



(21)申請案號：109133857 (22)申請日：中華民國 109 (2020) 年 09 月 29 日
(51)Int. Cl. : G06F30/20 (2020.01) G06Q10/08 (2024.01)
(30)優先權：2020/07/09 美國 16/924,499
2020/07/28 美國 16/941,484
2020/09/23 世界智慧財產權組織 PCT/US20/52183
(71)申請人：美商尤帕斯公司 (美國) UIPATH, INC. (US)
美國
(72)發明人：哈爾 安卓 HALL, ANDREW (US)
(74)代理人：閻啓泰；林景郁
(56)參考文獻：
CN 110428233A CN 111242577A
US 10521582B2 US 2009/0138937A1
US 2014/0019488A1 US 2016/0119304A1
US 2020/0067923A1 US 2020/0180155A1
US 2020/0184087A1
審查人員：陳延慶
申請專利範圍項數：20 項 圖式數：17 共 72 頁

(54)名稱
用於機器人流程自動化的機器人存取控制和管理

(57)摘要
揭露了用於機器人流程自動化 (RPA) 的機器人存取控制和管理。RPA 設計器應用程式 (諸如工作流程分析器) 的代碼分析器可以讀取用於 RPA 設計器應用程式的存取控制和管理策略規則，並且根據存取控制和管理策略規則來分析 RPA 設計器應用程式的 RPA 工作流程的活動。當 RPA 工作流程的一個或多個被分析的活動違反存取控制和管理策略規則時，代碼分析器將阻止 RPA 機器人的產生或 RPA 工作流程的發佈，直到 RPA 工作流程滿足存取控制和管理策略規則。當 RPA 工作流程的被分析的活動符合所有必需的存取控制和管理策略規則時，RPA 設計器應用程式可以產生實施 RPA 工作流程的 RPA 機器人或發佈 RPA 工作流程。

Robot access control and governance for robotic process automation (RPA) is disclosed. A code analyzer of an RPA designer application, such as a workflow analyzer, may read access control and governance policy rules for an RPA designer application and analyze activities of an RPA workflow of the RPA designer application against the access control and governance policy rules. When one or more analyzed activities of the RPA workflow violate the access control and governance policy rules, the code analyzer prevents generation of an RPA robot or publication of the RPA workflow until the RPA workflow satisfies the access control and governance policy rules. When the analyzed activities of the RPA workflow comply with all required access control and governance policy rules, the RPA designer application may generate an RPA robot implementing the RPA workflow or publish the RPA workflow.

指定代表圖：

符號簡單說明：

- 1500:系統
- 1502:計算系統/桌上型電腦
- 1504:計算系統/平板電腦
- 1506:計算系統/智慧型手機
- 1510:RPA 機器人
- 1520:網路
- 1530:伺服器
- 1532:檔案
- 1540:資料庫

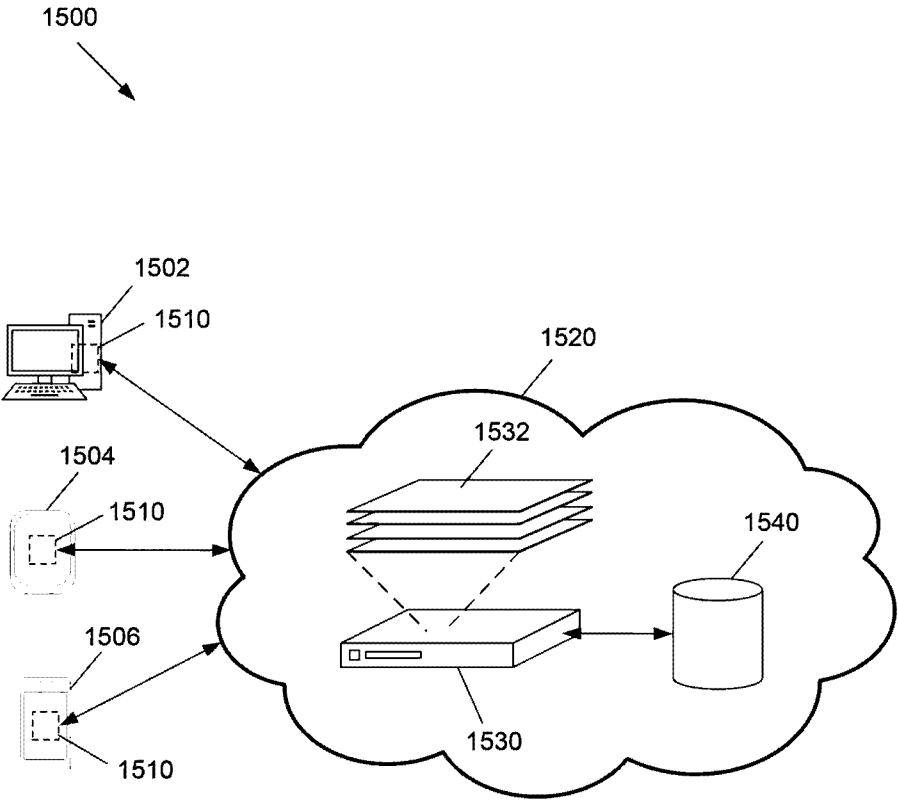


圖15



I861238

【發明摘要】

【中文發明名稱】 用於機器人流程自動化的機器人存取控制和管理

【英文發明名稱】 ROBOT ACCESS CONTROL AND GOVERNANCE FOR
ROBOTIC PROCESS AUTOMATION

【中文】

揭露了用於機器人流程自動化（RPA）的機器人存取控制和管理。RPA設計器應用程式（諸如工作流程分析器）的代碼分析器可以讀取用於RPA設計器應用程式的存取控制和管理策略規則，並且根據存取控制和管理策略規則來分析RPA設計器應用程式的RPA工作流程的活動。當RPA工作流程的一個或多個被分析的活動違反存取控制和管理策略規則時，代碼分析器將阻止RPA機器人的產生或RPA工作流程的發佈，直到RPA工作流程滿足存取控制和管理策略規則。當RPA工作流程的被分析的活動符合所有必需的存取控制和管理策略規則時，RPA設計器應用程式可以產生實施RPA工作流程的RPA機器人或發佈RPA工作流程。

【英文】

Robot access control and governance for robotic process automation (RPA) is disclosed. A code analyzer of an RPA designer application, such as a workflow analyzer, may read access control and governance policy rules for an RPA designer application and analyze activities of an RPA workflow of the RPA designer application against the access control and governance policy rules. When one or more analyzed activities of the RPA workflow violate the access control and governance policy rules, the code analyzer prevents generation of an RPA robot or publication of the RPA workflow until the RPA workflow satisfies the access control

and governance policy rules. When the analyzed activities of the RPA workflow comply with all required access control and governance policy rules, the RPA designer application may generate an RPA robot implementing the RPA workflow or publish the RPA workflow.

【指定代表圖】 圖15

【代表圖之符號簡單說明】

1500:系統

1502:計算系統/桌上型電腦

1504:計算系統/平板電腦

1506:計算系統/智慧型手機

1510:RPA機器人

1520:網路

1530:伺服器

1532:檔案

1540:資料庫

【特徵化學式】

無

【發明說明書】

【中文發明名稱】 用於機器人流程自動化的機器人存取控制和管理

【英文發明名稱】 ROBOT ACCESS CONTROL AND GOVERNANCE FOR
ROBOTIC PROCESS AUTOMATION

【技術領域】

【0001】 本發明大體上涉及機器人流程自動化（robotic process automation；RPA），並且更具體地涉及用於RPA的機器人存取控制和管理。

【0002】 本案主張於2020年7月28日提出申請的美國非臨時專利申請案第16/941,484號的權益，該申請案是2020年7月9日提出申請的美國非臨時專利申請案第16/924,499號的接續案，並且主張其權益。這些早前提出申請的申請案之標的以其整體內容通過引用的方式併入於本文中。

【先前技術】

【0003】 計算系統的使用者採取的某些動作可能會違反美國和其他國家的法律或協議，諸如歐盟通用資料保護條例（General Data Protection Regulation；GDPR）、美國健康保險可攜性和責任法案（Health Insurance Portability and Accountability Act；HIPAA）、第三方服務條款等。違反這些法律或協議的罰款可能會很高。例如，在採用GDPR時，較低層級的違規行為可能會被處以1000萬歐元的罰款或違規者全球年收入（而非利潤）的2%（以較高者為準）。更嚴重的違規行為將導致罰款2000萬歐元，或違反者年收入的4%（以較高者為準）。這些罰款可能會對公司造成經濟損失甚至致命。RPA對存取控制和管理提出了獨特的挑戰。因此，改進的RPA存取控制和管理方法會是有益的。

【發明內容】

【0004】 本發明的某些實施例可以對本領域中尚未被當前的RPA技術完全標識、理解或解決的問題和需求提供解決方案。例如，本發明的一些實施例涉及用於RPA的機器人存取控制和管理。

【0005】 在一個實施例中，一種用於執行用於RPA的機器人存取控制和管理電腦程式體現在非暫時性電腦可讀取媒體上。電腦程式被配置為引起至少一個處理器讀取用於RPA設計器應用程式的存取控制和管理策略規則，並且根據存取控制和管理策略規則來分析RPA設計器應用程式的RPA工作流程的活動。當RPA工作流程的一個或多個被分析的活動違反存取控制和管理策略規則時，電腦程式被配置為引起至少一個處理器阻止RPA機器人的產生或RPA工作流程的發佈，直到RPA工作流程滿足存取控制和管理策略規則。當RPA工作流程的被分析的活動符合所有必需的存取控制和管理策略規則時，電腦程式被配置為引起至少一個處理器產生實施RPA工作流程的RPA機器人或發佈RPA工作流程。

【0006】 在另一實施例中，一種用於執行用於RPA的機器人存取控制和管理電腦實施的方法包括由代碼分析器根據存取控制和管理策略規則來分析RPA設計器應用程式的RPA工作流程的活動。當RPA工作流程的一個或多個被分析的活動違反存取控制和管理策略規則時，該電腦實施的方法包括由代碼分析器阻止RPA機器人的產生或RPA工作流程的發佈，直到RPA工作流程滿足存取控制和管理策略規則。

【0007】 在又一實施例中，一種用於執行用於RPA的機器人存取控制和管理電腦程式體現在非暫時性電腦可讀取媒體上。電腦程式被配置為引起至少一個處理器從計算系統的註冊表項中確定通往包括存取控制和管理策略規則的檔案的連結，並且使用所確定的連結來下載檔案，或者從指揮器應用程式中下載管理策略規則。電腦程式還被配置為引起至少一個處理器從所下載的檔案中

讀取用於RPA設計器應用程式的存取控制和管理策略規則，並且根據存取控制和管理策略規則來分析RPA設計器應用程式的RPA工作流程的活動。當RPA工作流程的一個或多個被分析的活動違反存取控制和管理策略規則時，電腦程式被配置為引起至少一個處理器阻止RPA機器人的產生或RPA工作流程的發佈，直到RPA工作流程滿足存取控制和管理策略規則。當RPA工作流程的被分析的活動符合所有必需的存取控制和管理策略規則時，電腦程式被配置為引起至少一個處理器產生實施RPA工作流程的RPA機器人或發佈RPA工作流程。存取控制和管理策略規則包括一個或多個應用程式和/或統一資源定位符（universal resource locator；URL）限制、一個或多個套裝程式限制、一個或多個活動限制、一個或多個活動屬性要求、或其組合。

【圖式簡單說明】

【0008】 為了有助於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。然而應該要理解，這些圖式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定性和細節來描述和闡釋，在附圖中：

【0009】 [圖1]是示出根據本發明的實施例的機器人流程自動化（RPA）系統的架構圖。

【0010】 [圖2]是示出根據本發明的實施例的部署的RPA系統的架構圖。

【0011】 [圖3]是示出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。

【0012】 [圖4]是示出根據本發明的實施例的RPA系統的架構圖。

【0013】 [圖5]是示出根據本發明的實施例的被配置為執行用於RPA的機

器人存取控制和管理的計算系統的架構圖。

【0014】 [圖6]是示出根據本發明的實施例的用於UiPath Studio™的Windows®登錄編輯程式的資料夾的視圖的螢幕截圖。

【0015】 [圖7]是示出根據本發明的實施例的Windows Explorer® UI的一部分的螢幕截圖，其中檔案SampleGovernanceRules.dll安裝在UiPath Studio™的規則資料夾中。

【0016】 [圖8]是示出根據本發明的實施例的套裝程式管理介面的螢幕截圖。

【0017】 [圖9]是示出根據本發明的實施例的發送回饋圖示被禁用的RPA設計器應用程式的右上部分的螢幕截圖。

【0018】 [圖10A]-[圖10D]是示出根據本發明的實施例的工作流程分析器設置介面的螢幕截圖。

【0019】 [圖11]是示出根據本發明的實施例的RPA設計器應用程式設置介面的螢幕截圖。

【0020】 [圖12A]和[圖12B]是示出根據本發明的實施例的RPA工作流程的螢幕截圖。

【0021】 [圖13]示出了根據本發明實施例的錯誤清單介面。

【0022】 [圖14]是示出根據本發明的實施例的工作流程分析器介面的螢幕截圖。

【0023】 [圖15]是示出根據本發明的實施例的用於在機器人執行時執行用於RPA的存取控制和管理的系統的架構圖。

【0024】 [圖16]是示出根據本發明的實施例的用於執行用於RPA設計器應用程式的機器人存取控制和管理的流程的流程圖。

【0025】 [圖17]是示出根據本發明的實施例的用於執行正在執行的RPA機

器人的RPA的機器人存取控制和管理流程的流程圖。

【實施方式】

【0026】 一些實施例涉及用於RPA的機器人存取控制和管理。RPA管理是組織制定策略以阻止使用者開發RPA機器人的能力，這些機器人可能會採取有可能危害組織的行為，諸如違反GDPR、HIPAA、第三方應用程式服務條款等。由於開發人員可能會創建在執行其自動化時違反隱私法、服務條款等的RPA機器人，一些實施例在機器人和/或機器人設計應用程式層級實施存取控制和管理限制。在某些實施例中，這可以通過防止RPA開發人員依賴於可能會帶來安全風險或以違反策略、法規、隱私法和/或隱私權策略的方式工作的未經批准的軟體程式庫來為RPA流程開發管道提供更高層級的安全性和合規性。

【0027】 一些實施例防止RPA機器人採取使用者允許但機器人不允許的動作。傳統上，組織可以通過對系統的存取控制來控制使用者可以做什麼和不可以做什麼。但是，在RPA並且尤其是有人值守自動化的內容中，該控制可能會崩潰。機器人代表使用者執行任務，這表示後端系統無法區分機器人還是使用者正在存取系統。

【0028】 這可能是有問題的，原因有幾個。例如，後端系統可以允許使用者存取某些可能被濫用的資料。但是，良好的系統設計通常會使使用者難以在任何規模上引起問題，因為做他們不應當做的事情既緩慢又乏味。相比之下，如果使用RPA機器人，則該機器人由於其執行速度會大大增加資料濫用或違反策略的規模。

【0029】 考慮以下示例。組織使用客戶關係管理（customer relationship management；CRM）系統來聯繫客戶，該系統完全支援各種隱私選項，諸如通過檢查使用者是否選擇了“不與我聯繫”選項並且提供到隱私權原則和說明的連

結以請求組織“忘記他們”來支持GDPR的“選擇退出”和“忘記我”策略以及其他使用者保護法律。該組織已經選擇授予大部分員工存取CRM系統的許可權，但只授予少數員工在系統中發送電子郵件的許可權。

