

(19)日本国特許庁(JP)

(12)公表特許公報(A)

(11)公表番号

特表2024-523765
(P2024-523765A)

(43)公表日 令和6年7月2日(2024.7.2)

(51)国際特許分類

G 0 6 F 8/60 (2018.01)

F I

G 0 6 F 8/60

テーマコード(参考)

5 B 3 7 6

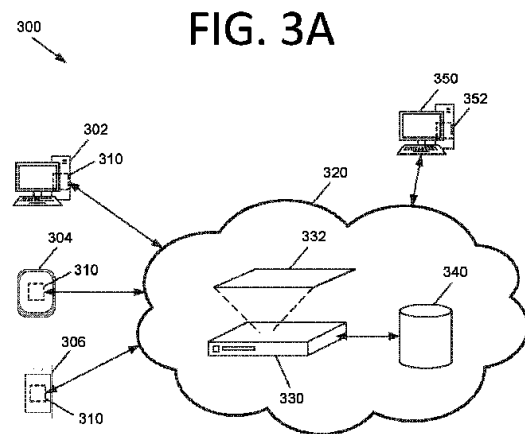
審査請求 未請求 予備審査請求 未請求 (全36頁)

(21)出願番号	特願2021-577290(P2021-577290)	(71)出願人	520262319 ユーアイパス, インコーポレイテッド UiPath, Inc. アメリカ合衆国, ニューヨーク 100 17, ニューヨーク, ヴァンダービルト アベニュー 1, シックスティース フロア 1 Vanderbilt Avenue , 60th Floor, New York, NY 10017, United States of America
(86)(22)出願日	令和3年10月28日(2021.10.28)	(74)代理人	100180781 弁理士 安達 友和
(85)翻訳文提出日	令和4年3月28日(2022.3.28)	(74)代理人	100182903 弁理士 福田 武慶
(86)国際出願番号	PCT/US2021/056929	(72)発明者	ミルシア グリゴアレ
(87)国際公開番号	WO2022/260694		
(87)国際公開日	令和4年12月15日(2022.12.15)		
(31)優先権主張番号	17/340,603		
(32)優先日	令和3年6月7日(2021.6.7)		
(33)優先権主張国・地域又は機関	米国(US)		
(31)優先権主張番号	17/341,512		
(32)優先日	令和3年6月8日(2021.6.8)		
(33)優先権主張国・地域又は機関	米国(US)		
(81)指定国・地域	AP(BW,GH,GM,KE,LR,LS,MW,MZ,NA 最終頁に続く		最終頁に続く

(54)【発明の名称】 ウェブベースのロボティックプロセスオートメーションデザインシステムならびに仮想マシン、セッション、およびコンテナのための自動化

(57)【要約】

RPA開発者がウェブサーバーレス自動化、ユーザーインターフェース(UI)自動化、および他の自動化を設計および実施することを可能にするウェブベースのロボティックプロセスオートメーション(RPA)デザインシステムが開示される。そのようなウェブベースのRPAデザインシステムは、開発者がクラウドを介してログインし、テンプレートプロジェクト、開発者が設計したプロジェクト、サービス、アクティビティなどのリストを取得することを可能にすることができる。したがって、RPA開発は集中型でクラウドベースであってもよく、ユーザーのコンピューティングシステム上のローカル処理およびメモリ要件を低減し、RPAデザイナー機能を集中化し、より良好なコンプライアンスを可能にする。ウェブベースのRPAデザインシステムによって生成された自動化は、仮想マシン(VM)、コンテナ、またはオペレーティングシステムセッションで展開および遂行することができる。



【特許請求の範囲】**【請求項 1】**

クラウドベースのシステムであって、
コンピュータプログラム命令を格納するメモリと、
前記コンピュータプログラム命令を遂行するように構成された少なくとも1つのプロセッサと、を含み、前記コンピュータプログラム命令は、前記少なくとも1つのプロセッサに、

実行時サービスとして複数のロボティックプロセスオートメーション（RPA）ロボットを作動させ、前記複数のRPAロボットは、オペレーティングシステムセッション、仮想マシン（VM）、または前記クラウドベースのシステムのコンテナに配置され、

前記複数のRPAロボットによって、複数の自動化を遂行させるように構成され、前記複数の自動化は、前記複数のRPAロボットによってアクセス可能であり、前記オペレーティングシステムセッション、前記VM、もしくは前記クラウドベースの生産サーバーの前記コンテナ内に配置されるか、または前記クラウドベースのシステムに対して遠隔に配置される、クラウドベースのシステム。

【請求項 2】

前記オペレーティングシステムセッション、前記VM、または前記コンテナは、前記オペレーティングシステム、前記VM、または前記コンテナのリソースの共有プールの一部である、請求項1に記載のクラウドベースのシステム。

【請求項 3】

前記複数の自動化のうちの少なくとも1つは、サーバーレス自動化である、請求項1に記載のクラウドベースのシステム。

【請求項 4】

前記複数の自動化のうちの1つの自動化は、前記複数のRPAロボットのうちの遂行中のRPAロボットに、クライアントコンピューティングシステムのウェブブラウザのウェブ拡張部へ1つまたは複数のコマンドを送信させるように構成され、

前記ウェブ拡張部は、

前記クライアントコンピューティングシステムの前記ウェブブラウザとインタラクトするか、前記クライアントコンピューティングシステムの前記ウェブブラウザの1つもしくは複数のタブ内の1つもしくは複数のウェブページから情報を取得するか、またはその両方を行い、

前記遂行中のRPAロボットに、前記要求されたインタラクションが発生したことの確認を提供するか、前記取得された情報を提供するか、またはその両方を提供する、ように構成される、

請求項1に記載のクラウドベースのシステム。

【請求項 5】

前記ウェブ拡張部は、前記自動化が前記コンテナ内で実行されるときにヘッドレスで遂行されるネイティブ機能を提供するように構成される、請求項4に記載のクラウドベースのシステム。

【請求項 6】

前記遂行中のRPAロボットは、前記ウェブブラウザとの前記インタラクションの前記確認、前記取得された情報、またはその両方を使用して、前記自動化に関連するプロセスを遂行するように構成される、請求項4に記載のクラウドベースのシステム。

【請求項 7】

前記複数の自動化のうちの1つの自動化は、前記複数のRPAロボットのうちの遂行中のRPAロボットに、クライアントコンピューティングシステムのローカルRPA拡張プロセスへ1つまたは複数の要求を送信させるように構成され、

前記ローカルRPA拡張プロセスは、

前記クライアントコンピューティングシステムの1つもしくは複数のアプリケーションおよび/またはプロセスとインタラクトするか、前記クライアントコンピューティングシ

10

20

30

40

50

システム上で実行されている前記 1 つもしくは複数のアプリケーションおよび / またはプロセスから情報を取得するか、またはその両方を行い、

前記遂行中の R P A ロボットに、前記要求されたインタラクションが発生したことの確認を提供するか、前記取得された情報を提供するか、またはその両方を提供する、ように構成される、

請求項 1 に記載のクラウドベースのシステム。

【請求項 8】

前記遂行中の R P A ロボットは、前記要求されたインタラクションが発生したという前記確認、前記取得された情報、またはその両方を使用して、前記自動化に関連するプロセスを遂行するように構成される、請求項 7 に記載のクラウドベースのシステム。

10

【請求項 9】

前記コンピュータプログラム命令は、前記少なくとも 1 つのプロセッサに、

前記複数の自動化のうちの 1 つの自動化を遂行するためのトリガイベント、条件またはコマンドを検出させ、

前記複数の R P A ロボットのうちの遂行中の R P A ロボットによって、ウェブブラウザとインタラクトするために前記ウェブブラウザのウェブ拡張部に 1 つまたは複数のコマンドを送信させるか、前記ウェブブラウザの 1 つまたは複数のタブ内の 1 つまたは複数のウェブページから情報を取得させるか、またはその両方を行わせ、

前記遂行中の R P A ロボットによって、前記ウェブ拡張部から、前記ウェブブラウザとの前記インタラクションの結果、前記取得された情報、またはその両方を受信させ、

20

前記遂行中の R P A ロボットによって、前記ウェブブラウザとの前記インタラクションの前記受信された結果、前記取得された情報、またはその両方を使用して、前記自動化に関連するプロセスを遂行させる、ようにさらに構成される、請求項 1 に記載のクラウドベースのシステム。

【請求項 10】

前記コンピュータプログラム命令は、前記少なくとも 1 つのプロセッサに、

前記複数の自動化のうちの 1 つの自動化を遂行するためのトリガイベント、条件またはコマンドを検出させ、

クライアントコンピューティングシステムの 1 つもしくは複数のアプリケーションおよび / またはプロセスとインタラクトするために、前記複数の R P A ロボットのうちの遂行中の R P A ロボットによって、前記クライアントコンピューティングシステムのローカル R P A 拡張プロセスへ 1 つまたは複数の要求を送信させるか、前記クライアントコンピューティングシステム上で動作している前記 1 つもしくは複数のアプリケーションおよび / またはプロセスから情報を取得させるか、あるいはその両方を行わせ、

30

前記要求されたインタラクションが発生したという確認を受信させるか、前記取得された情報を受信させるか、またはその両方を前記遂行中の R P A ロボットによって受信させ、

前記要求されたインタラクションが発生したという前記受信された確認、前記取得された情報、またはその両方を使用して、前記自動化と関連するプロセスを前記遂行中の R P A ロボットによって遂行させる、ようにさらに構成される、請求項 1 に記載のクラウドベースのシステム。

40

【請求項 11】

コンピュータプログラムを格納する非一時的コンピュータ可読媒体であって、前記コンピュータプログラムは、少なくとも 1 つのプロセッサに、

実行時サービスとしてロボティックプロセスオートメーション (R P A) ロボットを動作させ、前記 R P A ロボットは、オペレーティングシステムセッション、仮想マシン (V M)、またはコンテナに配置され、

前記 R P A ロボットを介して自動化を遂行させる、ように構成され、

前記自動化は、前記 R P A ロボットに、クライアントコンピューティングシステムのウェブブラウザのウェブ拡張部に 1 つまたは複数のコマンドを送信させるように構成され、

50

前記ウェブ拡張部は、

前記クライアントコンピューティングシステムの前記ウェブブラウザとインタラクトするか、前記クライアントコンピューティングシステムの前記ウェブブラウザの1つもしくは複数のタブ内の1つもしくは複数のウェブページから情報を取得するか、またはその両方を行い、

前記RPAロボットに、前記要求されたインタラクションが発生したことの確認を提供するか、前記取得された情報を提供するか、またはその両方を提供する、

ように構成される、非一時的コンピュータ可読媒体。

【請求項12】

前記自動化は、前記RPAロボットから遠隔に配置され、前記RPAロボットによってアクセス可能である、請求項11に記載の非一時的コンピュータ可読媒体。 10

【請求項13】

前記自動化は、サーバーレス自動化である、請求項11に記載の非一時的コンピュータ可読媒体。

【請求項14】

前記ウェブ拡張部は、前記自動化が前記コンテナ内で実行されるときにヘッドレスで遂行されるネイティブ機能を提供するように構成される、請求項11に記載の非一時的コンピュータ可読媒体。

【請求項15】

前記RPAロボットは、前記ウェブブラウザとの前記インタラクションの前記確認、前記取得された情報、またはその両方を使用して、前記自動化に関連するプロセスを遂行するように構成される、請求項11に記載の非一時的コンピュータ可読媒体。 20

【請求項16】

前記コンピュータプログラムは、前記少なくとも1つのプロセッサに、

前記自動化を遂行するためのトリガイベント、条件、またはコマンドを検出させ、

前記検出されたトリガイベント、条件、またはコマンドにตอบสนองして前記RPAロボットを遂行させる、ようにさらに構成される、請求項11に記載の非一時的コンピュータ可読媒体。

【請求項17】

クラウドベースのコンピューティングシステムによって、実行時サービスとしてロボティックプロセスオートメーション(RPA)ロボットを作動させるステップであって、前記RPAロボットは、オペレーティングシステムセッション、仮想マシン(VM)、または前記コンピューティングシステムのコンテナ内に配置される、ステップと、 30

前記RPAロボットによって自動化を遂行するステップと、を含み、

前記自動化は、前記RPAロボットに、クライアントコンピューティングシステムのローカルRPA拡張プロセスへ1つまたは複数の要求を送信させるように構成され、

前記ローカルRPA拡張プロセスは、

前記クライアントコンピューティングシステムの1つもしくは複数のアプリケーションおよび/またはプロセスとインタラクトするか、前記クライアントコンピューティングシステム上で実行されている前記1つもしくは複数のアプリケーションおよび/またはプロセスから情報を取得するか、またはその両方を行い、 40

前記RPAロボットに、前記要求されたインタラクションが発生したことの確認を提供するか、前記取得された情報を提供するか、またはその両方を提供する、ように構成される、

コンピュータにより実施される方法。

【請求項18】

前記自動化は、前記RPAロボットから遠隔に配置され、前記RPAロボットによってアクセス可能である、請求項17に記載のコンピュータにより実施される方法。

【請求項19】

前記自動化は、サーバーレス自動化である、請求項17に記載のコンピュータにより実 50

施される方法。

【請求項 20】

前記 RPA ロボットは、前記要求されたインタラクションが発生したという前記確認、前記取得された情報、またはその両方を使用して、前記自動化に関連するプロセスを遂行するように構成される、請求項 17 に記載のコンピュータにより実施される方法。

【請求項 21】

前記コンピューティングシステムによって、前記自動化を実行するためのトリガイベント、条件、またはコマンドを検出するステップと、

前記コンピューティングシステムによって、前記検出されたトリガイベント、条件、またはコマンドにตอบสนองして前記 RPA ロボットを遂行するステップと、

10

をさらに含む、請求項 17 に記載のコンピュータにより実施される方法。

【発明の詳細な説明】

【技術分野】

【0001】

(関連出願の相互参照)

本出願は、2021年6月8日に提出された米国特許出願第 17 / 341, 512 号の利益を主張し、上記出願は、2021年6月7日に提出された米国特許出願第 17 / 340, 603 号の継続であり、その利益を主張する。これらの先願の主題は、その全体が参照により本明細書に組み込まれる。

【0002】

20

本発明は、一般に、ロボティックプロセスオートメーション (RPA) に関し、より具体的には、RPA 開発者がウェブサーバーレス自動化、ユーザーインターフェース (UI) 自動化、および他の自動化を設計および実施することを可能にするウェブベースの RPA デザイナシステムに関する。

【背景技術】

【0003】

多くの組織のポリシーは、従業員が自らのコンピューティングシステムに直接追加のソフトウェアをインストールしたり既存のソフトウェアをアップグレードしたりすることを許可していない。そのようなポリシーは、米国および他の国における法律または協定 (例えば、EU 一般データ保護規則 (GDPR)、米国医療保険の相互運用性と説明責任に関する法律 (HIPAA)、第三者サービス利用規約など) のコンプライアンスを保証しようと試みるために定められ得る。現在の RPA 開発ソフトウェアは、ユーザーのコンピューティングシステム上にローカルに存在するデスクトップツールとして存在し、新しいバージョンまたは機能がユーザーに展開され得る前に、IT によってテストされ、コンプライアンス担当者によってレビューされるソフトウェアの新しいバージョンまたは機能を必要とする場合がある。また、多くの新しい機能およびサービスは、クラウドベースである。したがって、RPA に対する改善されたおよび / または代替的な手法が有益であり得る。

