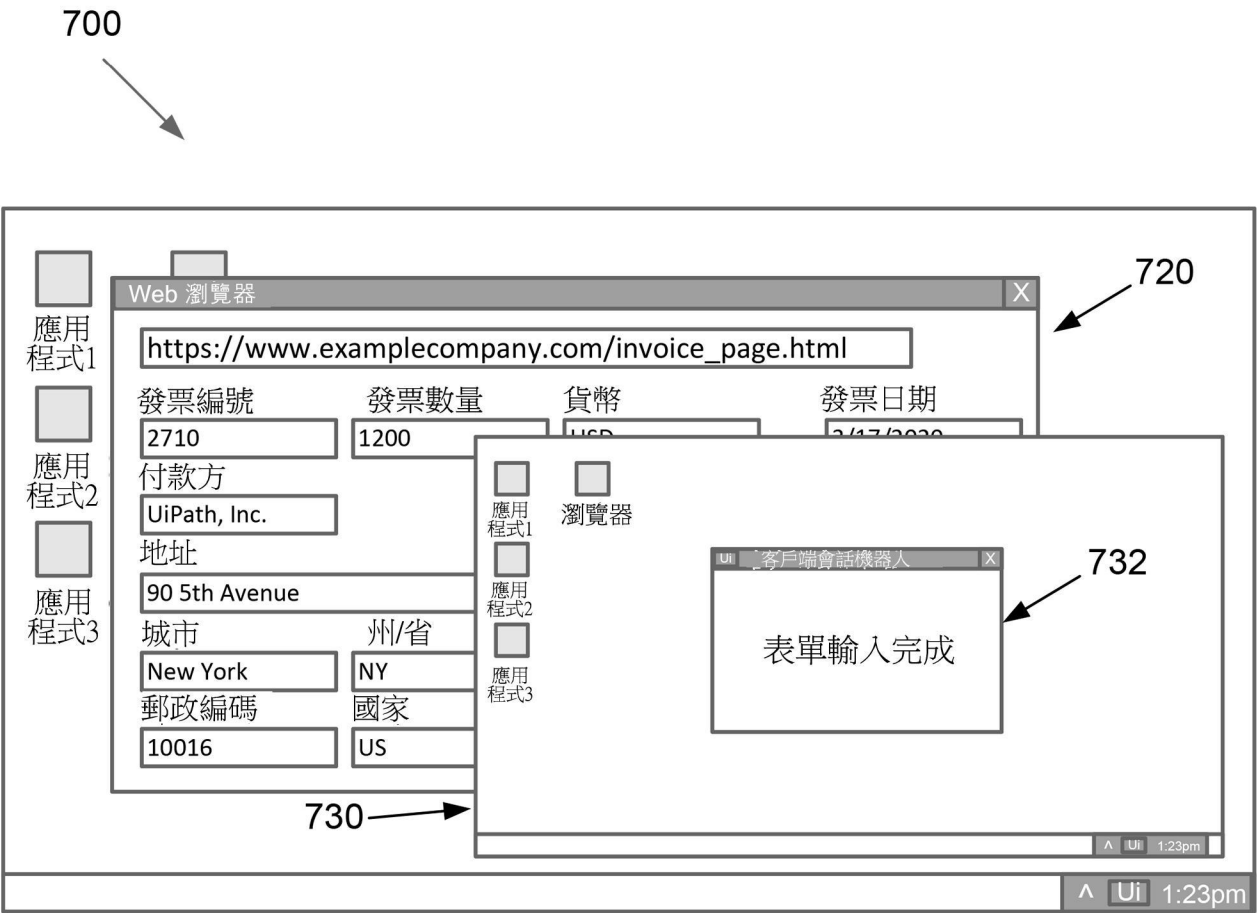


圖7F

