

(19)日本国特許庁(JP)

(12)特許公報(B2)

(11)特許番号
特許第7398189号
(P7398189)

(45)発行日 令和5年12月14日(2023.12.14)

(24)登録日 令和5年12月6日(2023.12.6)

(51)国際特許分類

G 0 5 B 23/02 (2006.01)**F I****G 0 5 B**

23/02

Z**請求項の数 15 外国語出願 (全36頁)**

(21)出願番号	特願2018-171534(P2018-171534)	(73)特許権者	512132022 フィッシュ - ローズマウント システムズ, インコーポレイテッド アメリカ合衆国 テキサス 78681 - 7430 ラウンド ロック ウエスト ルイス ヘナ ブルバード 1100 ビルディング 1 エマーソン プロセス マネジメント
(22)出願日	平成30年9月13日(2018.9.13)	(74)代理人	110002860 弁理士法人秀和特許事務所
(65)公開番号	特開2019-75098(P2019-75098A)	(74)代理人	100113608 弁理士 平川 明
(43)公開日	令和1年5月16日(2019.5.16)	(74)代理人	100138357 弁理士 矢澤 広伸
審査請求日	令和3年9月13日(2021.9.13)	(72)発明者	アーロン シー . ジョーンズ 最終頁に続く
(31)優先権主張番号	62/558,345		
(32)優先日	平成29年9月13日(2017.9.13)		
(33)優先権主張国・地域又は機関	米国(US)		
(31)優先権主張番号	62/567,225		
(32)優先日	平成29年10月2日(2017.10.2)		
(33)優先権主張国・地域又は機関	米国(US)		

(54)【発明の名称】 方法、コンピューティングデバイスおよびシステム

(57)【特許請求の範囲】**【請求項1】**

モジュール制御システムの構成を効率的に管理するための方法であって、

1つ以上のプロセッサによって、ワークステーションとモジュール制御システムとの間の通信連結を検出することと、

前記検出された通信連結に応答して、前記ワークステーションのメモリに格納されたアクティブプロジェクトが前記モジュール制御システムのメモリに格納された保存されたプロジェクトと一致するか否かを判定することであって、前記アクティブプロジェクト及び前記保存されたプロジェクトの各々が、前記モジュール制御システムの動作を制御するためのそれぞれの調整されたデータセットを含む、判定することと、

前記アクティブプロジェクトと前記保存されたプロジェクトとが一致しない場合に、

前記1つ以上のプロセッサによって、前記アクティブプロジェクトが前記モジュール制御システムに適正であるか否かを判定し、

前記アクティブプロジェクトが前記モジュール制御システムに対して不適正である場合に、前記モジュール制御システムの前記メモリから前記ワークステーションの前記メモリ内の前記アクティブプロジェクトに、前記保存されたプロジェクトの少なくとも一部分を転送するための制御を、対話型ユーザインターフェースを介して自動的に提供することと、を含む、方法。

【請求項2】

前記アクティブプロジェクトが前記モジュール制御システムに不適正である場合に、

- (i) 前記ワークステーションが、前記モジュール制御システムの前記メモリ内に前記アクティブプロジェクトを保存することを防止することと、
- (ii) 前記ワークステーションが、分散型制御システムの 1 つ以上のプロトコルを使用して、通信セッションを確立することを防止することと、
の少なくとも 1 つをさらに含む、請求項 1 に記載の方法。

【請求項 3】

前記アクティブプロジェクトが前記モジュール制御システムに対して不適正である場合に、前記ワークステーションの前記メモリに格納された複数の候補アクティブプロジェクトの中から異なるアクティブプロジェクトを選択するための制御を、前記ユーザインターフェースを介して自動的に提供することをさらに含む、請求項 1 または 2 に記載の方法。 10

【請求項 4】

前記モジュール制御システムの前記メモリに格納された前記保存されたプロジェクトに割り当てられた名前に基づいて、前記モジュール制御システムの前記メモリから転送された前記保存されたプロジェクトの前記少なくとも一部分について提案された名前を自動的に生成することをさらに含む、請求項 3 に記載の方法。

【請求項 5】

前記アクティブプロジェクトと前記保存されたプロジェクトとが一致しない場合に、前記アクティブプロジェクトと前記保存されたプロジェクトとの間で競合する項目のリストを生成することと、

前記アクティブプロジェクトに転送するための前記項目のリスト内の個々の項目を選択するための制御を、前記ユーザインターフェースを介して提供することと、をさらに含む、請求項 1 から 4 のいずれか 1 項に記載の方法。 20

【請求項 6】

前記項目のリスト内の各項目について、

(i) 項目名、(ii) 項目タイプ、(iii) 構成タイプ、及び(iv) 前記項目の最後の編集のタイムスタンプのうちの少なくとも 1 つの標示を提供することをさらに含む、請求項 5 に記載の方法。

【請求項 7】

前記項目のリストは、第 1 の項目のリストであり、

前記方法は、前記アクティブプロジェクトと前記保存されたプロジェクトとが一致しない場合に、 30

前記保存されたプロジェクトに含まれ、かつ前記アクティブプロジェクトに含まれない第 2 の項目のリストを生成することと、

前記アクティブプロジェクトに転送するための前記第 2 の項目のリスト内の個々の項目を選択するための制御を、前記ユーザインターフェースを介して提供することと、をさらに含む、請求項 5 または 6 に記載の方法。

【請求項 8】

前記アクティブプロジェクトと前記保存されたプロジェクトとが一致するか否かを判定することは、前記 1 つ以上のプロセッサによって、前記アクティブプロジェクト及び前記保存されたプロジェクトのそれぞれのタイムスタンプを比較することを含む、請求項 1 から 7 のいずれか 1 項に記載の方法。 40

【請求項 9】

前記モジュール制御システムから前記ワークステーションを切断するためのコマンドを、前記ユーザインターフェースを介して受信することと、

前記コマンドに応答して、前記保存されたプロジェクトの新しいバージョンとして、前記アクティブプロジェクトの少なくとも一部分を前記ワークステーションの前記メモリに転送するための制御を自動的に提供することと、をさらに含む、請求項 1 から 8 のいずれか 1 項に記載の方法。

【請求項 10】

前記保存されたプロジェクトの新しいバージョンとして、前記アクティブプロジェクト 50

の少なくとも一部分を前記ワークステーションの前記メモリに転送するための第1の制御を提供することと、

前記モジュール制御システムが前記保存されたプロジェクトのデータに従って動作し始めるように、前記モジュール制御システムの前記メモリに格納された前記アクティブプロジェクトを開発するための第2の制御を提供することと、をさらに含む、請求項1に記載の方法。

【請求項11】

前記検出された通信連結に応答して、

前記モジュール制御システムにアクセスするための認証情報を、前記ユーザインターフェースを介して自動的に要求することと、

前記認証情報が不適正である場合に、前記モジュール制御システムへのアクセスを防止することと、をさらに含む、請求項1から10のいずれか1項に記載の方法。

【請求項12】

前記検出された通信連結に応答して、

前記ワークステーションのネットワーク設定が適正であるか否かを判定することと、

前記ワークステーションの前記ネットワーク設定が不適正である場合に、訂正されたネットワーク設定を適用するために対話型制御を提供することと、をさらに含む、請求項1から11のいずれか1項に記載の方法。

【請求項13】

前記モジュール制御システムの前記メモリから前記ワークステーションの前記メモリに、前記保存されたプロジェクトの少なくとも一部分を転送するための制御を提供することは、前記モジュール制御システムの前記メモリから、前記ワークステーションの前記メモリに保存される新しいアクティブプロジェクトに、前記保存されたプロジェクトの全体を転送する制御を提供することを含む、請求項1から11のいずれか1項に記載の方法。

【請求項14】

コンピューティングデバイスであって、

1つ以上のプロセッサと、

ユーザインターフェースと、

前記コンピューティングデバイスをモジュール制御システムに通信可能に連結するインターフェースと、

(i) モジュール制御システムの動作を制御するための調整されたデータセットを含む少なくとも1つのアクティブプロジェクトと、(ii) 前記1つ以上のプロセッサによって実行されるときに、請求項1から13のいずれか1項に記載の方法を実行するように構成されたモジュール制御システムアシスタンントを実装する命令と、を格納する非一時的メモリと、を備える、コンピューティングデバイス。

【請求項15】

システムであって、

モジュール制御システムであって、

前記モジュール制御システムの制御論理を実装するモジュールコントローラであって、特に分散型制御システムのプラットフォームで使用されるために開発された、コントローラと、

前記制御論理に従ってプロセスプラント内で物理的機能を行うように構成される装置と、

前記モジュール制御システムの動作を制御するための調整されたデータセットを含む保存されたプロジェクトを格納するメモリと、を含む、モジュール制御システムと、

スタンドアロンワークステーションであって、

1つ以上のプロセッサと、

ユーザインターフェースと、

前記スタンドアロンワークステーションを前記モジュール制御システムに通信可能に連結するインターフェースと、

10

20

30

40

50

(i) 少なくとも 1 つのアクティブプロジェクトと、(i i) 前記 1 つ以上のプロセッサによって実行されるときに、前記スタンダロンワークステーションの前記メモリ内の前記少なくとも 1 つのアクティブプロジェクトと前記モジュール制御システムの前記メモリ内の前記保存されたプロジェクトとの間のデータを同期させるように構成されたモジュール制御システムアシスタントを実装する命令と、をその上に格納する非一時的メモリと、を含むスタンダロンワークステーションと、を備える、システム。

【発明の詳細な説明】

【技術分野】

【0 0 0 1】

関連出願の相互参照

10

本出願は、2017年9月13日及び2017年10月2日にそれぞれ出願された米国仮出願第62/558,345号及び第62/567,225号に対する優先権を主張するものであり、これらの開示全体は、参照により本明細書に明示的に組み込まれる。

【0 0 0 2】

本開示は、概して、プロセス制御においてスキッドマウントシステム等のモジュール制御システムを使用することに関し、具体的には、スタンダロンモードで動作するスキッドマウントシステムの構成を効率的に管理することに関する。

【背景技術】

【0 0 0 3】

今日、様々な業界で使用されているモジュール制御システムは、水の煮沸、液体のろ過、または熱交換の制御等の特定の機能を提供することのできる完全制御システムである。モジュール制御システムは、典型的には、スキッドマウントシステム、または単に「スキッド」として実装され、そう呼ばれる理由は、システムがフレーム内に閉じ込められ、容易に運ばれるからである。スキッドは、分解されて再び組み立てられることなく一体型ユニットとして工場に供給され、典型的には、製造業者によって事前に構成され得る。スキッドは、概して、例えば、プログラマブル論理コントローラ（PLC）、弁またはボイラー等の特殊装置、及び圧力または温度センサ等のセンサを含む。

20

【0 0 0 4】

一方、分散型制御システム（DCS）は、化学、石油化学、精製、製薬、飲食品、電力、セメント、上下水道、油及びガス、パルプ及び紙、ならびに鋼鉄を含む様々なプロセス産業においても使用され、単一の場所または遠隔地で動作するバッチ、フェドバッチ、及び連続プロセスを制御するために使用される。プロセスプラントは、典型的には、アナログ、デジタル、もしくはアナログ／デジタル一体型バスを介して、または無線通信リンクもしくはネットワークを介して、1つ以上のフィールドデバイスと通信可能に連結した1つ以上のプロセスコントローラを含む。集合的には、様々なデバイスは、監視、制御、及びデータ収集機能を行って、プロセス、安全停止システム、火災及びガス検出システム、機械健全性監視システム、保守システム、判定サポート、ならびに他のシステムを制御する。

30

【0 0 0 5】

例えば、バルブ、バルブポジショナ、スイッチ、及び送信器（例えば、温度、圧力、レベル、及び流速センサ）であり得るフィールドデバイスは、プロセス環境内に配置され、概して、バルブの開放または閉鎖、プロセスパラメータの測定等の物理的機能またはプロセス制御機能を行って、プロセスプラントまたはシステム内で実行中の1つ以上のプロセスを制御する。周知の Fieldbus プロトコルに準拠するフィールドデバイス等のスマートフィールドデバイスは、制御計算、アラーム機能、及びコントローラ内で一般に実装される他の制御機能も行い得る。プロセスコントローラも、典型的には、プラント環境内に位置し、このプロセスコントローラは、フィールドデバイスによって行われるプロセス測定を示す信号及び／またはフィールドデバイスに関する他の情報を受信し、例えば、プロセス制御判定を行い、受信した情報に基づいて制御信号を生成し、HART（登録商標）、Wireless HART（登録商標）、及び FOUNDATION（登録商標）

40

50

F i e l d b u s フィールドデバイス等のフィールドデバイスで行われる制御モジュールまたはブロックと連携する、異なる制御モジュールを起動するコントローラアプリケーションを実行する。コントローラの制御モジュールは、通信線またはリンクを通じて、制御信号をフィールドデバイスに送信し、それによって、プロセスプラントまたはシステムの少なくとも一部分の動作を制御する。

【 0 0 0 6 】

フィールドデバイス及びコントローラからの情報は、制御室もしくはより厳しいプラント環境から離れた他の場所に典型的に配置される、オペレータワークステーション、パソコンコンピュータもしくはコンピューティングデバイス、データヒストリアン、レポートジェネレータ、集中データベース、または他の集中管理コンピューティングデバイス等の 1 つ以上の他のハードウェアデバイスに対して、通常、データハイウェイを通じて利用可能になる。これらのハードウェアデバイスの各々は、典型的には、プロセスプラントにわたって、またはプロセスプラントの一部分にわたって集中化される。これらのハードウェアデバイスは、例えば、オペレータが、プロセス制御ルーチンの設定の変更、コントローラもしくはフィールドデバイス内の制御モジュールの動作の修正、プロセスの現在の状態の閲覧、フィールドデバイス及びコントローラにより生成されたアラームの閲覧、担当者の訓練もしくはプロセス制御ソフトウェアの試験を目的としたプロセスの動作のシミュレーション、構成データベースの保守及び更新等の、プロセスの制御及び / またはプロセスプラントの動作に関する機能を行うことを可能にし得るアプリケーションを起動する。ハードウェアデバイス、コントローラ、及びフィールドデバイスが利用するデータハイウェイは、有線通信バス、無線通信バス、または有線及び無線通信バスの組み合わせを含んでもよい。

10

【 0 0 0 7 】

例として、Emerson Process Management によって販売されている、DeltaV (商標) 制御システムは、プロセスプラント内の多様な場所に位置する異なるデバイス内に格納され、それらの異なるデバイスによって実行される多数のアプリケーションを含む。1 つ以上のワークステーションまたはコンピューティングデバイスに存在する構成アプリケーションは、ユーザによる、プロセス制御モジュールの作成または変更、及びデータハイウェイを介した、これらのプロセス制御モジュールの、専用分散型コントローラへのダウロードを可能にする。典型的には、これらの制御モジュールは、通信可能に相互接続された機能ブロックで構成され、これらの機能ブロックは、それに対する入力に基づき制御スキーム内で機能を行い、出力を制御スキーム内の他の機能ブロックに提供するオブジェクト指向プログラミングプロトコル内のオブジェクトである。また、構成アプリケーションは、データをオペレータに対して表示するため、かつプロセス制御ルーチン内の設定点等の設定のオペレータによる変更を可能にするために閲覧アプリケーションが使用するオペレータインターフェースを、構成エンジニアが作成または変更することを可能にし得る。各専用コントローラ、及び一部の場合においては、1 つ以上のフィールドデバイスは、実際のプロセス制御機能を実装するために、それらに割り当てられてダウロードされた制御モジュールを実行するそれぞれのコントローラアプリケーションを格納及び実行する。閲覧アプリケーションは、1 つ以上のオペレータワークステーション（またはオペレータワークステーション及びデータハイウェイと通信接続された 1 つ以上のリモートコンピューティングデバイス）上で実行されてもよく、この閲覧アプリケーションは、コントローラアプリケーションからデータハイウェイを介してデータを受信し、ユーザインターフェースを使用してこのデータをプロセス制御システム設計者、オペレータ、またはユーザに表示して、オペレータのビュー、エンジニアのビュー、技術者のビュー等の多数の異なるビューのうちのいずれかを提供してもよい。データヒストリアンアプリケーションは、典型的には、データハイウェイをわたって提供されたデータの一部または全てを収集及び格納するデータヒストリアンデバイスに格納され、それによって実行される一方で、構成データベースアプリケーションは、現在のプロセス制御ルーチン構成及びそれに関連付けられたデータを格納するために、データハイウェイに取り付けら