30

【発明の概要】

【0004】

40

本発明の特定の実施形態は、現在の RPA 技術によってまだ完全に識別、認識、または解決されていない当技術分野の問題および必要性に対するソリューションを提供することができる。例えば、本発明のいくつかの実施形態は、RPA 開発者がウェブサーバーレス自動化、ユーザーインターフェース (UI) 自動化、および他の自動化を設計および実施することを可能にするウェブベースの RPA デザイナシステムに関する。

【0005】

一実施形態では、システムは、ウェブベースの RPA デザイナアプリケーションを提供するウェブブラウザおよび開発サーバーを含む開発者コンピューティングシステムを含む。ウェブブラウザは、ウェブベースの RPA デザイナアプリケーションのウェブインターフェースにアクセスして表示するように構成される。ウェブインターフェースは、RPA

50

プロジェクトを作成し、RPAワークフローを構成し、RPAワークフロー構成をウェブベースのRPAデザイナーアプリケーションに提出するように構成される。ウェブベースのRPAデザイナーアプリケーションは、ウェブブラウザにウェブインターフェースを提供し、ウェブブラウザから提出されたRPAワークフロー構成に基づいて自動化を生成し、RPAロボットを介して生成された自動化を遂行し、生成された自動化を検証し、RPAロボットによる遂行および生成された自動化の検証の結果をウェブブラウザに提供するように構成される。生成された自動化は、オペレーティングシステムセッション、VM、またはコンテナでRPAロボットによって遠隔で遂行されるように構成される。

【0006】

別の実施形態では、非一時的コンピュータ可読媒体は、ウェブベースのRPAデザイナーアプリケーションのためのコンピュータプログラムを格納する。コンピュータプログラムは、少なくとも1つのプロセッサに、コンピューティングシステムのウェブブラウザにウェブインターフェースを提供させるように構成される。ウェブインターフェースは、RPAプロジェクトを作成し、RPAワークフローを構成し、RPAワークフロー構成をウェブベースのRPAデザイナーアプリケーションに提出するための機能を提供するように構成される。コンピュータプログラムはまた、少なくとも1つのプロセッサに、ウェブブラウザから提出されたRPAワークフロー構成に基づいて自動化を生成させ、RPAロボットを介して生成された自動化を遂行させ、生成された自動化を検証させるように構成される。コンピュータプログラムは、少なくとも1つのプロセッサに、RPAロボットの遂行および生成された自動化の検証の結果をウェブブラウザに提供させるようにさらに構成される。生成された自動化は、オペレーティングシステムセッション、VM、またはコンテナで実行時に生産RPAロボットによって遠隔で遂行されるように構成される。

【0007】

さらに別の実施形態では、ウェブベースのRPAのためのコンピュータにより実施される方法は、クラウドベースのコンピューティングシステムによって、ウェブブラウザから提出されたRPAワークフロー構成に基づいて自動化を生成するステップと、RPAロボットを介して生成された自動化を遂行するステップと、クラウドベースのコンピューティングシステムによって、生成された自動化を検証するステップと、を含む。コンピュータにより実施される方法はまた、クラウドベースのコンピューティングシステムによって、RPAロボットの遂行および生成された自動化の検証の結果をウェブブラウザに提供するステップを含む。生成された自動化は、オペレーティングシステムセッション、VM、またはコンテナで実行時に生産RPAロボットによって遠隔で遂行されるように構成される。

【0008】

別の実施形態では、クラウドベースのシステムは、コンピュータプログラム命令を格納するメモリと、コンピュータプログラム命令を遂行するように構成された少なくとも1つのプロセッサと、を含む。コンピュータプログラム命令は、少なくとも1つのプロセッサに、実行時サービスとして複数のRPAロボットを実行させるように構成される。複数のRPAロボットは、クラウドベースのシステムのオペレーティングシステムセッション、VM、またはコンテナに配置される。コンピュータプログラム命令はまた、複数のRPAロボットによって、少なくとも1つのプロセッサに自動化を遂行させるように構成される。複数の自動化は、複数のRPAロボットによってアクセス可能であり、オペレーティングシステムセッション、VM、またはクラウドベースの生産サーバーのコンテナに配置されるか、またはクラウドベースのシステムに対して遠隔に配置される。

【0009】

さらに別の実施形態では、非一時的コンピュータ可読媒体がコンピュータプログラムを格納する。コンピュータプログラムは、少なくとも1つのプロセッサに、実行時サービスとしてRPAロボットを実行させるように構成される。RPAロボットは、オペレーティングシステムセッション、VM、またはコンテナに配置される。コンピュータプログラムはまた、RPAロボットを介して、少なくとも1つのプロセッサに自動化を遂行させるよ

10

20

30

40

50

うに構成される。自動化は、RPAロボットに、クライアントコンピューティングシステムのウェブブラウザのウェブ拡張部に1つまたは複数のコマンドを送信させるように構成される。ウェブ拡張部は、クライアントコンピューティングシステムのウェブブラウザとインタラクトし、クライアントコンピューティングシステムのウェブブラウザの1つまたは複数のタブ内の1つまたは複数のウェブページから情報を取得し、またはその両方を行うように構成される。ウェブ拡張部はまた、要求されたインタラクションが発生したことの確認を提供するか、取得された情報を提供するか、またはその両方をRPAロボットに提供するように構成される。

【0010】

さらに別の実施形態では、コンピュータにより実施される方法は、クラウドベースのコンピューティングシステムによって、実行時サービスとしてRPAロボットを実行することを含む。RPAロボットは、コンピューティングシステムのオペレーティングシステムセッション、VM、またはコンテナに配置される。コンピュータにより実施される方法はまた、RPAロボットによって自動化を遂行することを含む。自動化は、RPAロボットにクライアントコンピューティングシステムのローカルRPA拡張プロセスへ1つまたは複数の要求を送信させるように構成される。ローカルRPA拡張プロセスは、クライアントコンピューティングシステムの1つもしくは複数のアプリケーションおよび/またはプロセスとインタラクトし、クライアントコンピューティングシステム上で実行されている1つもしくは複数のアプリケーションおよび/またはプロセスから情報を取得し、またはその両方を行うように構成される。ローカルRPA拡張プロセスはまた、要求されたインタラクションが発生したことの確認を提供するか、取得された情報を提供するか、またはその両方をRPAロボットに提供するように構成される。

【0011】

別の実施形態では、システムは、ウェブベースのRPAデザイナーアプリケーションを提供するウェブブラウザおよび開発サーバーを含む開発者コンピューティングシステムを含む。ウェブベースのRPAデザイナーアプリケーションは、ウェブブラウザにウェブベースのRPAデザイナーアプリケーションのコードをダウンロードさせ、ウェブブラウザにウェブベースのRPAデザイナーアプリケーションのウェブインターフェースを表示させるように構成される。ウェブインターフェースは、RPAプロジェクトを作成し、RPAワークフローを構成する機能を提供するように構成される。ウェブベースのRPAデザイナーアプリケーションはまた、ウェブブラウザに、ウェブブラウザで構成されたRPAワークフローの自動化を生成させ、ウェブブラウザで生成された自動化を遂行および検証させるように構成される。

【0012】

さらに別の実施形態では、非一時的コンピュータ可読媒体がコンピュータプログラムを格納する。コンピュータプログラムは、少なくとも1つのプロセッサに、ウェブベースのRPAデザイナーアプリケーションのためのコードをウェブブラウザにダウンロードさせ、ウェブベースのRPAデザイナーアプリケーションのためのウェブインターフェースをウェブブラウザに表示させるように構成される。ウェブインターフェースは、RPAプロジェクトを作成し、RPAワークフローを構成する機能を提供するように構成される。コンピュータプログラムはまた、少なくとも1つのプロセッサに、ウェブブラウザで構成されたRPAワークフローの自動化を生成させ、ウェブブラウザで生成された自動化を遂行および検証させるように構成される。

【0013】

さらに別の実施形態では、コンピュータにより実施される方法は、コンピューティングシステムによって、ウェブベースのRPAデザイナーアプリケーションのためのコードをウェブブラウザにダウンロードするステップを含む。コンピュータにより実施される方法はまた、コンピューティングシステムによって、ウェブベースのRPAデザイナーアプリケーションのためのウェブインターフェースをウェブブラウザに表示するステップを含む。ウェブインターフェースは、RPAプロジェクトを作成し、RPAワークフローを構成する

機能を提供するように構成される。コンピュータにより実施される方法は、コンピューティングシステムによって、ウェブブラウザで構成されたRPAワークフローの自動化を生成するステップと、コンピューティングシステムによって、ウェブブラウザで生成された自動化を遂行および検証するステップと、をさらに含む。ダウンロードされたコードには、生成された自動化を遂行および検証するためのコードが含まれる。

【0014】

別の実施形態では、クラウドベースのシステムは、コンピュータプログラム命令を格納するメモリと、コンピュータプログラム命令を遂行するように構成された少なくとも1つのプロセッサと、を含む。コンピュータプログラム命令は、少なくとも1つのプロセッサに、実行時に第1のウェブブラウザにおいて自動化を実行させるように構成される。第1のウェブブラウザは、オペレーティングシステムセッション、VM、またはクラウドベースのシステムのシステムのコンテナに配置される。

10

【0015】

さらに別の実施形態では、非一時的コンピュータ可読媒体がコンピュータプログラムを格納する。コンピュータプログラムは、少なくとも1つのプロセッサに、実行時に第1のウェブブラウザにおいて自動化を実行させるように構成される。第1のウェブブラウザは、オペレーティングシステムセッション、VM、またはクラウドベースのシステムのシステムのコンテナに配置される。自動化は、RPAロボットに、クライアントコンピューティングシステムの第2のウェブブラウザのウェブ拡張部に1つまたは複数のコマンドを送信させるように構成される。ウェブ拡張部は、クライアントコンピューティングシステムの第2のウェブブラウザとインタラクトし、クライアントコンピューティングシステムの第2のウェブブラウザの1つまたは複数のタブ内の1つまたは複数のウェブページから情報を取得し、またはその両方を行うように構成される。ウェブ拡張部はまた、要求されたインタラクションが発生したことの確認を提供するか、取得された情報を提供するか、またはその両方を第1のウェブブラウザに提供するように構成される。

20

【0016】

さらに別の実施形態では、コンピュータにより実施される方法は、コンピューティングシステムによって、実行時にウェブブラウザで自動化を実行することを含む。ウェブブラウザは、オペレーティングシステムセッション、VM、またはクラウドベースのシステムのシステムのコンテナに配置される。自動化は、ウェブブラウザにクライアントコンピューティングシステムのローカルRPA拡張プロセスへ1つまたは複数の要求を送信させるように構成される。ローカルRPA拡張プロセスは、クライアントコンピューティングシステムの1つもしくは複数のアプリケーションおよび/またはプロセスとインタラクトし、クライアントコンピューティングシステム上で実行されている1つもしくは複数のアプリケーションおよび/またはプロセスから情報を取得し、またはその両方を行うように構成される。ローカルRPA拡張プロセスはまた、要求されたインタラクションが発生したことの確認を提供するか、取得された情報を提供するか、またはその両方をウェブブラウザに提供するように構成される。

30

【図面の簡単な説明】

【0017】

本発明の特定の実施形態の利点が容易に理解されるように、上記で簡単に説明した本発明のより具体的な説明は、添付の図面に示されている特定の実施形態を参照することによって提供される。これらの図面は、本発明の典型的な実施形態のみを示しており、したがってその範囲を限定するものと見なされるべきではないことを理解されたいが、本発明は、添付の図面を使用することによって追加の具体性および詳細を伴って説明および説明される。

40

【0018】

【図1】本発明の一実施形態による、ウェブベースのRPAデザイナシステムを示すアーキテクチャ図である。

【0019】

50

【図 2】本発明の一実施形態による、ウェブベースの R P A デザイナシステムの一部または全部を実装するように構成されたコンピューティングシステムを示すアーキテクチャ図である。

【 0 0 2 0 】

【図 3 A】本発明の一実施形態による、ウェブベースの R P A システムを示すアーキテクチャ図である。

【 0 0 2 1 】

【図 3 B】本発明の一実施形態による、コンテナ、V M、またはオペレーティングシステムセッションで実行される自動化および R P A ロボットを示す図である。

【 0 0 2 2 】

【図 4 A】本発明の一実施形態による、ウェブベースの R P A 開発アプリケーションウェブページのホームページビューを示す図である。

【 0 0 2 3 】

【図 4 B】本発明の一実施形態による、新しい自動化ウィンドウが表示されたウェブベースの R P A 開発アプリケーションウェブページのホームページビューを示す図である。

【 0 0 2 4 】

【図 4 C】本発明の一実施形態による、ウェブベースの R P A 開発アプリケーションウェブページのキャンバスビューを示す図である。

【 0 0 2 5 】

【図 4 D】本発明の一実施形態による、アクティビティカテゴリおよびクイックアクションウィンドウを有するウェブベースの R P A 開発アプリケーションウェブページのキャンバスビューを示す図である。

【 0 0 2 6 】

【図 4 E】本発明の一実施形態による、E x c e l（登録商標）アクティビティウィンドウを有するウェブベースの R P A 開発アプリケーションウェブページのキャンバスビューを示す図である。

【 0 0 2 7 】

【図 4 F】本発明の一実施形態による、未構成の使用 E x c e l（登録商標）ファイルアクティビティを有するウェブベースの R P A 開発アプリケーションウェブページのキャンバスビューを示す図である。

【 0 0 2 8 】

【図 4 G】本発明の一実施形態による、使用 E x c e l（登録商標）ファイルアクティビティが構成された後のウェブベースの R P A 開発アプリケーションウェブページのキャンバスビューを示す図である。

【 0 0 2 9 】

【図 5】本発明の一実施形態による、設計時にウェブベースの R P A 開発を実行するためのプロセスを示すフローチャートである。

【 0 0 3 0 】

【図 6】本発明の一実施形態による、ウェブ拡張部とインタラクトする実行時に自動化を遂行するためのプロセスを示すフローチャートである。

【 0 0 3 1 】

【図 7】本発明の一実施形態による、V Mまたはコンテナ内で実行される実行時に自動化を遂行するためのプロセスを示すフローチャートである。

【 0 0 3 2 】

【図 8】本発明の一実施形態による、ローカルアプリケーションとインタラクトするように構成されたウェブベースの R P A システムを示すアーキテクチャ図である。

【 0 0 3 3 】

【図 9】本発明の一実施形態による、R P A ロボットとは異なるコンピューティングシステム上で実行されているローカル R P A 拡張プロセスとインタラクトする、実行時に自動化を遂行するためのプロセスを示すフローチャートである。

10

20

30

40

50

【 0 0 3 4 】

【 図 1 0 】本発明の一実施形態による、R P Aロボットなしでウェブブラウザを介して設計時にウェブベースのR P A開発を実行するためのプロセスを示すフローチャートである。

【 0 0 3 5 】

【 図 1 1 】本発明の一実施形態による、コンテナ、V M、またはオペレーティングシステムセッションのウェブブラウザにおいて実行時に自動化を遂行するためのプロセスを示すフローチャートである。