【0030】 但是，出於合法的商業目的，要求更廣泛的使用者群組能夠存取包括客戶的電子郵寄地址在內的客戶簡檔（例如，員工在與客戶合作時可能會在系統中創建或編輯包括客戶電子郵寄地址在內的錯誤資訊）。不使用CRM系統向客戶發送電子郵件是一種組織策略違規行為，但是有些員工希望瞭解一些資訊並且不想等待各種審批管道以獲取這些許可權，或者希望有適當許可權的員工代表他們這樣做。沒有RPA機器人，員工仍然可以違反此策略。但是因為手動將電子郵寄地址一對一地複製和貼上到系統之外很麻煩，即使員工違反了此策略，違規的規模也很小，因此，如果發生任何違規行為，則可以減輕組織的風險。但是，RPA機器人可以比人類快上很多數量級的速度提取該資料（例如，速度提高數千倍，速度提高數萬倍，速度提高數十萬倍，速度提高數百萬倍，等等）以創建和發送電子郵件，因此，在使用者可以手動發送5-10封違反策略的電子郵件，並且使公司面臨違反客戶“選擇退出”GDPR的風險的情況下，如果使用者為此創建了機器人，則可以發送大量違規電子郵件。

【0031】 作為另一示例，考慮使用受到嚴格保護的系統以符合HIPAA要求的方式儲存所有資料的醫療機構。員工可以在適當應用程式的上下文中合法存取該資料和在工作中使用該資料。對系統的低效感到沮喪的員工可能想編寫將患者資料提取到他或她的桌上型電腦上不受保護的Excel®試算表中的RPA機器人。現在，這違反了HIPAA，並且如果該資料發生了某些事情，諸如員工的電腦被盜和被存取，則可能使組織承擔重大的財務和法律責任。

【0032】 作為另一示例，考慮以下情況：第三方應用程式或服務的服務條款禁止使用自動化來在工作中使用該應用程式或服務。如果員工選擇編寫違反

這些服務條款進行工作的RPA機器人，則該公司可能承擔法律責任。

【0033】 這些問題和其他問題可能給採用RPA機器人的組織帶來重大風險。但是，一些實施例通過防止開發人員使用未經批准的程式庫和活動引用來減少或消除開發人員的安全和合規風險。RPA開發人員（無論是經過培訓的RPA開發人員（例如，使用UiPath Studio™）還是未經培訓的公民開發人員（例如，使用UiPath StudioX™））可以添加對被認為不安全或以違反策略/法規的方式工作的程式庫的引用。

【0034】 考慮以下示例。美國政府不允許將在俄羅斯或其他被認為不友好的國家/地區開發的技術安裝在美國政府的計算系統上。但是，RPA中使用了一些由俄羅斯公司開發的通用程式庫（例如，Abbyy® OCR引擎）。儘管開發人員可以選擇預設情況下這些程式庫在其計算系統上不可用，但他們無法明確阻止使用者依賴於將這些程式庫合併到RPA項目中。儘管RPA開發人員的網際網路存取可能會被阻止（由於開發軟體時的很多原因，這是不希望的），但是開發人員仍然可以通過另一種機制（諸如通用序列匯流排（USB）隨身碟）將未經批准的程式庫帶入計算系統。

【0035】 另一示例是，很多金融機構對哪些資訊可以或不可離開其內部網路和防火牆實施嚴格的策略。在現代時代，很多RPA程式庫（經由RPA活動被調用）通過與公共雲端上的伺服器通信來工作，諸如UiPath®文件理解活動。這些金融機構擔心使用者不會意識到不允許使用這些活動，因為他們可能會將敏感性資料發送到公司防火牆之外。但是，沒有先前存在的有效選項可以防止這種情況的發生。

【0036】 另一問題是阻止大規模RPA壓倒系統。RPA取得巨大成功的原因之一是，它可以實施傳統系統的自動化，而無需升級這些系統。由於傳統系統並非被設計成旨在自動化，因此其設計和性能可能只能以人類可以工作的速度

處理互動。但是，傳統應用程式並不總是能夠跟上RPA機器人可以與系統互動的速度。沒有被設計為在這些約束條件下運行的機器人根本上可以用傳統系統無法處理的更快的請求最終使不堪重負的舊系統癱瘓，從而根本上對系統執行意外的拒絕服務（DoS）攻擊。儘管UiPath Studio™中存在可以實現活動之間的延遲的設置，但是當前沒有中心化的實施機制可以確保開發人員在無法以RPA機器人速度運行的傳統系統上工作時使用這些設置。

【0037】 另一問題是確保組織策略得到遵守。很多組織都有要應用於自動化的特定規則。例如，一些示例包括強制將由自動化創建的電子郵件僅發送給內部員工，而不發送給外部收件人，強制將由自動化創建的所有電子郵件儲存為草稿，並且未經人工審核就不會發送等。在沒有適當管理的情況下，RPA機器人可能會違反這些策略。

【0038】 為了解決可能在RPA中發生的存取控制和管理問題，一些實施例為組織提供了定義和實施管理策略的能力。例如，這可以通過定義RPA機器人可以和/或不能存取那些部分的JavaScript物件標記法（JSON）檔案來實施。在一些實施例中，組織可以對哪些應用程式和/或統一資源定位符（URL）能夠和/或不能被自動化進行控制，對哪些活動能夠和/或不能被使用進行控制，對哪些套裝程式（例如，程式庫）能夠和/或不能被使用進行控制，提供創建和部署用於擴展管理功能的自訂規則的功能，提供可以控制例如使用者可以使用哪些套裝程式饋送來安裝程式庫的各種RPA設計器應用程式設置，等等。

【0039】 在一些實施例中，可以在設計時（即，當開發人員正在構建、測試和修改RPA機器人工作流程時）執行管理。但是，這可能使執行某些策略變得困難或不可能。例如，在僅在運行時才知道要應用該策略的值的的情況下，很難或不可能實施策略。例如，如果組織希望制定防止存取某些網站的策略，則設計時分析應當能夠知道要標記的URL。但是，在某些實施例中，可以構造RPA

程式以使得從動態來源（諸如外部檔案或詢問使用者的提示）中載入值。在這種情況下，設計時分析是不夠的，因為在設計時未知違反策略的值。另一示例是，如果組織希望阻止自動發送給組織外部客戶的電子郵件，但是電子郵寄地址清單來自試算表。在這種情況下，電子郵件在設計時是未知的，但是是應當在運行時被評估的系統的動態輸入。

【0040】 在某些實施例中，除了或代替設計時實施，在運行時實施管理。例如，這可以經由用於配置和部署策略的伺服器應用程式來完成。例如，當這些計算系統連接到伺服器指揮器應用程式時，策略可以被自動發送到使用者計算系統。RPA設計器應用程式可以自動將代碼插入RPA機器人中，以迫使它們獲取這些策略並且根據這些策略進行操作。

【0041】 在一些實施例中，可以基於組織、角色、群組、個體開發人員等為RPA設計器應用程式定義控制，以控制開發人員在開發RPA機器人時可以做什麼。可以定義策略並且實施控制，以控制RPA機器人在運行自動化時可以做什麼。這樣的實施例可以使得組織能夠更好地滿足關於法律、法規、協議和公司策略的合規性和管理要求。

【0042】 在一些實施例中，本地檔案只能由管理員修改。因此，非管理員使用者應當不能覆蓋或改變這些檔案。套裝程式饋送可以被鎖定，從而僅期望的饋送可用，而無法添加其他饋送。這樣可以防止使用者使用從其他地方獲取的套裝程式。在某些實施例中，作為策略的一部分，可以禁用“發送回饋”工具。

【0043】 如本文中使用的，“套裝程式饋送”是一個位置（遠端或本地），該位置包含對其他程式庫的存取權限，開發人員可以選擇安裝這些程式庫以獲取其自動化項目的附加功能。本文中使用的程式庫是包含電腦代碼的套裝程式，該電腦代碼為專案提供附加功能（例如，活動）。套裝程式饋送的一個示例是UiPath Connect™市場。這是一個開放的位置，社區成員可以在其中上傳包含

他們編寫的自訂活動的程式庫。如果UiPath® RPA開發人員希望使用這些功能，則該開發人員可以瀏覽並且向其專案添加程式庫引用。UiPath GSuite™活動是程式庫的一個示例。UiPath Studio™的預設安裝不能使RPA開發人員使用GSuite™。但是，開發人員可以安裝GSuite™活動程式庫，該活動程式庫現在使該項目能夠使用活動來與GSuite™一起使用（例如，Gmail®、Google® Sheets、Google® Drive等）。

【0044】 在一些實施例中，提供了代碼分析器，諸如程式分析器或RPA工作流程分析器，作為RPA設計器應用程式的審計功能。工作流程分析器可以定義使用者在運行RPA設計器應用程式時被允許的內容，並且可以實施策略。例如，可以指定允許使用者使用的URL，可以限制某些URL，等等。

【0045】 如本文中使用的，“代碼分析器”是運行一系列規則的電腦代碼，這些規則檢查RPA使用者編寫的代碼並且在應用程式不滿足規則時產生回饋（例如，消息、警告、錯誤等）。在一些實施例中，代碼分析器由RPA應用程式的賣方（例如，UiPath®）提供或由使用RPA應用程式的客戶編寫。UiPath Studio™中的“App/Url限制”規則是代碼分析器規則的一個示例，該規則使組織可以定義自動化專案中能夠或不能使用哪些桌面應用程式和網站URL。如果使用者違反這些規則之一，則他或她將收到具有在管理策略中指定的嚴重性的通知。在某些實施例中，這通常是一個錯誤，它將阻止自動化的發佈或運行。

【0046】 某些實施例可以被部署用於機器人流程自動化（RPA）。圖1是示出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施工作流程的設計器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）任務和業務IT流程的解決方案。設計器110可以支援自動化專案的開發，該自動化專案是業務流程的圖形表示。簡而言之，設計器110支援工作流程和機器人的開發和部署。

【0047】 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施基於規則的流程的自動化。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。

【0048】 在一些實施例中，工作流程可以被嵌套或嵌入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流程，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業務邏輯，從而實施通過多個分支邏輯運算子以更為不同的方式進行的決策的整合以及活動的连接。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的狀態，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誤差時確定工作流程行為且特別適合於偵錯（debug）流程。

【0049】 一旦工作流程在設計器110中被開發，業務流程的執行就由指揮器（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。

【0050】 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（attended）的機器人132、無人值守（unattended）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由使用者事件觸發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120一起用於集中流程部署和記錄媒體。有人值守的機器人132可以幫助人類使用者完成各種任務，並且可以由使用者事件觸發。在一些實施例中，無法從這種類型的機器人上的指揮器120開啟流程和/或這些流程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只可以從機器人託盤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。

【0051】 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流程自動化。無人值守的機器人134可以負責遠端執行、監測、調度並且為工作隊列提供支援。在一些實施例中，可以在設計器110中運行針對所有機器人類型的偵錯。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、網頁（web）應用程式、虛擬機（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（例如桌上型電腦和筆記型電腦應用程式、行動裝置應用程式、可穿戴電腦應用程式等）。

【0052】 指揮器120可以具有各種能力，包括但不限於供應（provisioning）、部署（deployment）、版本化（versioning）、配置（configuration）、佇列（queueing）、監測（monitoring）、日誌記錄（logging）和/或提供互連性（interconnectivity）。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套裝程式（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流程或配置的唯一實例的管理。配置可以包括維護和遞送機器人環境和流程配置。佇列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。

【0053】 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。

（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可以在本地系統帳戶下打開互動式Windows®會話，並且具有Windows®服務的許可權。

【0054】 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，這意味著其具有與已經安裝了給定機器人130的使用者相同的許可權。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。

【0055】 在一些實施例中，機器人130被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、代理以及命令列。SCM管理的機器人服務管理和監測Windows®會話，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）之間的委託。

【0056】 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。

【0057】 在一些實施例中，使用者模式機器人服務管理和監測Windows®會話，並且充當指揮器120與執行主機之間的委託。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。

【0058】 執行器可以在Windows®會話下運行給定作業（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。代理可

以是Windows® Presentation Foundation (WPF) 應用程式，其在系統託盤視窗中顯示可用的作業。代理可以是服務的用戶端。代理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。

【0059】 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的專案也可以獨立於瀏覽器縮放層級。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。

【0060】 圖2是示出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、伺服器或兩者都可以包括任何期望數量的計算系統。在用戶端，機器人應用程式210包括執行器212、代理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正在運行流程。如圖2中所示，若干業務專案可以同時運行。在該實施例中，代理214（例如Windows®服務）是所有執行器212的單個聯繫點。該實施例中的所有消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、索引器伺服器250或兩者進一步處理這些消息。如上文相對於圖1所討論的，執行器212可以是機器人元件。

【0061】 在一些實施例中，機器人表示機器名稱與使用者名之間的關聯。機器人可以同時管理多個執行器。在支持同時運行的多個互動式會話的計算系

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使用者名的單獨Windows®會話中。上文將此稱為HD機器人。

【0062】 代理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”消息）並且下載要被執行的包的所需版本。在一些實施例中，代理214與指揮器230之間的通信始終由代理214發起。在通知場景中，代理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）。

【0063】 在伺服器處，包括了表徵層（web應用程式232、開放資料協定（OData）表徵狀態轉移（REST）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/業務邏輯238）以及持久層（資料庫伺服器240和索引器伺服器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/業務邏輯238。在一些實施例中，使用者在指揮器230的介面中（例如經由瀏覽器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、調度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，使用者經由瀏覽器220與來自web應用程式232的網頁互動，以便執行各種動作來控制指揮器230。例如，使用者可以創建機器人群組、向機器人指派包、分析每個機器人和/或每個流程的日誌、開啟和停止機器人等。

【0064】 除了web應用程式232之外，指揮器230還包括暴露OData REST 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214兩者消耗。在該實施例中，代理214

是用戶端電腦上的一個或多個機器人的監督器。

【0065】 在該實施例中，REST API覆蓋配置、日誌記錄、監測以及佇列功能性。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）誤差、由機器人發送的顯式消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套裝程式版本。佇列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、從佇列中獲得事務、設置事務的狀態等。

【0066】 監測REST端點可以監測web應用程式232和代理214。通知和監測API 236可以用於註冊代理214、向代理214遞送配置設置並且用於從伺服器 and 代理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。

【0067】 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）和索引器伺服器250。在該實施例中，資料庫伺服器240儲存機器人、機器人群組、相關聯的流程、使用者、角色、調度等的配置。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（除了索引器伺服器250之外或代替索引器伺服器250）日誌記錄的消息。

【0068】 在一些實施例中可選的索引器伺服器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，索引器伺服器250可以通過配置設置來禁用。在一些實施例中，索引器伺服器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消息（例如使用比如日誌消息或寫入行的活動）可以通過（多個）日誌記錄的REST端點被發送到索引器伺服器250，在索引器伺服器250中這些消息被編入索引以供將來使用。

【0069】 圖3是示出根據本發明的實施例的設計器310、活動（使用者定義的活動320、UI自動化活動330）與驅動程式（driver）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。工作流程可以包括使用者定義的活動320和UI自動化活動330。一些實施例能夠標識圖像中的非文本視覺分量，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（type）、取得文本、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）、模糊文本匹配以及多定位點（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。取得文本可以標識特定文本的地點並且使用OCR來掃描該地點。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，可以存在可以在設計器310中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。

【0070】 UI自動化活動330是利用較低層級代碼編寫的特殊的較低層級活動（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、瀏覽器驅動程式344、VM驅動程式346、企業應用程式驅動程式348等。

【0071】 驅動程式340可以與OS進行低層級互動，進而尋找掛鉤、監測鍵等。驅動器可以有助於與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。

【0072】 圖4是示出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和可選的索引器伺服器440進行通信。

【0073】 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應用程式，該伺服器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。

【0074】 圖5是示出根據本發明的實施例的被配置為執行用於RPA的機器人存取控制和管理的計算系統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或用於傳送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（多個）處理器510。（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）、現場可程式閘陣列（FPGA）、圖形處理單元（GPU）、其多個實例和/或其任何組合。（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，可以使用多重平行處理。在某些實施例中，（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·諾依曼計算架構的典型組件。