20

30

40

50

れたさらに離れたコンピュータで起動し得る。あるいは、構成データベースは、構成アプリケーションと同じワークステーションに位置してもよい。

【0008】

プロセス制御及び産業オートメーションシステムで動作するデバイスは、有線または無線で相互接続され、FOUNDATION Fieldbus、HART（登録商標）、またはProfibus等の産業通信プロトコルを使用して通信し得る。さらに、Modbus等のプロトコルは、PLCを相互接続するために開発されている。またさらに、標準産業オートメーションプロトコルに加えて、プロセス制御システム内のノードを相互接続するための独自のプロトコルが存在する。Deltavはそのようなプロトコルの一例である。一般に、これらのプロトコルは、測定、アラート、及び状態報告、プロセス変数またはオートメーションパラメータに影響を及ぼすコマンド、デバイスをアクティブ化または非アクティブ化するコマンド等を搬送するためのフォーマットを指定する。典型的な産業通信プロトコルはまた、プロトコルの構文に従って特定のデバイスの製造業者によって定義された定義済みコマンドまたは複数の定義済みコマンドを介して、デバイス構成をサポートする。

【0009】

一般に、分散型制御システムは非常に複雑で機能豊富である。ストレージ及びデブロイメントは、典型的には、DCSで分離されており、多数のデータベースは構成データを格納し、共有アクセスポイントがない場合がある。DCSとは対照的に、PLCは、典型的には、簡易的なソフトウェア、限られた機能、及び非常に簡易的なプロジェクトストレージを有し、ランタイムデブロイメントと緊密に連結されることが多い。さらに、PLCデバイスは、典型的には、ランタイム及び構成データをデバイス自体に格納する。

【0010】

モジュール制御システムは、従来のPLCよりも堅牢であるが、DCSのように多くの機能は含まれておらず、モジュール制御システムの管理に利用可能であるデータベース及びソフトウェアアプリケーションがそれほど多くない。モジュール制御システムがDCSのノードとしてではなくスタンドアロンモードで動作する状況では、エンジニアは、複雑なソフトウェアスイートまたは多数のデータベースの複雑さを伴わずに、DCSで利用可能な機能に類似した機能にアクセスすることを望み得る。

【発明の概要】

【0011】

スキッドマウントシステム等の本開示のモジュール制御システムは、スタンドアロンモードで、場合によっては分散型制御システム内のノードとして動作するように構成される。モジュール制御システムは、他のコントローラとは独立してモジュール制御システムの制御論理を実行することができるモジュールコントローラを含む。モジュール制御システムがスタンドアロンモードで動作するとき、オペレータは、例えば、ラップトップコンピュータまたはタブレットコンピュータ等のコンピューティングデバイス上で実装されるワークステーションを使用してモジュール制御システムにアクセスすることができる。以下スキッドアシスタントまたはモジュールアシスタントと呼ばれるアシスタントソフトウェアアプリケーションによって、モジュール制御システムの構成及び管理が容易になる。様々な実装及び/またはシナリオでは、モジュールアシスタントによって、オペレータがモジュール制御システム構成を更新し、構成の特定のバージョンとの同期を確認し、修正またはコピーのための構成をリトリーブし、構成にエラーが含まれているか否かを判定することが可能になる。いくつかの実装では、アシスタントソフトウェアアプリケーションは、モジュール制御システムに新しい構成を提供するタイミングを制御して、モジュールコントローラがモジュール制御システムのメモリにアクセスする頻度を減少させるが、いくつかの実装では比較的遅い。他の実装では、アシスタントソフトウェアアプリケーションは、即座に、例えば、実質的にリアルタイムで、構成データをモジュール制御システムに提供する。

【0012】

10

20

30

40

50

スキッドアシスタントは、ワークステーションと、モジュール制御システム、またはより一般的には、制御システムのノードとの間の複数のレベルの通信連結を提供することができる。例示的に実装において、スキッドアシスタントは、物理層における接続が検出されたときに、適正な連結が接続段階におけるものであるかどうかを確認し、その後、再度、認証段階において連結を確認して、オペレータがモジュール制御システムと対話することを適切に承認されているかどうかを判定し、次いで、再度、プロジェクト検証段階において、アクティブプロジェクトがモジュール制御システムに適正であるかどうかを判定する。このように、スキッドアシスタントは、モジュール制御システムのリムーバブルメモリカード、または別のタイプの好適なメモリへのアクティブプロジェクトの誤ったバックアップを防止する。さらに、スキッドアシスタントは、分散型制御システムのプロトコルを使用して、分散型制御システムとモジュール制御システムとの間の誤った通信を防止し、かつ、そうでなければ、分散型制御システムによって使用される情報を交換する。例えば、スキッドアシスタントは、モジュール制御システムのノードへの間違った構成のダウンロード、またはワークステーションを介したモジュール制御システムのノードに対する間違ったランタイム情報の表示を防止することができる。

【 0 0 1 3 】

これらの技術の一例示的実施形態は、モジュール制御システムの構成を効率的に管理するための方法である。この方法は、1つ以上のプロセッサによって、ワークステーションとモジュール制御システムとの間の通信連結を検出することと、検出された通信連結に応答して、ワークステーションのメモリに格納されたアクティブプロジェクトがモジュール制御システムのメモリに格納された保存されたプロジェクトと一致するか否かを判定することであって、アクティブプロジェクト及び保存されたプロジェクトの各々が、モジュール制御システムの動作を制御するためのそれぞれの調整されたデータセットを含む、判定することと、アクティブプロジェクトと保存されたプロジェクトとが一致しない場合に、アクティブプロジェクトがモジュール制御システムに適正であるか否かを判定し、アクティブプロジェクトがモジュール制御システムに不適正である場合に、モジュール制御システムのメモリからワークステーションのメモリ内のアクティブプロジェクトに、保存されたプロジェクトの少なくとも一部分を転送するための制御を、対話型ユーザインターフェースを介して自動的に提供することと、を含む。

【 0 0 1 4 】

これらの技術の別の例示的実施形態は、1つ以上のプロセッサと、ユーザインターフェースと、コンピューティングデバイスをモジュール制御システムに通信可能に連結するインターフェースと、非一時的メモリとを備えるコンピューティングデバイスである。メモリは、モジュール制御システムの動作を制御するための調整されたデータセットを含む少なくとも1つのアクティブプロジェクトを格納する。メモリはまた、1つ以上のプロセッサによって実行されるときに、(i) コンピューティングデバイスとモジュール制御システムとの間の通信連結を検出し、(i i) アクティブプロジェクトが、モジュール制御システムのメモリに格納された保存されたプロジェクトに一致するか否かを判定し、(i i i) アクティブプロジェクトと保存されたプロジェクトとが一致しない場合に、アクティブプロジェクトがモジュール制御システムに適正であるか否かを判定し、アクティブプロジェクトがモジュール制御システムに不適正である場合に、モジュール制御システムのメモリからコンピューティングデバイスのメモリ内のアクティブプロジェクトに保存されたプロジェクトの少なくとも一部分を転送するための制御を、ユーザインターフェースを介して自動的に提供する、ように構成されたモジュール制御システムアシスタントを実装する命令も格納する。

【 0 0 1 5 】

これらの技術のさらに別の例示的実施形態は、モジュール制御システム及びスタンドアロンワークステーションを備えるシステムである。モジュール制御システムは、モジュール制御システムの制御論理を実装するモジュールコントローラであって、分散型制御システムのプラットフォーム上にネイティブに構築される、コントローラと、制御論

10

20

30

40

50

理に従ってプロセスプラント内の物理的機能を行うように構成された装置と、モジュール制御システムの動作を制御するための調整されたデータセットを含む保存されたプロジェクトを格納するメモリと、を含む。スタンドアロンワークステーションは、1つ以上のプロセッサと、ユーザインターフェースと、スタンドアロンワークステーションをモジュール制御システムに通信可能に連結するインターフェースと、(i)少なくとも1つのアクティブプロジェクト及び(i i)1つ以上のプロセッサによって実行されるときに、スタンドアロンワークステーションのメモリ内の少なくとも1つのアクティブプロジェクトとモジュール制御システムのメモリ内の保存されたプロジェクトとの間でデータを同期させるように構成されたモジュール制御システムアシスタントを実装する命令を格納する非一時的メモリと、を含む。

10

【図面の簡単な説明】

【0 0 1 6】

【図1】スタンドアロンモードで動作するように構成されたモジュール制御システムと、モジュール制御システムの構成を管理するように構成されたアシスタントアプリケーションを実行するエンジニアリングステーションとして動作するコンピュータとのブロック図である。

【図2】多数のモジュール制御システムのブロック図であり、図1のアシスタントアプリケーションの単一のインスタンスがその構成を管理し得る。

【図3】エンジニアリングワークステーションとモジュール制御システムのメモリとの間の、図1のアシスタントアプリケーションを介した相互作用を概略的に示す。

20

【図4】いくつかのシナリオにおいてモジュール制御システムを統合し得る分散型プロセス制御システムのブロック図である。

【図5 A】図1のアシスタントアプリケーションが、接続、構成管理、切断、及び管理機能にそれぞれアクセスを提供するようにオペレータに提示し得る例示的対話型スクリーンを示す。

【図5 B】図1のアシスタントアプリケーションが、接続、構成管理、切断、及び管理機能にそれぞれアクセスを提供するようにオペレータに提示し得る例示的対話型スクリーンを示す。

【図5 C】図1のアシスタントアプリケーションが、接続、構成管理、切断、及び管理機能にそれぞれアクセスを提供するようにオペレータに提示し得る例示的対話型スクリーンを示す。

30

【図5 D】図1のアシスタントアプリケーションが、接続、構成管理、切断、及び管理機能にそれぞれアクセスを提供するようにオペレータに提示し得る例示的対話型スクリーンを示す。

【図6 A】オペレータがコミッショニングされたモジュール制御システムに接続するときに、図1のアシスタントアプリケーションがオペレータに提示し得る対話型スクリーンの例示的シーケンスを示す。

【図6 B】オペレータがコミッショニングされたモジュール制御システムに接続するときに、図1のアシスタントアプリケーションがオペレータに提示し得る対話型スクリーンの例示的シーケンスを示す。

【図6 C】オペレータがコミッショニングされたモジュール制御システムに接続するときに、図1のアシスタントアプリケーションがオペレータに提示し得る対話型スクリーンの例示的シーケンスを示す。

40

【図6 D】オペレータがコミッショニングされたモジュール制御システムに接続するときに、図1のアシスタントアプリケーションがオペレータに提示し得る対話型スクリーンの例示的シーケンスを示す。

【図6 E】オペレータがコミッショニングされたモジュール制御システムに接続するときに、図1のアシスタントアプリケーションがオペレータに提示し得る対話型スクリーンの例示的シーケンスを示す。

【図6 F】オペレータがコミッショニングされたモジュール制御システムに接続するときに、

50

図1のアシスタントアプリケーションがオペレータに提示し得る対話型スクリーンの例示的シーケンスを示す。

【図7 A】オペレータがバックアップストレージから新しいアクティブプロジェクトに構成をリトリーブするときに、図1のアシスタントアプリケーションがオペレータに提示し得る対話型スクリーンの例示的シーケンスを示す。

【図7 B】オペレータがバックアップストレージから新しいアクティブプロジェクトに構成をリトリーブするときに、図1のアシスタントアプリケーションがオペレータに提示し得る対話型スクリーンの例示的シーケンスを示す。

【図8 A】オペレータがモジュール制御システムをデコミッショナ化せずにコミッショナ化されたモジュール制御システム上でパスワードを変更することを望む場合に、図1のアシスタントアプリケーションがオペレータに提示し得る対話型スクリーンの例示的シーケンスを示す。
10

【図8 B】オペレータがモジュール制御システムをデコミッショナ化せずにコミッショナ化されたモジュール制御システム上でパスワードを変更することを望む場合に、図1のアシスタントアプリケーションがオペレータに提示し得る対話型スクリーンの例示的シーケンスを示す。

【図9 A】オペレータがコミッショナ化されたモジュール制御システムに接続するときに、図1のアシスタントアプリケーションがオペレータに提示し得る対話型スクリーンの例示的シーケンスを示す。

【図9 B】オペレータがコミッショナ化されたモジュール制御システムに接続するときに、図1のアシスタントアプリケーションがオペレータに提示し得る対話型スクリーンの例示的シーケンスを示す。
20

【図9 C】オペレータがコミッショナ化されたモジュール制御システムに接続するときに、図1のアシスタントアプリケーションがオペレータに提示し得る対話型スクリーンの例示的シーケンスを示す。

【図9 D】オペレータがコミッショナ化されたモジュール制御システムに接続するときに、図1のアシスタントアプリケーションがオペレータに提示し得る対話型スクリーンの例示的シーケンスを示す。
30

【図9 E】オペレータがコミッショナ化されたモジュール制御システムに接続するときに、図1のアシスタントアプリケーションがオペレータに提示し得る対話型スクリーンの例示的シーケンスを示す。

【図10 A】オペレータがバックアップストレージから構成をリトリーブするときに、図1のアシスタントアプリケーションがオペレータに提示し得る対話型スクリーンの例示的シーケンスを示す。

【図10 B】オペレータがバックアップストレージから構成をリトリーブするときに、図1のアシスタントアプリケーションがオペレータに提示し得る対話型スクリーンの例示的シーケンスを示す。

【図11】アクティブプロジェクトが取り付けられたコントローラに不適正である場合に、図1のアシスタントアプリケーションがオペレータに提示し得る例示的対話型スクリーンを示す。
40

【図12 A】オペレータがアクティブプロジェクトをバックアップするときに、図1のアシスタントアプリケーションがオペレータに提示し得る対話型スクリーンの例示的シーケンスを示す。

【図12 B】オペレータがアクティブプロジェクトをバックアップするときに、図1のアシスタントアプリケーションがオペレータに提示し得る対話型スクリーンの例示的シーケンスを示す。

【図12 C】オペレータがアクティブプロジェクトをバックアップするときに、図1のアシスタントアプリケーションがオペレータに提示し得る対話型スクリーンの例示的シーケンスを示す。

【図12 D】オペレータがアクティブプロジェクトをバックアップするときに、図1のア
50

クリーンを示す。

【図20】オペレータがドライブから新しいプロジェクトを作成するときに、図1のアシスタントアプリケーションがオペレータに提示し得る例示的対話型スクリーンを示す。

【図21】図1のシステムで実装可能なモジュール制御システムの構成を効率的に管理するための例示的方法のフロー図である。

【発明を実施するための形態】

【0017】

図1は、Ethernet接続14を介してコンピュータ12に連結された、スキッドマウントシステムで動作し得るモジュールコントローラ10の一例を示す。モジュールコントローラ10は、例えば、汎用（例えば、CPU、GPU）または専用（例えば、ASIC、FPGA）ハードウェアを含み得る1つ以上のプロセッサ20を含む。1つ以上のプロセッサ20は、メモリモジュール22、ユーザインターフェース24、ローカル構成データベース26、ローカルエリアネットワークインターフェース28、及び産業オートメーションインターフェース29に連結される。10

【0018】

メモリ22は、オペレーティングシステム30を格納する。いくつかの実装では、オペレーティングシステム30は、モジュールコントローラ10が多数のサービス及びアプリケーションを同時に実行することを可能にするQNX等のリアルタイムオペレーティングシステムである。例えば、構成モジュール32は、制御論理モジュール34と同時に実行し得る。構成モジュール32は、モジュール制御システム11をオフラインにすることを必要とせずに、コンピュータ12から構成データを受信し、受信した構成データをローカル構成データベース26等に格納してローカル構成データベース26からデータをリトリーブし、リトリーブしたデータをコンピュータ12等に提供し得る。制御論理モジュール34は、フィールドデバイス40A～Cにコマンドを送信し、かつフィールドデバイス40A～Cからデータを受信することによって制御論理を実行し続けてもよい。モジュールコントローラ10は、フィールドデバイス40A～Cと共に、モジュール制御システム11を実装する。20