【 0 0 3 6 】

別段の指示がない限り、同様の符号は、添付の図面を通して一貫して対応する特徴を示す。

【 発明を実施するための形態 】

【 0 0 3 7 】

いくつかの実施形態は、R P A開発者がウェブサーバーレス自動化、U I自動化、および他の自動化を設計および実装することを可能にするウェブベースのR P Aデザイナシステムに関する。いくつかの実施形態では、ウェブベースのR P Aデザイナシステム、システムによって生成された自動化、またはその両方は、例えばクラウドベースの環境で提供される。そのような実施形態は、デスクトップインストールを必要とせず、ユーザーがクラウドを介してログインし、テンプレートプロジェクト、開発者が設計したプロジェクト、サービス、アクティビティなどのリストを取得することを可能にすることができる。したがって、いくつかの実施形態では、R P A開発は集中型でクラウドベースであってもよく、ユーザーのコンピューティングシステム上のローカル処理およびメモリ要件を低減し、R P Aデザイナ機能を集中化し、より良好なコンプライアンスを可能にする。そのような実施形態はまた、R P Aサービスプロバイダが自らR P Aデザイナシステムを保守およびアップグレードすることを可能にし、アップグレードに関連する保守問題およびコストを削減し、顧客による新しい機能を追加することができる。これはまた、より迅速かつより定期的な更新、新しい特徴の展開、バグ修正などを可能にすることができる。

【 0 0 3 8 】

いくつかの実施形態では、R P A開発者は、ウェブブラウザを介してクラウドベースのR P Aデザイナシステムにアクセスすることができる。ユーザーは、最初にホームページ（例えば、図7 Aを参照）で出迎えられ、そこからプロジェクトを作成ことができ、ユーザーが構築したいものに関する情報を提供することができる。次いで、ユーザーが所望のR P A自動化を構築することができる「キャンパス」ウェブページが表示され得る（例えば、図7 Cを参照）。例えば、いくつかの実施形態では、ユーザーは、R P AワークフローでR P Aアクティビティを構成し、自動化がどのように実行されるか（例えば、手動、トリガに基づく、スケジュールによる、など）を制御することができる。R P Aワークフローが意図通りに設計された後に、いくつかの実施形態では、ユーザーはR P Aワークフローをパブリッシュすることができ、自動化は必要に応じて展開することができる。

【 0 0 3 9 】

いくつかの実施形態では、ウェブベースのR P Aデザイナシステムは、サーバー側ワークフロー、サーバーレスワークフロー、長期実行ワークフロー（例えば、特定のプロセスが完了するのを待つか、または動作を再開する前に特定の入力を待つことができるワークフロー）、オーケストレーションプロセス、ヘッドレスワークフロー、バックグラウンドプロセス、非U I自動化ワークフロー、一般にユーザーとのインタラクションを必要としないワークフローなどを設計、編集、およびテストする能力をユーザーに提供する。例えば、ウェブベースのR P Aデザイナシステムで作成されたサーバーレスワークフローをクラウドに保存することができ、ユーザーはウェブブラウザを介してサーバー側ワークフローを手動で実行およびデバッグすることができる。パッケージは、クラウド（例えば、サーバー側）オーケストレータおよびアシスタントに公開されてもよい。プロジェクトは、パッケージとして公開されたときにコンパイルされてもよい。

10

20

30

40

50

【 0 0 4 0 】

いくつかの実施形態におけるトリガは、アプリケーションプログラミングインターフェース (API) トリガである。API トリガは、アプリケーションまたはオペレーティングシステム (OS) から来ることができ、ユーザーがユーザーインターフェースとインタラクトし (例えば、ボタンをクリックする)、または特定の情報を入力すること、電子メールが受信されたという通知、画面上に表示されるグラフィカル要素、押されているキー、ファイル変更、要素属性変更、OS レベルのイベントなどの、様々なユーザーまたはソフトウェアイベントから生じ得るプログラム間コールである。トリガは、イベントがクラスのインスタンスであるクラスとして実装することができ、イベントは、イベントリスナーを介して検出することができる。

10

【 0 0 4 1 】

いくつかの実施形態のウェブベースの RPA デザインシステムは、プッシュトリガおよびプルトリガなどの少なくとも2つのタイプのAPI トリガに応答するように構成されたUI 自動化を開発することができる。プルトリガは、(例えば、OS が情報を探しているプロセスをサイクルするたびに、毎秒ごと、毎分ごと、10分ごとなど) 繰り返し実行され、新しい電子メールが受信されたことを示すフラグ、テキストが変更されたかどうかを確認するためのテキストフィールドの値のチェックなど、そのトリガに関連する変更が発生したかどうかをチェックすることができる。プッシュトリガは、アプリケーションによって生成されたイベント (例えば、ウェブフック) であり、トリガが発せられると、リスナーが反応する。これは、サーバーからの要求、ユーザーがボタンをクリックしたことなどにより、新しい電子メールが受信されたことに応答しても起こり得る。

20

【 0 0 4 2 】

ウェブトリガと「デスクトップ」(ローカル) トリガとの間にも違いがある。デスクトップトリガは、ユーザーのコンピューティングシステム上で発生していることに基づく傾向がある。一方、ウェブトリガはウェブサーバー上で発生し、グローバルであり得る。複数のクライアント (例えば、2人、10人、すべてなど) がウェブトリガを受信することができる。

【 0 0 4 3 】

しかしながら、いくつかの実施形態では、UI 自動化は、ウェブ上に存在するアプリケーションの上のウェブインターフェースから構築されてもよい。例えば、Workday (登録商標) のウェブページにタブがある場合、RPA 開発者は、いくつかの実施形態のクラウドベースの RPA デザインシステムを使用してこのページとインタラクトし、その上に自動化を作成することができる。この自動化は、本明細書で後により詳細に説明するように、サーバー側 (例えば、クラウドにおいて) で実行され、仮想マシン (VM) で遂行されてもよい。複数の VM は単一のサーバー上で動作することができ、各 VM は通常、独自のオペレーティングシステムおよびアプリケーションを有する。

30

【 0 0 4 4 】

いくつかの実施形態では、UI 自動化のために、ユーザーがボタン、テキストフィールドなどとインタラクトすることを可能にするUI 要素の分類法が提供されてもよい。分類法は階層的であり、どの要素がどのページおよび/またはページの一部に配置されているかなど、UI 要素間の関係を指定することができる。例えば、ユーザーは、分類法からOK ボタンを選択し、自動化にそのボタンをクリックさせることができる。

40

【 0 0 4 5 】

いくつかの実施形態では、ウェブベースのUI 自動化は、情報を取得してローカルにUI 自動化を実施するためにウェブブラウザにインストールしてウェブブラウザ内のタブと通信することができるウェブブラウザ拡張 (例えば、Google Chrome (登録商標) 拡張機能、別のウェブブラウザのプラグイン、またはウェブブラウザの機能を拡張するための任意の他の適切な機構) によって支援される。例えば、そのような拡張は、ブラウザをクリックし、ブラウザ要素などからテキストを取得し、この情報をRPA ロボットに提供することができる。いくつかの実施形態では、ウェブブラウザおよびRPA デザ

50

イナアプリケーションは両方とも、RPA開発者のコンピューティングシステム上に配置されてもよい。しかしながら、特定の形態では、ウェブベースのRPAデザイナーアプリケーションは遠隔に配置され、RPA開発者はウェブブラウザを介してウェブベースのRPAデザイナーアプリケーションとインタラクトする。これにより、ウェブベースのRPAデザイナーアプリケーションが、RPAデザイナーアプリケーションを作成した会社によって維持およびアップグレードされ、アプリケーションを自動的に最新の状態に保つことができる。

【0046】

実行時に、拡張はユーザーのコンピューティングシステム上に配置されてもよく、自動化コードは遠隔に（例えば、サーバー上、クラウドベースの環境など）配置されてもよい。オペレーティングシステムセッションは、ウェブアプリケーションが開いて実行されているユーザーのローカルコンピューティングシステム上に存在することができる。UI自動化は、ウェブ拡張部を介して、所望のアクションを実行するようにウェブアプリケーションに命令する。上記のように、いくつかの実形態では、自動化は、サーバー上のVM内で実行することができ、その機能を実装するためにユーザーコンピューティングシステム上のウェブブラウザ拡張とインタラクトする必要がない場合がある。

【0047】

いくつかの実形態では、「ヘッドレスブラウザ」を使用することができ、Windows（登録商標）または他の何らかのオペレーティングシステムでのインタラクティブなセッションは必要ない。そのようなヘッドレスブラウザは、Linux（登録商標）コンテナなどのいくつかの実形態では、任意のOS上で実行することができる。コンテナは、アプリケーションが実行されるホストシステムから分離されるようにアプリケーションを含む。自動化は、それが使用するライブラリおよび他の依存関係と共に、コンテナにパッケージ化することができる。自動化は、ウェブベースのRPAデザイナーシステムで設計することができるが、APIから呼び出すことができ、呼び出しはサーバーレスセッションに変換することができる。したがって、ヘッドレスブラウザ自動化は、例えば、オープンウェブブラウザ、インタラクティブなセッション、およびWindows（登録商標）VMを必要としない。

【0048】

いくつかの実形態では、サーバーレスセッションは、例えば、コンテナ、VM、またはWindows（登録商標）セッション内であってもよい。コンピューティングシステムがそのハードウェアリソースで処理できる限り多くのサーバーレスセッションを実行することができる。これらのサーバーレスワークフローは、例えば、オペレーティングシステムにおいて実行中のプロセスとして現れることができる。

【0049】

ヘッドレスブラウザのためのUI自動化を設計するために、デスクトップRPAデザイナーアプリケーションと同様の設計体験を提供することができ、ウェブブラウザとのユーザーインタラクションを記録することができ、および/または画面上の機能、キャプチャ要素機能などを示すことができる。レコーダは、ユーザーがインタラクトしている現在のブラウザタブの1つ上のレベルに配置されてもよい。これは、ブラウザ自体へのアクセスを提供するChrome（登録商標）拡張または他の何らかのブラウザ拡張を介して実装することができる。拡張は、設計時にロードされたページのUI自動化（例えば、要素のクリック、テーブルの作成など）を設計するためにオープンタブとインタラクトすることができる。公開されるか、そうでなければ実行時使用に利用可能になると、ヘッドレスブラウザUI自動化は、例えばVMセッションではなくLinux（登録商標）コンテナで遂行されるようにスケジューリングすることができる。これはここでは設計方法を考慮して「UI自動化」と呼ばれるが、ヘッドレスブラウザUI自動化が実行されるときにユーザーに実際に表示されるUIはない。

【0050】

いくつかの実形態のウェブベースのRPAデザイナーシステムは、ユーザーがUI自動

10

20

30

40

50

化および他の自動化を構築することを可能にする。本明細書で使用される場合、「自動化」は、いくつかの実施形態のウェブベースのRPAデザイナシステムによって開発されたものなどのRPAワークフロー論理を実装するコードである。自動化が実行される環境は、自動化が何を行うかに依存し得る。自動化が、UIインタラクションを実行して可視化されるように意図されている、および/またはユーザーに入力を求める、注目された自動化である場合には、UI自動化を正常に実行できるように、Windows（登録商標）がインストールされたVMを使用することができる。しかしながら、UI自動化を可視化したり、ヘッドレスブラウザ自動化などのユーザー入力を要求したりする必要がない場合、そのような自動化は、例えばLinux（登録商標）コンテナで実行することができる。ウェブ拡張部は、エンドユーザーのコンピューティングシステムからの情報が必要とされ、UI自動化がヘッドレスであるウェブベースのアプリケーションに使用することができる。そうでなければ、必要に応じて、この機能は、VMをスピンアップすることによって実装されてもよい。

10

【0051】

VMまたはLinux（登録商標）コンテナは、いくつかの実施形態において、RPA開発者がUI自動化または別の自動化を作成し、ユーザーのコンピューティングシステムにローカルに自動化をインストールして実行することなくそれを直接実行したい場合に使用される。そうでなければユーザーのコンピューティングシステム上で発生するであろうアクションは、代わりにVMまたはコンテナ内で発生する。そのようなVMまたはコンテナを管理するために、自動的にまたは手動でプロビジョニングされたプールを使用することができる。自動的にプロビジョニングされるプールは弾力的であり得、プール内のVMまたはコンテナの数は、需要に応じて増減され得る。一方、手動でプロビジョニングされたプールは、人間（例えば、管理者）によって指定されたVMまたはコンテナの数を含むことができる。

20

【0052】

「サーバーレス」（無人）自動化は、典型的なサーバーが実行する必要はなく、例えば、Linux（登録商標）コンテナで実行することができる。いくつかの実施形態のウェブベースのRPAデザイナシステムを使用して開発されたサーバーレス自動化は、特定の時間に行われるようにスケジュールされてもよく、そうでなければユーザーインタラクションなしで実行されてもよい。例えば、特定の電子メールが受信されると、メッセージがスラック（登録商標）を介して送信されてもよい。コンダクタアプリケーションでジョブが失敗した場合、ジョブを再実行することができ、またはジョブが特定のエラーで失敗した場合、ジョブを再実行する代わりに電子メールを管理者に送信することができる。本発明の範囲から逸脱することなく、任意の適切な自動化を開発および実施することができる。UI自動化はまた、特定の時間に、またはユーザーの要求もしくはアクションに 응답して実行されてもよい。

30

【0053】

いくつかの実施形態では、ウェブ機能を自動化するために専用のUIアクティビティのセットが提供される。例えば、ウェブコンポーネントを認識して相互作用するアクティビティは、潜在的にブラウザ間の視覚的差異を考慮して、ウェブアプリケーションに存在する傾向がある形状およびレイアウトをより正確に検出するように調整することができる。

40

【0054】

いくつかの実施形態では、ウェブベースのRPAデザイナアプリケーションとデスクトップベースのRPAデザイナアプリケーションとの間で統合を提供することができる。例えば、所与のUI自動化のためのプロジェクトデータは、XAMLなどの両方のデザイナアプリケーションバージョンが読み取ることができる共通フォーマットで保存することができる。これにより、RPA開発者は、移動中に自分のラップトップ上でオフラインでUI自動化に取り組み、インターネットアクセスが利用可能であるときにプロジェクトをクラウドにアップロードし、その後、ウェブブラウザを使用してUI自動化に取り組み続けることができる。いくつかの実施形態では、データはまた、デザイナアプリケーション

50

のウェブバージョンとデスクトップバージョンとの間で同期されてもよい。また、ウェブベースのデザイナーアプリケーションの使用は、各ユーザーのローカルメモリのフットプリントおよび処理要件を低減し、複数のウェブユーザー間の設計協調を容易にすることができる。

【0055】

いくつかの実施形態では、自動化は、特定のプラットフォーム用の自動化パッケージを変更することなく、VM、コンテナ、またはオペレーティングシステムセッションで遂行されるRPAロボットによって遂行されるように設計およびパッケージ化される。言い換えれば、同じパッケージを、例えば、パッケージを変更することなく、VMware（登録商標）VM、Linux（登録商標）コンテナ、またはWindows（登録商標）セッションで実行することができる。RPAロボットは、特定のターゲットプラットフォーム用に設計され、その上に展開されてもよいが、自動化を実行するために共通のフォーマットが使用されてもよい。例えば、ワークフローは、XAMLまたは各プラットフォーム上のRPAロボットがコンパイルされたマシンコード（例えば、デジタルリンクライブラリ（DLL））を読み取って遂行するように設計され得る他の何らかのフォーマットで記述されてもよい。これは、コードの難読化およびセキュリティを提供し、自動化をより高速に実行する可能性がある。しかしながら、特定の実施形態では、自動化は、XAML、XML、プレーンテキスト、または他の何らかの適切なフォーマットのスクリプトであってもよい。