【0075】 計算系統500還包括用於儲存將由（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）、快閃記憶體、快取、靜態儲存裝置（諸如磁片或光碟）或任何其他類型的非暫時性電腦可讀取媒體或其組合的任何組合組成。非暫時性電腦可讀取媒體是可以由（多個）處理器510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。

【0076】 附加地，計算系統500包括通信裝置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，通信裝置520可以被配置為使用分頻多重存取（FDMA）、單載波FDMA（SC-FDMA）、分時多重存取（TDMA）、分碼多重存取（CDMA）、正交分頻多工（OFDM）、正交分頻多重存取（OFDMA）、全球行動通信系統（GSM）通信、通用封包無線服務（GPRS）、通用行動通信系統（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）、長期演進技術（LTE）、LTE高級（LTE-A）、802.11x、Wi-Fi、Zigbee、超寬頻（UWB）、802.16x、802.15、家庭基站（HnB）、藍牙、射頻辨識（RFID）、紅外線資料協會（IrDA）、近場通信（NFC）、第五代（5G）、新無線電（NR）、其任何組合和/或任何其他現有或將來實施的通信標準和/或協議。在一些實施例中，在不脫離本發明的範圍的情況下，通信裝置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。

【0077】 （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、高清顯示器、Retina®顯示器、平面內切換（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、紅外線、光學成像、色散信號技術、聲學脈衝識別、受抑全內反射等的觸摸（觸覺）顯示器、三維（3D）觸摸顯示器、多輸入觸摸顯示器、多觸摸顯示器等。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。

【0078】 鍵盤530和游標控制裝置535（諸如電腦滑鼠、觸控面板等）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介面連接。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/或觸控面板（未示出）與裝置互動。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，使用者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。

【0079】 記憶體515儲存在被（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。這些模組還包括被配置為執行本文中描述的全部或部分流程或其衍生物的機器人存取控制和管理模組545。計算系統500可以包括一個或多個附加功能模組550，該功能模組包括附加功能性。

【0080】 所屬技術領域具有通常知識者應瞭解，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、平板計算裝置、量子計算系統或任何其他合適的計算裝置或裝置的組合。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，本文中所揭露的方法、系統和裝置可以按照與包括雲端計算系統的計算技術一致的當地語系化和分散式形式來實施。

【0081】 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施獨立性。例如，模組可以被實施為包括定制的超大型積體（VLSI）電路或閘陣列、現有的半導體（諸如邏輯晶片、電晶體或其他離散元件）的硬體電路。模組也可以被實施在可程式硬體裝置中，諸如現場可程式閘陣列、可程式陣列邏輯、可程式邏輯裝置、圖形處理單元等。

【0082】 模組也可以至少部分地被實施在軟體中以便由各種類型的處理器執行。例如，可執行程式碼的所標識的單元可以包括電腦指令的一個或多個

實體塊或邏輯塊，這些電腦指令可以例如被組織成物件、流程或功能。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地點中的全異的指令，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施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驅動器、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時性電腦可讀取媒體。

【0083】 實際上，可執行程式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分佈。類似地，運算元據在本文中可以被標識和圖示為在模組內，並且該運算元據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。運算元據可以被收集為單個資料集，或可以分佈於包括在不同存放裝置上的不同地點上，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。

【0084】 在一些實施例中，用於RPA設計器應用程式的管理資訊可以儲存在RPA開發人員無法編輯的檔案中。例如，圖6是示出用於RPA設計器應用程式的Windows®登錄編輯程式的資料夾600的視圖的螢幕截圖（在這種情況下為UiPath Studio™）。在路徑Computer\HKEY_CURRENT_USER\Software\UiPath下，設置了名為GovernanceSource的註冊表項610（例如，當安裝了RPA設計器應用程式時），該註冊表項指向經由關聯連結620可用的遠端策略檔案。在該連結處可用的策略可以被唯讀到計算系統，並且將由RPA設計器應用程式或RPA機器人用來控制某些許可權。該策略可以在個體使用者層級、群組層級、角色層級、組織層級等定義。通過包括經由遠端連結的策略，可以在任何需要時對策略進行更新，並且通過消費RPA機器人和/或RPA設計器應用程式將自動存取更新後的策略。在該示例中，策略檔案提供了針對開發設定檔案定義不同策略群

組的能力。在一些實施例中，可以為經驗豐富的程式師和可能不具有程式設計知識的公民開發人員設置不同的策略，因為這些開發人員可能具有不同的需求。下面包括UiPath Studio™的示例策略檔案的一部分，該檔案為開發使用者、商業使用者 and StudioPro 使用者定義了規則。該示例中，粗體部分指定了可以定義附加規則或計數器的地方。

```
{
  "Metadata": {
    "Version": "1.0",
  },
  "Info": {
    "Name": "Sample Policy Registry Key"
  },
  "Profiles": {
    "Development": {
      "Shell": {
        "IsFeedbackEnabled": true,
        "HideGettingStartedScreen": false
      },
      "SourceControl": {
        "CheckInBeforePublish": false
      },
      "Workflow": {
        "DockedAnnotations": true,
        "AnalyzeOnPublish": false,
```

```
"AnalyzeOnRun": false
},
"PackageManager": {
  "AllowAddRemoveFeeds": false,
  "AllowEnableDisableFeeds": false,
  "AllowOrchestratorFeeds": true,
  "Feeds": [
    {
      "Name": "Local",
      "Source": "C:\\Program Files (x86)\\UiPath\\Studio\\Packages",
      "IsEnabled": true
    },
    {
      "Name": "Official",
      "Source": "https://www.myget.org/F/workflow/",
      "IsEnabled": true
    },
    {
      "Name": "Connect",
      "Source": "https://gallery.uipath.com/api/v2",
      "IsEnabled": true
    }
  ]
},
```

```
"Analyzer": {
  "AllowEdit": false,
  "ReferencedRulesConfigFile": null,
  "EmbeddedRulesConfig": {
    "Rules": [
      {
        "Id": "ST-NMG-001",
        "IsEnabled": true,
        "Parameters": [
          {
            "Name": "Regex",
            "Value": null
          }
        ],
        "ErrorLevel": "Off"
      },
      <ADDITIONAL RULES DEFINED HERE>
    ],
    "Counters": [
      {
        "Id": "ST-NMG-001",
        "IsEnabled": true,
        "Parameters": []
      },
    ],
```

```

    <ADDITIONAL COUNTERS DEFINED HERE>

    ]

  }

}

},

"Business": {

  "Shell": {

    "IsFeedbackEnabled": true,

    "HideGettingStartedScreen": false

  },

  "SourceControl": {

    "CheckInBeforePublish": false

  },

  "Workflow": {

    "DockedAnnotations": true,

    "AnalyzeOnPublish": false,

    "AnalyzeOnRun": false

  },

  "PackageManager": {

    "AllowAddRemoveFeeds": false,

    "AllowEnableDisableFeeds": false,

    "AllowOrchestratorFeeds": true,

    "Feeds": [

      {
```

```
"Name": "Local",
"Source": "C:\\Program Files (x86)\\UiPath\\Studio\\Packages",
"IsEnabled": true
},
{
  "Name": "Official",
  "Source": "https://www.myget.org/F/workflow/",
  "IsEnabled": true
},
{
  "Name": "Connect",
  "Source": "https://gallery.uipath.com/api/v2",
  "IsEnabled": true
}
]
},
"Analyzer": {
  "AllowEdit": false,
  "ReferencedRulesConfigFile": null,
  "EmbeddedRulesConfig": {
    "Rules": [
      {
        "Id": "ST-NMG-001",
        "IsEnabled": true,
```

```

    "Parameters": [
      {
        "Name": "Regex",
        "Value": null
      }
    ],
    "ErrorLevel": "Off"
  },
  <ADDITIONAL RULES DEFINED HERE>
],
"Counters": [
  {
    "Id": "ST-NMG-001",
    "IsEnabled": true,
    "Parameters": []
  },
  <ADDITIONAL COUNTERS DEFINED HERE>
]
}
}
},
"StudioPro": {
  "Shell": {
    "IsFeedbackEnabled": true,
```

```
"HideGettingStartedScreen": false
},
"SourceControl": {
  "CheckInBeforePublish": false
},
"Workflow": {
  "DockedAnnotations": true,
  "AnalyzeOnPublish": false,
  "AnalyzeOnRun": false
},
"PackageManager": {
  "AllowAddRemoveFeeds": false,
  "AllowEnableDisableFeeds": false,
  "AllowOrchestratorFeeds": true,
  "Feeds": [
    {
      "Name": "Local",
      "Source": "C:\\Program Files (x86)\\UiPath\\Studio\\Packages",
      "IsEnabled": true
    },
    {
      "Name": "Official",
      "Source": "https://www.myget.org/F/workflow/",
      "IsEnabled": true
    }
  ]
}
```

```
    },  
    {  
      "Name": "Connect",  
      "Source": "https://gallery.uipath.com/api/v2",  
      "IsEnabled": true  
    }  
  ]  
},  
"Analyzer": {  
  "AllowEdit": false,  
  "ReferencedRulesConfigFile": null,  
  "EmbeddedRulesConfig": {  
    "Rules": [  
      {  
        "Id": "ST-NMG-001",  
        "IsEnabled": true,  
        "Parameters": [  
          {  
            "Name": "Regex",  
            "Value": null  
          }  
        ],  
        "ErrorLevel": "Off"  
      }  
    ],  
    "ErrorLevel": "Off"  
  },  
}
```

<ADDITIONAL RULES DEFINED HERE>

],

"Counters": [

{

"Id": " ST-NMG-001",

"IsEnabled": true,

"Parameters": []

},

<ADDITIONAL RULES DEFINED HERE>

]

}

}

}

}

}

【0085】 應當注意，儘管此處示出的規則和計數器對於每種使用者類型都是相同的，但不一定是這種情況，通常在實際的實施中也不是。還應當注意，在一些實施例中，儘管使用註冊表項來獲取包括控制和管理策略規則的檔案，但是在某些實施例中，檔案或其他策略資訊格式是從指揮器應用程式傳遞的。在某些實施例中，伺服器UI允許管理員定義策略，該策略將經由連接來傳遞（例如，作為.json有效載荷）。但是，在某些實施例中，該策略不一定是管理員明確編寫的“檔案”。

【0086】 在一些實施例中，可以將定制管理規則開發為工作流程分析器流程的一部分。例如，在一些實施例中，RPA設計器應用程式的安裝腳本可以將自

訂管理規則安裝到使用者的計算系統。參見例如圖7的螢幕截圖700，其示出了Windows Explorer® UI的一部分，其中檔案SampleGovernanceRules.dll 710安裝在UiPath Studio™的規則資料夾中。因為檔案SampleGovernanceRules.dll 710位於Program Files資料夾中，所以只有管理員可以修改它。在其電腦上沒有管理特權的最終使用者不能。因此，沒有這種特權的最終使用者將擁有管理策略，並且將無法覆蓋或改變它。

【0087】 圖8是示出根據本發明的實施例的套裝程式管理介面800的螢幕截圖。在該實施例中，用於開發人員的套裝程式饋送已經被鎖定。更具體地，開發人員僅有權存取官方套裝程式饋送810，並且開發人員不能啟用或禁用官方套裝程式饋送810。開發人員也不能為使用者定義的套裝程式輸入名稱820或來源830，使用添加按鈕840來添加使用者定義的套裝程式，或者使用儲存按鈕850來儲存已修改套裝程式設置。這樣可以防止開發人員從組織不希望的任何地方獲取活動套裝程式。

【0088】 一些組織可能擔心開發人員在開發流程中可能將敏感資訊洩漏給RPA設計器應用程式的提供者。因此，在一些實施例中，作為策略問題，在RPA設計器應用程式中禁用了發送回饋功能。參見例如圖9，該圖是螢幕截屏，其示出了RPA設計器應用程式的右上部分900，其中發送回饋圖示被禁用並且指示正是這種情況的訊息910。

【0089】 圖10A是示出根據本發明的實施例的工作流程分析器設置介面1000的螢幕截圖。在一些實施例中，工作流程分析器設置來自策略設定檔。從圖10A中可以看出，規則窗格1002中列出了各種規則，且具有相應的代碼（Code）、名稱（Name）、範圍（Scope）和預設動作（Default action）。針對所選擇的規則的特定設置選項被示出在規則設置窗格1004中。

【0090】 在該實施例中，存在四種類型的存取限制——應用程式/URL限

制1010、套裝程式限制1020、活動限制1030和活動屬性要求1040。但是，在沒有偏離本發明的範圍的情況下，可以使用任何數目和/或類型的存取限制。應用程式/URL限制1010控制RPA設計器應用程式的使用者被允許和/或不允許使用哪些應用程式和/或URL。例如，參考圖10B，如果有任何應用程式被禁止，則被禁止的應用程式將被示出在禁止應用程式欄位1012中，並且被允許的應用程式被示出在允許應用程式欄位1014中。如果使用者向下滾動，則被禁止的URL和被允許的URL也將被示出在相應欄位中。自然，給定應用程式或URL不能既被禁止（即，將其列入黑名單）又被允許（即，將其列入白名單）。在某些實施例中，如果指定了被允許的應用程式，則使用者只能使用那些應用程式。在該實施例中，這些欄位是唯讀的。

【0091】 套裝程式限制1020控制禁止哪些套裝程式。例如，參考圖10C，可以在禁止套裝程式欄位1022中列出被禁止的套裝程式。在一些實施例中，這可以限制組織不希望使用者存取的套裝程式的類別。在某些實施例中，除了或代替禁止套裝程式欄位1022，可以列出被允許的套裝程式。在該實施例中，禁止套裝程式欄位1022是唯讀的。

【0092】 活動限制1030控制允許和/或不允許使用者使用哪些活動。例如，參考圖10D，如果有任何活動被禁止，則被禁止的活動將被示出在禁止活動欄位1032中；如果有任何活動被列出為被允許，則被允許的活動將被示出在允許活動欄位1034中。因此，如果在整個套裝程式的層級不需要限制，但是某些活動不應當被允許和/或僅某些活動應當被允許，則這可以經由活動限制1030來指定。在該實施例中，這些字段是唯讀的。

【0093】 在一些實施例中，如果沒有活動被指定為被允許和/或不允許，則默認所有活動被允許。在某些實施例中，僅實施允許或不允許，而不是兩者。例如，在這樣的實施例中，如果同時包括允許和不允許活動，則“允許”集可以優

先，並且可以忽略該不允許集，因為具體地指定被允許的活動被認為是限制性更大的。

【0094】 活動屬性要求1040允許組織定義自訂規則。返回圖10A，規則配置欄位1042的字段的值為“Property:*Draft*,Activity:*MailX,Value:True”。在該示例中，對於在UiPath StudioX™中與Outlook®一起使用的MailX活動，必須將Draft屬性設置為“True”。與其他限制類型一樣，規則配置欄位1042僅對使用者唯讀。如在圖10A-D中可以看到，一些規則被設置為預設動作是提供錯誤，而其他規則被設置為提供警告。

【0095】 如圖11的螢幕截圖所示，在RPA設計器應用程式設置介面1100中，可以針對RPA工作流程設計實施各種設置。在這種情況下，工作流程必須在運行之前和發佈之前都通過工作流程分析器。可以分別在設置1110和1120中看到該實施。這表示，對於在工作流程分析器設置介面1000中設置為規則的管理策略，工作流程必須遵守所有設置為在不合規情況下拋出錯誤的規則。

【0096】 圖12A和12B是示出根據本發明的實施例的RPA工作流程1200的螢幕截圖。使用者正在嘗試使鏈接1210（此處為“<https://www.uipath.com/>”）處的內容自動化。考慮以下情況：基於工作流程分析器的規則，這是不允許的URL。還考慮存在規則要求選擇“儲存為草稿”（Save as draft）核取方塊1220的情況。當使用者試圖驗證工作流程時，在該實施例中出現錯誤清單介面1300。參見圖13。錯誤清單介面1300指示“儲存為草稿”屬性不符合組織的指南，並且<https://www.uipath.com/>不是組織為UI自動化而定義的被允許應用程式。