【0019】

メモリカード23は、モジュールコントローラ10に取り外し可能に連結され得る。メモリ23は、例えば、セキュアデジタル（SD）カードであり得る。いくつかの実装では、メモリ23は、アクティブプロジェクトのバックアップバージョン（以下に論じるように、様々な構成データを含む）を格納し得る。より一般的には、モジュールコントローラ10は、任意の適切な種類のリムーバブルメモリを含み得る。30

【0020】

ユーザインターフェース24は、オペレータがモジュールコントローラ10及び/またはフィールドデバイス40A～Cの状態を閲覧することを可能にするパネルであってもよい。パネルは、モジュールコントローラ10のハウ징内に配置されたタッチスクリーンであってもよい。

【0021】

構成データベース26は、メモリモジュール22に、または別個のモジュールに実装されてもよい。いくつかの実装では、構成データベース26が実装されるメモリモジュールはフラッシュメモリである。構成データベース26は、比較的小さなテーブルのセットとして、または例えばSQLクエリをサポートするより堅牢なリレーションナルデータベースとして実装され得る。40

【0022】

LANインターフェースは、有線Ethernetインターフェースであってもよい。いくつかの実装では、LANインターフェースは、セキュリティ強化のために無線通信をサポートしない。産業オートメーションインターフェース28は、Profibus、Modbus、HART等のプロトコルをサポートし得る。

【0023】

50

いくつかのシナリオでは、図 4 を参照して以下に論じるように、システム 11 等の本開示のモジュール制御システムは、プロセス制御システムによって使用されるプラットフォームとは異なるプラットフォーム上に構築されたプロセス制御システムに統合され得る。モジュール制御システムのプラットフォームは、独占的である場合がある。この場合、モジュールコントローラは、オープンプラットフォーム通信（O P C）通信プロトコルまたは M o d b u s 等の標準プロトコルを使用して、プロセス制御システムのノードと通信し得る。

【 0 0 2 4 】

コンピュータ 12 は、ラップトップコンピュータ、タブレットコンピュータ、ハンドヘルド専用電子デバイス等であり得る。コンピュータ 12 は、1 つ以上のプロセッサ 42、ユーザインターフェース 44、及びメモリ 46 を含み得る。場合によっては、コンピュータ 12 は、モジュール制御システム 11 にアクセスするために、アクセスキーや他のセキュリティ情報を格納するハードウェアドングル（混乱を回避するために図示せず）に接続される。10

【 0 0 2 5 】

メモリ 46 は、本明細書においてスキッドアシスタント、PKアシスタント、またはモジュール制御システム（簡略化のために「モジュールアシスタント」と呼ばれ得るアシスタントソフトウェアアプリケーション 50 を格納することができ、1 つ以上のプロセッサ 42 上で実行可能な命令のセットとして実装される。メモリ 46 はまた、多数のモジュール制御システム構成 52 も格納し得る。モジュールアシスタント 50 は、ユーザが様々なバージョン及び様々なモジュール制御システムのモジュール制御システム構成に名前を付けることを可能にし得る。より一般的には、本開示のモジュールアシスタント 50 は、任意の適切なモジュール制御システムで動作し得る。20

【 0 0 2 6 】

図 2 を参照すると、モジュールアシスタント 50 等のアシスタントソフトウェアアプリケーションは、多数のモジュール制御システム 62A～C の構成を管理するためにコンピュータ 60 において動作し得る。コンピュータ 60 は、上述のコンピュータ 12 と同様であり得る。異なる状況では、モジュール制御システム 62A～C の構成は、互いに同じまたは異なっていてもよい。例えば、モジュールアシスタント 62A によって、ユーザがモジュール制御システム 62A から構成 A を効率的にリトリープし、リトリープされた構成 A をモジュール制御システム 62C に適用することが可能になり得る。モジュール制御システム 62C は、異なる構成 B を有することができます。30

【 0 0 2 7 】

オペレータは、同じコンピュータ 60 を使用して、システム 62A～C の構成を管理し、構成の差異（例えば、構成 A 対構成 B）をチェックし、構成のエラーをチェックし（例えば、構成 A 対、コンピュータ 62 のメモリに格納されたベンチマーク構成）、構成をコピーし、構成を削除し、システム 62A～C のそれぞれのモジュールコントローラへの構成の適用をアクティブ化すること等ができる。

【 0 0 2 8 】

集中位置から個々のコントローラに構成がダウンロードされる（または、ハンドヘルドが使用される場合、コントローラからワークステーションまで伝搬され、集中データベースに格納される）D C S とは異なり、スタンドアロンモードで動作するモジュール制御システムは、他のシステムと自動的に同期されない。これにより、D C S には一般的に存在しない問題が発生する可能性があり、アシスタントソフトウェアアプリケーションは、構成データのバージョンを管理し、かつコピー、編集、エラーチェック、及び他の機能を提供することにより、エラーの確率を低下させ得る。40

【 0 0 2 9 】

図 3 は、コンピュータ 12 または 60 等のエンジニアリングワークステーションに格納され得るアクティブプロジェクト 70 と、11、62A、62B、または 62C 等のモジュール制御システムのメモリに格納され得るアクティブプロジェクトバックアップ 72 と

10

20

30

40

50

の間の、モジュールアシスタント 50 を介した相互作用を概略的に示す。エンジニアリングワークステーションは、スタンドアロンマシンとして動作し、モジュールアシスタント 50 を介してモジュール制御システムのコントローラとプロジェクトデータを交換するように構成される。プロジェクトは、分散型制御システム（例えば、Deltav）の構成ファイル、分散型制御システムの動作情報（例えば、リアルタイムデータ、アラーム、セキュリティデータ）、デバイス設定、資産管理データ等の調整されたデータセットであり得る。プロジェクトは、モジュールコントローラのためのプレースホルダを含み得る。プレースホルダは、プロジェクトにおいてコミッショニングまたはデコミッショニングされ得る。

【0030】

モジュールアシスタント 50 によって、プロジェクト全体またはプロジェクトの選択された部分の交換が容易になり得る。この場合、セーブ（またはバックアップ）動作は、概念的にダウンロード動作から分離される。言い換えれば、モジュールアシスタント 50 は、この構成をモジュール制御システムにダウンロードせずに、上述のメモリカード 23 等のモジュール制御システムのメモリに新しい構成を保存し得る。これにより、任意のフェーズ及び任意の手順のプロジェクトのサポートが可能になる。さらに、このアプローチにより、モジュール制御システムがいつでもモジュールにアクセスするかをユーザがより良く制御することが可能になる。

10

【0031】

図 3 に示すように、モジュールアシスタント 50 は、モジュールコントローラのリムバブルメモリにアクティブプロジェクト 70 を転送し、アクティブプロジェクトバックアップ 72 として保存するバックアップ動作 80 をサポートし得る。モジュールアシスタント 50 はまた、アクティブプロジェクトバックアップ 72 をリトリーブするためにリトリーブ動作 82 をサポートし得る。次いで、モジュールアシスタント 50 は、プロジェクト 72 とプロジェクト 70 とを比較し、プロジェクト 72 とプロジェクト 70 とをマージし、またはプロジェクト 70 にプロジェクト 72 をコピーして、例えば、対応するデータを他のモジュール制御システムに転送し得る。

20

【0032】

一般に、モジュールアシスタント 50 は、モジュールコントローラのコミッショニング状態とアクティブプロジェクトのコミッショニング状態との様々な組み合わせをサポートし、場合によっては、エンジニアリングワークステーションのオペレータに特定のアクションを自動的に提案し得る。モジュールアシスタント 50 は、ワークステーションがモジュールコントローラに接続されていると判定すると、これらの可能なアクションを識別し得る。例えば、モジュールアシスタント 50 を実行するスタンドアロンエンジニアリングワークステーションがモジュールコントローラに接続する場合、及びエンジニアリングワークステーションのネットワーク設定が適正である場合、モジュールアシスタント 50 は、アクティブプロジェクト 70 のモジュールコントローラのプレースホルダがコミッショニングされていない場合、及び物理的モジュールコントローラがコミッショニングされていない場合に、コミッショニングステップを行う必要があることをオペレータに通知し得る。アクティブプロジェクト 70 のプレースホルダがコミッショニングされているが、物理的モジュールコントローラがコミッショニングされていない場合、モジュールアシスタント 50 は、コミッショニングステップを行うようにユーザにプロンプトし得る。この場合、ユーザはまずプレースホルダをデコミッショニングしてから、プレースホルダを再コミッショニングし得る。

30

【0033】

アクティブプロジェクト 70 のプレースホルダとモジュールコントローラとの両方がコミッショニングされると、モジュールアシスタント 50 は、コミッショニングされた構成データ（すなわち、プロジェクト）を比較し得る。一致しなければ、モジュールアシスタント 50 は、ユーザにパスワードをプロンプトし、アクティブプロジェクト 70 を取り付けられたモジュールコントローラ内のプロジェクトと整合させるように実行可能な異なるアクションを通じてオペレータを案内し得る。具体的には、モジュールアシスタント 50 は、取り付けられたモジュールコントローラに適正であるプロジェクトに切り替えることを提案し

40

50

得る（上述のように、コンピュータ 1 2 または 6 0 は、異なるモジュールコントローラと共に使用可能である多数のプロジェクトを格納し得る）。代替として、モジュールアシスタント 5 0 は、構成データをモジュールコントローラから新しいアクティブプロジェクト内にリトリープすることを提案し得る。さらに別の代替として、モジュールアシスタント 5 0 は、オペレータが異なるモジュールコントローラに接続できることを提案し得る。そうではなく、一致する場合、モジュールアシスタント 5 0 は、ユーザにパスワードをプロンプトし、アクティブプロジェクト 7 0 と取り付けられたモジュールコントローラ内のプロジェクトとが整合していることをユーザに通知し得るだけである。

【 0 0 3 4 】

アクティブプロジェクト 7 0 のプレースホルダがコミッショニングされていない場合、及びモジュールコントローラがコミッショニングされている場合、モジュールアシスタント 5 0 はユーザにパスワードをプロンプトし、アクティブプロジェクト 7 0 をモジュールコントローラの構成に整合させ、オペレータがモジュールコントローラに適正であるプロジェクトに切り替えることを提案し、またはモジュールコントローラの構成を新しいアクティブプロジェクトにリトリープし、もしくはモジュールコントローラの構成を現在のアクティブプロジェクト 7 0 にリトリープし、またはワークステーションを異なるモジュールコントローラに接続し得る。

10

【 0 0 3 5 】

モジュールシステム 5 0 がスタンドアロンモードで動作するモジュール制御システムと相互作用する例示的シナリオの前に、モジュール制御システムの分散型制御システムへの例示的統合について、図 4 を参照して考察される。上述のように、実装のうちの少なくともいくつかのモジュール制御システムは、スタンドアロンノードとして動作し、または分散型制御システムにシームレスに統合されてもよく、分散型ノードとして動作し得る。

20

【 0 0 3 6 】

図 4 は、分散型制御システム 1 2 2 を実装する例示的プロセスプラント 1 0 0 を示す。典型的であるように、分散型プロセス制御システム 1 2 2 は、例えば、Fieldbus インターフェース、Profibus インターフェース、HART インターフェース、標準 4 ~ 20 mA インターフェース等であり得る入力 / 出力 (I/O) デバイスまたはカード 1 4 8 を介して、1 つ以上のフィールドデバイスまたはスマートデバイス 1 4 4 及び 1 4 6 に各々が接続された 1 つ以上のコントローラ 1 4 0 を有する。コントローラ 1 4 0 はまた、データハイウェイ 1 5 4 を介して 1 つ以上のホストまたはオペレータワークステーション 1 5 0 、 1 5 2 に連結され、データハイウェイ 1 5 4 は、例えば、Ethernet リンクまたはローカルエリアネットワーク (LAN) リンクに適切な別のリンクであってもよい。プロセスデータベース 1 5 8 は、データハイウェイ 1 5 4 に接続されてもよく、パラメータ、状態、ならびにプラント 1 0 0 内のコントローラ及びフィールドデバイスに関連付けられた他のデータを収集及び格納するように動作する。プロセスプラント 1 0 0 の動作中、プロセスデータベース 1 5 8 は、コントローラ 1 4 0 から、及び間接的に、データハイウェイ 1 5 4 を介してデバイス 1 4 4 ~ 1 4 6 からプロセスデータを受信し得る。

30

【 0 0 3 7 】

構成データベース 1 6 0 は、コントローラ 1 4 0 ならびにフィールドデバイス 1 4 4 及び 1 4 6 内にダウンロードされてそこに格納される、プラント 1 0 0 内のプロセス制御システム 1 2 2 の現在の構成を格納する。構成データベース 6 0 は、プロセス制御システム 1 2 2 のうちの 1 つまたはいくつかの制御ストラテジ、デバイス 1 4 4 及び 1 4 6 の構成パラメータ、プロセス制御機能へのデバイス 1 4 4 及び 1 4 6 の割り当て、ならびにプロセスプラント 1 0 0 に関する他の構成データを定義するプロセス制御機能を格納する。構成データベース 1 6 0 は、さらに、プロセスプラント 1 0 0 の要素の様々なグラフィック表示を提供するために、グラフィックオブジェクトをさらに格納し得る。格納されたグラフィックオブジェクトのうちのいくつかは、プロセス制御機能に対応する場合があり（例えば、特定の PID ループに開発されたプロセスグラフィック）、他のグラフィックオ

40

50

プロジェクトは、デバイス固有であり得る（例えば、圧力センサに対応するグラフィック）。

【0038】

プロセスプラント100はまた、混乱を回避するために図4には示されていないデータハイウェイ154に連結された他のデータベースを含み得る。例えば、データヒストリアンは、イベント、アラーム、コメント、及びオペレータが取った一連のアクションを格納し得る。イベント、アラーム、及びコメントは、個々のデバイス（例えば、バルブ、送信器）、通信リンク（例えば、有線Fieldbusセグメント、WirelessHART通信リンク）、またはプロセス制御機能（例えば、所望の温度設定点を維持するためのPID制御ループ）に関連し得る。さらに、知識レポジトリは、参照、オペレータログブック入力、ヘルプトピック、またはオペレータ及び保守技術者がプロセスプラント100の監督時に有用とみなしえるこれら及び他の文書へのリンクを格納し得る。またさらに、ユーザデータベースは、オペレータまたは保守技術者等のユーザに関する情報を格納し得る。

【0039】

これらのデータベースの各々は、任意の所望の種類のメモリ及びデータを格納するための任意の所望のまたは既知のソフトウェア、ハードウェア、またはファームウェアを有する、任意の所望の種類のデータストレージまたは収集ユニットであり得る。当然ながら、データベースは、別個の物理的デバイス内に存在する必要はない。したがって、いくつかの実施形態では、これらのデータベースのいくつかは、共有データプロセッサ上に実装される。一般に、より多いまたはより少ないデータベースを利用して、上述のデータベースによって集合的に格納及び管理されたデータを格納することが可能である。

【0040】

コントローラ140、I/Oカード148、ならびにフィールドデバイス144及び146は、典型的には、時に厳しいプラント環境全体に分散される一方で、オペレータワークステーション150及び152ならびにデータベース158、160等は、通常、制御室またはコントローラ、保守、及び他の様々なプラント担当者が容易に評価できる他により厳しくない環境に位置する。しかしながら、場合によっては、ハンドヘルドデバイスを使用してこれらの機能を実装してもよく、これらのハンドヘルドデバイスは、典型的には、プラント内の様々な場所に持ち運ばれる。