10

【0056】

図1は、本発明の一実施形態による、ウェブベースのRPAデザイナーシステム100を示すアーキテクチャ図である。RPAシステム100は、開発者がクラウドベースのウェブアプリケーション120を介してRPAワークフローを設計および実装することを可能にするウェブブラウザ110（例えば、RPA開発者のコンピューティングシステム上で実行される）を含む。ウェブブラウザ110は、ウェブアプリケーション120を介して、アプリケーション統合、ならびに第三者アプリケーション、管理情報技術（IT）タスク、およびビジネスITプロセスを自動化するためのソリューションを提供することができる。ウェブブラウザ110およびウェブアプリケーション120は、プロセスのグラフィカル表現を含む自動化プロジェクトの開発を容易にすることができる。簡単に言えば、ウェブブラウザ110およびウェブアプリケーション120は、RPAワークフローの開発およびサービスとして実行されるRPAロボットによって実施することができる自動化のデプロイメントを容易にする。

20

30

【0057】

自動化プロジェクトは、本明細書で「アクティビティ」と定義される、ワークフローで開発されたステップのカスタムセット間の遂行順序および関係の制御をRPA開発者に与えることによって、ルールベースのプロセスの自動化を可能にする。ウェブアプリケーション120の一実施形態の商用例は、UiPath Studio Web（商標）である。各アクティビティは、ボタンのクリック、ファイルの読み取り、ログパネルへの書き込みなどのアクションを含むことができる。いくつかの実施形態では、ワークフローはネストまたは埋め込みされてもよい。

40

【0058】

いくつかのタイプのワークフローは、シーケンス、フローチャート、有限状態機械（FSM）、および/またはグローバル例外ハンドラを含むことができるが、これらに限定されない。シーケンスは、ワークフローを乱すことなく1つのアクティビティから別のアクティビティへの流れを可能にする線形プロセスに特に適し得る。フローチャートは、より複雑なビジネスロジックに特に適しており、複数の分岐論理演算子を介してより多様な方法で決定の統合およびアクティビティの接続を可能にする。FSMは、大規模なワークフローに特に適している場合がある。FSMは、条件（すなわち、遷移）またはアクティビティによってトリガされる有限数の状態をそれらの遂行において使用することができる。グローバル例外ハンドラは、遂行エラーに遭遇したときのワークフローの挙動を判定し、

50

プロセスをデバッグするのに特に適し得る。

【0059】

ワークフローがウェブブラウザ110で開発されると、自動化の遂行は、ウェブアプリケーション120を使用してテストすることができる。しかしながら、いくつかの実施形態では、ウェブブラウザ110は、ウェブアプリケーション120の自動化生成およびテスト機能を含む。自動化が適切に機能し、生産サーバー140上にデプロイメントする準備ができた後に、自動化は、機械読み取り可能なコードまたはスクリプトの形態で、生産サーバー140のコンテナ、VM、またはサーバーオペレーティングシステムセッションに展開される。いくつかの実施形態では、ロボット130はコンダクタ150によって編成される。次いで、RPAロボット130は、必要に応じて展開された自動化を遂行することができる。管理され得るロボット130のタイプは、これらに限定されないが、アテンディッドロボット132、アンアテンディッドロボット134、開発ロボット（アンアテンディッドロボット134と同様であるが、開発および試験の目的で使用される）、および非生産ロボット（アテンディッドロボット132と同様であるが、開発および試験の目的で使用される）を含む。アテンディッドロボット132は、ユーザーイベントによってトリガされ、同じコンピューティングシステム上で人間と一緒に動作する。例えば、アテンディッドロボット132は、UI自動化を実行し、充填可能な形態などを提供することができる。アテンディッドロボット132は、人間のユーザーが様々なタスクを達成するのを助けることができ、ユーザーイベントによってトリガすることができる。特定の実施形態では、アテンディッドロボット132は、ロボットレイまたはコマンドプロンプトからのみ起動することができる。いくつかの実施形態では、アテンディッドロボット132は人間の監督下で動作するべきである。

10

20

【0060】

アンアテンディッドロボット134は、仮想環境、コンテナ、またはオペレーティングシステムセッションで無人で動作することができ、多くのプロセスを自動化することができる。アンアテンディッドロボット134は、例えば、リモート遂行、監視、スケジューリング、および作業待ち行列のサポートの提供を担当することができる。いくつかの実施形態では、すべてのロボットタイプのデバッグをウェブブラウザ110およびウェブアプリケーション120で実行することができる。アテンディッドロボットおよびアンアテンディッドロボットの両方は、メインフレーム、ウェブアプリケーション、VM、コンテナ、エンタープライズアプリケーション（例えば、SAP（登録商標）、Sales Force（登録商標）、Oracle（登録商標）などによって製造されたもの）、およびコンピューティングシステムアプリケーション（例えば、デスクトップおよびラップトップアプリケーション、モバイルデバイスアプリケーション、ウェアラブルコンピュータアプリケーションなど）を含むがこれらに限定されない様々なシステムおよびアプリケーションを自動化することができる。

30

【0061】

ロボット130は、ウェブアプリケーション120を介してウェブブラウザ110に構築されたワークフローを実行する遂行エージェントである。ロボット130のいくつかの実施形態の一商用例は、UiPath Robots（商標）である。いくつかの実施形態では、ロボット130は、デフォルトでMicrosoft Windows（登録商標）Service Control Manager（SCM）管理サービスをインストールする。結果として、そのようなロボット130は、ローカルシステムアカウントの下でインタラクティブなWindows（登録商標）セッションを開き、Windows（登録商標）サービスの権利を有することができる。

40

【0062】

いくつかの実施形態では、ロボット130は、ユーザーモードで設置することができる。このようなロボット130の場合、これは、所与のロボット130が設置されているユーザーと同じ権利を有することを意味する。この特徴は、その最大の可能性で各機械の完全な利用を保証する高密度（HD）ロボットにも利用可能であり得る。いくつかの実施形

50

態では、任意のタイプのロボット130をHD環境で構成することができる。

【0063】

エグゼキュータは、Windows（登録商標）セッション下で所与のジョブを遂行することができる（すなわち、エグゼキュータはワークフローを遂行することができる）。エグゼキュータは、モニターごとのドット/インチ（DPI）設定を認識することができる。エージェントは、システムトレイウィンドウに利用可能なジョブを表示するWindows（登録商標）Presentation Foundation（WPF）アプリケーションであってもよい。エージェントは、サービスのクライアントであってもよい。エージェントは、ジョブの開始または停止および設定の変更を要求することができる。コマンドラインは、サービスのクライアントである。コマンドラインは、ジョブの開始を要求し、その出力を待つことができるコンソールアプリケーションである。

10

【0064】

上記で説明したようにロボット130のコンポーネントを分割することは、開発者、サポートユーザー、およびコンピューティングシステムが各コンポーネントが遂行しているものをより容易に実行、識別、および追跡するのに役立つ。このようにして、エグゼキュータおよびサービスに対して異なるファイアウォールルールを設定するなど、特別な挙動をコンポーネントごとに構成することができる。エグゼキュータは、いくつかの実施形態では、モニターごとにDPI設定を常に認識することができる。結果として、ワークフローは、それらが作成されたコンピューティングシステムの構成にかかわらず、任意のDPIで遂行され得る。いくつかの実施形態では、ウェブブラウザ110およびウェブアプリケーション120からのプロジェクトは、ブラウザのズームレベルとは無関係であってもよい。DPIを認識していない、または意図的に認識していないとマークされたアプリケーションの場合、いくつかの実施形態ではDPIを無効にすることができる。

20

【0065】

いくつかの実施形態では、コンダクタ150は、プロビジョニング、デプロイメント、構成、キューイング、監視、ロギング、および/または相互接続性の提供を含むがこれらに限定されない様々な機能を有することができる。特定の実施形態では、コンダクタ150は、生産サーバー140に遠隔配置されてもよい。プロビジョニングは、（例えば、例えば、コンダクタウェブアプリケーションを介して）ロボット130とコンダクタ150との間の接続の作成および保守を含むことができる。デプロイメントは、遂行のために割り当てられたロボット130へのパッケージバージョンの正しい配信を保証することを含むことができる。構成は、ロボット環境およびプロセス構成の維持および配信を含むことができる。キューイングは、キューおよびキュー項目の管理を提供することを含むことができる。監視は、ロボット識別データを追跡し、ユーザー権限を維持することを含むことができる。ロギングは、データベース（例えば、SQLデータベース）および/または別のストレージ機構（例えば、大規模なデータセットを格納し、迅速にクエリする能力を提供するElastic Search（登録商標））へのログの格納およびインデックス付けを含むことができる。コンダクタ150は、第三者のソリューションおよび/またはアプリケーションのための通信の集中ポイントとして作用することによって相互接続性を提供することができる。コンダクタ150は、展開されたロボットおよび展開されたロボットが動作するコンピューティングシステムを含むインフラストラクチャを管理することができる。

30

40

【0066】

図2は、本発明の一実施形態による、ウェブベースのRPAデザイナシステムの一部または全部を実装するように構成されたコンピューティングシステム200を示すアーキテクチャ図である。いくつかの実施形態では、コンピューティングシステム200は、本明細書に図示および/または記載されたコンピューティングシステムのうちの1つまたは複数であってもよい。コンピューティングシステム200は、情報を通信するためのバス205または他の通信機構と、情報を処理するためにバス205に結合されたプロセッサ210と、を含む。プロセッサ210は、中央プロセッシングユニット（CPU）、特定用

50

途集積回路（ASIC）、フィールドプログラマブルゲートアレイ（FPGA）、グラフィックスプロセッシングユニット（GPU）、それらの複数のインスタンス、および/またはそれらの任意の組み合わせを含む、任意のタイプの汎用または専用プロセッサであってもよい。プロセッサ210はまた、複数の処理コアを有してもよく、コアの少なくともいくつかは、特定の機能を実行するように構成されてもよい。いくつかの実施形態では、複数並列処理が使用されてもよい。特定の実施形態では、プロセッサ210の少なくとも一方は、生体ニューロンを模倣する処理要素を含むニューロモーフィック回路であってもよい。いくつかの実施形態では、ニューロモーフィック回路は、フォン・ノイマン・コンピューティングアーキテクチャの典型的なコンポーネントを必要としない場合がある。

【0067】

10

コンピューティングシステム200は、プロセッサ210によって遂行される情報および命令を格納するためのメモリ215をさらに含む。メモリ215は、ランダムアクセスメモリ（RAM）、読み出し専用メモリ（ROM）、フラッシュメモリ、キャッシュ、磁気もしくは光ディスクなどの静的ストレージ、または任意の他のタイプの非一時的コンピュータ可読媒体、またはそれらの組み合わせの任意の組み合わせで構成することができる。非一時的コンピュータ可読媒体は、プロセッサ210によってアクセス可能な任意の利用可能な媒体であってもよく、揮発性媒体、不揮発性媒体、またはその両方を含んでもよい。媒体はまた、取り外し可能、取り外し不能、またはその両方であってもよい。

【0068】

さらに、コンピューティングシステム200は、無線接続および/または有線接続を介して通信ネットワークへのアクセスを提供するためのトランシーバなどの通信デバイス220を含む。いくつかの実施形態では、通信デバイス220は、周波数分割多元接続（FDMA）、シングルキャリアFDMA（SC-FDMA）、時分割多元接続（TDMA）、符号分割多元接続（CDMA）、直交周波数分割多重方式（OFDM）、直交周波数分割多元接続（OFDMA）、移動体用グローバルシステム（GSM）通信、汎用パケット無線サービス（GPRS）、ユニバーサル移動体通信システム（UMTS）、cdma2000、広帯域CDMA（W-CDMA）、高速ダウンリンクパケットアクセス（HSDPA）、高速アップリンクパケットアクセス（HSUPA）、高速パケットアクセス（HSPA）、ロングタームエボリューション（LTE）、LTEアドバンスド（LTE-A）、802.11x、Wi-Fi、ZigBee、超広帯域無線（UWB）、802.16x、802.15、ホームノードB（HNB）、Bluetooth、無線周波数識別（RFID）、赤外線データ協会（IrDA）、近距離通信（NFC）、第5世代（5G）、新無線（NR）、それらの任意の組み合わせ、ならびに/あるいは本発明の範囲から逸脱することなく、任意の他の現在存在する、または将来実施される通信規格および/またはプロトコルを使用するように構成されてもよい。いくつかの実施形態では、通信デバイス220は、本発明の範囲から逸脱することなく、単一、アレイ、位相、切り替え、ビームフォーミング、ビームステア、それらの組み合わせ、および/または任意の他のアンテナ構成である1つまたは複数のアンテナを含むことができる。

20

30

【0069】

プロセッサ210はさらに、バス205を介して、プラズマディスプレイ、液晶ディスプレイ（LCD）、発光ダイオード（LED）ディスプレイ、電界放出ディスプレイ（FED）、有機発光ダイオード（OLED）ディスプレイ、フレキシブルOLEDディスプレイ、フレキシブル基板ディスプレイ、プロジェクションディスプレイ、4Kディスプレイ、高精細ディスプレイ、Retina（登録商標）ディスプレイ、インプレーンスイッチング（IPS）ディスプレイ、またはユーザーに情報を表示するための任意の他の適切なディスプレイなどのディスプレイ225にさらに結合される。ディスプレイ225は、抵抗性、容量性、表面弾性波（SAW）容量性、赤外線、光学イメージング、分散信号技術、音響パルス認識、フラストレート全内部反射などを使用して、タッチ（触覚）ディスプレイ、3次元（3D）タッチディスプレイ、マルチ入力タッチディスプレイ、マルチタッチディスプレイなどとして構成され得る。本発明の範囲から逸脱することなく、任意の

40

50

適切なディスプレイデバイスおよび触覚 I / O が使用され得る。

【 0 0 7 0 】

キーボード 2 3 0 およびコンピュータマウス、タッチパッドなどのカーソル制御デバイス 2 3 5 は、ユーザーがコンピューティングシステム 2 0 0 とインターフェースすることを可能にするためにバス 2 0 5 にさらに結合される。しかしながら、特定の実施形態では、物理的なキーボードおよびマウスが存在しなくてもよく、ユーザーは、ディスプレイ 2 2 5 および / またはタッチパッド (図示せず) のみを介してデバイスとインタラクトすることができる。入力デバイスの任意のタイプおよび組み合わせを、設計上の選択事項として使用することができる。特定の実施形態では、物理的な入力デバイスおよび / またはディスプレイは存在しない。例えば、ユーザーは、それと通信する別のコンピューティングシステムを介してコンピューティングシステム 2 0 0 と遠隔でインタラクトすることができる、またはコンピューティングシステム 2 0 0 は自律的に動作することができる。

10

【 0 0 7 1 】

メモリ 2 1 5 は、プロセッサ 2 1 0 によって遂行されると機能を提供するソフトウェアモジュールを格納する。モジュールは、コンピューティングシステム 2 0 0 のためのオペレーティングシステム 2 4 0 を含む。モジュールは、本明細書に記載のプロセスまたはその派生物の全部または一部を実行するように構成されたウェブベースの R P A モジュール 2 4 5 をさらに含む。コンピューティングシステム 2 0 0 は、追加の機能を含む 1 つまたは複数の追加の機能モジュール 2 5 0 を含むことができる。