【0097】 根據以上內容，在一些實施例中，可以基於使用者的技術成熟度來定義策略。例如，與UiPath StudioX™（可以由具有任何程式設計知識或甚至缺乏該知識的個人使用）相比，可以為UiPath Studio™（由擁有程式設計知識的RPA開發人員使用）定義更複雜的規則。例如，技術上更成熟的RPA設計器應用

程式的使用者可以存取更多的套裝程式饋送，具有禁用饋送的能力，具有更豐富的規則集，等等。

【0098】 考慮圖14的工作流程分析器介面1400。在這種情況下，為RPA開發人員提供了豐富的規則集，包括自訂規則。實施了自訂的套裝程式版本要求規則（Package Version Requirements）1410。郵件活動“UiPath.Mail.Activities”必須具有規則配置欄位1412中的最低版本1.6.0和最高版本1.6.0，並且如果引用其他版本，則將拋出錯誤。這要求UiPath.Mail.Activities必須只能是該版本。另外，允許預發行套裝程式欄位（Allow Prelease Package（True/False））1414被設置為“真”（True）。

【0099】 根據以上內容，在一些實施例中，將管理作為RPA設計器應用程式的安裝腳本的一部分來啟用。在某些實施例中，可以經由註冊金鑰來設置遠端策略檔案，以使RPA設計器應用程式始終具有管理策略的當前版本。替代地，策略檔可以位於RPA設計器應用程式所在的本地計算系統上的受保護位置。在一些實施例中，作為安裝腳本的一部分，將自訂規則部署到受保護的“規則”資料夾中。

【0100】 在一些實施例中，為精通技術的使用者和沒有牢固程式設計知識的使用者實施單獨的策略。在某些實施例中，基於給定使用者、使用者角色、使用者群組等應用不同的策略。在某些實施例中，使用者無法運行或發佈不合規的RPA工作流程。

【0101】 圖15是示出根據本發明的實施例的用於在機器人執行時執行用於RPA的存取控制和管理的系統1500的架構圖。系統1500包括使用者計算系統，諸如桌上型電腦1502、平板電腦1504和智慧型手機1506。但是，在不脫離本發明的範圍的情況下，可以使用任何期望的計算系統，包括但不限於智慧手錶、膝上型電腦、物聯網（IoT）裝置、車輛計算系統等。

【0102】 每個計算系統1502、1504、1506上運行有執行一個或多個自動化的RPA機器人1510。然而，在某些實施例中，計算系統1502、1504、1506可以執行不同的機器人。當執行其流程時，RPA機器人經由網路1520（例如，區域網路（LAN）、行動通信網路、衛星通信網路、網際網路及其任何組合等）向伺服器1530請求某些資訊（例如，向在伺服器1530上運行的指揮器應用程式發送請求）。例如，在不脫離本發明的範圍的情況下，RPA機器人1510可以與伺服器1530通信以發送或接收資料（例如，從資料佇列中推/拉），取回安全儲存的證書/資產資訊（例如，使用者名/密碼、存取權杖等），日誌記錄來自活動的資訊（例如，執行日誌），報告“心跳狀態”（例如，RPA機器人1510正在正確運行），檢查“編排命令”（例如，“運行該命令”，“中止/取消執行”等），或任何其他合適的通信。在一些實施例中，伺服器1530可以運行指揮器應用程式，並且資料可以作為心跳消息的一部分被週期性地發送。伺服器1530從一個或多個檔案1532、資料庫1540或兩者獲取用於給定機器人的存取控制和管理規則。

【0103】 伺服器1530根據存取控制和管理規則檢查將由機器人1510執行的動作和/或由RPA機器人請求的資訊。如果允許機器人1510採取該動作或獲取該資訊，則伺服器1530將所請求的資訊提供給機器人1510和/或將資訊發送到相應計算系統上的驗證應用程式，以指示該動作是可接受的。如果不是，則伺服器1530不將所請求的資訊提供給機器人1510和/或將資訊發送到相應計算系統上的驗證應用程式，以指示機器人1510嘗試的動作是不可接受的。在某些實施例中，驗證應用程式可以結束與機器人1510相關聯的流程。

【0104】 在一些實施例中，不允許的動作可以是針對人類使用者允許、但針對RPA機器人不允許的動作。在某些實施例中，伺服器1530可以驗證機器人1510對於一個或多個傳統系統暫停足夠長的時間，並且可以延遲獲取機器人所請求的資訊或從機器人接受新的資訊請求，直到延遲時段到期。在某些實施例

中，除了設計時實施，還在運行時實施管理。在一些實施例中，例如，當計算系統1502、1504、1506連接到伺服器1530的伺服器指揮器應用程式時，策略可以被自動發送到計算系統1502、1504、1506。在一些實施例中，RPA設計器應用程式可以自動將代碼插入RPA機器人中，從而迫使它們獲取這些策略並且根據這些策略進行操作。

【0105】 圖16是示出根據本發明的實施例的用於執行RPA設計器應用程式的機器人存取控制和管理流程1600的流程圖。該流程開始於在1605從計算系統的註冊表條目中確定通往包括存取控制和管理策略規則的檔案的連結，並且在1610使用所確定的連結下載檔案。在一些實施例中，存取控制和管理策略規則經由用於RPA設計器應用程式的安裝腳本來實施。在1615，工作流程分析器從檔案中讀取用於RPA設計器應用程式的存取控制和管理策略規則。

【0106】 在一些實施例中，存取控制和管理策略規則包括：關於哪些應用程式和/或統一資源定位符（URL）能夠和/或不能被自動化的控制、關於哪些活動能夠和/或不能在RPA工作流程中使用的控制、關於哪些套裝程式能夠和/或不能用於RPA工作流程的控制、或其組合。在某些實施例中，基於組織、角色、群組、個體開發人員或其組合為RPA設計器應用程式定義存取控制和管理策略規則。在一些實施例中，當存取控制和管理策略規則被計算系統的作業系統實施時，不能由RPA設計器應用程式的使用者修改，其中該計算系統上執行有RPA設計器應用程式。在某些實施例中，存取控制和管理策略規則包括一個或多個應用程式和/或URL限制、一個或多個套裝程式限制、一個或多個活動限制、一個或多個活動屬性要求、或其組合。

【0107】 在一些實施例中，在1620，可以顯示各種介面，並且可以阻止未經許可的使用者修改。例如，可以顯示包括可以由RPA工作流程的活動存取的套裝程式的套裝程式管理介面，並且基於存取控制和管理策略規則，可以阻止RPA

設計器應用程式的使用者修改允許的套裝程式或添加不允許的新套裝程式。作為另一示例，可以顯示列出存取控制和管理策略規則的工作流程分析器設置介面，並且可以阻止RPA設計器應用程式的使用者修改存取控制和管理策略規則。

【0108】 在1625，根據存取控制和管理策略規則來分析RPA設計器應用程式的RPA工作流程的活動。在一些實施例中，對RPA工作流程的活動的分析包括驗證要在RPA工作流程活動中存取的一個或多個程式庫是否被包括在白名單中或未被包括在黑名單中。如果驗證在1630成功（即，RPA工作流程活動符合所有必需的存取控制和管理策略規則），則在1635，產生實施RPA工作流程的RPA機器人或發佈RPA工作流程。但是，如果驗證在1630失敗（即，RPA工作流程的一個或多個被分析的活動違反了存取控制和管理策略規則），則阻止RPA機器人的產生或RPA工作流程的發佈，直到RPA工作流程滿足存取控制和管理策略規則。在該實施例中，在1640，RPA工作流程分析器/RPA設計器應用程式等待使用者對RPA工作流程進行修改。然後，該流程返回到步驟1625以分析RPA工作流程的修改後的活動。在一些實施例中，僅再次分析已經修改的活動。

【0109】 圖17是示出根據本發明的實施例的用於執行正在執行的RPA機器人的RPA的機器人存取控制和管理流程1700的流程圖。該流程開始於在1710根據存取控制和管理規則來檢查將由RPA機器人執行的動作和/或由RPA機器人請求的資訊。例如，該檢查可以由用戶端或伺服器的驗證應用程式來執行。如果在1720允許RPA機器人執行操作和/或獲取資訊，則在1730，將所請求的資訊提供給RPA機器人和/或授權操作。然後，在1740驗證應用程式等待，直到接收到下一RPA機器人通信。

【0110】 如果在1720不允許RPA機器人採取行動和/或獲取資訊，則在1750驗證應用程式阻止RPA機器人獲取資訊和/或採取行動。在一些實施例中，在1760，可以將關於不允許存取資訊和/或動作的通知發送給RPA機器人。在某

些實施例中，在1770，驗證應用程式可以自動結束RPA機器人的執行（例如，通過終止與RPA機器人相關聯的進程）。在一些實施例中，不允許的動作可以是針對人類使用者允許、但針對RPA機器人不允許的動作。在某些實施例中，驗證應用程式可以驗證RPA機器人對於一個或多個傳統系統暫停足夠長的時間，並且可以延遲獲取RPA機器人請求的資訊或從RPA機器人接受新的資訊請求，直到延遲時段到期。

【0111】 根據本發明的實施例，圖16和圖17中執行的流程步驟可以由電腦程式執行，該電腦程式編碼有指令，該指令用於使（多個）處理器執行圖16和圖17中所描述的（多個）流程的至少一部分。電腦程式可以被實施在非暫時性電腦可讀取媒體上。電腦可讀取媒體可以是但不限於硬式磁碟機、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的組合。電腦程式可以包括用於控制計算系統的（多個）處理器（例如圖5的計算系統500的（多個）處理器510）以實施圖16和圖17中所描述的流程步驟中的全部或一部分的編碼指令，這些編碼指令還可以被儲存在電腦可讀取媒體上。

【0112】 電腦程式可以被實現在硬體、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或任何其他合適的裝置上操作。

【0113】 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。

【0114】 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，短語“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。

【0115】 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味著可以用本發明實施的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。

【0116】 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。所屬技術領域具有通常知識者應認識到，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。

【0117】 所屬技術領域具有通常知識者應容易地理解，可以利用不同順序的步驟和/或利用與所揭露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，但對於所屬技術領域具有通常知識者而言將明顯的是，某些修改、變型和替代性構造將是明顯的，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】

【0118】

100: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
216:設計器
220:瀏覽器
230:指揮器
232:Web應用程式
234:OData REST API端點
236:通知和監測API
238:API實施/業務邏輯
240:資料庫伺服器
250:索引器伺服器
300:關係
310:設計器
320:使用者定義的活動
330:UI自動化活動
340:驅動程式
342:OS驅動程式

344:瀏覽器驅動程式
346:VM驅動程式
348:企業應用程式驅動程式
400:RPA系統
410:用戶端計算系統
420:指揮器計算系統
430:資料庫伺服器
440:索引器伺服器
500:計算系統
505:匯流排
510:處理器
515:記憶體
520:通信裝置
525:顯示器
530:鍵盤
535:游標控制裝置
540:作業系統
545:機器人存取控制和管理模組
550:附加功能模組
600:Windows®登錄編輯程式的資料夾
610:註冊表項
620:關聯連結
700:螢幕截圖
710:檔案SampleGovernanceRules.dll

800:套裝程式管理介面
810:官方套裝程式饋送
820:名稱
830:來源
840:添加按鈕
850:儲存按鈕
900:RPA設計器應用程式的右上部分
910:訊息
1000:工作流程分析器設置介面
1002:規則窗格
1004:規則設置窗格
1010:應用程式/URL限制
1012:禁止應用程式欄位
1014:允許應用程式欄位
1020:套裝程式限制
1022:禁止套裝程式欄位
1030:活動限制
1032:禁止活動欄位
1034:允許活動欄位
1040:活動屬性要求
1042:規則配置欄位
1100:RPA設計器應用程式設置介面
1110:設置
1120:設置

- 1200:RPA工作流程
- 1210:鏈接
- 1220:核取方塊
- 1300:錯誤清單介面
- 1400:工作流程分析器介面
- 1410:自訂的套裝程式版本要求規則
- 1412:規則配置欄位
- 1414:允許預發行套裝程式欄位
- 1500:系統
- 1502:計算系統/桌上型電腦
- 1504:計算系統/平板電腦
- 1506:計算系統/智慧型手機
- 1510:RPA機器人
- 1520:網路
- 1530:伺服器
- 1532:檔案
- 1540:資料庫
- 1600:流程
- 1605:步驟
- 1610:步驟
- 1615:步驟
- 1620:步驟
- 1625:步驟
- 1630:步驟

- 1635:步驟
- 1640:步驟
- 1700:流程
- 1710:步驟
- 1720:步驟
- 1730:步驟
- 1740:步驟
- 1750:步驟
- 1760:步驟
- 1770:步驟

【發明申請專利範圍】

【請求項1】一種用於執行用於機器人流程自動化（RPA）的執行期間機器人存取控制和管理的方法，包括：

根據存取控制和管理規則檢查要由RPA機器人執行的一個或多個動作、由所述RPA機器人請求的資訊、或這兩者；

當基於所述存取控制和管理規則，所述RPA機器人被允許執行所述一個或多個動作，接收所請求的所述資訊，或這兩者時：

授權所述RPA機器人執行所述一個或多個動作和/或向所述RPA機器人提供所請求的所述資訊；以及

當基於所述存取控制和管理規則，所述RPA機器人不被允許執行所述一個或多個動作，接收所請求的所述資訊，或這兩者時：

阻止所述RPA機器人採取所述一個或多個動作和/或獲取所請求的所述資訊。

【請求項2】根據請求項1所述的電腦實施的方法，其中當所述RPA機器人不被允許執行所述一個或多個動作，接收所請求的所述資訊，或這兩者時，所述方法還包括：

向所述RPA機器人或用戶端的應用程式發送指示所述一個或多個動作和/或對所述資訊的存取不被允許的通知。

【請求項3】根據請求項1所述的電腦實施的方法，其中當所述RPA機器人不被允許執行所述一個或多個動作，接收所請求的所述資訊，或這兩者時，所述方法還包括：

自動結束所述RPA機器人的執行。

【請求項4】根據請求項1所述的電腦實施的方法，其中所述方法由用戶端的驗證應用程式來執行。

【請求項5】根據請求項1所述的電腦實施的方法，其中所述方法由伺服器的驗證應用程式來執行。

【請求項6】根據請求項1所述的電腦實施的方法，還包括：

當與所述RPA機器人相關聯的計算系統連接到伺服器的指揮器應用程式時，自動將所述存取控制和管理規則發送到所述計算系統。

【請求項7】根據請求項1所述的電腦實施的方法，其中所述一個或多個動作中的至少一個動作被允許用於人類使用者，但是不被允許用於所述RPA機器人。

【請求項8】根據請求項1所述的電腦實施的方法，還包括：

驗證所述RPA機器人至少暫停一段時間以使一個或多個傳統系統執行一個或多個操作；以及

當所述RPA機器人沒有至少暫停所述一段時間時：

延遲獲取由所述RPA機器人請求的所述資訊或延遲接受來自所述RPA機器人的新的資訊請求，直到所述一段時間到期。

【請求項9】根據請求項1所述的電腦實施的方法，其中除了在所述存取控制和管理規則的設計實施期間之外，在所述存取控制和管理規則的執行實施期間也被執行。

【請求項10】根據請求項1所述的電腦實施的方法，其中所述RPA機器人包括被自動插入的代碼，所述被自動插入的代碼迫使所述RPA機器人獲取所述存取控制和管理規則和/或根據所述存取控制和管理規則進行操作。

【請求項11】根據請求項1所述的電腦實施的方法，其中所述存取控制和管理規則包括一個或多個應用程式和/或統一資源定位符（URL）限制、一個或多個套裝程式限制、一個或多個活動限制、一個或多個活動屬性要求、或其組合。