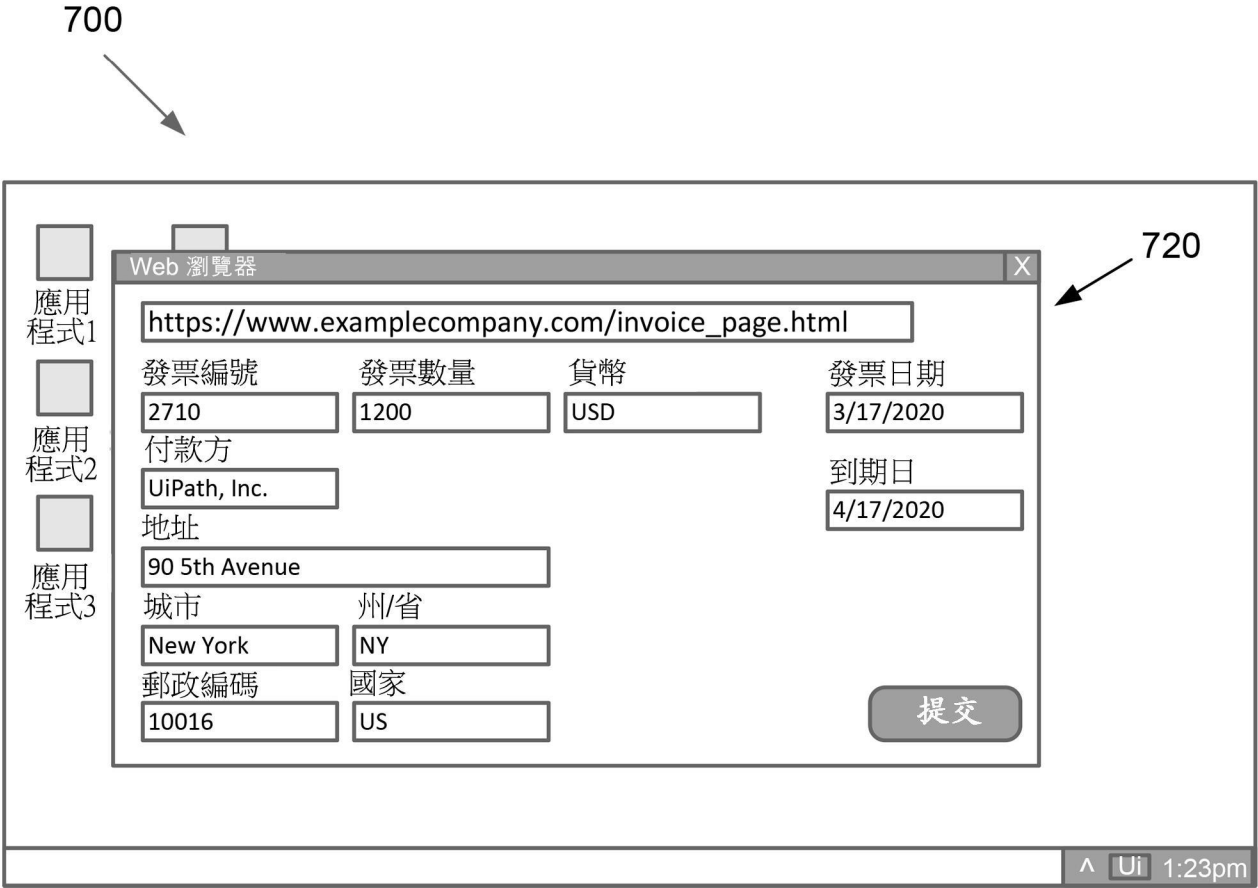


圖7G

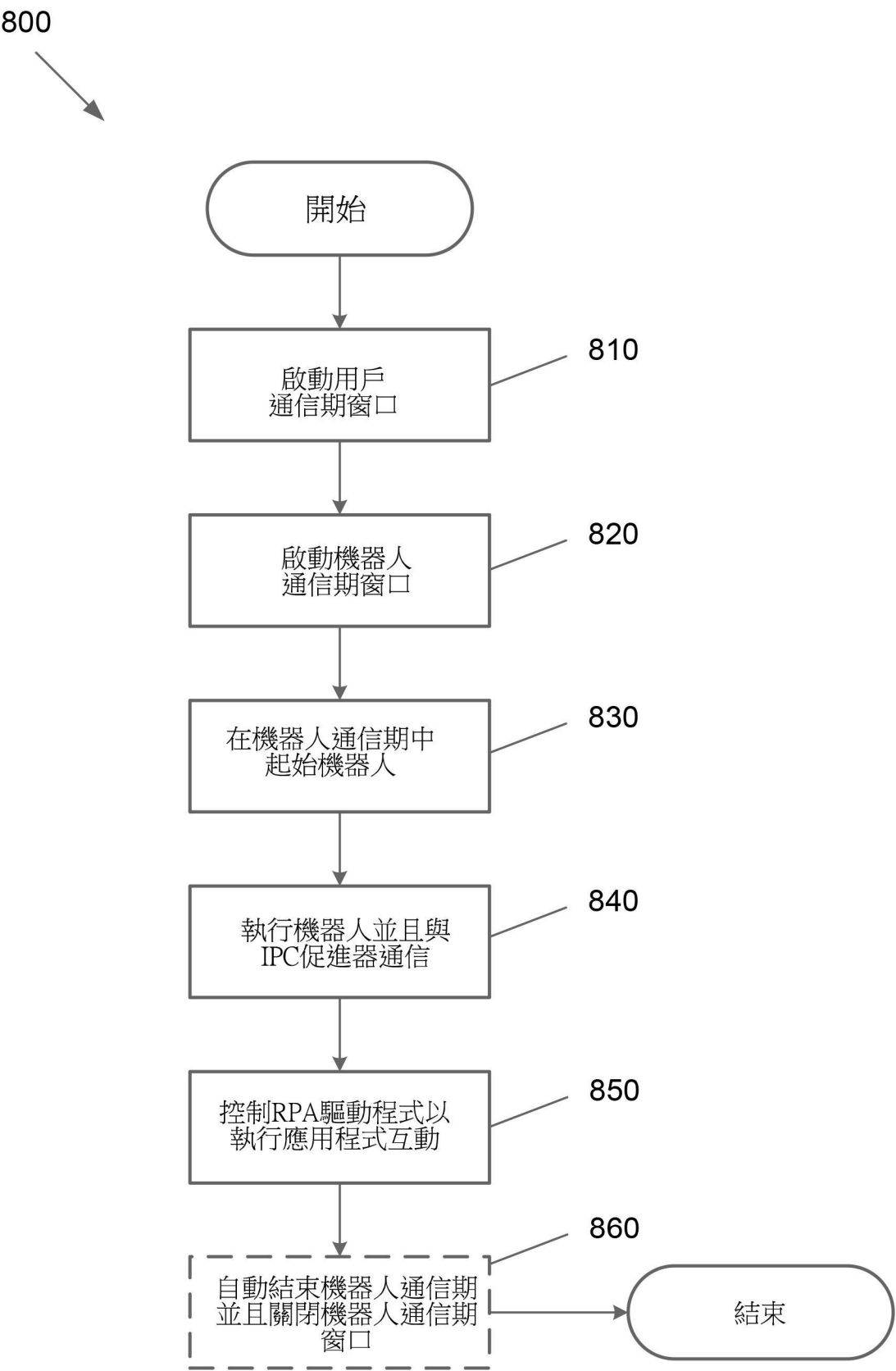