【 0 0 7 2 】

当業者は、「システム」が、本発明の範囲から逸脱することなく、サーバー、組込みコンピューティングシステム、パーソナルコンピュータ、コンソール、パーソナルデジタルアシスタント (P D A)、携帯電話、タブレットコンピューティングデバイス、量子コンピューティングシステム、もしくは任意の他の適切なコンピューティングデバイス、またはデバイスの組み合わせとして具現化され得ることを理解するであろう。上記の機能を「システム」によって実行されるものとして提示することは、本発明の範囲を決して限定することを意図するものではなく、本発明の多くの実施形態の一例を提供することを意図している。実際、本明細書に開示する方法、システム、および装置は、クラウドコンピューティングシステムを含むコンピューティング技術と一致する局所化された形態および分散された形態で実装されてもよい。コンピューティングシステムは、ローカルエリアネットワーク (L A N)、移動通信ネットワーク、衛星通信ネットワーク、インターネット、パブリッククラウドもしくはプライベートクラウド、ハイブリッドクラウド、サーバーファーム、それらの任意の組み合わせなどの一部であるが、またはそれらによってアクセス可能であり得る。本発明の範囲から逸脱することなく、任意の局所化されたまたは分散されたアーキテクチャが使用されてもよい。

20

30

【 0 0 7 3 】

本明細書に記載されたシステム特徴のいくつかは、それらの実装の独立性をより具体的に強調するために、モジュールとして提示されていることに留意されたい。例えば、モジュールは、カスタムの超大規模集積 (V L S I) 回路またはゲートアレイ、ロジックチップ、トランジスタ、または他のディスクリートコンポーネントなどの既製の半導体を含むハードウェア回路として実装されてもよい。モジュールはまた、フィールドプログラマブルゲートアレイ、プログラマブルアレイロジック、プログラマブルロジックデバイス、グラフィックスプロセッシングユニットなどのプログラマブルハードウェアデバイスに実装されてもよい。

40

【 0 0 7 4 】

モジュールはまた、様々なタイプのプロセッサによって実行するためのソフトウェアに少なくとも部分的に実装されてもよい。遂行可能コードの識別されたユニットは、例えば、オブジェクト、プロシージャ、または関数として編成することができるコンピュータ命令の 1 つまたは複数の物理ブロックまたは論理ブロックを含むことができる。それにもかかわらず、識別されたモジュールの遂行可能ファイルは、物理的に共に配置される必要は

50

ないが、論理的に共に結合されたときにモジュールを含み、モジュールの記載された目的を達成する異なる場所に格納された異なる命令を含むことができる。さらに、モジュールは、コンピュータ可読媒体に格納されてもよく、それは、例えば、ハードディスクドライブ、フラッシュデバイス、RAM、テープ、および/または本発明の範囲から逸脱することなくデータを格納するために使用される任意の他のそのような非一時的コンピュータ可読媒体であってもよい。

【0075】

実際、遂行可能コードのモジュールは、単一の命令、または多くの命令であってもよく、いくつかの異なるコードセグメント、異なるプログラム、およびいくつかのメモリデバイスに分散されてもよい。同様に、操作データは、本明細書ではモジュール内で識別および図示されてもよく、任意の適切な形態で具現化され、任意の適切なタイプのデータ構造内に編成されてもよい。操作データは、単一のデータセットとして収集されてもよく、または異なるストレージデバイスを含む異なる場所に分散されてもよく、少なくとも部分的に、システムまたはネットワーク上の電子信号としてのみ存在してもよい。

【0076】

図3Aは、本発明の一実施形態による、ウェブベースのRPAシステム300を示すアーキテクチャ図である。システム300は、デスクトップコンピュータ302、タブレット304、およびスマートフォン306などのユーザーコンピューティングシステムを含む。しかしながら、これらに限られるわけではないがスマートウォッチ、ラップトップコンピュータ、モノのインターネット(IoT)デバイス、車両コンピューティングシステムなどを含む任意の所望のコンピューティングシステムを、本発明の範囲から逸脱することなく使用することができる。

【0077】

各コンピューティングシステム302、304、306には、ウェブブラウザ310がインストールされている。ウェブブラウザ310は、マウスクリックの実行、テキストフィールドへのテキストの入力、スクリーンショットの提供、テキストフィールドまたはウェブページの他のコンポーネントに表示される情報の提供などのUI自動化のローカル部分を実行する拡張を含むことができる。そのような拡張は、例えば、注目された自動化に有用であり得る。特定の実施形態では、コンピューティングシステム302、304、306は、UIオブジェクトリポジトリからPowerPoint(登録商標)、Outlook(登録商標)などの要素を検索するウェブベースのRPAデザイナシステムを使用して開発されたRPAロボットを実行することができる。例えば、米国特許出願第16/922,289号を参照されたい。ターゲットアプリケーションの要素(例えば、アプリケーション、画面、およびUI要素などのオブジェクトリポジトリ内のアイテムであって、アプリケーションが画面をグループ化し、画面がUI要素をグループ化する、アイテム)をワークフローに追加することができる(例えば、要素をアクティビティにドラッグアンドドロップすることによって)。ウェブブラウザ310は、本発明の範囲から逸脱することなく、任意の所望のタイプのウェブブラウザとすることができる。

【0078】

UIオブジェクトリポジトリは、アプリケーション、アプリケーションバージョン、アプリケーション画面、UI要素の集合、それらの組み合わせなどによってグループ化され得るUIオブジェクトのUIオブジェクトライブラリを含むことができる。いくつかの実施形態におけるUIオブジェクトライブラリのUIオブジェクトリポジトリは、プロジェクト内のUI記述子の管理、再利用、および信頼性の向上を容易にすることができる。UIオブジェクトを再利用可能にするために、UIオブジェクトは、RPAプロセスによって参照され得るUIオブジェクトライブラリに抽出されてもよい。例えば、セレクトまたは他のUI記述子がアプリケーションの新しいバージョンに起因して変更される場合、ライブラリは、変更されたUI記述子を含むように再作成(または再公開)されてもよい。次いで、UIオブジェクトライブラリを使用するRPAプロセスは、修正されたUI記述子バージョンを呼び出すことができる。

10

20

30

40

50

【 0 0 7 9 】

UIオブジェクトライブラリからのUI記述子は、RPAワークフローアクティビティに直接追加することができ、そうでなければアクティビティのカスタムセクタを作成するのに必要とされ得る開発者の時間を節約する。オブジェクトブラウザは、UI記述子の再利用性を可能にするために、作成されたセクタをオブジェクトライブラリに格納するデータベースを提供することができる。オブジェクトライブラリは、本明細書では、アプリケーションの特定のバージョンからの1つまたは複数の画面に対応するUI記述子の集合として定義される。UI記述子は、UI要素を見つけるための命令のセットである。

【 0 0 8 0 】

ウェブブラウザ310は、ウェブアプリケーションを実行し、ネットワーク320（例えば、ローカルエリアネットワーク（LAN）、移動通信ネットワーク、衛星通信ネットワーク、インターネット、それらの任意の組み合わせなど）を介してこれらのウェブアプリケーションに関する情報をクラウド環境のサーバー330に提供する。サーバー330は、拡張を介してウェブブラウザ310とインタラクトし、および/またはウェブブラウザ310とは別個の独自のプロセスを実装する自動化334およびRPAロボット336を含むホスティング環境332（例えば、コンテナ、VM、またはオペレーティングシステムセッション）を実行する。図6Bを参照されたい。例えば、自動化334およびRPAロボット336は、VMまたはコンテナ内で実行されてもよく、いくつかの実施形態ではヘッドレスであってもよい。サーバー330は、いくつかの実施形態では、自動化のためのデータ334（例えば、異なるバージョンの自動化334、オブジェクトリポジトリなど）をデータベース340に格納し、それらをロードするか、そうでなければそれらをRPAロボット336による遂行のために利用可能にすることができる。

【 0 0 8 1 】

RPAロボット336は、RPAロボット336がサーバー330などの所与のコンピューティングシステムにインストールされ実行される機械側サービスであるという点で、自動化334とは異なる。RPAロボット336は、コンダクタアプリケーション（例えば、UiPath Orchestrator（商標））、RPAデザイナーアプリケーションなどからの要求を待ち、要求の受信後に要求された自動化を実行することができる。それぞれのRPAロボット336は、機械読み取り可能なコードまたはスクリプトの形態で自動化334の定義を受信し、それを実行する。一方、自動化334は、例えば、パッケージの形態のワークフローの実施形態である。いくつかの実施形態では、パッケージはRPAロボット336によって使用され実行される。

【 0 0 8 2 】

自動化334は、コンピューティングシステム350上で実行されるウェブベースのRPAデザイナーアプリケーション352を使用してRPA開発者によって設計されてもよい。しかしながら、いくつかの実施形態では、ウェブベースのRPAデザイナーアプリケーションは、コンピューティングシステム350に対して遠隔であり、ウェブブラウザを介してアクセスされる。RPA開発者は、プロジェクトの作成または選択、RPAワークフローの作成または修正、RPAワークフローアクティビティの追加/修正/削除などを行うことができる。いくつかの実施形態では、ウェブベースのRPAデザイナーアプリケーション352は、RPA開発者がUIオブジェクトリポジトリから要素を選択し、それぞれのUI記述子を介してRPAワークフローのアクティビティでこれらの要素を使用することを可能にする。RPA開発者はまた、自動化334をサーバー330上で生成させ、テストを実行させ、エンドユーザーによって使用される準備ができ、サーバー330によってプールの一部として使用され、コンテナ、VM、またはオペレーティングシステムセッション332などで展開されると、自動化334を展開することができる。いくつかの実施形態では、自動化は、遠隔で実行されるのではなく、コンピューティングシステム350のウェブブラウザでローカルに生成され、そこでテストおよび検証される。

【 0 0 8 3 】

図4Aは、本発明の一実施形態による、ウェブベースのRPA開発アプリケーションウ

10

20

30

40

50

ウェブページ400のホームページビューを示すスクリーンショットである。様々なサンプルプロジェクトが作成されている場合、これらをユーザーに表示することができる。RPA開発者は、新しい自動化ボタン410をクリックすることによって新しい自動化を作成することができる。いくつかの実施形態では、プロセス、ライブラリプロジェクト、テスト自動化プロジェクト、トリガベースのプロセスなどを含み得るがこれらに限定されないプロジェクトタイプのセットなど、他のオプションが提示されてもよい。ユーザーはまた、既に作成されたプロジェクトを再び開き、編集し、再発行することができる。

【0084】

RPA開発者が新しい自動化ボタン410をクリックすると、新しい自動化ウィンドウ420が現れる。図4Bを参照されたい。ユーザーは、作成ボタンをクリックすることによって、自動化の名前および説明を入力し、新しいプロジェクトを作成することができる。しかしながら、開発者、対象企業など、本発明の範囲から逸脱することなく、任意の他の所望の情報を提供することができる。

10

【0085】

RPA開発者が新しい自動化プロジェクトを作成した後に、自動化を設計することができるキャンバス430が現れる。図4Cを参照されたい。開始オプション指定432は、自動化がどのように開始されるかをユーザーが示すことを可能にする。ここで、自動化は手動で開始するように構成されている。しかしながら、ユーザーは、トリガ（例えば、ユーザーがウェブページ上のボタンをクリックすること、特定のウェブページが開かれていること、ウェブフォームが提出されていることなど）に基づいて、所望の開始時間に基づいて、所望の定期的実行時間（例えば、毎月1日の午前零時）に基づいてなど、開始条件を別のオプションに変更することができる。アクティビティは、ユーザーが空のアクティビティ434をクリックすることによって追加することができる。

20

【0086】

ユーザーが空のアクティビティ434をクリックすると、アクティビティカテゴリおよびクイックアクションウィンドウ440が表示される。図4Dを参照されたい。ユーザーは、列挙されたカテゴリを検索または選択することができる。例えば、ユーザーがExcel（登録商標）をクリックすると、Excel（登録商標）アクティビティウィンドウ450が、Excel（登録商標）アクティビティのオプションと共に表示される。図4Eを参照されたい。

30

【0087】

ユーザーがExcel（登録商標）ファイルを使用することを選択した場合、ユーザーが今すぐ接続ボタン461をクリックした後に接続情報を提供すると構成することができるExcel（登録商標）ファイル使用アクティビティ460が作成される。特定のアプリケーションでは、アプリケーションおよび他のプロセスにアクセスするために、ウェブベースのRPAデザイナアプリケーション、ロボットトレイプロセス（例えば、UiPath Assistant（商標））、または他のサービスに許可が付与される必要がある。資格情報は、将来の自動化のために保存および再利用することができる。RPAロボットが自動化を遂行すると、RPAロボットは、自動化が必要とする接続と、自動化が実行される資格情報と、を確認することができる。RPAロボットは、基礎となるワークフロー論理を変更することなくこれらの資格情報にアクセスするように構成することができる、それらが見つからないかまたは適切に機能しない場合には資格情報を入力するようにユーザーに求めることができる。ファイルは、クラウドでホストされてもよく、リモートデータベースなどに格納されてもよい。アクティビティの終了は、使用終了Excel（登録商標）ファイルインジケータ462によって指定される。図4Fを参照されたい。

40

【0088】

他のアクティビティ、条件付きロジックなどは、空のアクティビティ464をクリックすることにより、使用Excel（登録商標）ファイルアクティビティ460内に追加および構成することができる。これは、図4Dのウィンドウ440と同じまたは同様のウィンドウを表示することができる。テスト自動化を実行するためのオプション436および

50

UI自動化プロジェクトを共有するためのオプション438も表示される。これにより、ユーザーは、ワークフローが作成されるときにワークフローを検証し、他のユーザーとワークフローの作業を共有することができる。

【0089】

ユーザーがアクティビティを設定すると、様々な選択肢が現れる。例えば、ユーザーが「今すぐ接続」ボタン461をクリックし、リモートファイルおよび/またはアプリケーションにアクセスするためのログイン情報を提供する場合、オプションはオプションペイン470の右側に表示される。図4Gを参照されたい。この場合、ユーザーは、ファイル選択メニュー471から、Excel（登録商標）ファイル「Apollocomponentslist.xlsx」を選択する。ファイル名は、ファイル名リスト472

10

【0090】

いくつかの実施形態では、ユーザーは、管理規則に応じて、解析エラーを伴う自動化を公開または実行することを許可されてもされなくてもよい。検証エラーは、いくつかの実施形態では実行および公開の自動化の前提条件であり、すべてのユーザーに許可され得る。しかしながら、静的コード解析エラーは許可されない場合があり、そのように識別される場合がある。

20

【0091】

図4A～図4Gに示すようなウェブベースのRPAデザイナーによって開発された自動化をテストするために、遂行は、開発者のコンピューティングシステム上ではなくサーバー側（例えば、クラウドにおいて）で行われてもよい。自動化を設計して遂行する以外に、RPA開発者がさらなる構成を必要とすることはない。遂行は、例えばUiPath Orchestrator（登録商標）などのコンダクタアプリケーション内のRPA開発者によって構成されてもされなくてもよいリモートVM上で行われてもよい。RPA開発者は、ウェブベースのRPAデザイナーのウェブブラウザ部分を介して、成功した遂行を通知されるか、または遂行エラーおよび/または管理規則の違反を提供され得る。特定の実施形態では、RPAワークフローは、自動化が生産用途（すなわち、実行時使用）のために承認および公開される前に、適切な権限を有する個人によってレビューされ得る。