【請求項12】一種非暫時性電腦可讀取媒體，儲存用於執行用於機器人流

程自動化（RPA）的執行期間機器人存取控制和管理的電腦程式，所述電腦程式被配置為引起至少一個處理器：

根據存取控制和管理規則檢查要由RPA機器人執行的一個或多個動作、由所述RPA機器人請求的資訊、或這兩者；以及

當基於所述存取控制和管理規則，所述RPA機器人不被允許執行所述一個或多個動作，接收所請求的所述資訊，或這兩者時：

阻止所述RPA機器人採取所述一個或多個動作和/或獲取所請求的所述資訊。

【請求項13】根據請求項12所述的非暫時性電腦可讀取媒體，其中當基於所述存取控制和管理規則，所述RPA機器人被允許執行所述一個或多個動作，接收所請求的所述資訊，或這兩者時，所述電腦程式還被配置為引起所述至少一個處理器：

授權所述一個或多個動作和/或向所述RPA機器人提供所請求的所述資訊。

【請求項14】根據請求項12所述的非暫時性電腦可讀取媒體，其中當所述RPA機器人不被允許執行所述一個或多個動作，接收所請求的所述資訊，或這兩者時，所述電腦程式還被配置為引起所述至少一個處理器：

向所述RPA機器人或用戶端的應用程式發送指示所述一個或多個動作和/或對所述資訊的存取不被允許的通知。

【請求項15】根據請求項12所述的非暫時性電腦可讀取媒體，其中所述一個或多個動作中的至少一個動作被允許用於人類使用者，但是不被允許用於所述RPA機器人。

【請求項16】根據請求項12所述的非暫時性電腦可讀取媒體，其中所述電腦程式還被配置為引起所述至少一個處理器：

驗證所述RPA機器人至少暫停一段時間以使一個或多個傳統系統執行一個

或多個操作；以及

當所述RPA機器人沒有至少暫停所述一段時間時：

延遲獲取由所述RPA機器人請求的所述資訊或延遲接受來自所述RPA機器人的新的資訊請求，直到所述一段時間到期。

【請求項17】根據請求項12所述的非暫時性電腦可讀取媒體，其中所述RPA機器人包括被自動插入的代碼，所述被自動插入的代碼迫使所述RPA機器人獲取所述存取控制和管理規則和/或根據所述存取控制和管理規則進行操作。

【請求項18】一種非暫時性電腦可讀取媒體，儲存用於執行用於機器人流程自動化（RPA）的執行期間機器人存取控制和管理電腦程式，所述電腦程式被配置為引起至少一個處理器：

根據存取控制和管理規則檢查要由RPA機器人執行的一個或多個動作、由所述RPA機器人請求的資訊、或這兩者；以及

當基於所述存取控制和管理規則，所述RPA機器人不被允許執行所述一個或多個動作，接收所請求的所述資訊，或這兩者時：

阻止所述RPA機器人採取所述一個或多個動作和/或獲取所請求的所述資訊，以及

向所述RPA機器人或用戶端的應用程式發送指示所述一個或多個動作和/或對所述資訊的存取不被允許的通知，自動結束所述RPA機器人的執行，或這兩者。

【請求項19】根據請求項18所述的非暫時性電腦可讀取媒體，其中所述電腦程式還被配置為引起所述至少一個處理器：

驗證所述RPA機器人至少暫停一段時間以使一個或多個傳統系統執行一個或多個操作；以及

當所述RPA機器人沒有至少暫停所述一段時間時：

延遲獲取由所述RPA機器人請求的所述資訊或延遲接受來自所述RPA機器人的新的資訊請求，直到所述一段時間到期。

【請求項20】根據請求項18所述的非暫時性電腦可讀取媒體，其中所述RPA機器人包括被自動插入的代碼，所述被自動插入的代碼迫使所述RPA機器人獲取所述存取控制和管理規則和/或根據所述存取控制和管理規則進行操作。