圖8

900

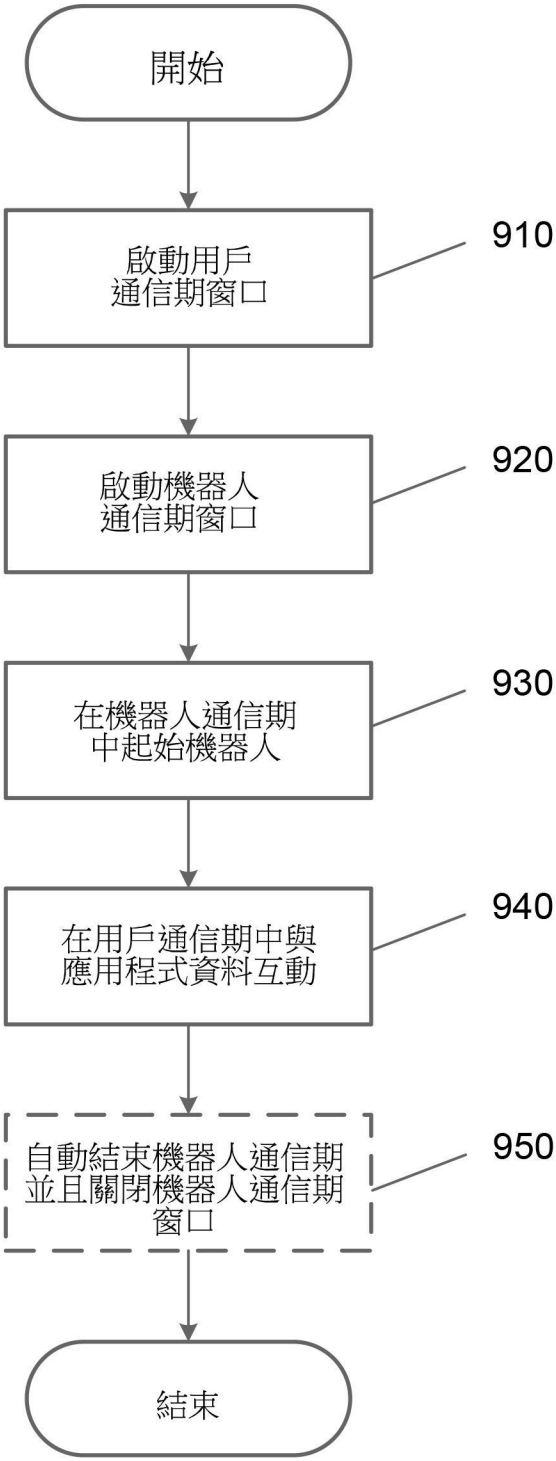