30

【0092】

図4A～図4Gは例としてのみ提供されており、本発明の範囲から逸脱することなく、様々な他のアクティビティおよび条件を構成することができることに留意されたい。例えば、RPA開発者は、異なる条件が存在する場合には、Excel（登録商標）ファイルに対して異なるアクションを実行することを選択することができる（例えば、コンポーネントの期限が過ぎている場合には、コンポーネントの期限に関する電子メールを送信するのに対して、新しい最終コンポーネントを含む電子メールを送信し、コンポーネントの期限が過ぎている場合には、Excel（登録商標）のステータスを更新するセルに書き込み、電子メールおよび/またはメッセージを他の非電子メールアプリケーションで送信するなど）。自動化は、ウェブページとインタラクトし、ウェブページから情報を取得し、ユーザーのウェブページとのインタラクション、サーバー側の他のRPAロボットとのインタラクション、またはユーザーコンピューティングシステム上のローカルでのインタラクション、データベースからの情報の取得、または本発明の範囲から逸脱することなく任意の他のRPAプロセスを含み得る、サーバー側で所望のRPAロボットアクションを実行することができるいくつかの実施形態のウェブベースのRPAデザイナーを使用して設

40

50

計され得る。上記のように、特定の実施形態では、自動化は、VMまたはコンテナを介してサーバー側で部分的または全体的に実行されてもよい。

【0093】

図5は、本発明の一実施形態による、設計時にウェブベースのRPA開発を実行するためのプロセスを示すフローチャート500である。プロセスは、510において、ウェブブラウザ上のウェブベースのデザイナーアプリケーションのウェブインターフェースにアクセスして表示することから始まる。ウェブインターフェースは、ユーザーがRPAプロジェクトを作成し、RPAワークフローを構成し、RPAワークフロー構成をサーバー側のウェブベースのRPAデザイナーアプリケーションに提出して、遠隔で自動化を生成、遂行、および検証することを可能にする。520で、ウェブベースのデザイナーアプリケーションによってウェブインターフェースを介してRPAワークフローが設計および構成される。

10

【0094】

ユーザーがRPAワークフローに満足した後に、またはそうでなければユーザーがテストおよび検証を実行したい場合、ユーザーは、ウェブインターフェースを介して、サーバー側のウェブベースのRPAデザイナーアプリケーションに、530でワークフローを実装する自動化を生成させる。次いで、サーバー側のウェブベースのRPAデザイナーアプリケーションは、540において、RPAロボットを介して自動化を遂行し、自動化を検証する。550で検証が失敗した場合、ユーザーは520でRPAワークフローの設計および構成を継続することができる。検証が成功した場合、560でRPAワークフロー/プロジェクトを生産（実行時）使用のために発行することができる。例えば、これは、自動化コードを生成し、VM、コンテナ、またはオペレーティングシステムセッション内のRPAロボットを介して自動化コードを実行することを含み得る。いくつかの実施形態では、プロセスは次に図6、図7、または図9のステップに進むことができる。

20

【0095】

図6は、本発明の一実施形態による、ウェブ拡張部とインタラクトする実行時に自動化を遂行するためのプロセス600を示すフローチャートである。プロセスは、610においてRPAロボットを介してサーバー側自動化を遂行することから始まる。サーバー側自動化は、スケジュールされてもよいし、手動で遂行されてもよいし、トリガなどに基づいて遂行されてもよい。トリガは、アプリケーションイベント、発生しているウェブページとのユーザーインタラクション、特定の時間の経過などであり得る。その後、RPAロボットは、620において、ウェブブラウザとインタラクトし、ウェブブラウザの1つまたは複数のタブ内の1つまたは複数のウェブページから情報を取得し、またはその両方を行うために、1つまたは複数のコマンドをウェブブラウザのウェブ拡張部（例えば、クライアント側コンピューティングシステム上、VM内、コンテナ内など）に送信する。その後、ウェブ拡張部は、630においてRPAロボットによって要求されるように、ウェブブラウザとインタラクトし、ウェブブラウザの1つまたは複数のタブ内の1つまたは複数のウェブページから情報を取得し、またはその両方である。いくつかの実施形態では、ウェブ拡張部は、例えば自動化がコンテナ内で実行されるときに、およびこの拡張部によって定義された結果として、ヘッドレスで遂行することができるネイティブ機能を提供する。そのような機能は、デスクトップ/ラップトップコンピュータ、シンクライアント、シックスクライアントなどのためのいくつかの実施形態において実装することができる。情報は、ローカルコンピューティングシステムから取得されてもよく、および/またはブラウザインタラクションは、ローカルで実行されてもよい。

30

40

【0096】

そして、ウェブ拡張部は、要求されたインタラクションが発生したことの確認を提供し、要求された情報を提供し、または640で問題があった場合にはエラーメッセージを提供し、それをRPAロボットが受信する。いくつかの実施形態では、これは、自動化が達成するように設計されたタスクであり得る。しかしながら、自動化が、データベース検索の実行、電子メールの送信、ファイルへの書き込みおよび保存、ウェブ拡張部からの追加

50

のアクションおよび/または情報の要求など、ウェブ拡張部からの情報に基づいて何らかの追加の処理または他のアクションを実行するように設計されていた場合には、RPAロボットは、650でこれらのアクションを実行することができる。プロセスはまた、1つまたは複数の他のコマンドについてステップ620に戻ることができる。

【0097】

図7は、本発明の一実施形態による、VMまたはコンテナ内で実行される実行時に自動化を遂行するためのプロセス700を示すフローチャートである。プロセスは、710において、それぞれの自動化を実施するVMおよび/またはコンテナのプールを作成することから始まる。プールは、例えば、弾性的にプロビジョニングされてもよいし、手動でプロビジョニングされてもよい。次いで、サーバー側自動化が、720において、RPAロボットによってVM、コンテナ、またはオペレーティングシステムセッションで遂行される。サーバー側自動化は、スケジュールされてもよいし、手動で遂行されてもよいし、トリガなどに基づいて遂行されてもよい。その後、RPAロボットは、730において、自動化ワークフローで設計されたアクションを実行する。

10

【0098】

いくつかの実施形態では、ウェブベースのデザイナシステムを使用して、ローカルRPA拡張プロセスを介してコンピューティングシステム上で実行されているアプリケーションのローカルインスタンスとインタラクトする自動化を設計することができる。ローカルRPA拡張プロセスは、コンテナ、VM、またはオペレーティングシステムセッションで遠隔で動作するRPAロボットから通信を受信し、ローカルアプリケーションとインタラクトし、要求しているRPAロボットに情報を送り返すことができる。図8は、本発明の一実施形態による、ローカルアプリケーションとインタラクトするように構成されたウェブベースのRPAシステム800を示すアーキテクチャ図である。図3Aおよび図3Bのシステム300と同様に、システム800は、デスクトップコンピュータ802、タブレット804、スマートフォン806、ネットワーク820、サーバー830、ホスティング環境832、データベース840、開発者コンピューティングシステム850、およびウェブベースのRPAデザイナアプリケーション852を含む。システム300と同様に、ホスティング環境832は、自動化834およびRPAロボット836を含む。

20

【0099】

しかしながら、システム800では、デスクトップコンピュータ802、タブレット804、およびスマートフォン806は、デスクトップコンピュータ802、タブレット804、およびスマートフォン806上で実行されているアプリケーションとインタラクトするためにRPAロボット836の機能を拡張するローカルRPA拡張プロセス810を実行する。ローカルRPA拡張プロセス810は、ネットワーク820を介して自動化834を遂行するRPAロボット836から要求を受信することができる。例えば、ローカルRPA拡張プロセス810は、通信のためにポートを聴取していてもよい。本発明の範囲から逸脱することなく、任意の適切な通信プロトコル(例えば、伝送制御プロトコル(TCP)/インターネットプロトコル(IP)、ファイル転送プロトコル(FTP)など)を使用することができる。

30

【0100】

ローカルRPA拡張プロセス810は、RPAロボット836からの要求を解析し、関連するアクションを実行し、および/または要求された情報を取り出すことができる。例えば、RPAロボット836からの要求は、ローカルRPA拡張プロセス810に、「請求書」という用語についてユーザーの電子メールを検索し、一致する結果を検索し、その結果をRPAロボット836に送り返すために、Outlook(登録商標)用のAPIを使用するよう依頼することができる。その後、RPAロボット836は、この情報を使用して、自動化834のロジックを実行し続けることができる。

40

【0101】

いくつかの実施形態では、ローカルRPA拡張プロセス810は、プロセス間通信(IPC)プロトコルを使用してそれぞれのコンピューティングシステム上で実行されている

50

他のプロセスと通信することができる。IPCプロトコルは、オペレーティングシステムで実行されているプロセスが、潜在的に異なるセッションで、そのオペレーティングシステムで実行されている他のプロセスと通信することができる機構である。これらのプロトコルは、ネットワーク820、パイプ、コンポーネントオブジェクトモデル(COM)、リモートプロシージャコール(RPC)、ソケットなどを介した通信を促進することができる。IPCプロトコルを使用して、実行中のプロセス間で情報、要求、コマンドなどを送信することができる。

【0102】

図9は、本発明の一実施形態による、RPAロボットとは異なるコンピューティングシステム上で実行されているローカルRPA拡張プロセスとインタラクトする、実行時に自動化を遂行するためのプロセス900を示すフローチャートである。プロセスは、910において、RPAロボットを介してサーバー側自動化を遂行することから始まる。サーバー側自動化は、スケジュールされてもよいし、手動で遂行されてもよいし、トリガなどに基づいて遂行されてもよい。トリガは、アプリケーションイベント、発生しているウェブページとのユーザーインタラクション、特定の時間の経過などであり得る。次いで、RPAロボットは、ローカルRPA拡張プロセスに対してローカルに実行されている1つまたは複数のアプリケーションおよび/またはプロセス(例えば、オペレーティングシステムプロセス、アプリケーションに関連するプロセスなど)とインタラクトするために、920において、リモートコンピューティングシステム上(例えば、クライアント側コンピューティングシステム上、リモートVM内、リモートテナ内など)で実行されているローカルRPA拡張プロセスに対して1つまたは複数の要求を送信する。次いで、ローカルRPA拡張プロセスは、930において、アプリケーションにおいて所望の動作を実行するように、(例えば、APIを介して)1つもしくは複数のアプリケーションおよび/またはプロセスを制御し、アプリケーションおよび/またはコンピューティングシステム上の他のプロセスから情報を取得し、またはその両方を行う。次いで、ローカルRPA拡張アプリケーションは、940において、アプリケーションとの要求されたインタラクションが発生したことの確認を提供し、要求された情報を提供し、またはその両方を行う。いくつかの実施形態では、RPAロボットが受信する問題がある場合、エラーメッセージが提供されてもよい。

【0103】

いくつかの実施形態では、これは、自動化が達成するように設計されたタスクであり得る。しかしながら、自動化が、データベース検索の実行、電子メールの送信、ファイルへの書き込みおよび保存、ウェブ拡張部からの追加のアクションおよび/または情報の要求など、ウェブ拡張部からの情報に基づいて何らかの追加の処理または他のアクションを実行するように設計されていた場合には、RPAロボットは、950でこれらのアクションを実行することができる。プロセスはまた、1つまたは複数の他の要求についてステップ920に戻ることができる。

【0104】

図10は、本発明の一実施形態による、RPAロボットなしで(すなわち、ロボットレス)ウェブブラウザを介して設計時にウェブベースのRPA開発を実行するためのプロセス1000を示すフローチャートである。本明細書で使用される「ロボットレス」は、RPAロボットのような機能がウェブブラウザに含まれているが、RPAロボットが別個のサービスとして実行されていないことを意味する。したがって、ウェブブラウザが実行されるコンピューティングシステムにローカルクライアント側バイナリをインストールする必要はなく、いくつかの実施形態ではウェブブラウザのみが必要である。ロボット様バイナリがタブ内に含まれ、ロード時にウェブサーバーから提供されてもよい。その際、HTMLやJavaScript(商標)などをロードしてもよい。

【0105】

プロセスは、1010でウェブベースのRPAデザイナーアプリケーションのコードをウェブブラウザにダウンロードすることから始まる。コードは、ユーザーが自動化を作成お

10

20

30

40

50

よび遂行することを可能にする。このコードは、ウェブブラウザのタブにロードされるDLLなどの遂行可能バイナリを含むことができる。JavaScript（商標）などの他のフォーマットが通常ロードされるため、これは非定型的である。しかしながら、JavaScript（商標）、Servlets（商標）などの自動化遂行機能を含む任意の適切なコードが、本発明の範囲から逸脱することなく使用され得ることに留意されたい。ウェブベースのRPAデザイナーアプリケーションのウェブインターフェースは、1020でウェブブラウザに表示される。ウェブインターフェースは、RPAプロジェクトを作成し、RPAワークフローを構成し、そこから自動化を生成し遂行する機能を提供するように構成される。1030で、ウェブベースのデザイナーアプリケーションによってウェブインターフェースを介してRPAワークフローが設計および構成される。

10

【0106】

ユーザーがRPAワークフローに満足した後に、またはそうでなければユーザーが試験および検証を実行したい場合、ウェブブラウザは、1040において、構成されたRPAワークフローの自動化を生成する。この自動化は、通常のブラウザプロセスではなく、ブラウザワーカプロセスとして実行されてもよい。その後、ウェブブラウザは、1050において、生成された自動化を遂行して検証する。1060で検証が失敗した場合（例えば、エラーがあるか、または管理規則に違反している）、ユーザーは1030でRPAワークフローの設計および構成を継続することができる。検証が成功した場合、1070でRPAワークフロー/プロジェクトを生産（実行時）使用のために発行することができる。例えば、これは、ウェブブラウザの自動化コードを生成し、自動化コードをVM、コンテナ、またはオペレーティングシステムセッションでウェブブラウザで実行することを含むことができる。いくつかの実施形態では、プロセスは次に図11のステップに進むことができる。

20

【0107】

自動化を遂行およびデバッグするための機構は、この実施形態ではウェブブラウザ自体にロードされる。これは、例えば、リモートVM、コンテナ、またはオペレーティングシステムセッションを必要としない。言い換えれば、ユーザーが自動化を実行するとき、コードは、リモートコンテナ、リモートVM、またはリモートオペレーティングシステムセッションで遂行されるように転送されない。代わりに、自動化設計時間および実行時機能はウェブブラウザ内にあり、いくつかの実施形態では潜在的に同じタブ内にある。自動化を遂行するための機能は、ウェブベースのRPAデザイナーアプリケーションのウェブページがロードされたときにウェブブラウザにロードすることができる。これは、いくつかの実施形態ではウェブブラウザ拡張を必要としない。したがって、タブ自体は本質的に設計時RPAデザイナーと実行時RPAロボットの両方になり、ユーザーは自動化を作成、遂行、およびデバッグすることができる。いくつかの実施形態では、生産RPAロボットによって遂行されるのではなく、生産サーバーのウェブブラウザで実行される自動化を設計することもできる。