【發明圖式】

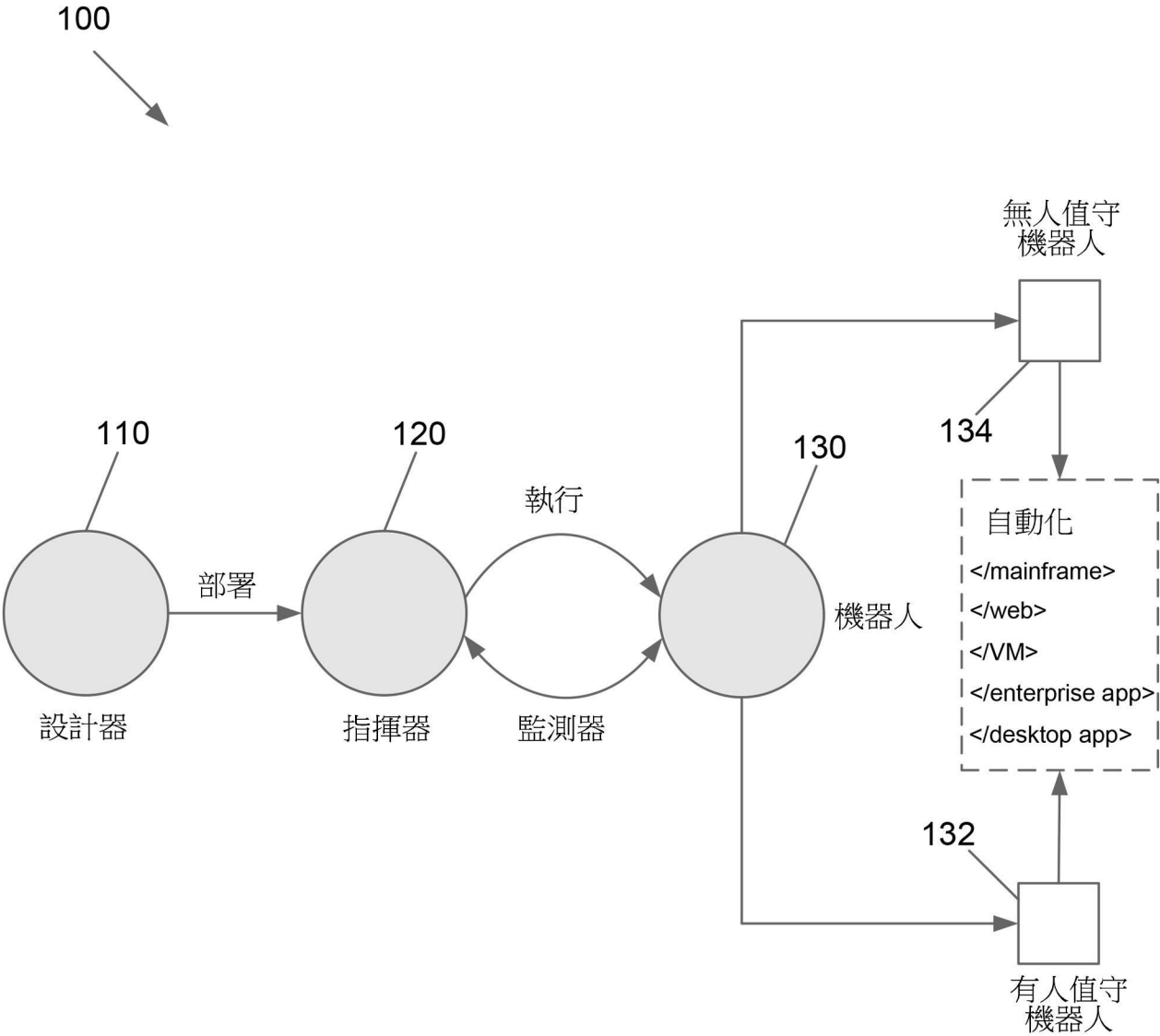


圖1

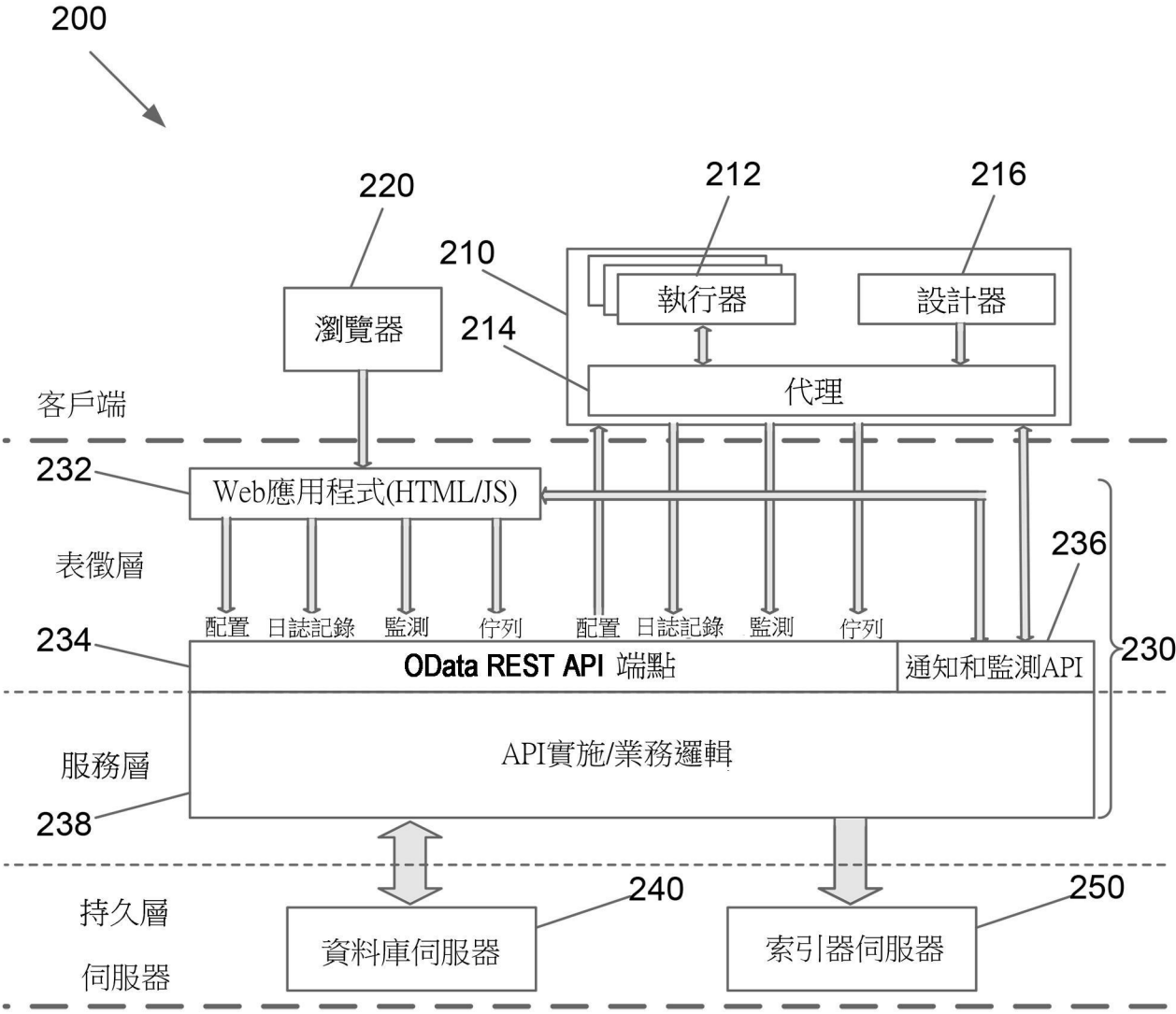


圖2

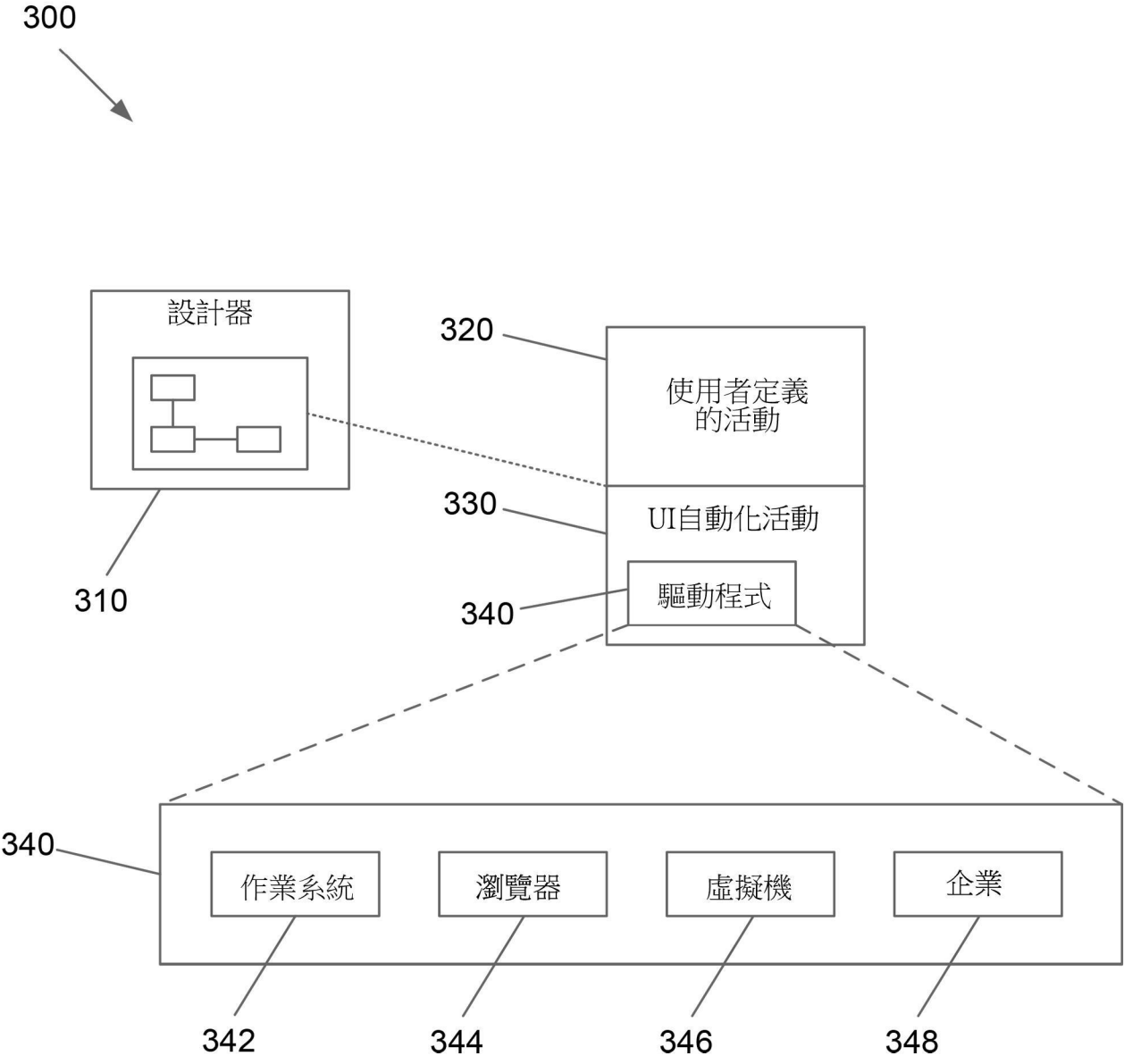


圖3

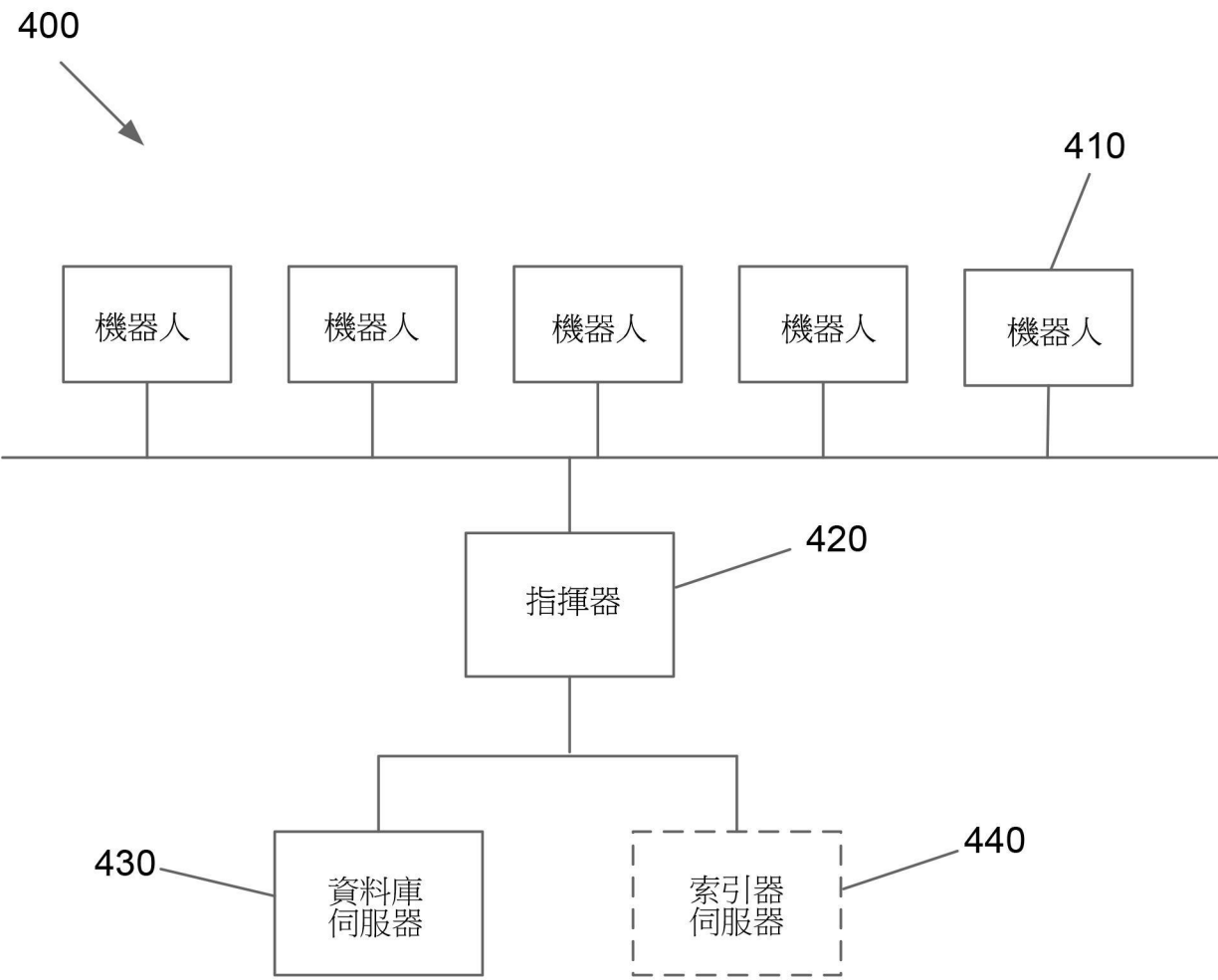


圖4

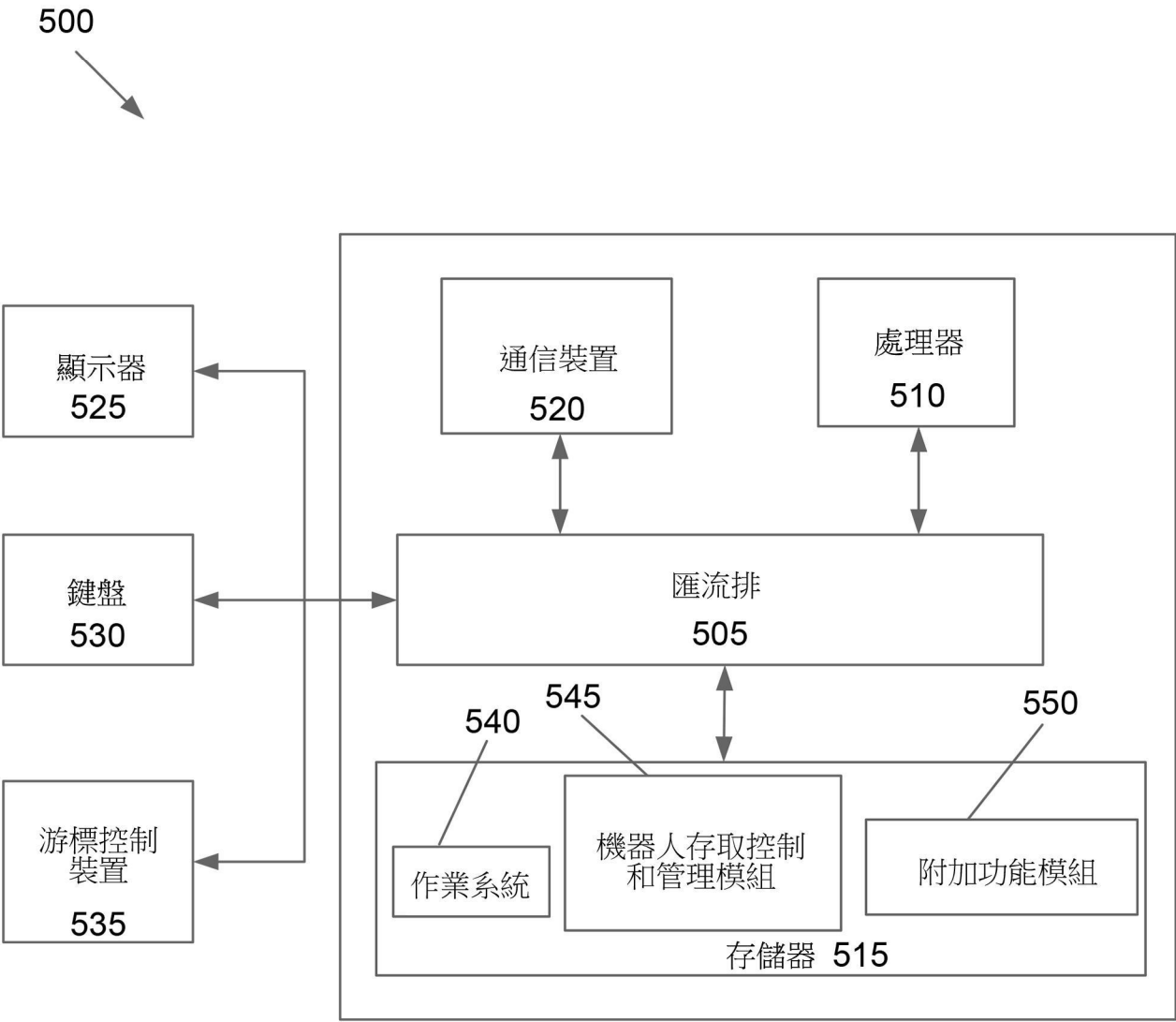


圖5

600

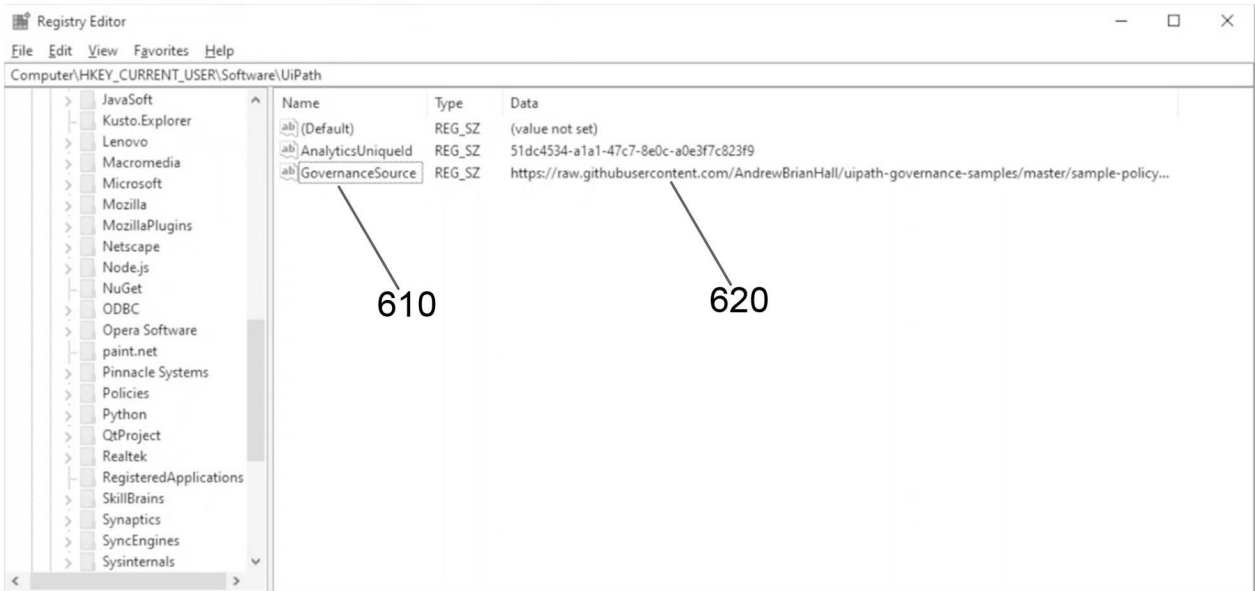
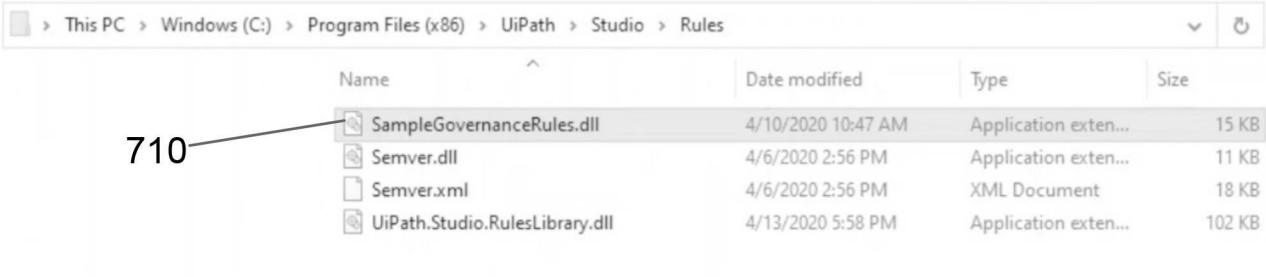