【0041】

例としてEmerson Process Managementによって販売されているDeltaV（商標）コントローラであり得るコントローラ140の各々は、任意の数の、独立して実行される異なる制御モジュールまたはブロック170を使用して、制御ストラテジを実装するコントローラアプリケーションを格納及び実行する。制御モジュール170の各々は、一般に機能ブロックと呼ばれるもので構成され得、各機能ブロックは、制御ルーチン全体の一部またはサブルーチンであり、他の機能ブロックと共に動作して（リンクと呼ばれる通信を介して）、プロセスプラント100内でプロセス制御ループを実装する。周知のように、オブジェクト指向プログラミングプロトコル中のオブジェクトであり得る機能ブロックは、典型的には、送信器、センサ、もしくは他のプロセスパラメータ測定デバイスに関連付けられるもの等の入力機能、PID、ファジー論理等の制御を行う制御ルーチンに関連付けられるもの等の制御機能、またはバルブ等の何らかのデバイスの動作を制御して、プロセスプラント10内で何らかの物理的機能を行う出力機能のうちの1つを行う。当然ながら、ハイブリッド及び他の種類の複雑な機能ブロック、例えばモデル予測コントローラ（MPC）、オプティマイザ等が存在する。Fieldbusプロトコル及びDeltaVシステムプロトコルが、オブジェクト指向プログラミングプロトコル中で設計かつ実装された制御モジュール及び機能ブロックを使用する一方、制御モジュールは、例えば、逐次機能ブロック、ラダー論理等を含む任意の所望の制御プログラミングスキームを使用して設計され得、機能ブロックまたは任意の他の特定のプログラミング技術を使用した設計及び実装に限定されない。コントローラ40の各々はまた、アプリケーションのAMS（登録商標）シートをサポートしてもよく、予測知能を使用して、機械設備、電気システム、プロセス装置、計器、フィールド及びスマートフィールドデ

バイス 144、146、ならびにバルブを含むプロダクションアセットの可用性及び性能を改善してもよい。

【0042】

図 1 に示されるプラント 100 では、コントローラ 112 に接続されたフィールドデバイス 144、146 は、標準 4 ~ 20 mA デバイスであってもよく、プロセッサ及びメモリを含む、HART、Profibus、もしくは FOUNDATION (商標) Fieldbus フィールドデバイス等のスマートフィールドデバイスであってもよく、または任意の他の所望の種類のデバイスであってもよい。Fieldbus フィールドデバイス (図 1 の参照番号 146 で表示される) 等のこれらのデバイスのいくつかは、コントローラ 140 に実装された制御ストラテジに関する付けられたモジュールまたは機能ブロック等のサブモジュールを格納及び実行してもよい。図 1 に Fieldbus フィールドデバイス 146 の 2 つの異なるデバイスに配置されて示される機能ブロック 172 は、周知のように、コントローラ 140 内の制御モジュール 170 の実行と共に実行されて、プロセス制御を実装してもよい。当然ながら、フィールドデバイス 144 及び 146 は、センサ、バルブ、送信器、ポジショナ等の任意の種類のデバイスであってもよく、I/O デバイス 148 は、任意の所望の通信またはコントローラプロトコル、例えば HART、Fieldbus、Profibus 等に準拠する、任意の種類の I/O デバイスであってもよい。

10

【0043】

ワークステーション 150 及び 152 は、メモリ 180 に格納された命令を実行する 1 つ以上のプロセッサ 182 を含み得る。この命令は、プロセスプラント 10 の動作中に様々な表示を提供する閲覧アプリケーション 184 を部分的に実装し、オペレータがプロセスプラント 100 内、またはより大きなプラントでは一般的であるように、プロセスプラント 100 のセクション内の様々な動作を閲覧及び制御することを可能にし得る。閲覧アプリケーション 184 は、制御診断アプリケーション、チューニングアプリケーション、レポート生成アプリケーション、または制御機能を行う際にオペレータを支援するために使用され得る任意の他の制御サポートアプリケーション等のサポートアプリケーションを含み得るか、またはそれと連携し得る。さらに、閲覧アプリケーション 184 によって、プラント 100 の保守ニーズを監督する保守技術者が、例えば、様々なデバイス 140、144、及び 146 の動作状態または作動状態を閲覧することが可能になる。閲覧アプリケーションはまた、保守診断アプリケーション、較正アプリケーション、振動解析アプリケーション、報告生成アプリケーション、またはプラント 100 内の保守機能を行う際に保守技術者を支援するために使用され得る任意の他の保守サポートアプリケーション等のサポートアプリケーションを含んでもよい。

20

【0044】

引き続き図 4 を参照すると、例示的モジュール制御システム 200 は、図 1 のモジュール制御システム 1 と同様に実装され得る。モジュール制御システム 200 は、モジュールコントローラ 202、構成データベース 204、及びフィールドデバイス 210 を含み得る特殊装置を含む。モジュール制御システム 200 は、デバイス 202、204、及び 210 ~ 114 が物理的フレーム 220 内に存在するスキッドマウントシステムであり得る。モジュール制御システム 200 は、スタンドアロンモードで動作し、制御された方式でポンプ液等のプラント内で比較的複雑な機能を行い、水を加熱してタンク内の一定温度を維持し、ろ過機能等を行うように構成され得る。このために、モジュール制御システム 200 は、バルブ、タンク、センサ等を含み得る。

30

【0045】

モジュールコントローラ 202 は、分散型制御システム 122 のプラットフォーム上にネイティブに構築され得る。言い換えれば、モジュールコントローラ 202 は、特に分散型制御システム 122 での使用のために開発されている一方で、自律的な動作も可能である。このために、モジュールコントローラ 202 は、仲介を必要としないファームウェア及び / またはソフトウェア機能 (ファームウェア及び / またはソフトウェアのパーティング / アダプテーションレイヤまたは対応するアプリケーションプログラミングインターフ

40

50

エイス(A P I)機能等)を含み、分散型制御システム 122 のノードと相互作用し得る。特定のソフトウェアーキテクチャのモジュールコントローラ 202 は、分散型制御システム 122 の他のコントローラと 1 つ以上のソフトウェアレイヤを共有する。いずれにしても、分散型制御システム 122 のプラットフォームにネイティブであるため、モジュールコントローラ 202 の構成、セキュリティ機構、及び通信は、分散型制御システム 122 と完全に互換性がある。

【 0046 】

構成データベース 204 は、例えば、ハードディスクまたはフラッシュドライブ等の非一時的コンピュータ可読メモリに格納され得る。コンピュータ可読メモリ及びモジュールコントローラ 202 は、実装に応じて、別々の単一のチップセットの一部として提供され得る。

10

【 0047 】

製造業者は、モジュール制御システム 200 を組み立てて、P I D ループのゲイン値、フィールドデバイス 210 の名前及びタグ、ライセンス、ロケール等の設定値及び他のパラメータを用いてモジュール制御システム 200 のパラメータを構成することができる。製造業者はまた、パラメータセキュリティを設定することもできる。例えば、製造業者は、ゲイン値を制限された制御の一部とし、この変数のロック解除のために適切なキーを提供したときにのみ、この値が変更されることを必要とする。場合によっては、製造業者は、モジュール制御システム 200 を一体型ユニットとして輸送するために完全に組み立てることができる。

20

【 0048 】

図 4 に示すように、モジュール制御システム 200 は、データハイウェイ 154 を介してプラント 100 及び分散型制御システム 122 に連結され得る。例えば、ラップトップコンピュータであり得るホスト 240 は、マージアシスタントアプリケーションを実行して、モジュール制御システム 200 を分散型制御システム 122 にマージすることを容易にし得る。あるいは、マージアシスタント 250 は、オペレータワークステーション 250 または 252 に実装され得る。動作中、マージアシスタント 250 は、構成データベース 204 から集中構成データベース 160 に構成データを効率的かつ正確にマージすることによってオペレータを支援する。場合によっては、マージアシスタント 250 はまた、構成データベース 160 内項目との競合を解決するために、構成データベース 204 内の項目の名前の変更または再ラベル付けに関する自動提案も生成する。次いで、マージアシスタント 250 は、更新された構成データを構成データベース 204 にロードする。場合によっては、ホスト 240 は、構成データが更新されている間にモジュール制御システム 200 の構成データを格納する。

30

【 0049 】

別の実装によると、マージアシスタント 250 は、モジュール制御システムを分散型制御システム 122 に統合することによって導入された新しい構成データを分散し、それによって構成データベース 160 に新しいデータをインポートする必要性を排除する。具体的には、構成データをコピーするのではなく、マージアシスタント 250 は、分散型制御システム 122 の範囲内の一意性を確実にするために、構成データベース 204 を新しいタグによりその場で更新し得る。このように、マージアシスタント 250 は、モジュール制御システムを分散型制御システムに統合するのに必要な時間を短縮することができ、場合によっては「インスタントインポート」を提供することさえできる。

40

【 0050 】

図 4 はモジュール制御システム 200 を 1 つだけ示しているが、多数のモジュール制御システムを分散型プロセス制御システム 122 に統合してもよい。マージアシスタント 250 は、この場合、データハイウェイ 154 に連結されたモジュール制御システムを検出し、対応する構成を集中構成データベース 160 に反復的にマージすることができる。

【 0051 】

図 5 A ~ 20 に示すように、モジュールアシスタント 50 は、スタンドアロンモードで

50

動作するモジュール制御システムに関する様々な動作を容易にするための様々な対話型スクリーンを提供し得る。

【0052】

具体的には、図5A～Dは、接続、構成管理、切断、及び管理機能へのアクセスを提供するように、モジュールアシスタント50がユーザに提示し得る例示的対話型スクリーンを示す。「接続」機能は、概して、モジュールアシスタント50が実行しているスタンドアロンコンピュータ（コンピュータ20または60等）が、スタンドアロンコンピュータが接続されているモジュールコントローラと整合することを可能にする。モジュールアシスタント50は、スタンドアロンコンピュータ及びモジュールコントローラ内の構成データが一致するか否かをユーザに通知するための「ワンストップ場所」を有利に提供し得る。図5Aの例示的対話型スクリーンは、物理的モジュールコントローラとの接続の要約を提供し、該当する場合は潜在的な問題をユーザに通知する。一般に、モジュールアシスタント50は、ユーザがパスワードを入力したとき、アクティブプロジェクトが適正であるとき、及びスタンドアロンコンピュータのネットワーク設定が適正であるときに、適切な整合を検出する。さらに、モジュールアシスタント50は、アクティブプロジェクトが取り付けられたモジュールコントローラに不適正であるときに、特定のアイコンを表示し得る。モジュールアシスタント50はまた、例えば、他の問題が検出されたときに、フラグの形式でアイコンを提供し得る。

10

【0053】

本例示的実装では、モジュールアシスタント50は、切断機能300を提供して、リムーバブルメモリカード（例えば、図1のメモリカード23を参照）にデータをバックアップすることをユーザにリマインドする。構成管理機能302は、ユーザがプロジェクトを比較及びリトリーブすることを可能にする。プロジェクト管理機能304は、ユーザがどのプロジェクトが利用可能であるかを見て、作成、削除、コピー等の動作を行うことを可能にする。図5Dに示すように、モジュールアシスタント50は、「類似アクティブ」フラグの適切な値を表示して、プロジェクトにアクティブプロジェクトと同じコミッショニングデータがあるか否かを標示し得る。また、図5Dに示すように、モジュールアシスタント50は、「類似アクティブ」フラグの適切な値を表示して、特定のプロジェクトが取り付けられたモジュールコントローラに適正であるか否かを標示し得る。モジュールアシスタント50は、プロジェクトが取り付けられたモジュールコントローラに不適正である場合に、情報310または別の適切な推奨を表示することができる。

20

【0054】

例示的シナリオでは、ポータブルコンピュータを使用するユーザは、コミッションされたモジュールコントローラに接続する。ポータブルコンピュータのネットワークカード設定は、モジュールコントローラとの通信目的では不適正であり、アクティブプロジェクトは、モジュールコントローラに不適正である。モジュールアシスタント50は、最初に、モジュールコントローラが検出されているが、ユーザのコンピュータのネットワーク設定が完全な通信を許可していないことをユーザに通知し得る。例示的実装では、モジュールアシスタント50は、エクスプローラウィンドウ（例えば、D e l t a V エクスプローラ ウィンドウと同様であってもよい）内にネットワークの階層ビューを表示する。階層ビューは、物理ネットワークの下に、モジュールコントローラとポータブルコンピュータとが接続されていない制御ネットワークカテゴリがあることを標示する。

30

【0055】

次いで、ユーザは、制御320をアクティブ化して、D e l t a V（図6A）等の分散型制御システムのネットワーク設定を適用しようと試みることができる。適用されたネットワーク設定が適正である場合、モジュールコントローラはモジュールアシスタント50を介して認証を要求する（図6B）。次いで、モジュールアシスタント50は、アクティブプロジェクトが不適正であると判定し、適切な通知をユーザに提供し得る（図6C）。次いで、ユーザは適切なプロジェクトに切り替えることができる（図6D及び6E）。次いで、モジュールアシスタント50は、ポータブルコンピュータとモジュールコントローラーとの接続を確認する（図6F）。

40

50

ラとの間の接続が適正であることをユーザに通知する（図 6 F）。

【 0 0 5 6 】

次いで、ユーザは、プロジェクトバックアップ（図 3 に示すプロジェクト 72 等）から新しいアクティブプロジェクトに構成をリトリーブすることを選択してもよい。モジュールアシスタント 50 は、それぞれのラジオボタンを介して選択可能ないくつかの対話型オプションを提供することができ、ユーザは、所望のオプションを選択し、ユーザは、制御 350 を続行してもよい（図 7 A）。次いで、モジュールアシスタント 50 は、モジュールコントローラのプロジェクトバックアップからのプロジェクトの名前で名前を自動入力し得る（図 7 B）。

【 0 0 5 7 】

別の例示的シナリオでは、ユーザは、コミッショニングされたモジュールコントローラ上でパスワードを変更することを望む場合がある。このシナリオにおけるユーザは、コミッショニングされたモジュールコントローラで既に認証されている。ユーザは、モジュールコントローラをデコミッショニングせずにパスワードを変更することを望む。ユーザは、適切な制御をアクティブ化させ（図 8 A）、モジュールアシスタント 50 は、パスワードの変更が成功したときに適切な通知を提供する（図 8 B）。

【 0 0 5 8 】

別の例示的シナリオでは、ユーザは、バックアップされていない特定の変更を有し、別のユーザも同様に変更を行っている。具体的には、ユーザは、自分のポータブルコンピュータをモジュールコントローラに接続する前に、いくつかの編集を行っている。ユーザがモジュールコントローラに接続し、モジュールアシスタント 350 は、モジュールコントローラがユーザのパーソナルコンピュータに格納された構成データから欠落している特定の変更を有することをユーザに通知する。ユーザは、モジュールコントローラのプロジェクトストレージからこの構成をリトリーブすることを判定する。ユーザは、モジュールアシスタント 50 によって提供された制御 370 をアクティブ化する（図 9 A）。それに応答して、モジュールアシスタント 50 は、競合する項目のレポート 380 を生成する（図 9 B）。モジュールアシスタント 50 は、より新しい日付及びタイムスタンプを視覚的に強調し得る（例えば、図 9 B に示すように、ハイライトまたは下線によって）。次いで、ユーザは、特定の項目のチェックボックス 400 をアクティブにし、これらの項目をリトリーブすることができる（図 9 C）。次いで、モジュールアシスタント 50 は、選択された項目をリトリーブし、進行状況を表示し得る（図 9 D 及び 9 E）。

【 0 0 5 9 】

別のシナリオによると、ユーザは基本ライセンスを有する。ユーザは、バックアッププロジェクトからリトリーブする（図 10 A）が、モジュールアシスタント 50 は詳細な比較を表示しない（図 10 B）。

【 0 0 6 0 】

別の例示的シナリオでは、アクティブプロジェクトは、取り付けられたモジュールコントローラに不適正である。モジュールアシスタント 50 は、図 11 に示すような適切な通知を提供する。