圖9

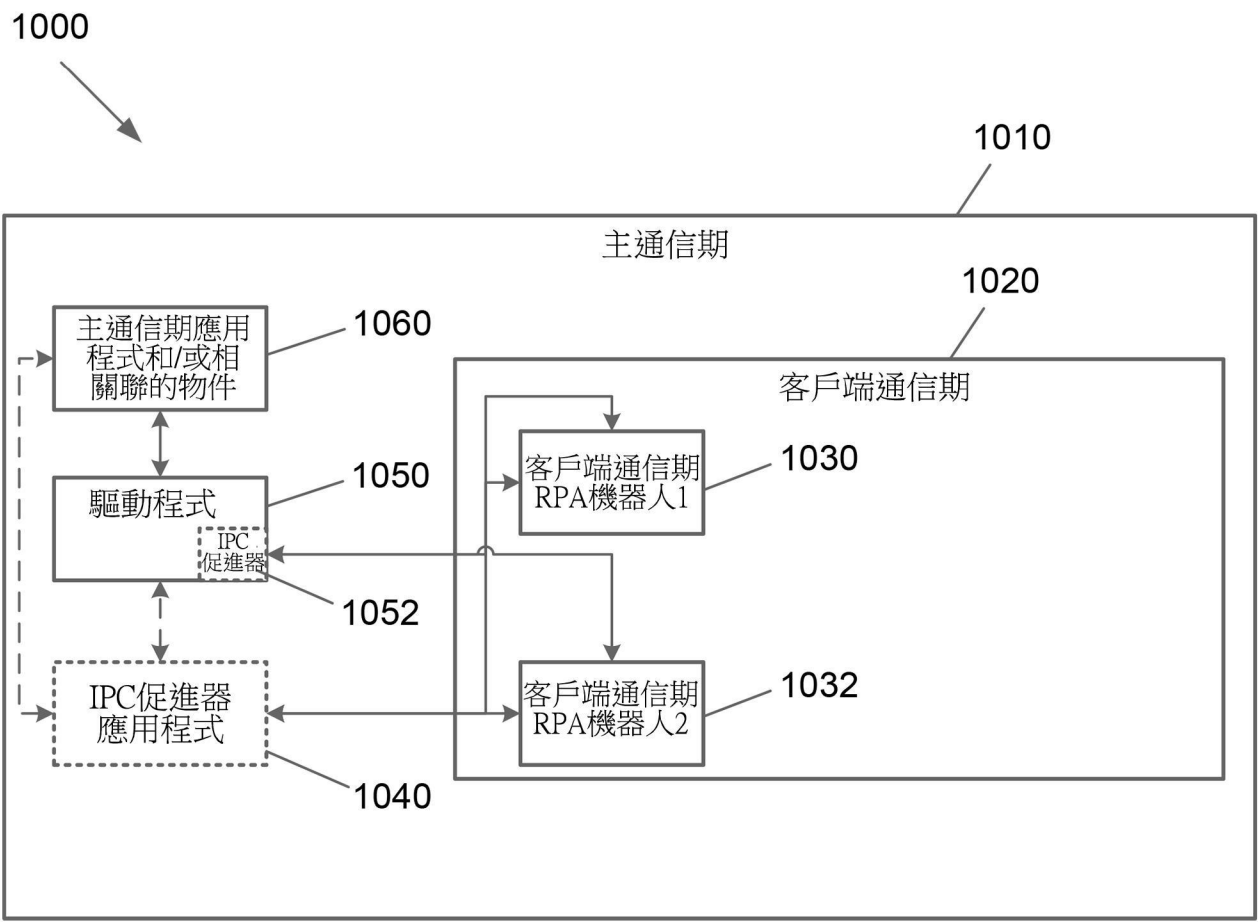


圖10

1100