圖6

700



710

圖7

800

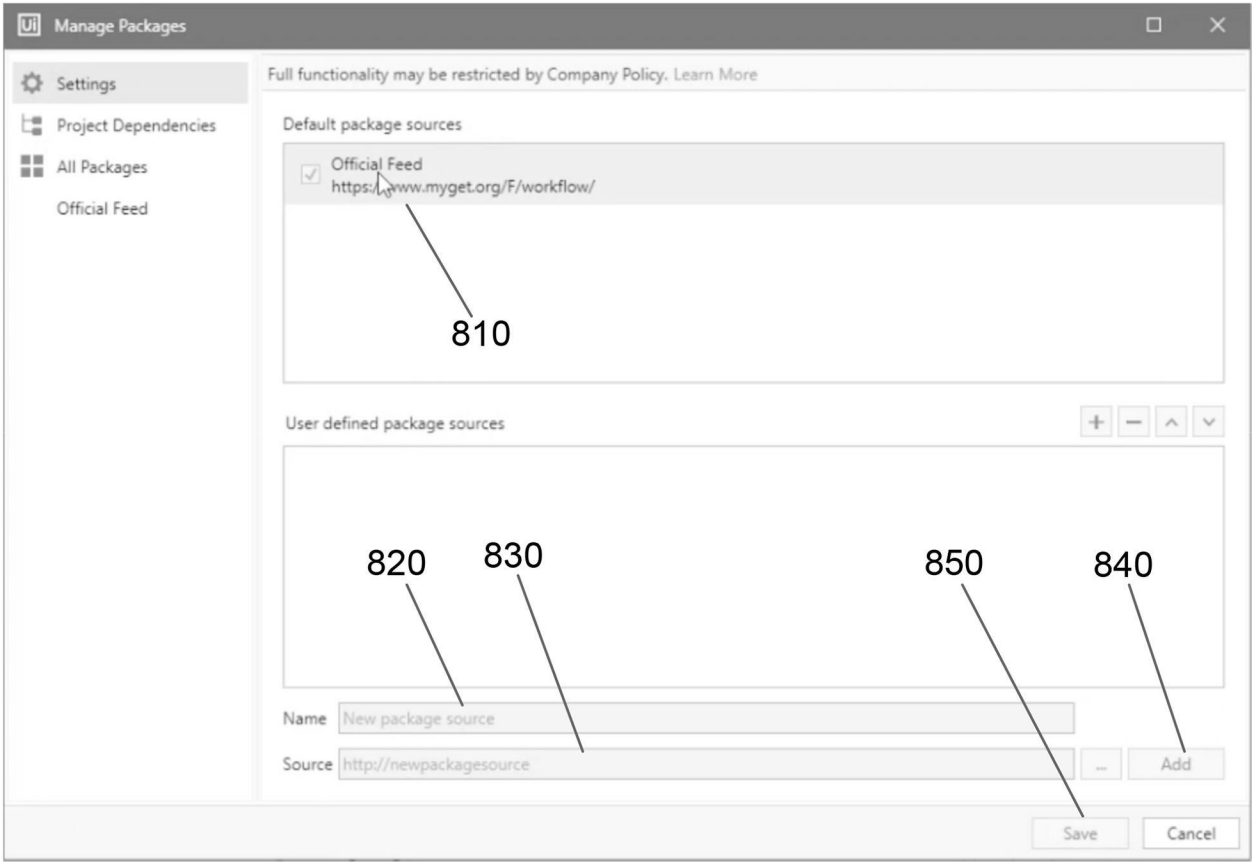


圖8

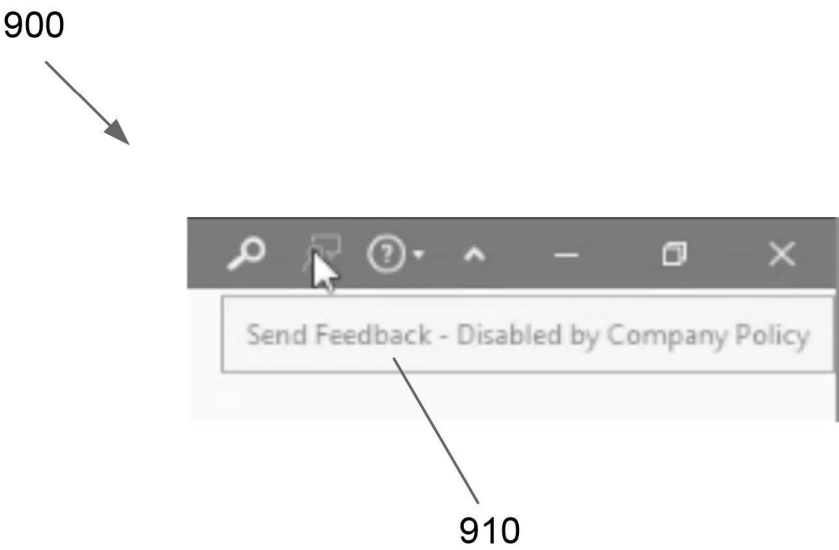


圖9

1000

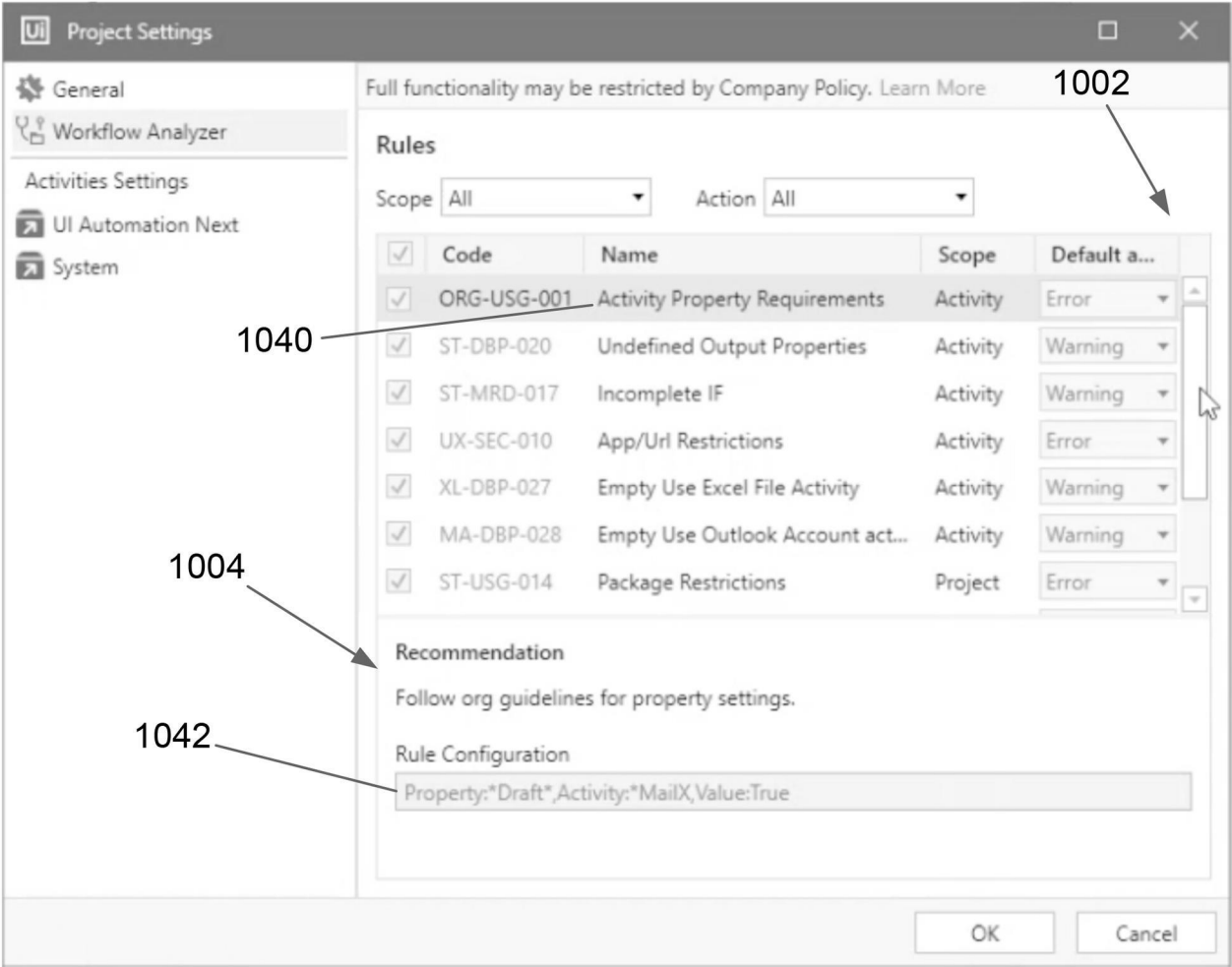


圖10A

1000

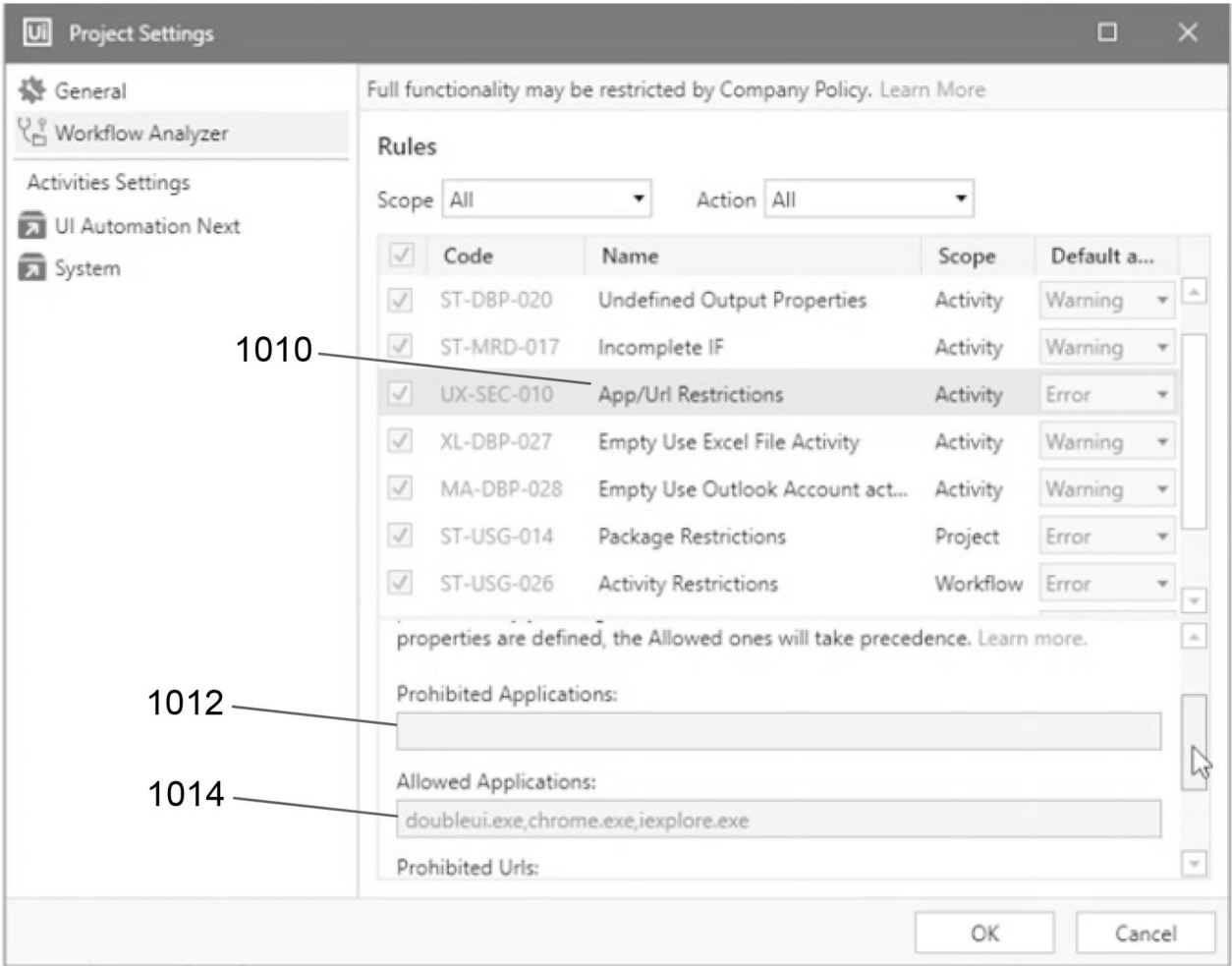


圖10B

1000

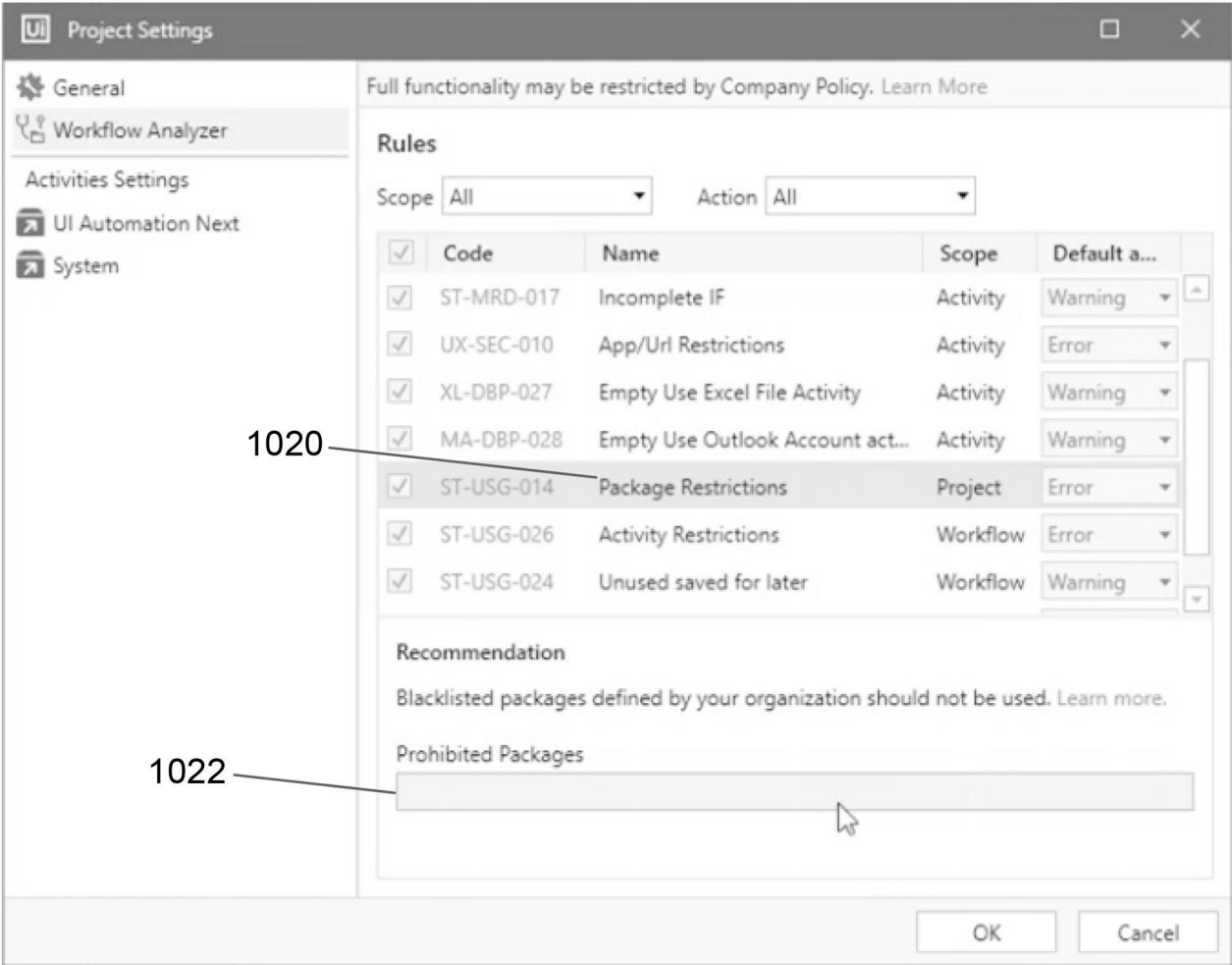
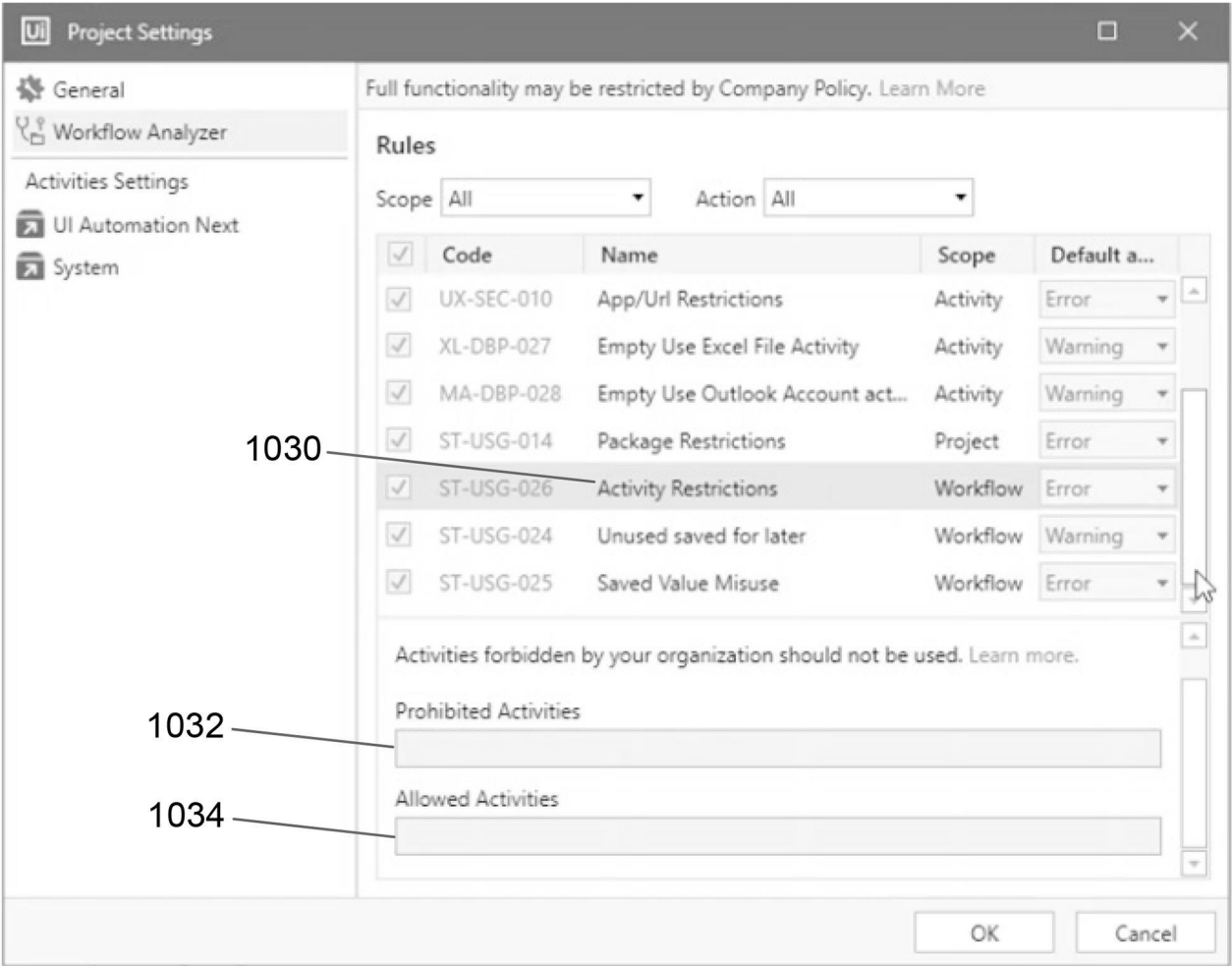