【 0 0 6 1 】

別の例示的シナリオでは、ユーザは、アクティブプロジェクトをバックアップする。ユーザは、モジュールアシスタント 50 によって提供された制御を使用し（図 12 A）、モジュールアシスタント 50 は、バックアップの効果に関する警告を提供する（図 12 B）。モジュールアシスタント 50 は、バックアップが進行中であることをユーザに通知する。バックアップが完了すると、モジュールアシスタント 50 は、「キャンセル」ボタンを「閉じる」に変更し、「バックアップ」ボタンを無効にし得る（図 12 C 及び 12 D）。モジュールアシスタント 50 は、もしあれば失敗、及びそれに対応する理由をユーザに通知し得る（図 12 E）。

【 0 0 6 2 】

ユーザは、時々、アクティブプロジェクトのバックアップを忘れる場合がある。例示的

10

20

30

40

50

シナリオでは、ユーザのポータブルコンピュータは、コミッショナされたモジュールコントローラに取り付けられる。ユーザがケーブルを切断するか、またはポータブルコンピュータとモジュールコントローラとの間の接続を切断すると、モジュールアシスタント50は通知（例えばポップアップウィンドウ）を自動的に生成し、アクティブプロジェクトがバックアップされていないことをユーザに通知し得る。ポップアップウィンドウは、図12A～Eに示すスクリーン等の切断されたスクリーンにユーザを直接案内するための対話型制御を含み得る。いくつかの実装では、モジュールアシスタント50は、タスクバーにコンテキストメニューoptionを提供して、例えば、ユーザが切断されたスクリーンに直接進むことを可能にし得る。

【0063】

別のシナリオでは、ユーザはプロジェクトを作成することを望む。図5Dを参照して上述したシナリオと同様に、モジュールアシスタント50は、ユーザが「プロジェクト管理」タブを選択したときに、「類似アクティブ」フラグ及び「適正」フラグを表示し得る。（図13A）モジュールアシスタント50は、ユーザにプロジェクト名をプロンプトし（図13B）、新しいプロジェクト名を受信し（図13D）、プロジェクトを作成し得る（図13E）。具体的には、モジュールアシスタント50は、最初に「作成」ボタンを無効にして、有効なプロジェクト名が入力された場合にのみこのボタンを有効にし得る。このために、モジュールアシスタント50は、長さ、名前及び日付制限、プロジェクトタイプ制限等の任意の適切な基準を適用し得る。プロジェクトが作成されるとき（図13D）、モジュールアシスタント50は、プロセスの中止を回避するために「キャンセル」及び「作成」ボタンを無効にし得る。モジュールアシスタント50は、完了時に「キャンセル」ボタンを「閉じる」に変更し得る（図13E）。

10

【0064】

別のシナリオでは、ユーザはプロジェクトのプロパティを編集することを望む。モジュールアシスタント50は、ユーザに、アクティブプロジェクトのプロパティを編集することを促し得る（図14A）。ユーザが「適用」をクリックすると、モジュールアシスタント50は動作を開始し、プロセスの中止を防止するためにボタンを無効にする（図14B）。完了すると、モジュールアシスタント50は、ボタンを有効にし得る（図14C）。

20

【0065】

別のシナリオでのユーザは、アクティブプロジェクトを切り替える。モジュールアシスタント50は、状態 / 警告ウィンドウ450を生成する。モジュールアシスタント50は、上記のシナリオと同様の進捗をユーザに通知する（図15B及び15C）。

30

【0066】

ユーザがアクティブプロジェクトを削除することを望む場合、モジュールアシスタント50は適切な警告を生成し得る（図16A）。削除を望まない場合、モジュールアシスタント50は、上記のシナリオと同様の進捗をユーザに通知する（図16B及び16C）。

【0067】

ユーザがプロジェクトをコピーすることを望む場合、モジュールアシスタント50は非アクティブプロジェクトのコピーを許可し得る（図17A）。次いで、モジュールアシスタント50は、上記のシナリオと同様の進捗をユーザに通知する（図17B及び17C）。

40

【0068】

モジュールアシスタント50はまた、ユーザが、アクティブプロジェクトをポータブルコンピュータまたは別の適切なストレージ上のドライブにバックアップすることを可能にし得る（図18）。ユーザがドライブからのプロジェクトを比較及びリトリーブすることを望む場合、モジュールアシスタント50は比較 / リトリーブダイアログを提供し得る（図19）。ユーザがドライブ内のバックアップに基づいて新しいプロジェクトを作成することを望む場合、モジュールアシスタント50は適切なダイアログボックスを提供し得る（図20）。

【0069】

いくつかの実装では、モジュールアシスタント50は、ユーザがドングルを有するモジ

50

ユールコントローラだけと通信することを要求し得る。ポータブルコンピュータに取り付けられていない場合、モジュールアシスタント 50 は、特定の機能を非アクティブ化し、ユーザに他の機能の提供を継続し得る。例えば、モジュールアシスタント 50 は、「接続」タブ下のネットワーク設定セクションの機能を有効にし、システムドングルが認証セクションに存在しない（及び全ての制御を無効にする）というメッセージを提供し、プロジェクト情報セクションのアクティブプロジェクトの名前のみを表示し得る。「管理構成」及び「切断」タブの下では、モジュールアシスタント 50 は、欠落しているドングルに関するメッセージを表示し、全ての制御を無効にし得る。「プロジェクト管理」タブの下では、モジュールアシスタント 50 は、ドングルが取り付けられたときと同じ機能を引き続き提供し得る。

10

【0070】

次に、図 21 は、モジュール制御システムの構成を効率的に管理するための例示的方法 500 のフロー図を示す。方法 500 は、例えば、図 1 のシステムに実装され得るより具体的な例として、方法 500 は、モジュール制御システムアシスタント 50 において実装され得る。

【0071】

方法 500 は、スタンドアロンワークステーションとモジュールコントローラとの間の接続が検出されるブロック 502 で開始する。例えば、図 1 を参照すると、モジュール制御システムアシスタント 50 は、コンピュータ 12 がモジュールコントローラ 10 に通信可能に接続されていることを検出し得る。例えば、図 5A に示すように、モジュール制御システムアシスタント 50 は、コントローラがワークステーションのユーザインターフェースを介して取り付けられているという通知を提供し得る。

20

【0072】

場合によっては、モジュール制御システムアシスタント 50 は、モジュールコントローラへの接続を検出するが、ワークステーションがモジュールコントローラと情報を交換できないと判定する。例えば、スタンドアロンワークステーションのネットワーク設定によって、通信スタックの特定のレイヤーのアクティブ化が妨げられ得る。モジュール制御システムアシスタント 50 は、ブロック 504 でネットワーク設定が適正であるか否かを検出し、設定が不適正である場合に、モジュール制御システムアシスタント 50 は、分散型制御システムに適用可能なネットワーク設定を適用するための対話型スクリーンを提供し得る（例えば、図 5 参照）。これらの設定は、IP アドレス、サブネットマスク、デフォルトゲートウェイ等に影響する可能性がある。

30

【0073】

ブロック 506 において、オペレータは認証され得る。例えば、図 6A 及び 6B に示すように、認証情報は、ログイン情報及びパスワード情報を含み得る。

【0074】

次に、ブロック 508 において、スタンドアロンワークステーションのメモリに格納されたプロジェクト（「アクティブプロジェクト」）は、モジュールコントローラのメモリ（例えば、メモリカード 23）内のプロジェクトと比較される。このために、任意の適切なファイル比較技術が使用され得、またはモジュール制御システムアシスタント 50 は、対応するバージョン識別子を比較し得る。2つのプロジェクトが同一でない場合、モジュール制御システムアシスタント 50 は、アクティブプロジェクトがモジュールコントローラに適正であるか否かを判定し得る。モジュール制御システムアシスタント 50 は、例えば、アクティブプロジェクトが、モジュールコントローラが動作するモジュール制御システムの種類、モジュール制御システムが装備されているコンポーネントのタイプ等を指定するメタデータを含むか否かを判定し得る。

40

【0075】

アクティブプロジェクトがモジュールコントローラに適正であることが判定された場合、ブロック 510 でモジュール制御システムアシスタント 50 は、モジュールコントローラから構成をリトリープするための1つ以上の制御を提供し得る（例えば、図 9A を参照

50

)。場合によっては、モジュールコントローラに保存されたプロジェクトの一部分のみが、ワークステーションに保存されたアクティブプロジェクトにマージされる。ブロック 512において、モジュール制御システムアシスタント 50 は、アクティブプロジェクトに適用するための個々の要素または項目を選択するための制御を提供し得る。例えば、図 9B に示すように、モジュール制御システムアシスタント 50 は、個々の項目の名称、構成の種類等を指定し、オペレータが様々な特徴に基づいてこれらの項目をグループ化することを可能にし得る。

【0076】

「processing (処理すること)」、「computing (コンピューティングすること)」、「calculating (計算すること)」、「determining (判定すること)」、「presenting (提示すること)」、「displaying (表示すること)」等の単語を本明細書において使用して議論することは、1つ以上のメモリ(例えば、揮発性メモリ、不揮発性メモリ、またはそれらの組み合わせ)内の物理的(例えば、電子的、磁気的、もしくは光学的)量として表されるデータを操作または変換する機械(例えば、コンピュータ)、レジスタ、または情報を受信、格納、送信、もしくは表示する他の機械構成要素のアクションまたはプロセスを指す場合がある。

【0077】

ソフトウェアに実装される場合、本明細書に記載されるアプリケーション、サービス、エンジン、ルーチン、及びモジュールはいずれも、コンピュータもしくはプロセッサのRAMもしくはROM等における磁気ディスク、レーザディスク、固体メモリデバイス、分子メモリストレージデバイス、光ディスク、または他のストレージ媒体等の、任意の有形の非一時的コンピュータ可読メモリに格納されてもよい。本明細書に開示される例示的システムは、他の構成要素の中でも、ハードウェア上で実行されるソフトウェア及び/またはファームウェアを含むように開示されているが、そのようなシステムが単に例証的であるに過ぎず、限定的であるとみなされるべきではないことに留意されたい。例えば、これらのハードウェア、ソフトウェア、及びファームウェア構成要素のうちのいずれかまたは全てが、ハードウェアにのみ、ソフトウェアにのみ、あるいはハードウェア及びソフトウェアの任意の組み合わせで、埋め込まれ得ることが企図される。したがって、当業者は、提供された例がこのようなシステムを実装する唯一の方式ではないことを容易に理解するであろう。

【0078】

したがって、本開示の技術について具体的な例を参照して説明したが、これらの例は単に例証的であることを意図し、本発明の限定であることを意図せず、変更、追加、または削除が、本発明の趣旨及び範囲から逸脱することなく、開示された実施形態に対して行われてもよいことが当業者には明らかであろう。