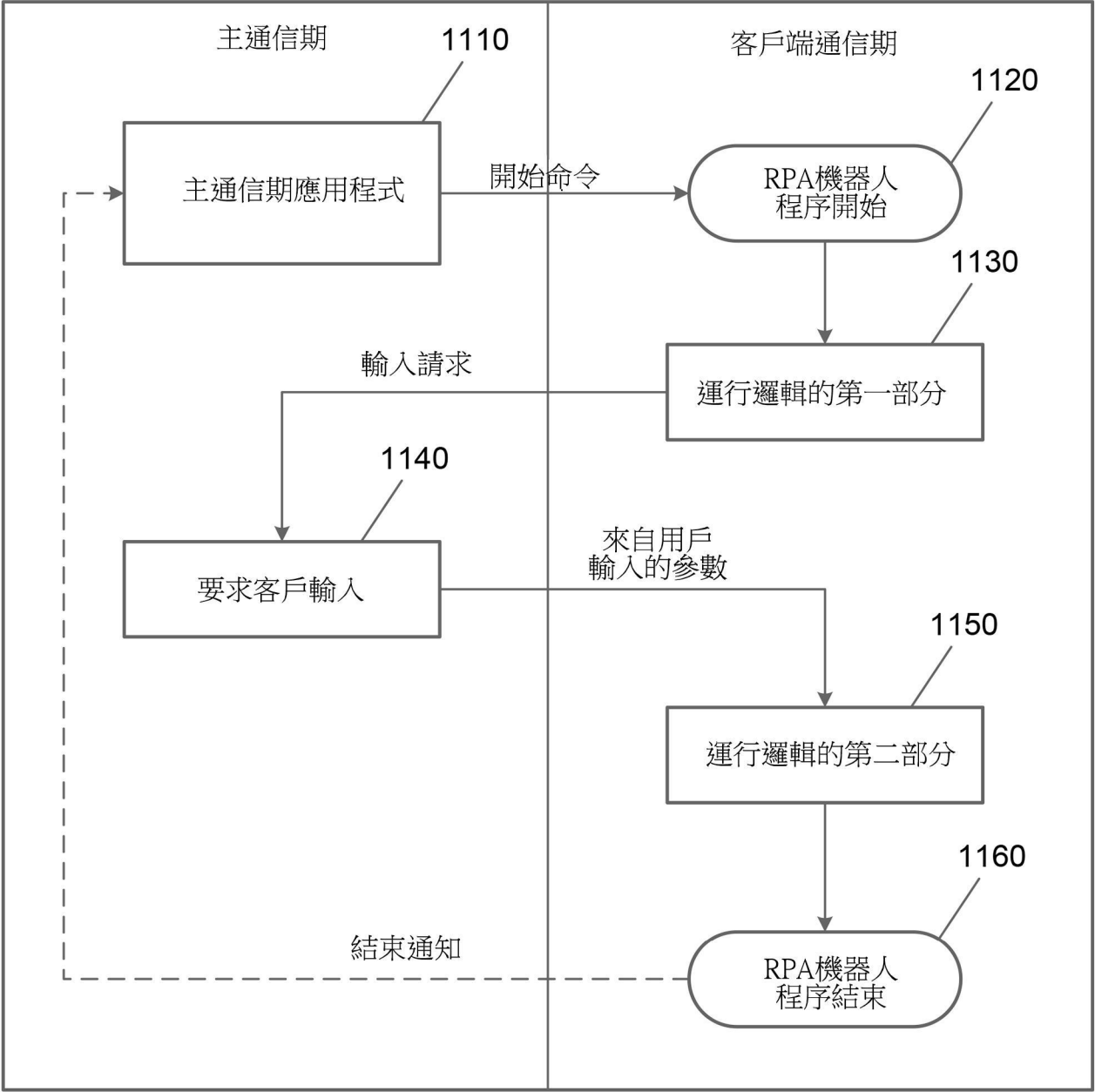


圖11

1200