30

【0108】

図11は、本発明の一実施形態による、コンテナ、VM、またはオペレーティングシステムセッションのウェブブラウザにおいて実行時に自動化を遂行するためのプロセス1100を示すフローチャートである。プロセスは、1110において、コンテナ、VM、またはサーバーの動作セッションで実行されているウェブブラウザを介してサーバー側自動化を遂行することから始まる。この形態の自動化のみが新規である。しかしながら、いくつかの実施形態では、自動化は、図6および図8のプロセス600および800とそれぞれ同様に、1120において、リモートコンピューティングシステム上で実行されるウェブ拡張部またはローカルRPA拡張プロセスにコマンドを送信する。次いで、ウェブ拡張部またはローカルRPA拡張プロセスは、1130において、要求されたアクションを実行し、および/または要求された情報を取得し、1140において、サーバーのコンテナ、VM、または動作セッション内のウェブブラウザに確認および/または情報を提供する。その後、ウェブブラウザは、1150において、自動化に関連する追加のアクション

40

50

を実行してもよい。

【0109】

図5～図7および図9～図11で実行されるプロセスステップは、本発明の実施形態に従って、プロセッサが図5～図7および図9～図11で説明したプロセスの少なくとも一部を実行するための命令を符号化するコンピュータプログラムによって実行されてもよい。コンピュータプログラムは、非一時的コンピュータ可読媒体上で具現化されてもよい。コンピュータ可読媒体は、ハードディスクドライブ、フラッシュデバイス、RAM、テープ、および/またはデータを格納するために使用される任意の他のそのような媒体または媒体の組み合わせであってもよいが、これらに限定されない。コンピュータプログラムは、コンピュータ可読媒体に格納することもできる、図5～図7および図9～図11で説明したプロセスステップの全部または一部を実施するようにコンピューティングシステムのプロセッサ（例えば、図2のコンピューティングシステム200のプロセッサ210）を制御するための符号化命令を含むことができる。

10

【0110】

コンピュータプログラムは、ハードウェア、ソフトウェア、またはハイブリッド実装で実施することができる。コンピュータプログラムは、互いに動作可能に通信し、表示するために情報または命令を渡すように設計されたモジュールから構成することができる。コンピュータプログラムは、汎用コンピュータ、ASIC、または任意の他の適切なデバイス上で動作するように構成することができる。

【0111】

本発明の様々な実施形態のコンポーネントは、本明細書の図に一般的に記載および図示するように、多種多様な異なる構成で配置および設計されてもよいことが容易に理解されよう。したがって、添付の図面に表される本発明の実施形態の詳細な説明は、特許請求される本発明の範囲を限定することを意図するものではなく、本発明の選択された実施形態を単に代表するものである。

20

【0112】

本明細書を通して説明される本発明の特徴、構造、または特性は、1つまたは複数の実施形態において任意の適切な方法で組み合わせることができる。例えば、本明細書全体を通して「特定の実施形態」、「いくつかの実施形態」、または同様の文言への言及は、実施形態に関連して説明される特定の特徵、構造、または特性が本発明の少なくとも1つの実施形態に含まれることを意味する。したがって、本明細書全体を通して、「特定の実施形態では」、「いくつかの実施形態では」、「他の実施形態では」、または同様の文言の出現は、必ずしもすべてが同じ実施形態のグループを指すわけではなく、記載された特徴、構造、または特性は、1つまたは複数の実施形態において任意の適切な方法で組み合わせることができる。

30

【0113】

本明細書を通して特徴、利点、または同様の文言を参照することは、本発明で実現され得る特徴および利点のすべてが本発明の任意の単一の実施形態であるべきであること、または本発明の任意の単一の実施形態であることを意味するものではないことに留意されたい。むしろ、特徴および利点に言及する文言は、一実施形態に関連して説明される特定の特徵、利点、または特性が本発明の少なくとも1つの実施形態に含まれることを意味すると理解される。したがって、本明細書を通して、特徴および利点、ならびに同様の文言の説明は、必ずしもそうとは限らないが、同じ実施形態を参照することができる。

40

【0114】

さらに、本発明の記載された特徴、利点、および特性は、1つまたは複数の実施形態において任意の適切な方法で組み合わせることができる。当業者は、特定の実施形態の特定の特徵または利点の1つまたは複数なしで本発明を実施できることを認識するであろう。他の例では、本発明のすべての実施形態には存在しない可能性がある特定の実施形態において、追加の特徴および利点が認識され得る。

【0115】

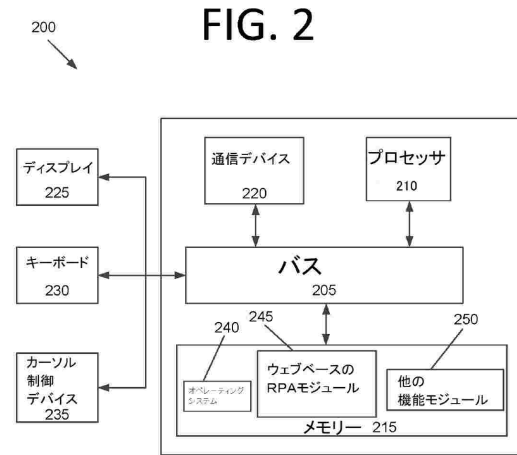
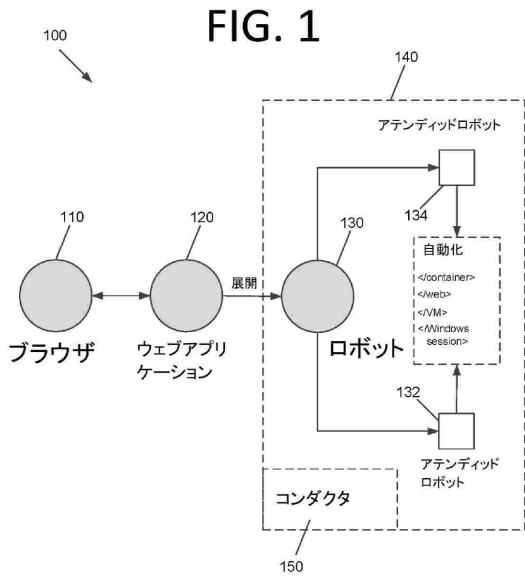
50

当業者は、上述の本発明が、異なる順序のステップ、および/または開示されているものとは異なる構成のハードウェア要素を用いて実施され得ることを容易に理解するであろう。したがって、本発明をこれらの好ましい実施形態に基づいて説明してきたが、本発明の趣旨および範囲内に留まりながら、特定の修正、変形、および代替構築が明らかであることは、当業者には明らかであろう。したがって、本発明の範囲を決定するために、添付の特許請求の範囲を参照すべきである。