10

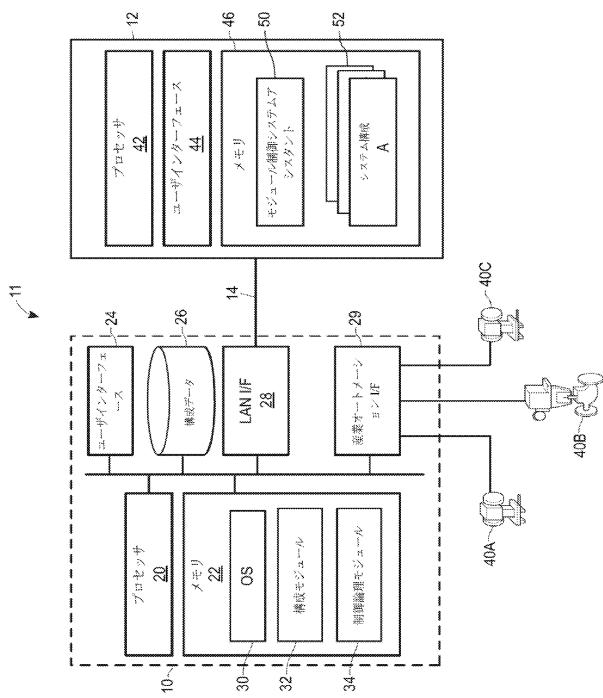
20

30

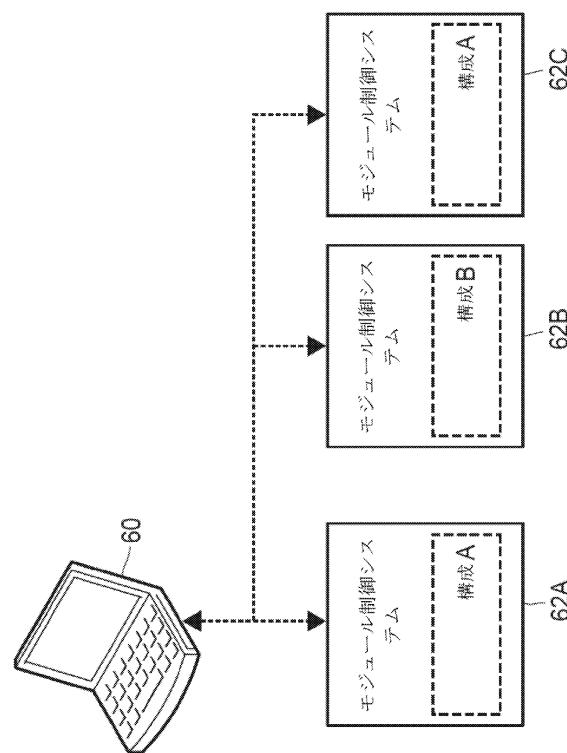
40

50

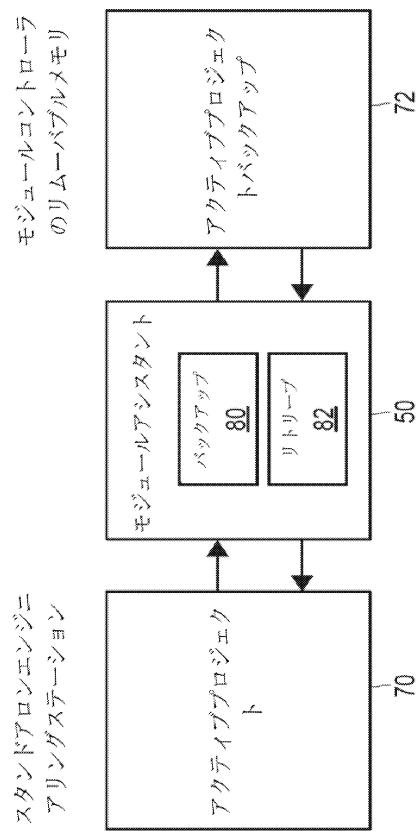
【図面】
【図 1】



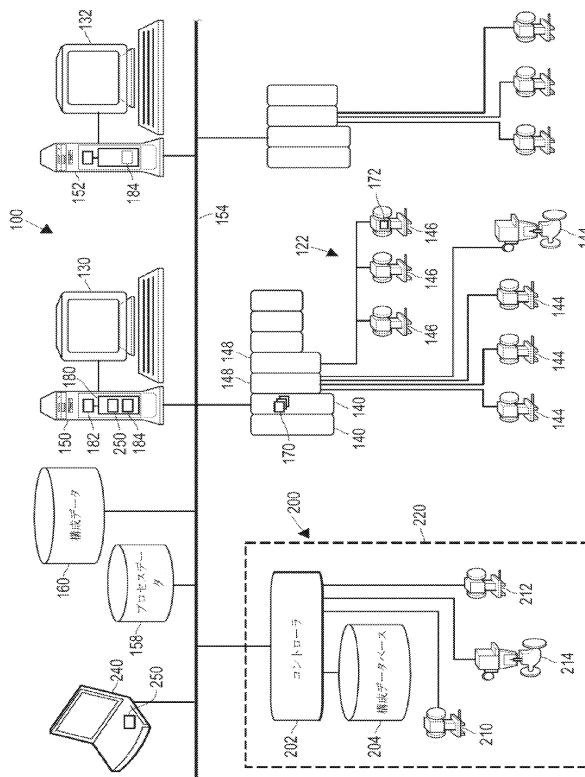
【図2】



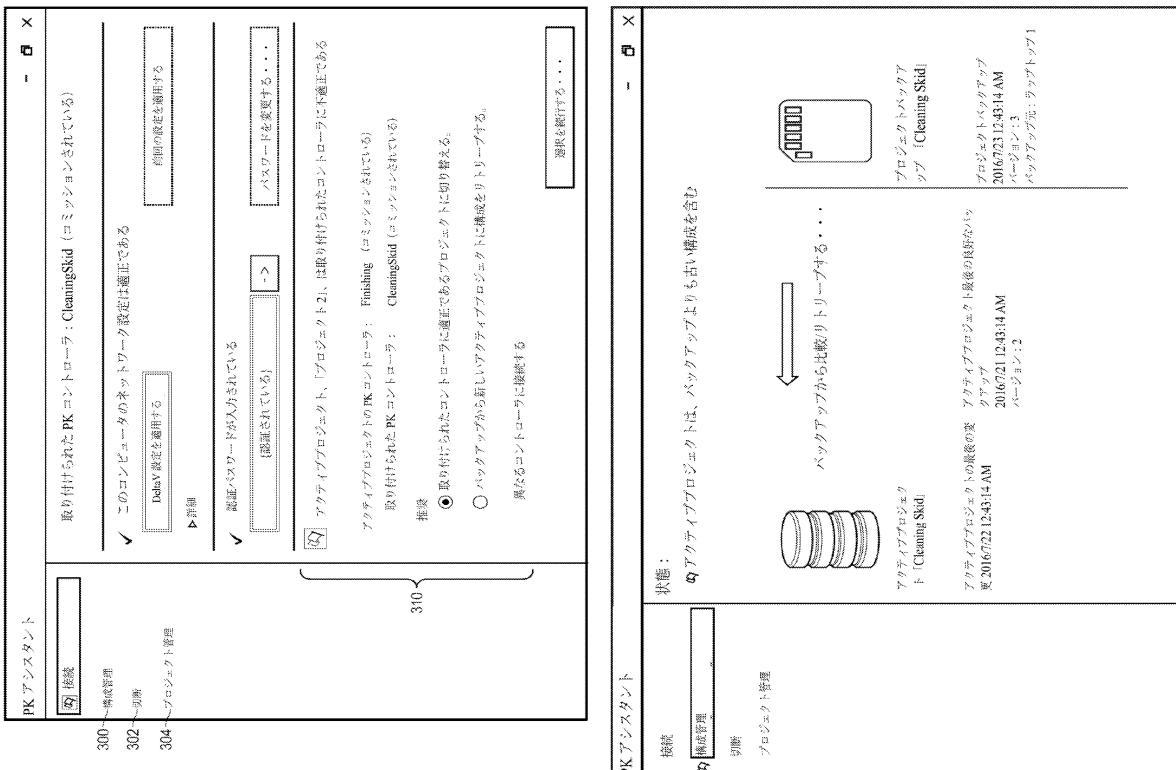
【 図 3 】



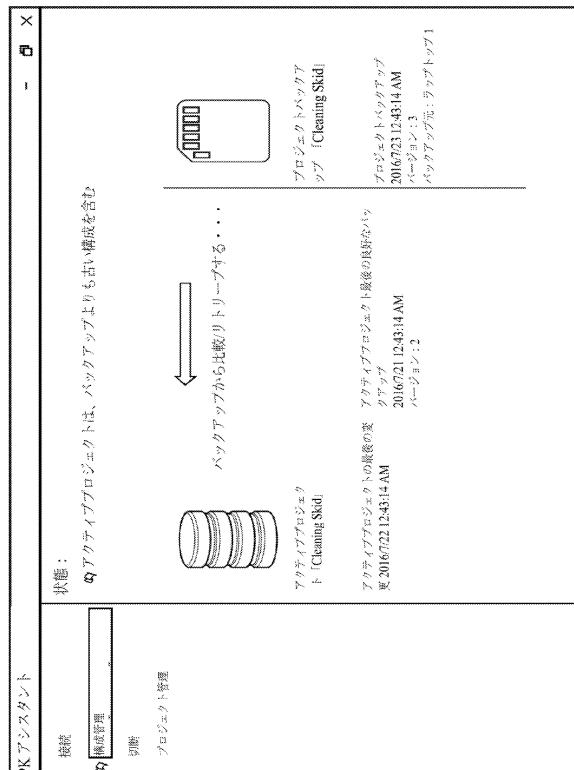
【図4】



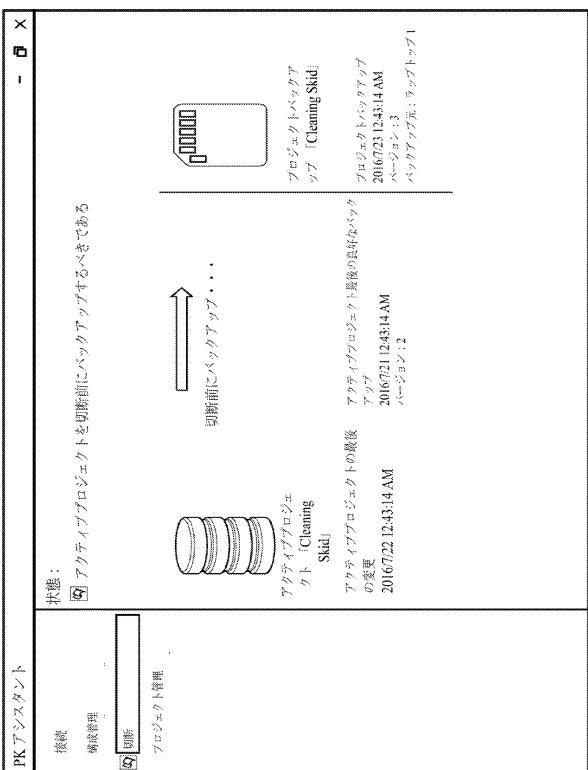
【図 5 A】



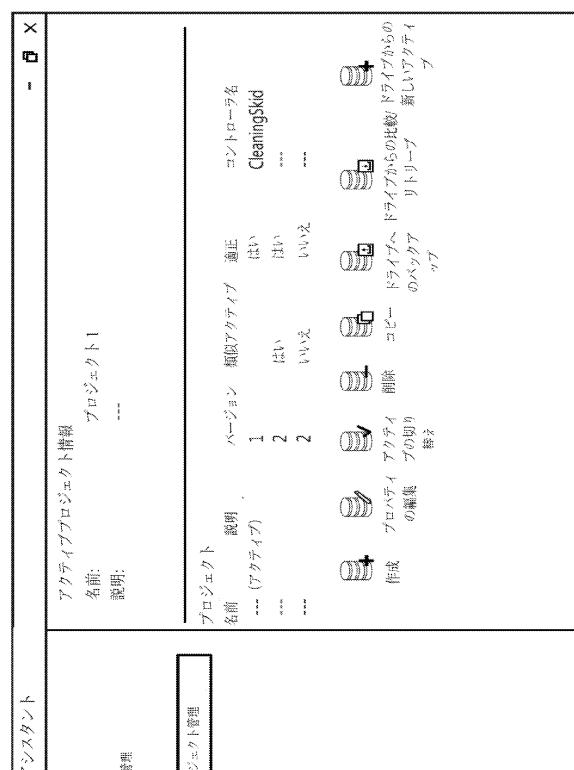
【図 5 B】



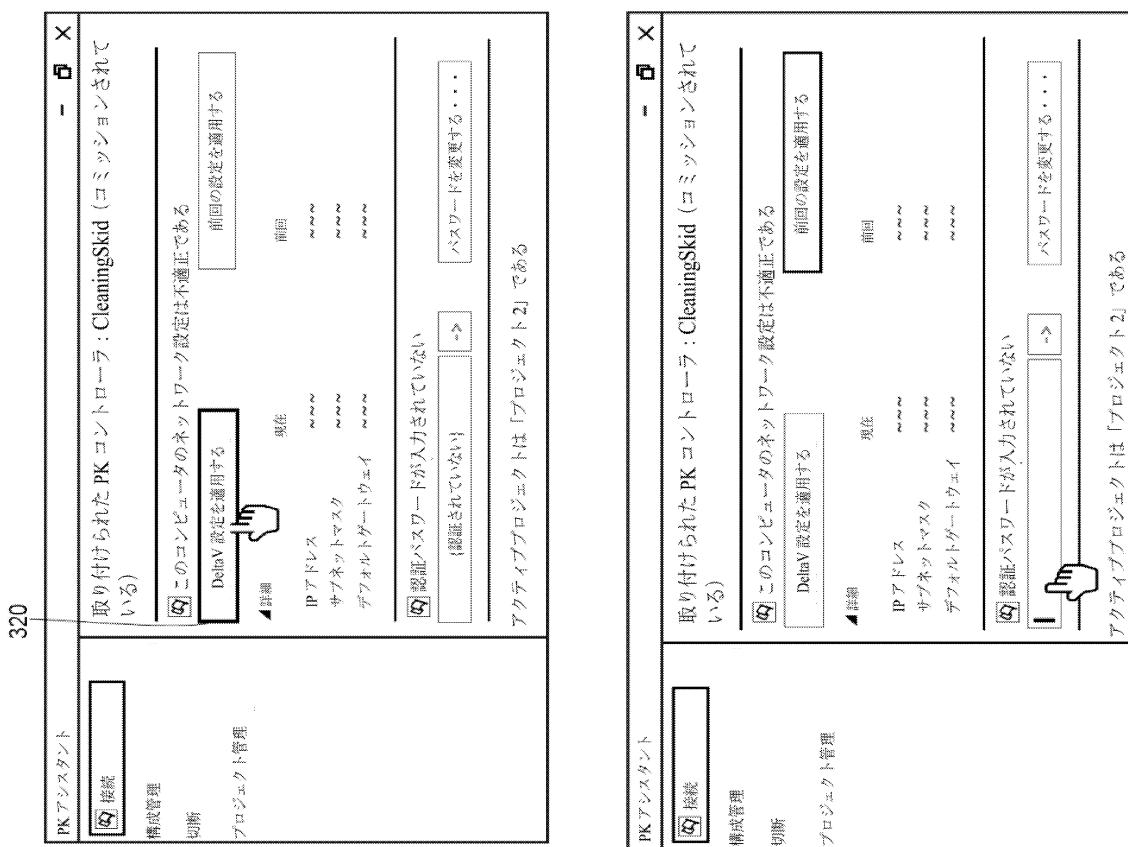
【図 5 C】



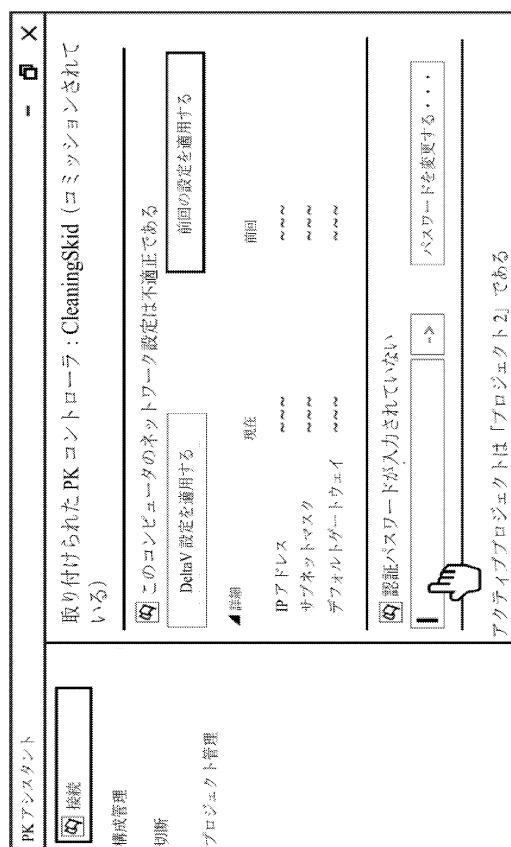
【図 5 D】



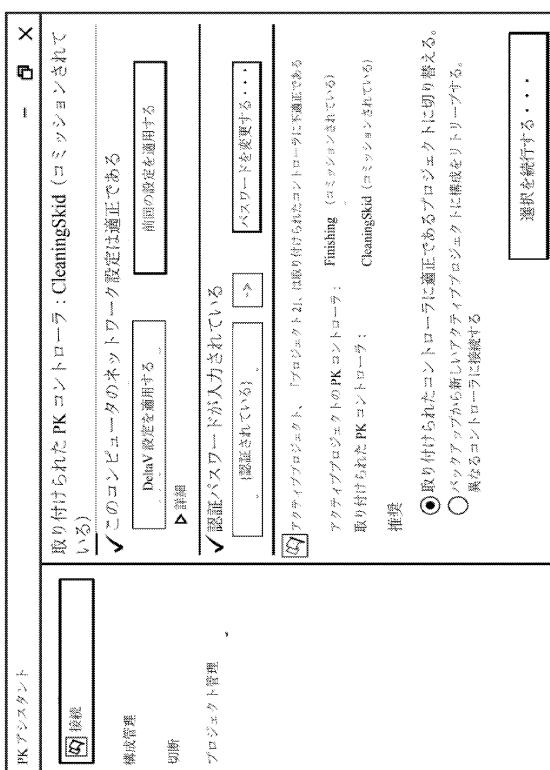
【図 6 A】