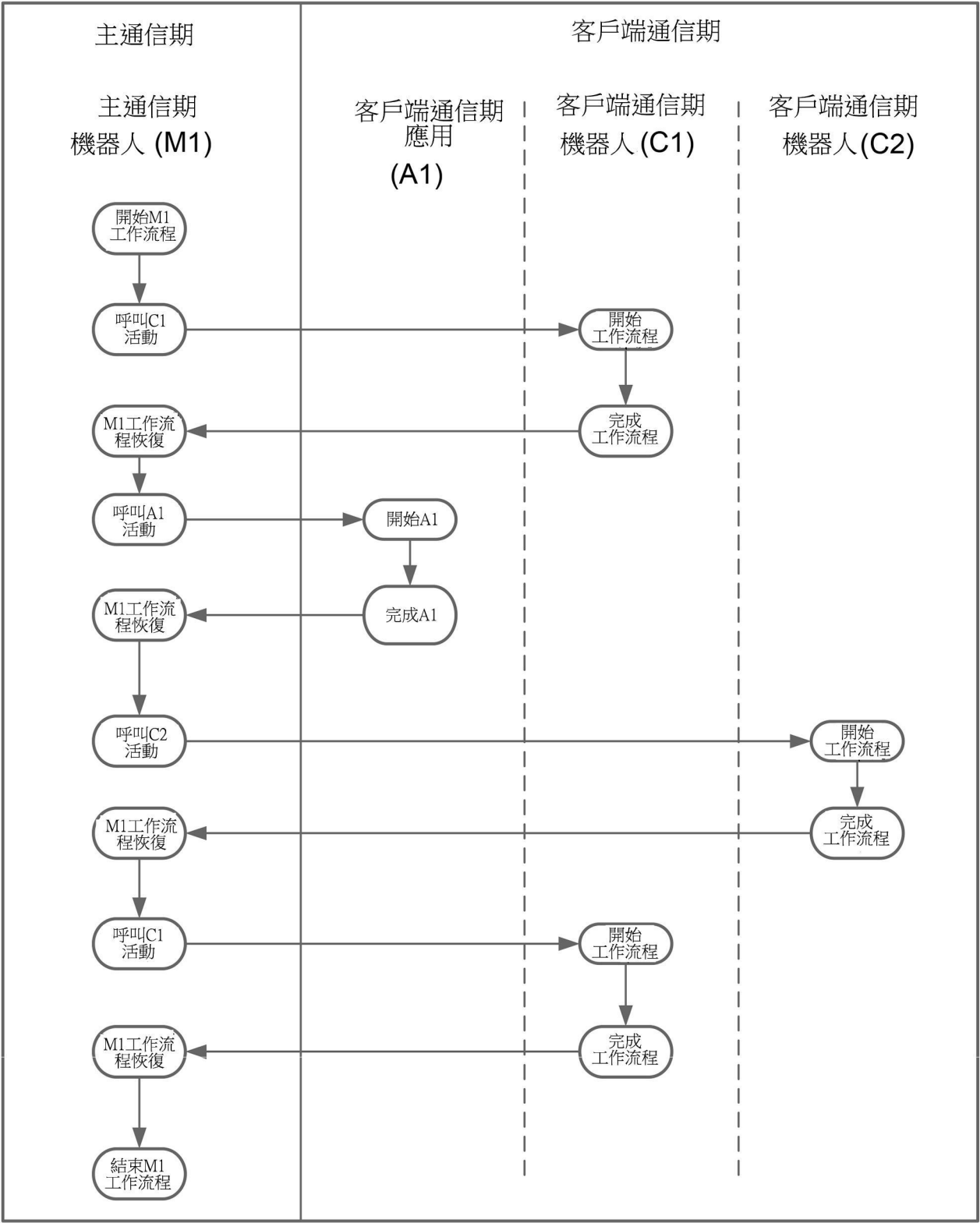


圖12

1300

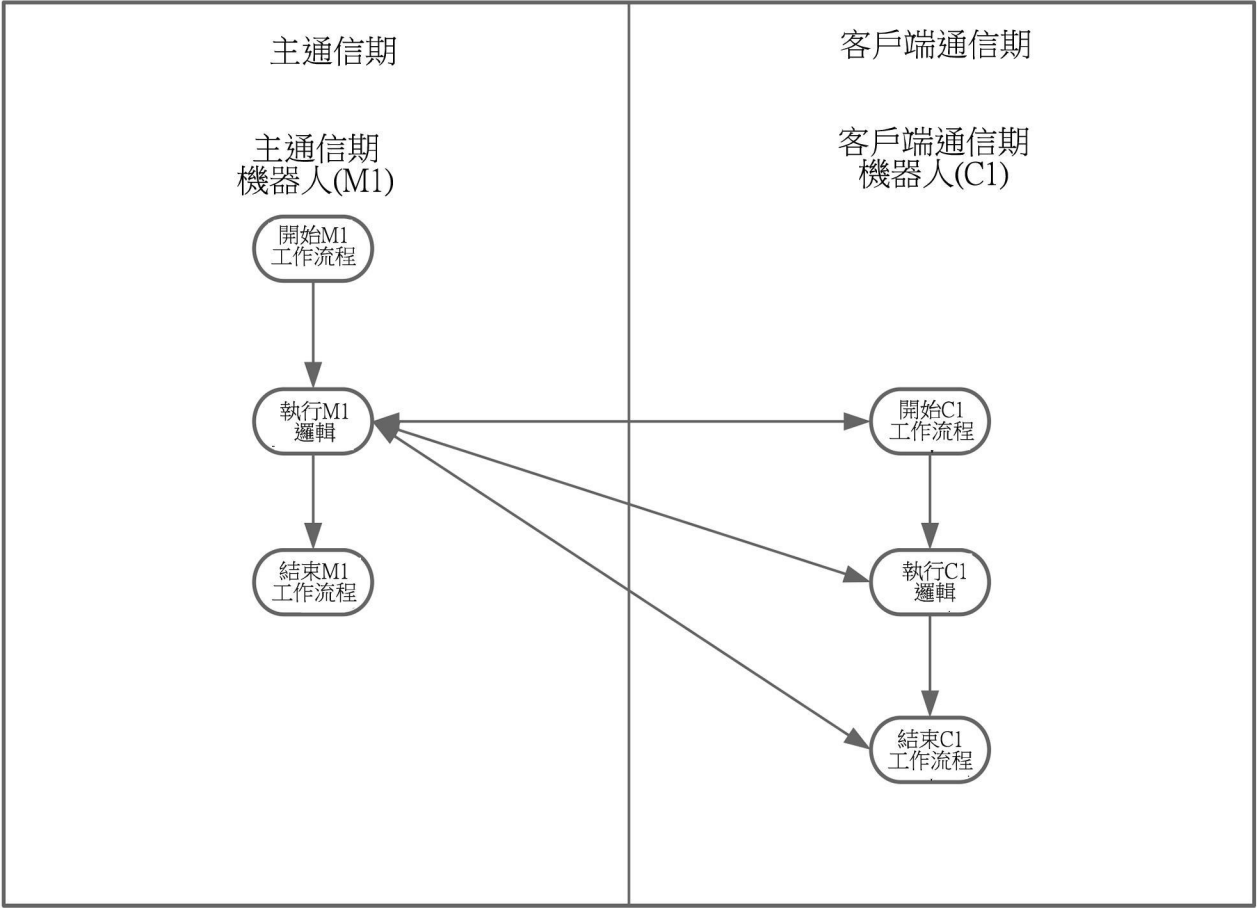