【図面】

【図 1】

【図 2】



10

20

30

40

50

【 3 A 】

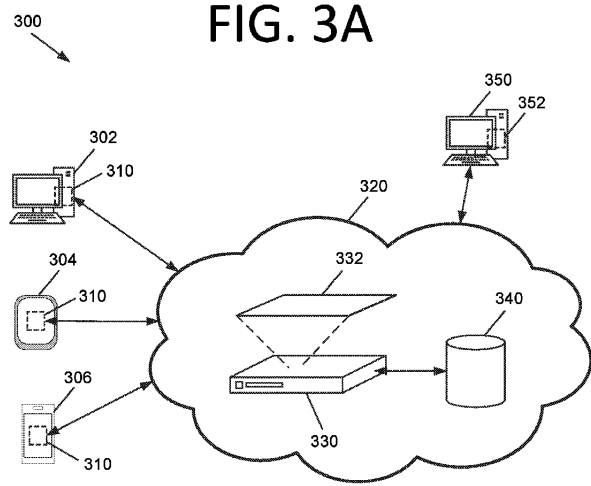


FIG. 3A

【 3 B 】

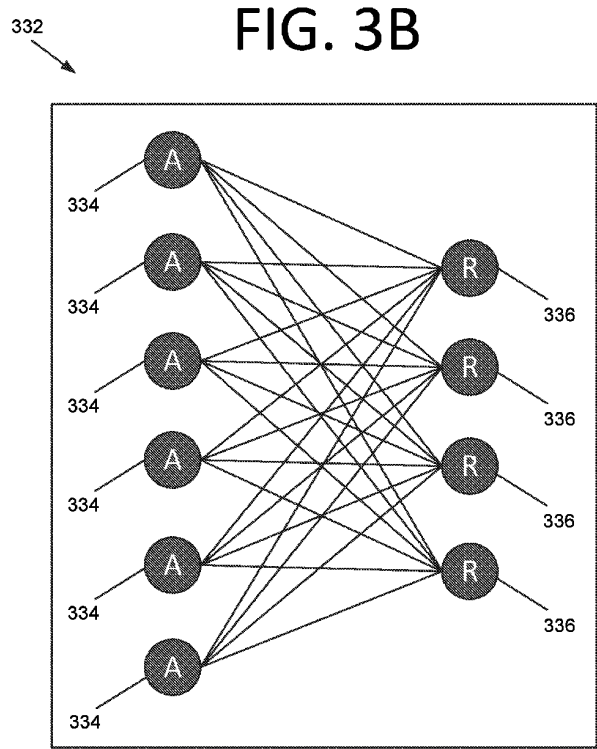


FIG. 3B

10

20

【 4 A 】

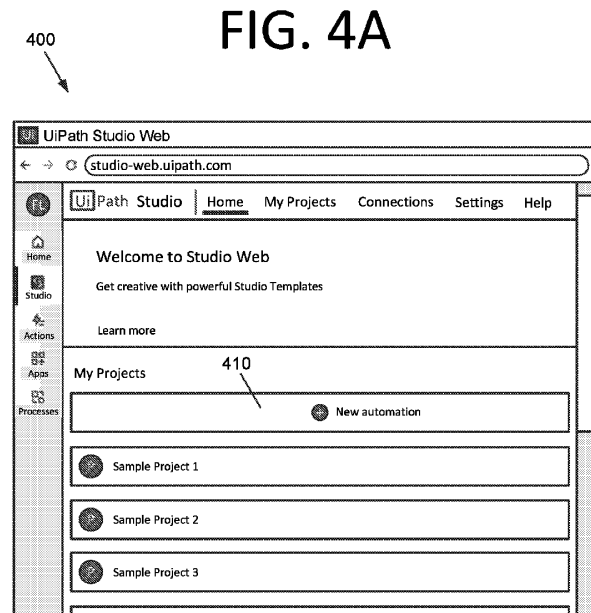


FIG. 4A

【 4 B 】

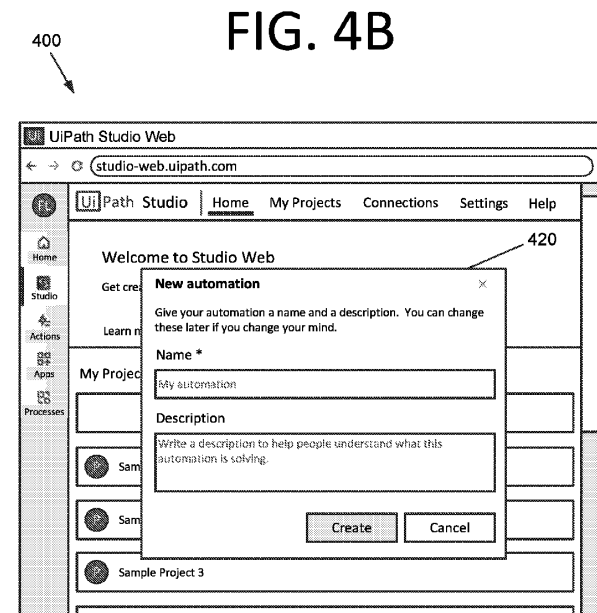


FIG. 4B

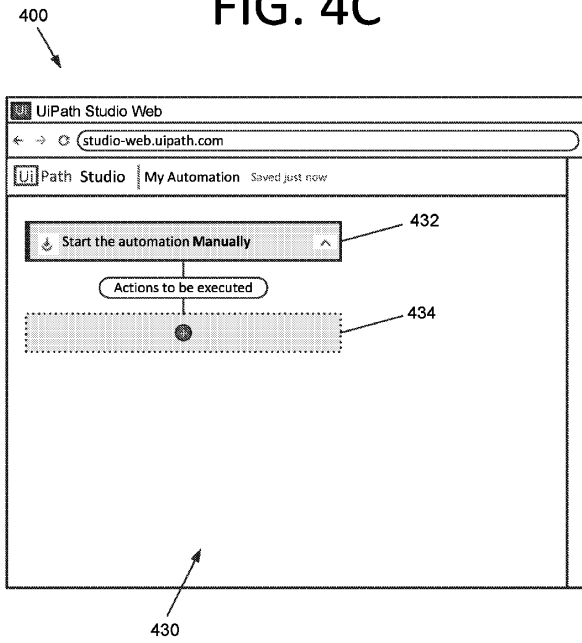
30

40

50

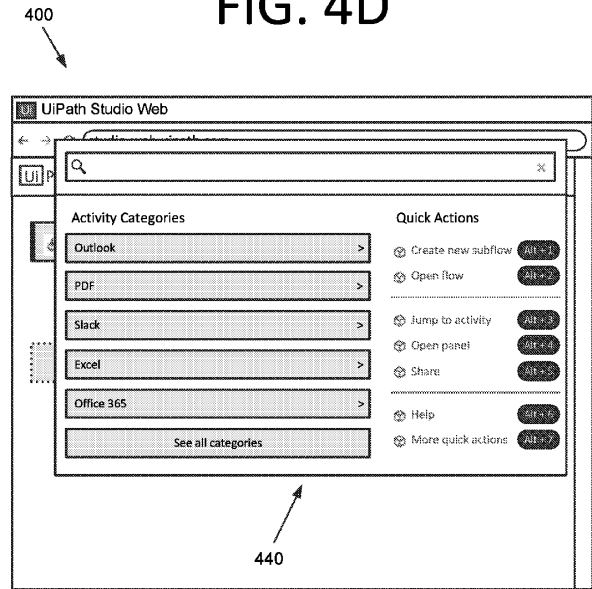
【 4 C 】

FIG. 4C



【 4 D 】

FIG. 4D

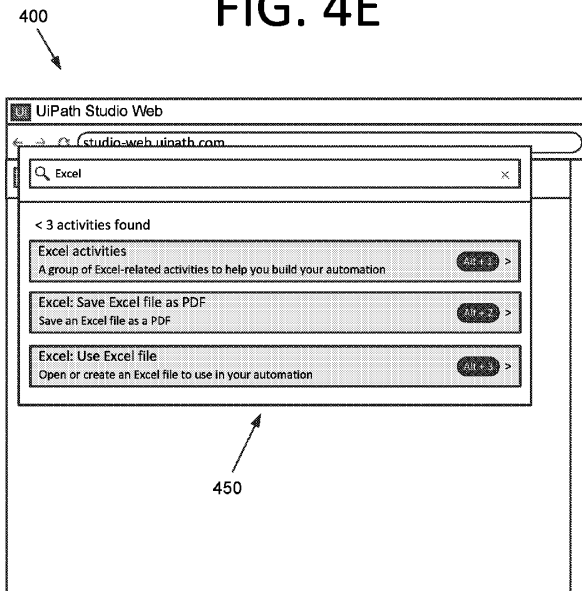


10

20

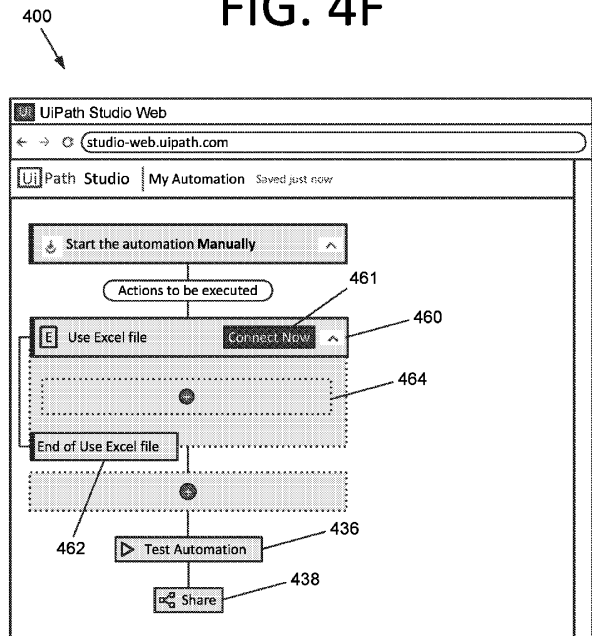
【 4 E 】

FIG. 4E



【 4 F 】

FIG. 4F



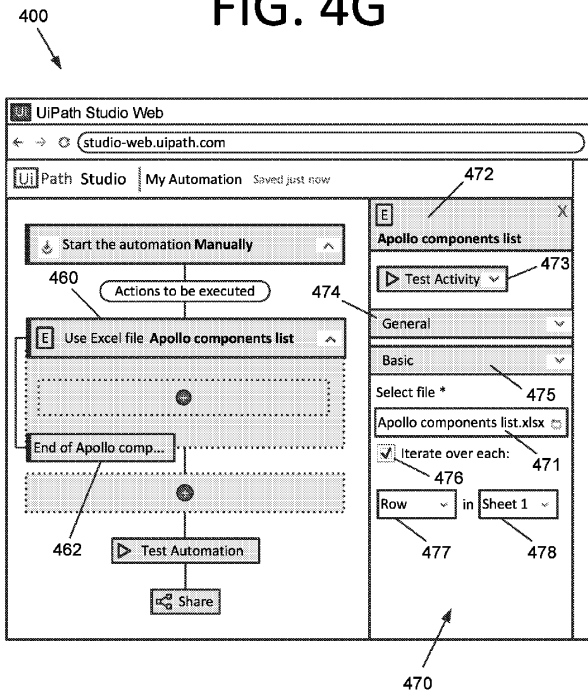
30

40

50

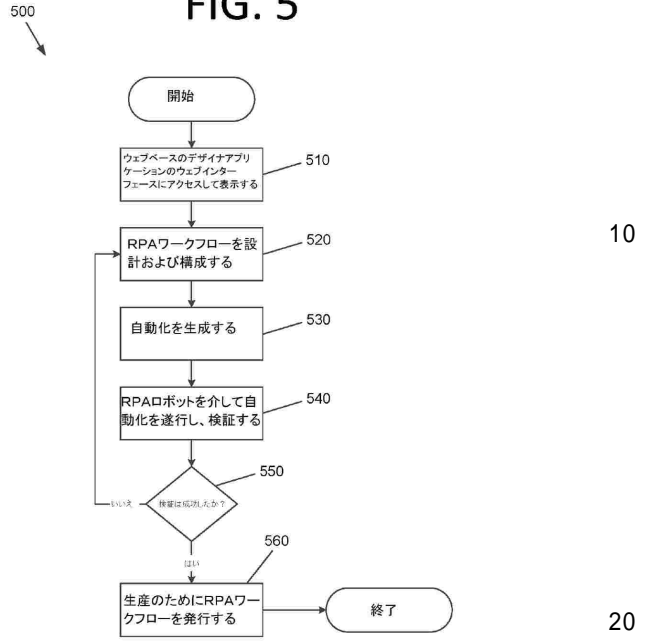
【 図 4 G 】

FIG. 4G



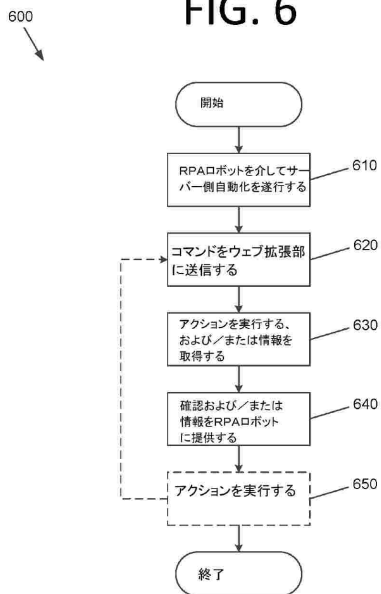
【 図 5 】

FIG. 5



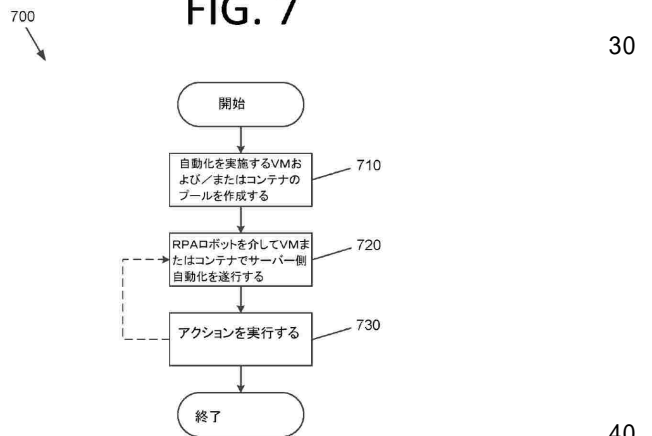
【 図 6 】

FIG. 6



【 図 7 】

FIG. 7



10

20

30

40

50

【 図 8 】

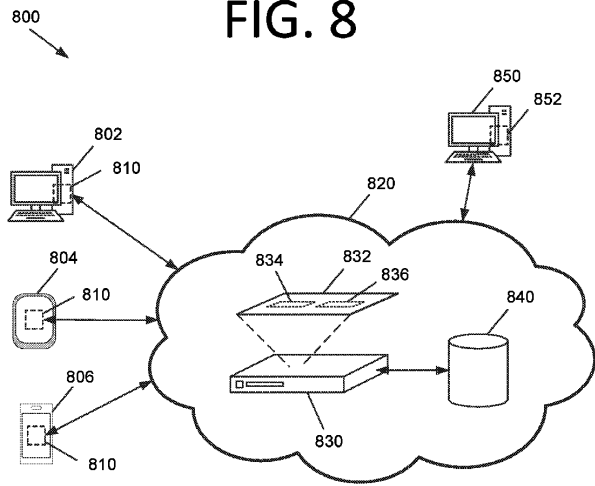


FIG. 8

【 図 9 】

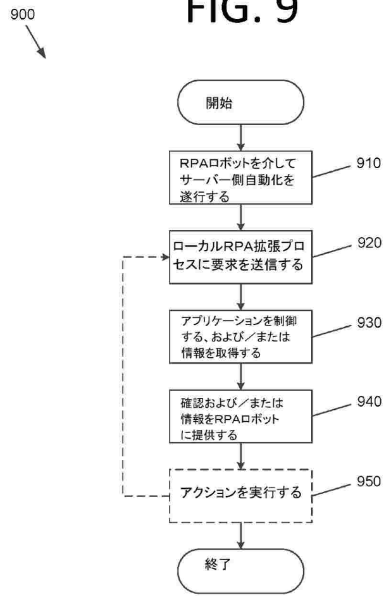


FIG. 9

10

20

【 図 10 】

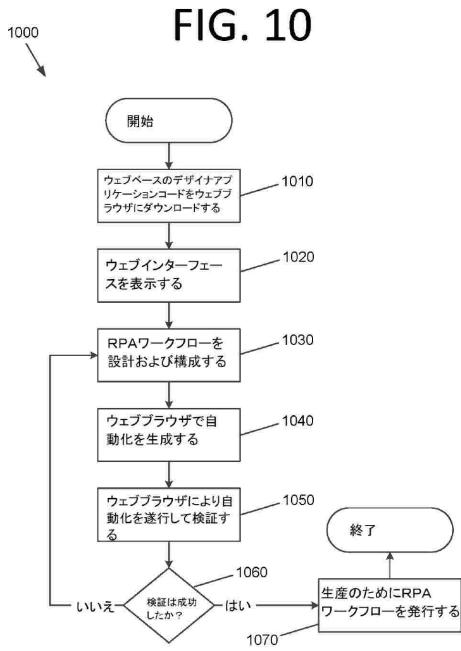


FIG. 10

【 図 11 】

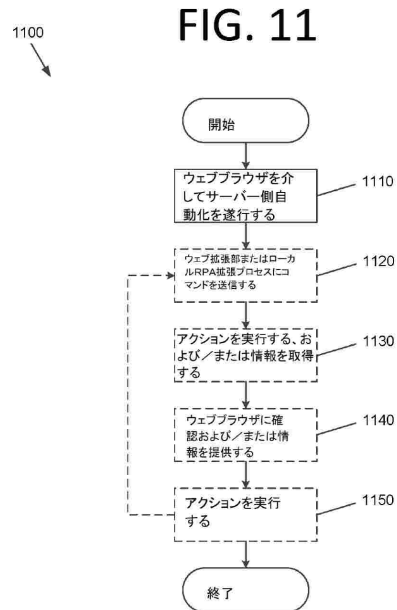


FIG. 11

30

40

50

【 国際調査報告 】

INTERNATIONAL SEARCH REPORT		International application No. PCT/US2021/056929
A. CLASSIFICATION OF SUBJECT MATTER G06F 9/448(2018.01); G06F 9/455(2006.01); G05B 19/042(2006.01); G06Q 10/10(2012.01); According to International Patent Classification (IPC) or to both national classification and IPC		
B. FIELDS SEARCHED Minimum documentation searched (classification system followed by classification symbols) G06F 9/448(2018.01); G06F 3/0482(2013.01); G06F 3/0489(2013.01); G06F 9/455(2006.01); G06F 9/50(2006.01); H04L 29/08(2006.01); H04L 9/32(2006.01) Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched Korean utility models and applications for utility models Japanese utility models and applications for utility models Electronic data base consulted during the international search (name of data base and, where practicable, search terms used) eKOMPASS(KIPO internal) & Keywords: cloud, automation, web browser, operating system, virtual machine (VM), container, serverless, tab, RPA (robotic process automation) robot, web extension, interact, confirmation, tab, application, headless, trigger		
C. DOCUMENTS CONSIDERED TO BE RELEVANT		
Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
Y	WILLIAM SCOTT et al., 'freedom.js: an Architecture for Serverless Web Applications', In: UW Technical Report, 03 May 2013 [retrieved on 2022.02.08]. Retrieved from<https://norfolk.cs.washington.edu/htbin-post/unrestricted/tr/list.cgi?sortby=date>. pages 1-9, 13; and figure 1	1-21
Y	KAREN OSMUNDSEN et al., 'Organizing Robotic Process Automation: Balancing Loose and Tight Coupling', ISBN: 978-0-9981331-2-6, In: Proceedings of the 52nd Hawaii International Conference on System Sciences, 2019 abstract; and section 2	1-21
A	US 2021-0089358 A1 (UNIVERSITY OF WASHINGTON) 25 March 2021 (2021-03-25) paragraphs [0005]-[0008]; and claims 1-4	1-21
A	US 2017-0177171 A1 (MICROSOFT TECHNOLOGY LICENSING, LLC) 22 June 2017 (2017-06-22) paragraphs [0002]-[0003], [0015]; and claims 1-4	1-21
<input checked="" type="checkbox"/> Further documents are listed in the continuation of Box C. <input checked="" type="checkbox"/> See patent family annex.		
* Special categories of cited documents: "A" document defining the general state of the art which is not considered to be of particular relevance "D" document cited by the applicant in the international application "E" earlier application or patent but published on or after the international filing date "L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified) "O" document referring to an oral disclosure, use, exhibition or other means "P" document published prior to the international filing date but later than the priority date claimed		"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention "X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone "Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art "&" document member of the same patent family
Date of the actual completion of the international search 23 February 2022		Date of mailing of the international search report 23 February 2022
Name and mailing address of the ISA/KR Korean Intellectual Property Office 189 Cheongsa-ro, Seo-gu, Daejeon 35208, Republic of Korea Facsimile No. +82-42-481-8578		Authorized officer YANG, Jeong Rok Telephone No. +82-42-481-5709

Form PCT/ISA/210 (second sheet) (July 2019)

10

20

30

40

50

INTERNATIONAL SEARCH REPORT

International application No.

PCT/US2021/056929**C. DOCUMENTS CONSIDERED TO BE RELEVANT**

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	US 10958732 B1 (MICHAEL JEFFREY PROCOPIO) 23 March 2021 (2021-03-23) claims 1-16	1-21

10

20

30

40

50

INTERNATIONAL SEARCH REPORT
Information on patent family members

International application No.
PCT/US2021/056929

Patent document cited in search report			Publication date (day/month/year)	Patent family member(s)			Publication date (day/month/year)
US	2021-0089358	A1	25 March 2021	None			
US	2017-0177171	A1	22 June 2017	CN	108369600	A	03 August 2018
				EP	3391250	A2	24 October 2018
				US	10845950	B2	24 November 2020
				WO	2017-105975	A2	22 June 2017
				WO	2017-105975	A3	10 August 2017
US	10958732	B1	23 March 2021	None			

10

20

30

40

50

フロントページの続き

,RW,SD,SL,ST,SZ,TZ,UG,ZM,ZW),EA(AM,AZ,BY,KG,KZ,RU,TJ,TM),EP(AL,AT,BE,BG,CH,CY,CZ,DE,D
K,EE,ES,FI,FR,GB,GR,HR,HU,IE,IS,IT,LT,LU,LV,MC,MK,MT,NL,NO,PL,PT,RO,RS,SE,SI,SK,SM,TR),O
A(BF,BJ,CF,CG,CI,CM,GA,GN,GQ,GW,KM,ML,MR,NE,SN,TD,TG),AE,AG,AL,AM,AO,AT,AU,AZ,BA,B
B,BG,BH,BN,BR,BW,BY,BZ,CA,CH,CL,CN,CO,CR,CU,CZ,DE,DJ,DK,DM,DO,DZ,EC,EE,EG,ES,FI,GB,GD
,GE,GH,GM,GT,HN,HR,HU,ID,IL,IN,IR,IS,IT,JO,JP,KE,KG,KH,KN,KP,KR,KW,KZ,LA,LC,LK,LR,LS,LU,
LY,MA,MD,ME,MG,MK,MN,MW,MX,MY,MZ,NA,NG,NI,NO,NZ,OM,PA,PE,PG,PH,PL,PT,QA,RO,RS,
RU,RW,SA,SC,SD,SE,SG,SK,SL,ST,SV,SY,TH,TJ,TM,TN,TR,TT,TZ,UA,UG,US,UZ,VC,VN,WS,ZA,ZM,Z
W

(特許庁注：以下のものは登録商標)

1 . Z I G B E E

2 . B L U E T O O T H

3 . W C D M A

アメリカ合衆国 1 0 0 1 6 ニューヨーク , ニューヨーク , パーク アベニュー 9 0 , トウエン
テイス フロア

Fターム(参考) 5B376 AA07 AA21 AA38 AB01 EA17