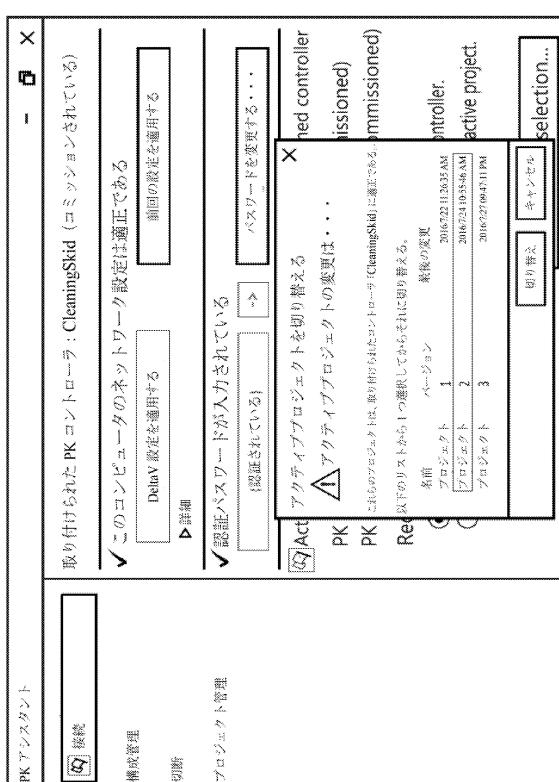
【図 6 B】



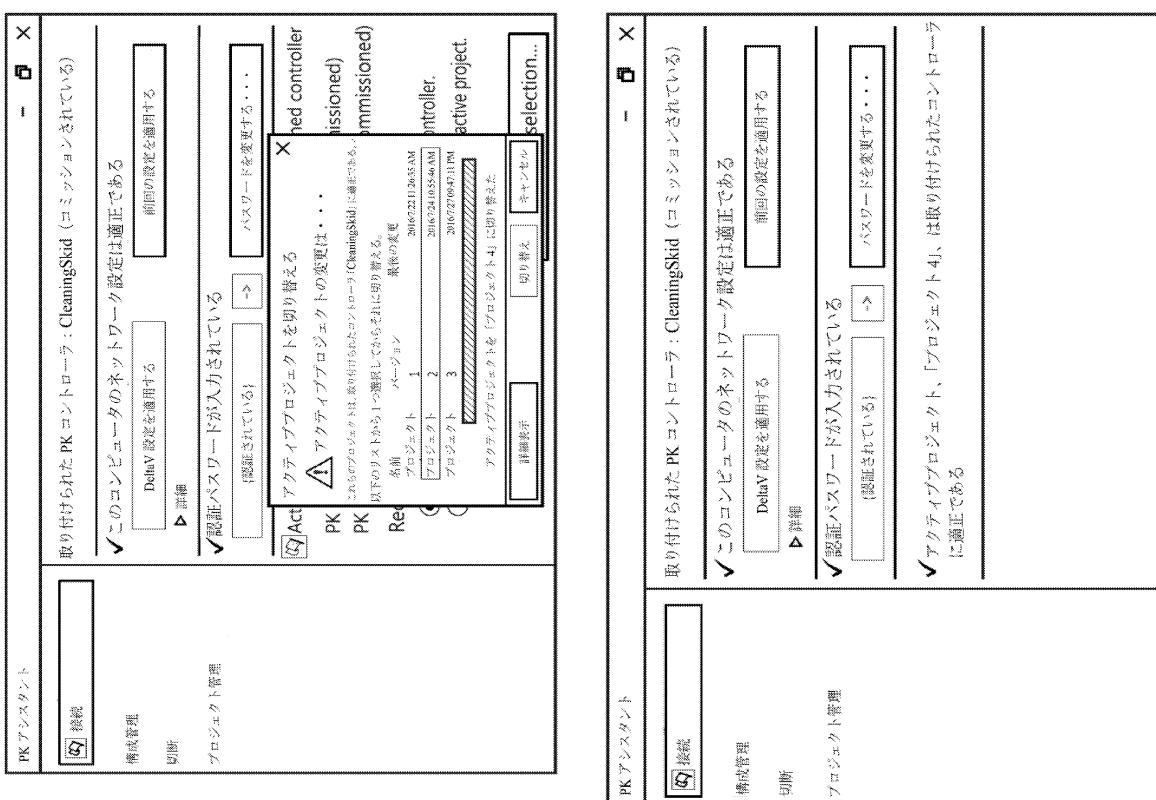
【図 6 C】



【図 6 D】

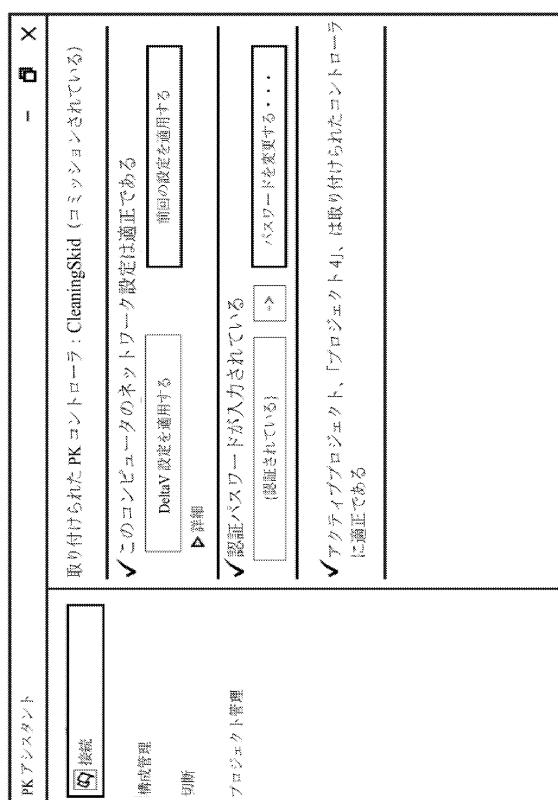


【図 6 E】



(26)

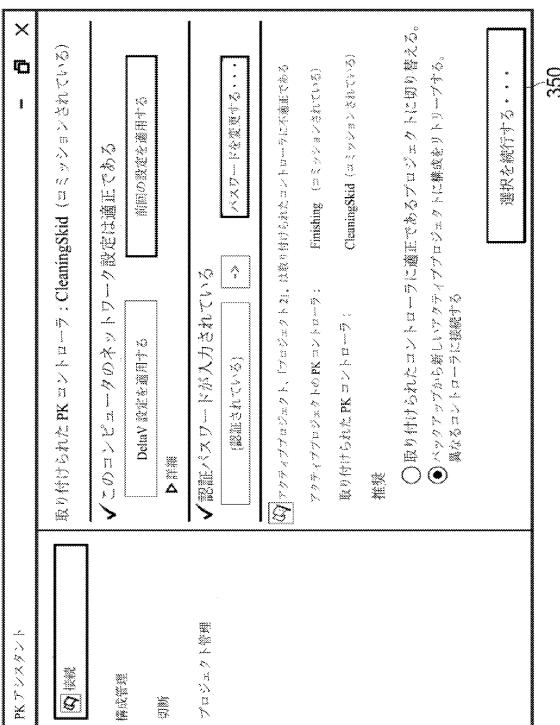
【図 6 F】



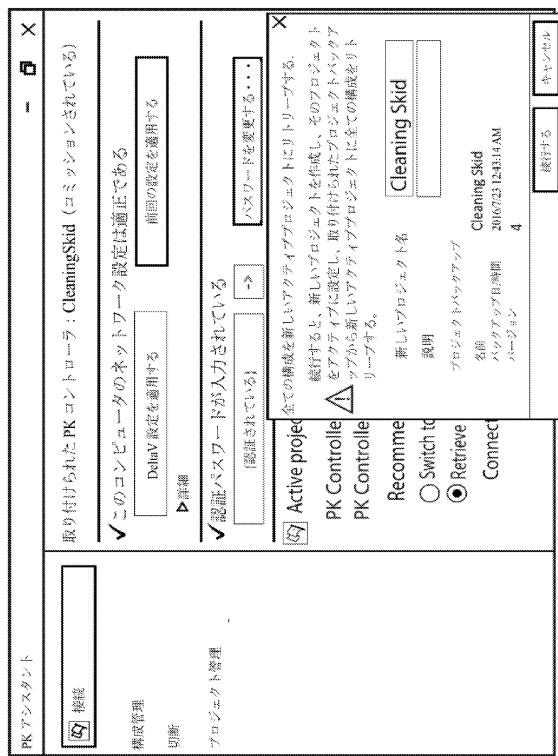
10

20

【図 7 A】



【図 7 B】

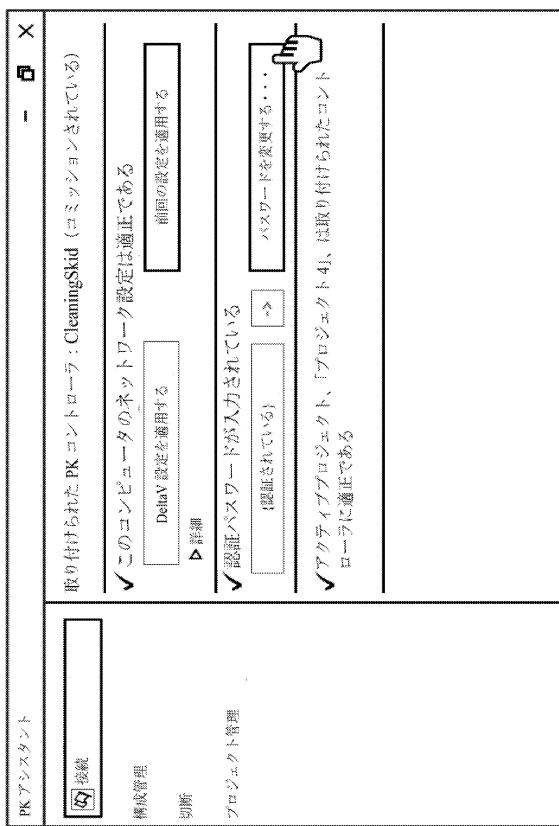


30

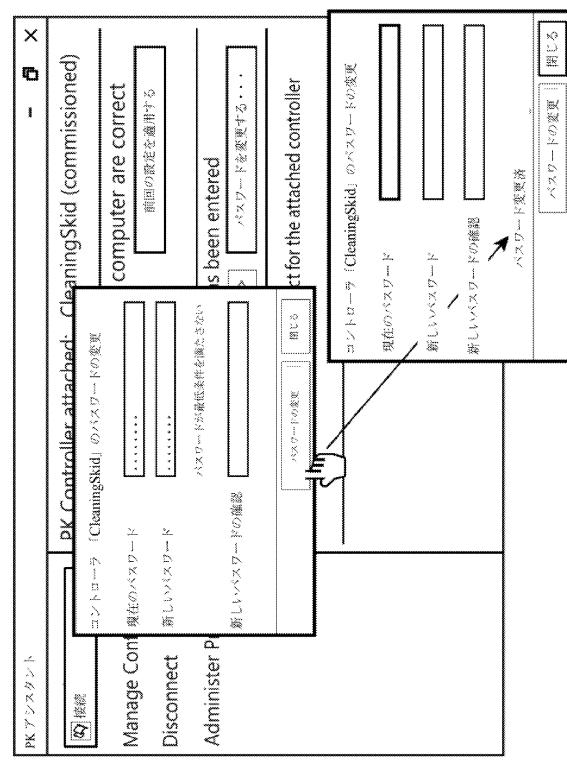
40

50

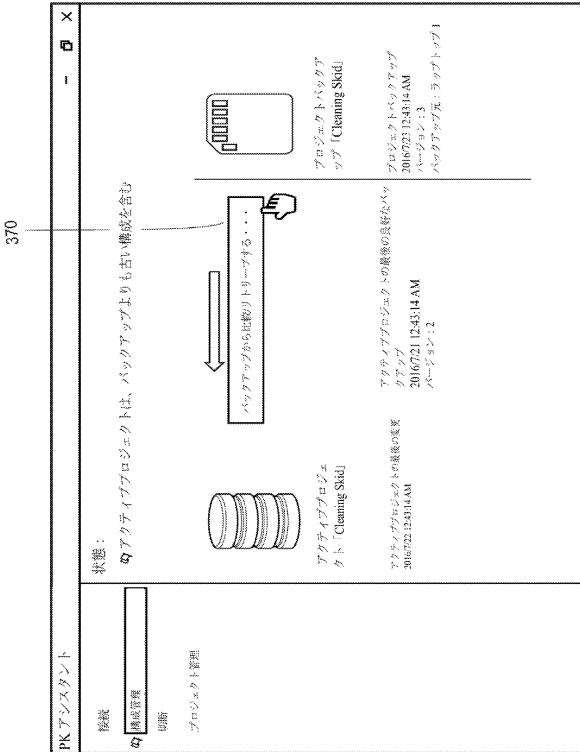
【図 8 A】



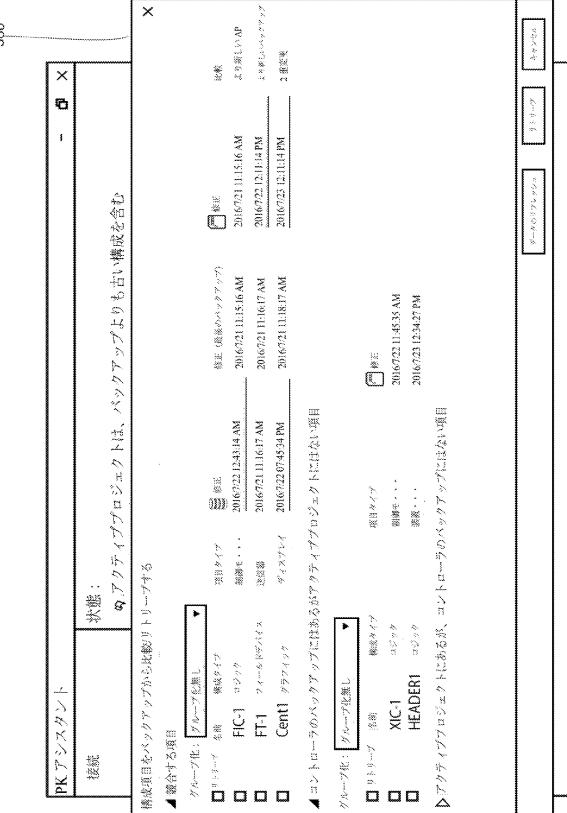
【図 8 B】



【図 9 A】



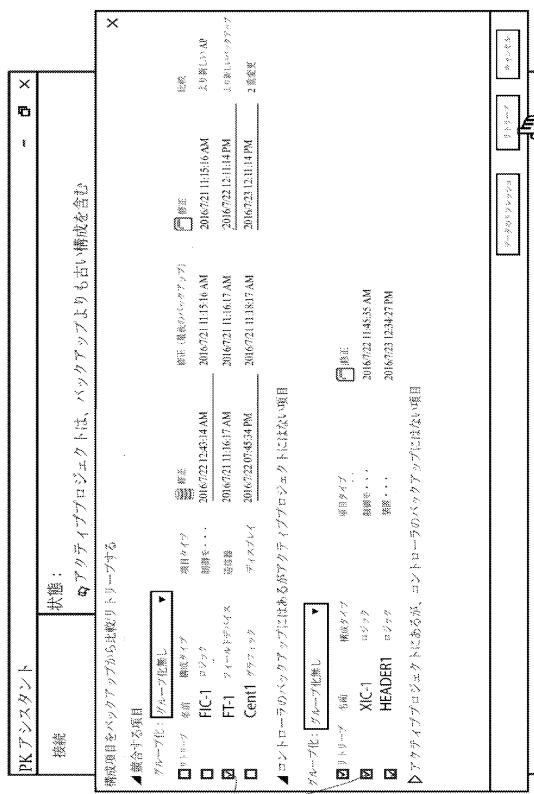
【図 9 B】



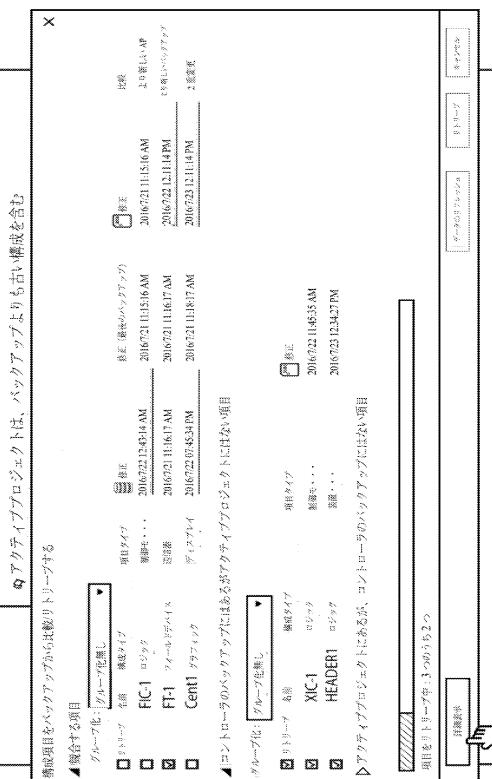
(28)