圖13

1400

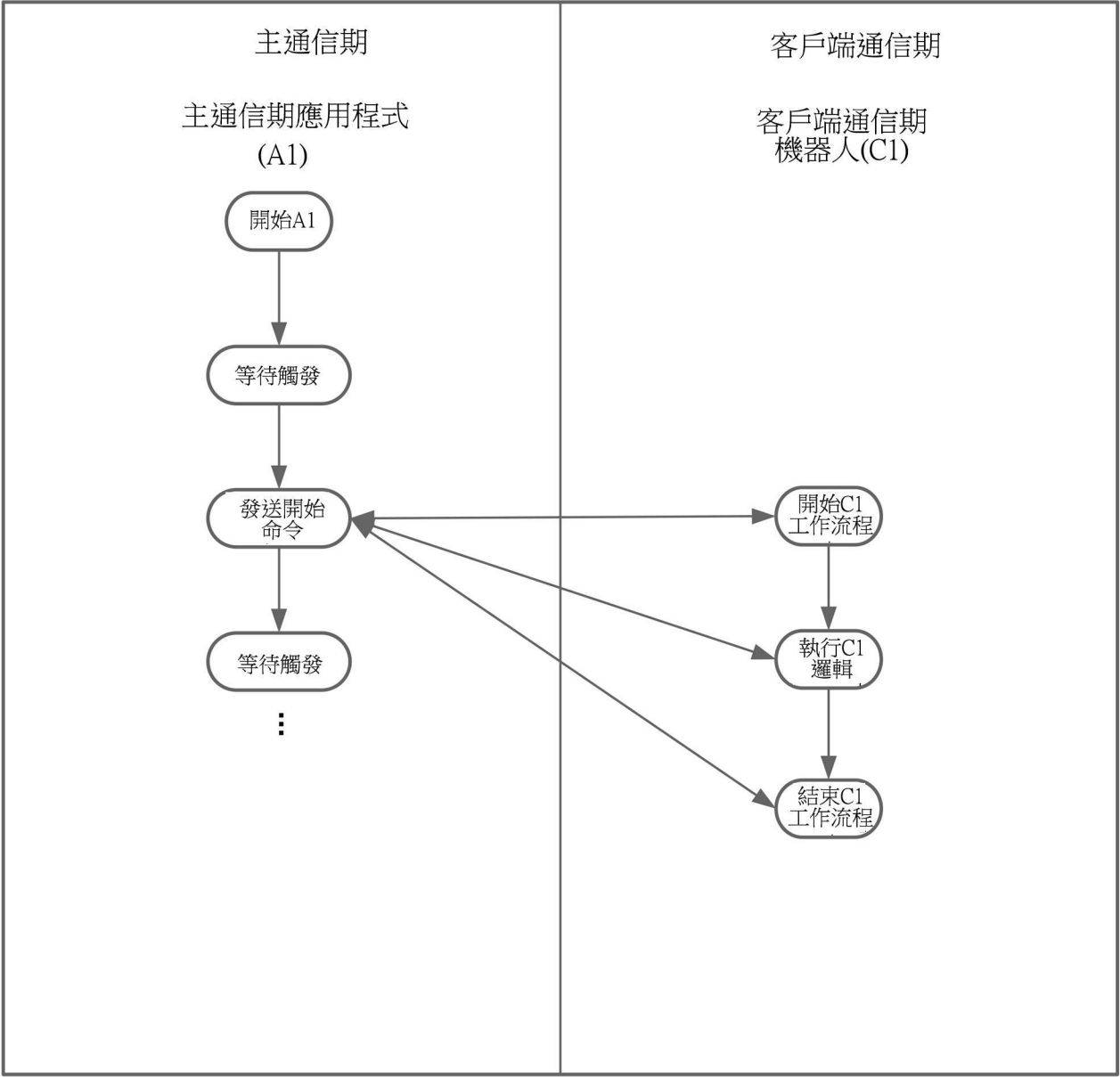