圖10C

1000



1030

1032

1034

圖10D

1100



Settings

 General

 Design

 Locations

 Manage Sources

 License and Profile

Design

Save and Publish

Auto backup interval ⓘ

30000

milliseconds

Publish project timeout ⓘ

30

seconds

Open Project Behavior ⓘ

No Tabs

Enforce Analyzer before Publish ⓘ

Yes

Enforce Analyzer before Run ⓘ

Yes

Execution

Output console size ⓘ

2000

lines

Minimize on run ⓘ

Yes

Activities Design Style

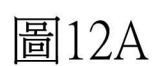
Create docked annotations ⓘ

No

1120

1110

圖11



1200



Browser URL

https://www.uipath.com/

Use Outlook Account

Account

Default Email Account

Reference as

Outlook

Send Outlook Email

Account

Outlook

To

me@mailinator.com

Cc

Select a value

Subject

Select a value

Body ☒ Text ☐ Use Word Document

Select a value

☐ Save as draft

Attachments ☒ Files ☐ Folder

Select a value

1220



圖12B

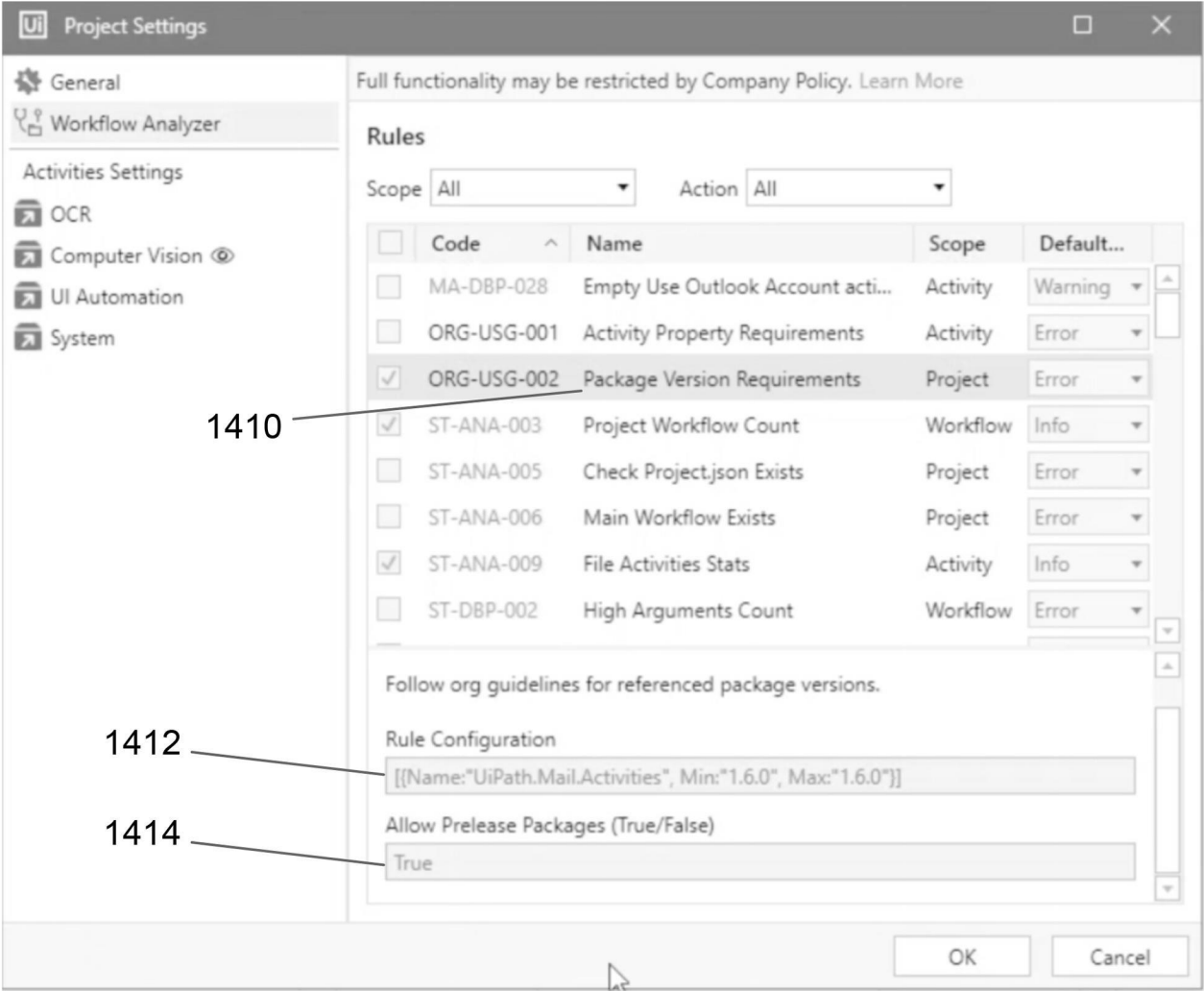
1300



Error List					
Scope:		Project ▼	Rule:		All Rules ▼
	v	Code	Name	Description	File
▶	⚠	ORG-USG-001	Activity Property Requirements	'Save as draft' property of the UiPath.Mail.Activities.Business.Send MailX' activity does not meet your organization's guidelines	Main.xml
▶	⚠	UX-SEC-010	App/Url Restrictions	UI automation detected on 'https://www.uipath.com' which is not an allowed application as defined by your organization	Main.xml

圖13

1400



1410

1412

1414

圖14

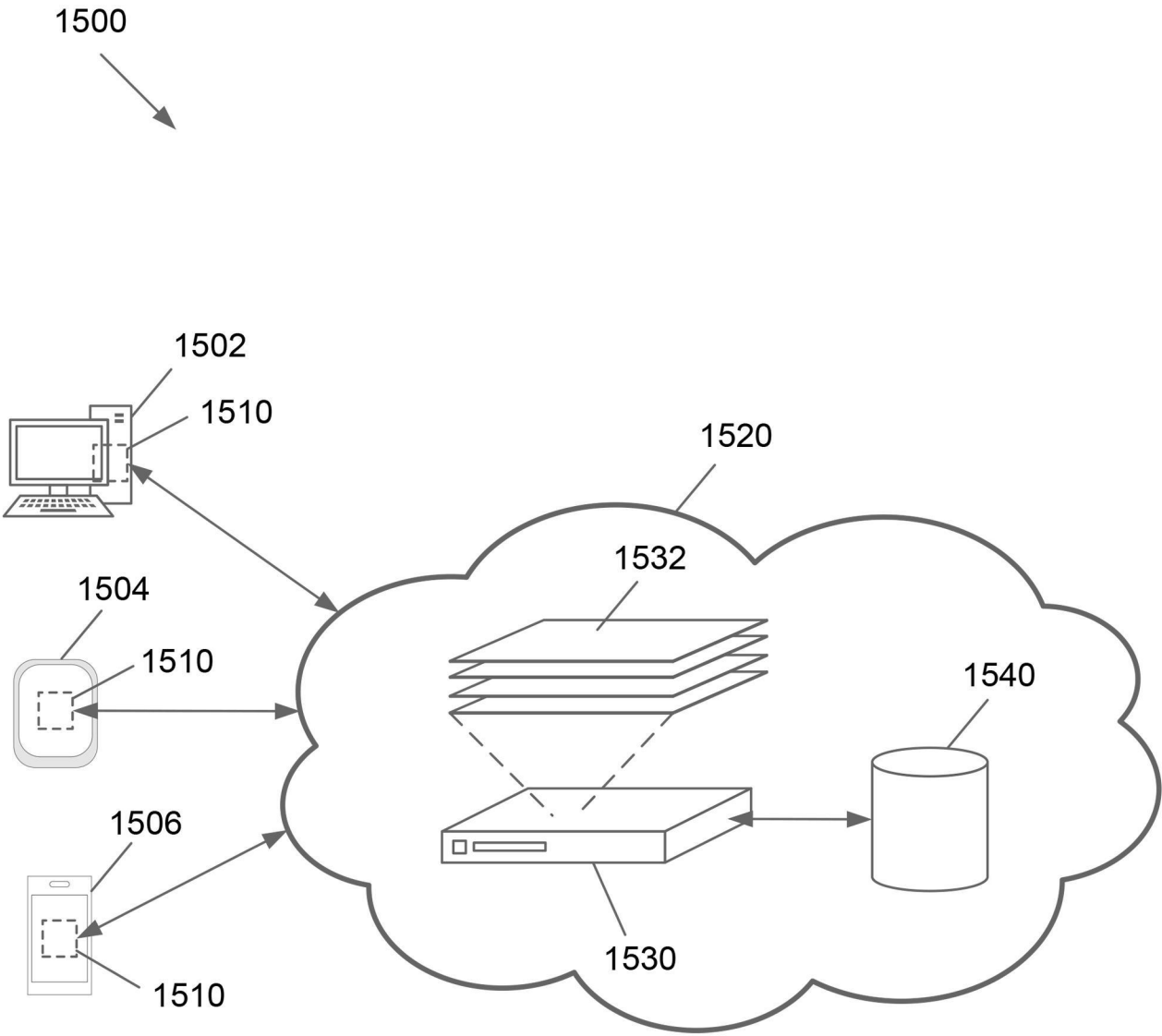


圖15

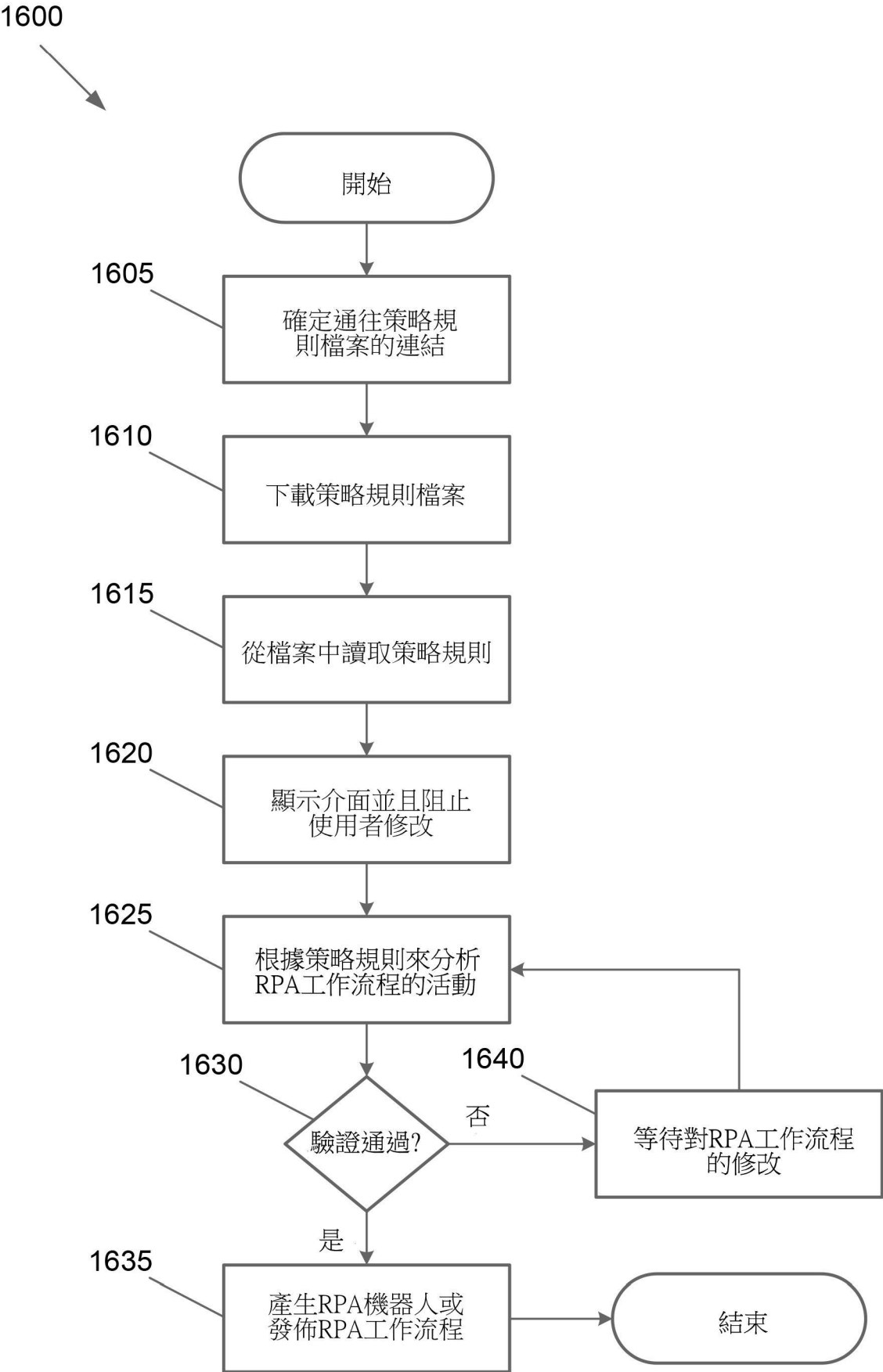


圖16

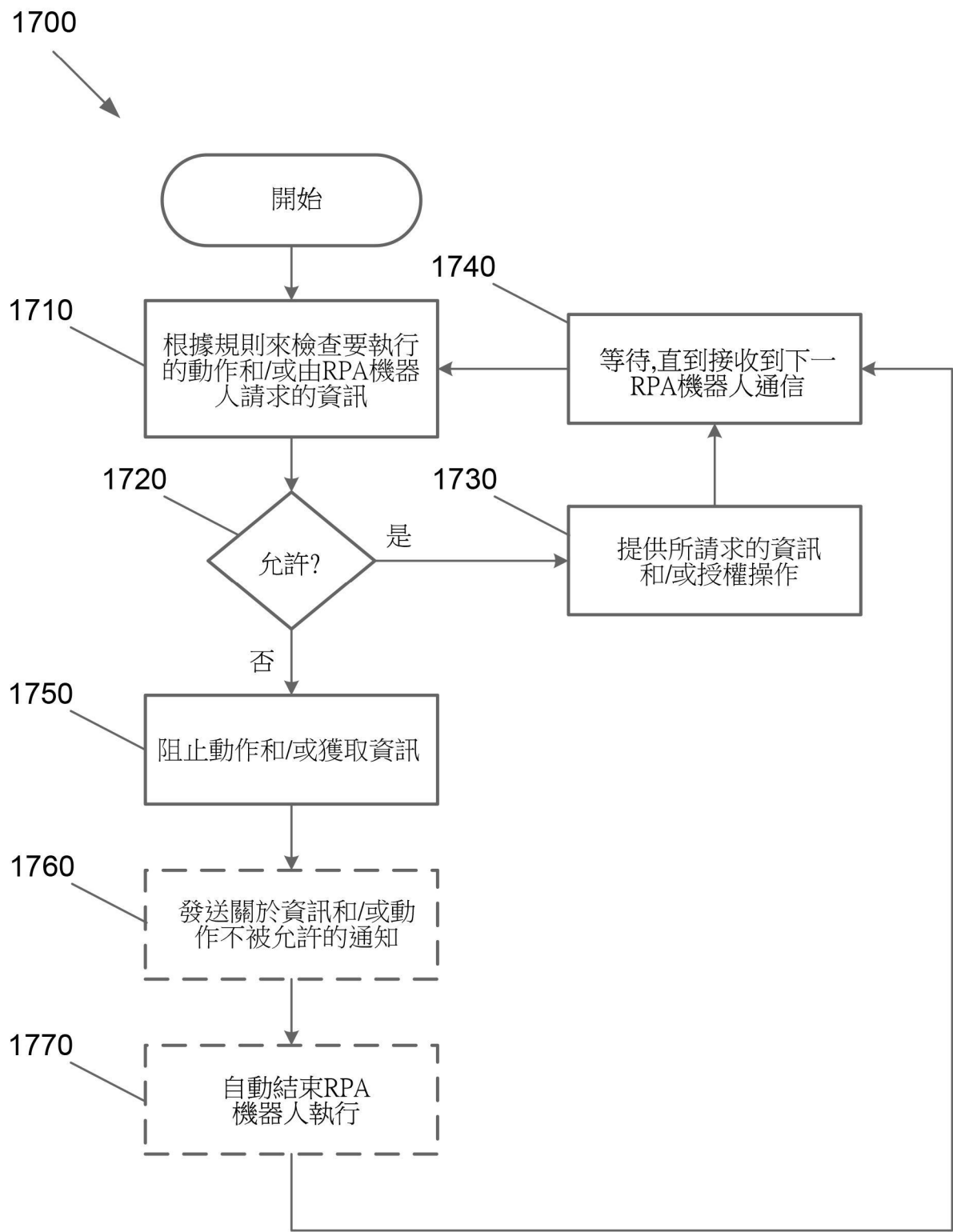


圖17