圖14

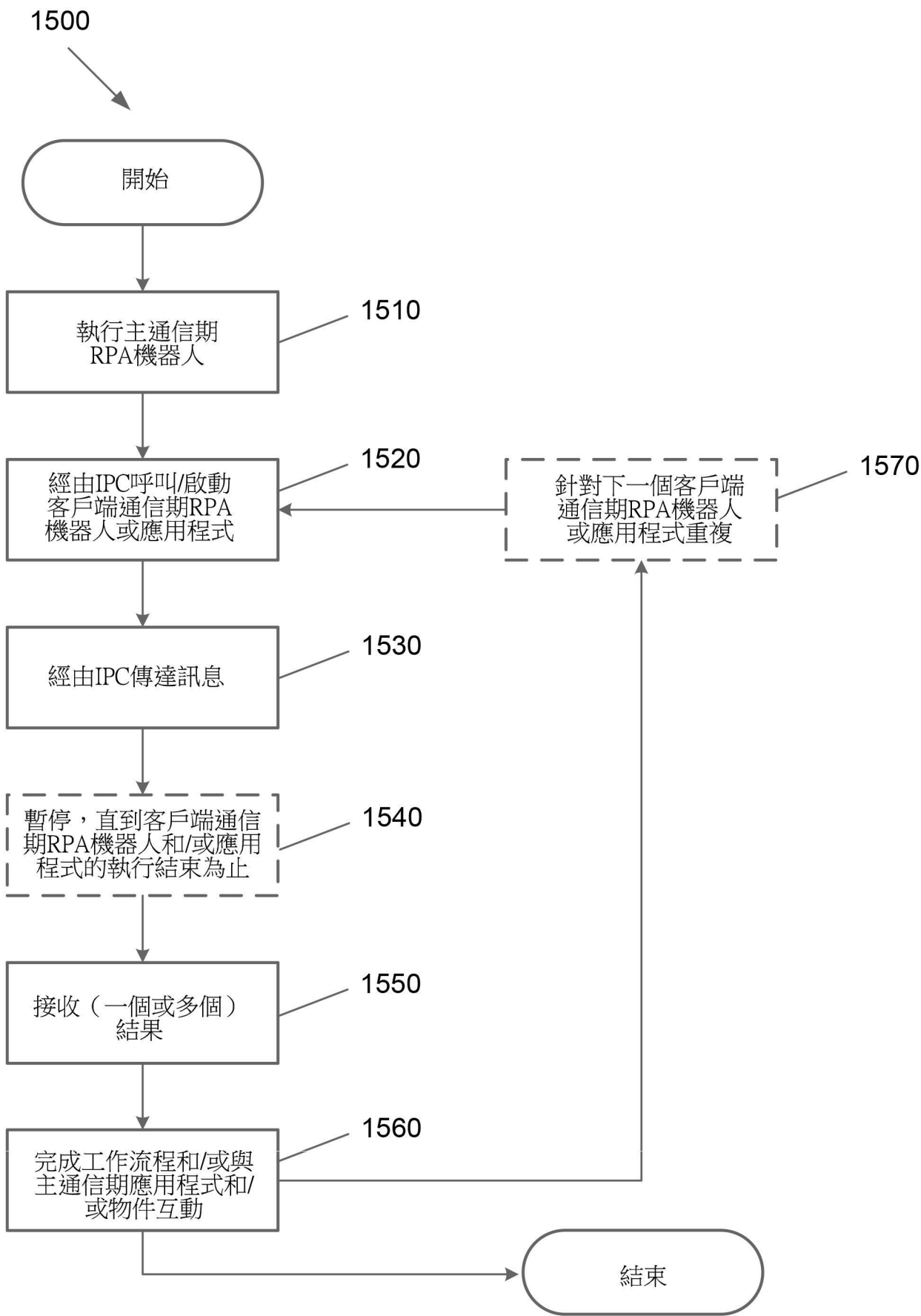


圖15