JP 7398189 B2 2023.12.14

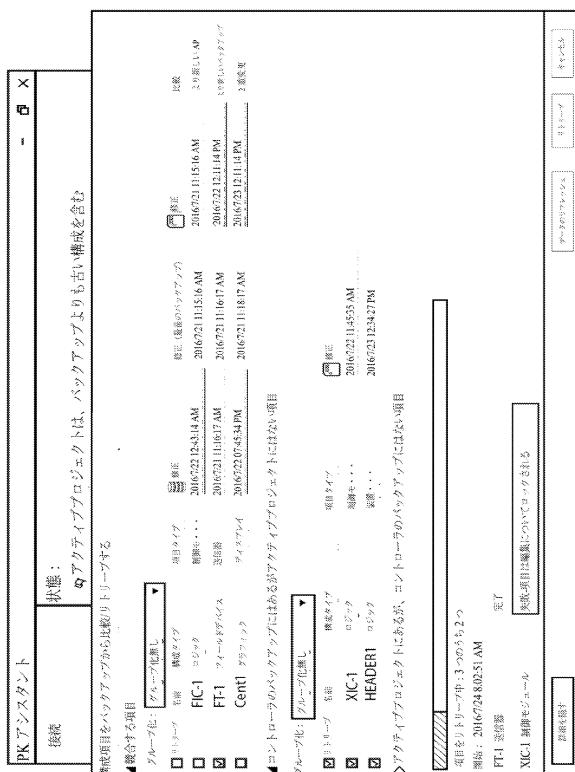
【図 9 C】



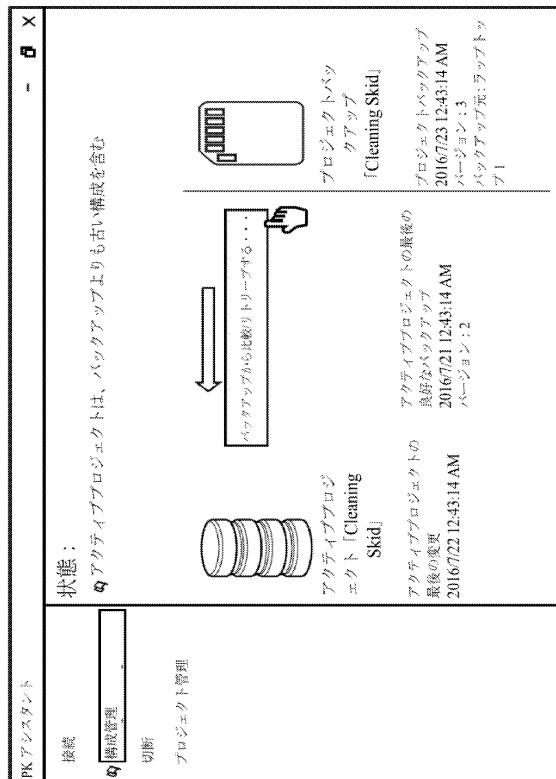
【図9D】



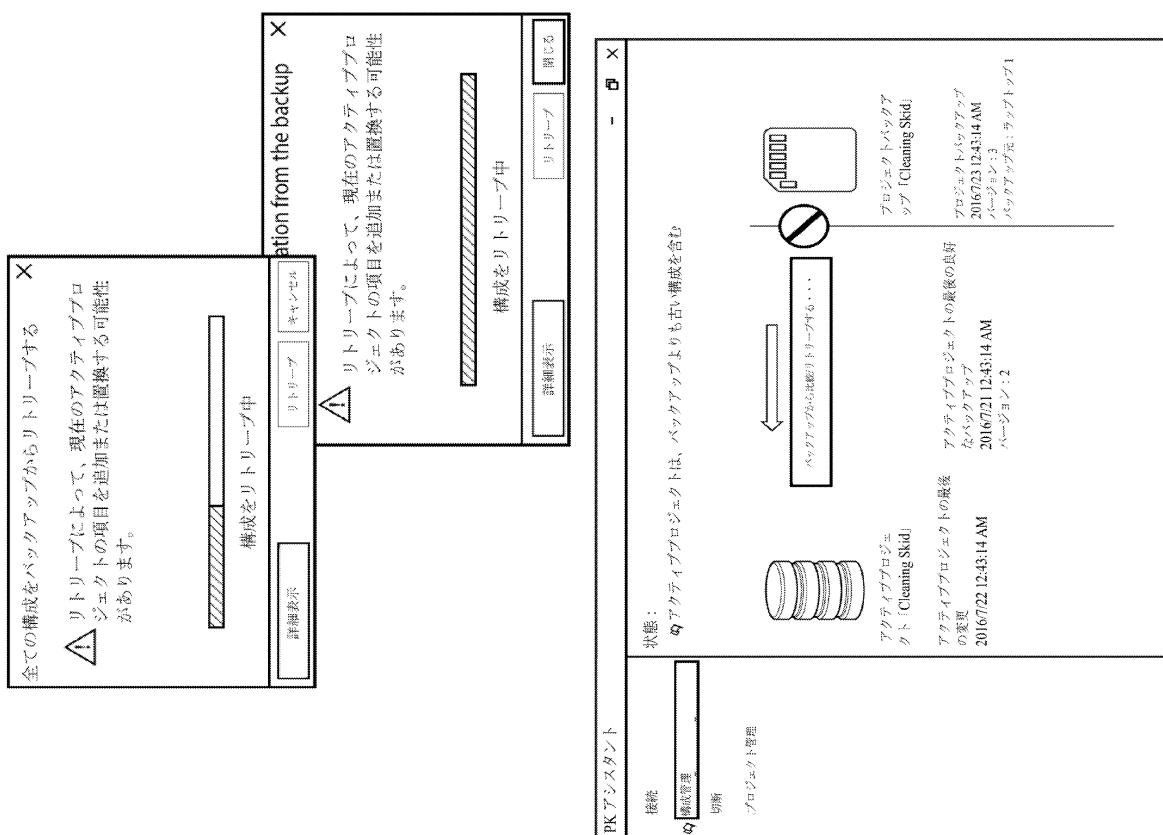
【図9E】



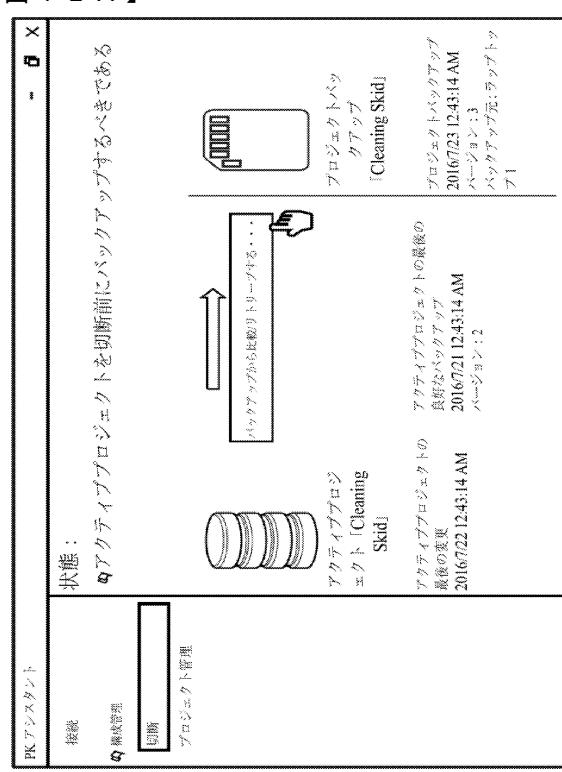
【 10A 】



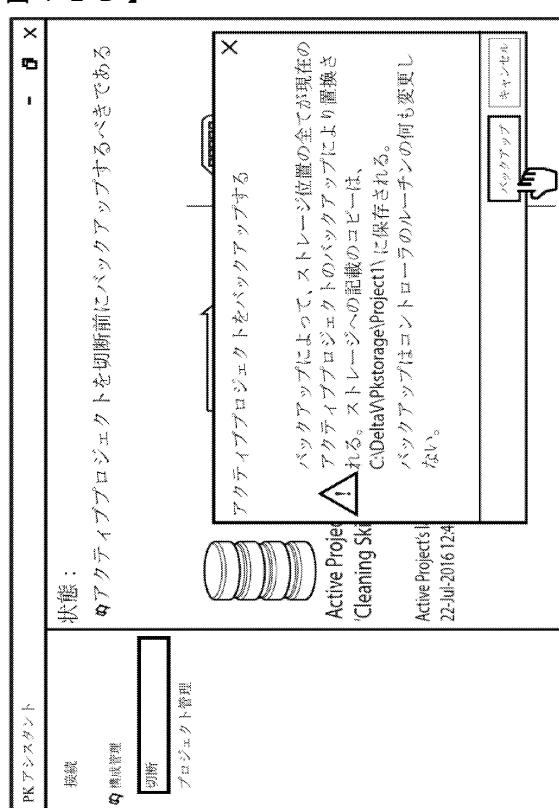
【図 10B】



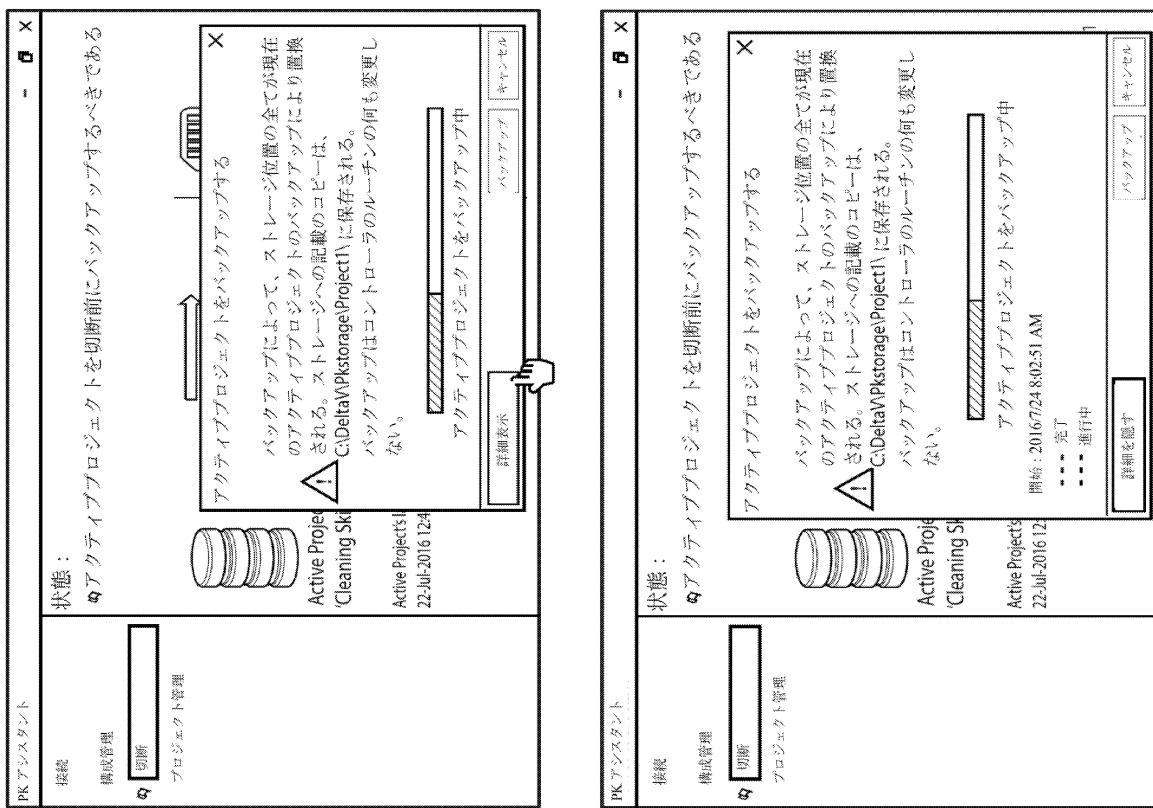
【図 12A】



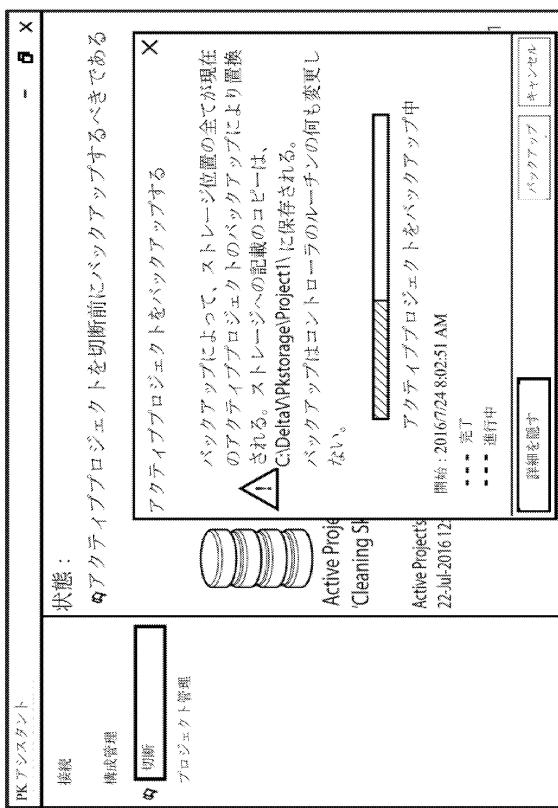
【図 12B】



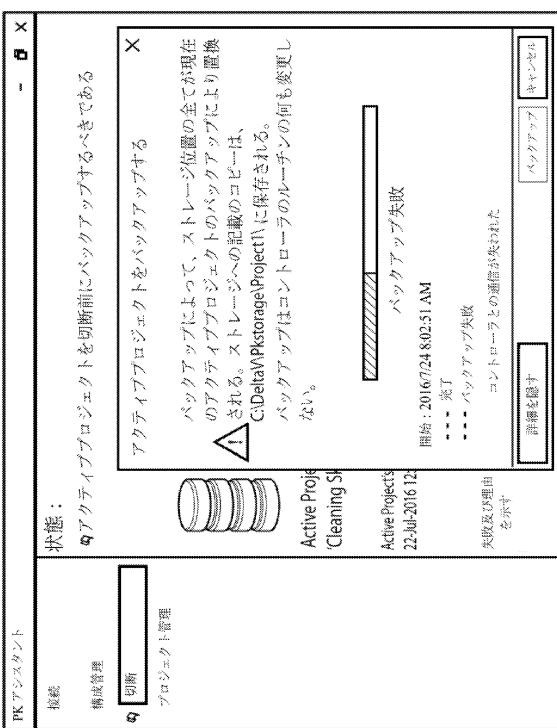
【図 1 2 C】



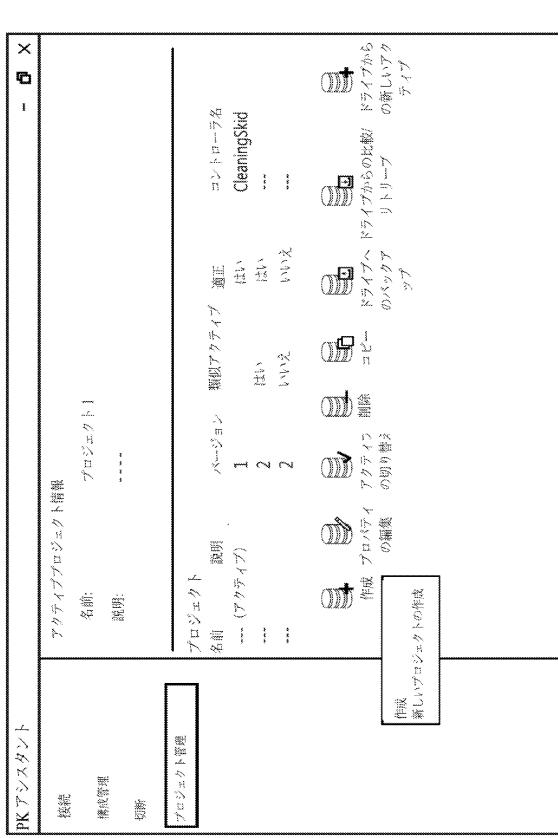
【図 1 2 D】



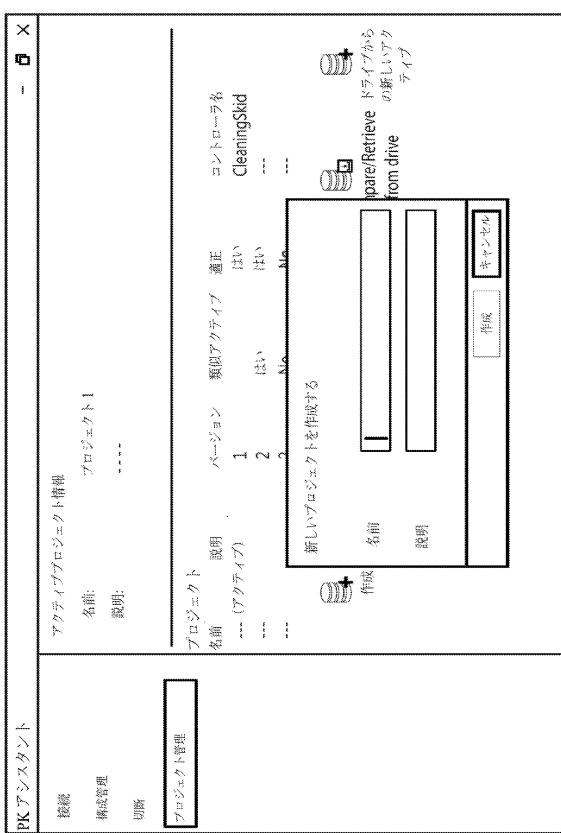
【図 1 2 E】



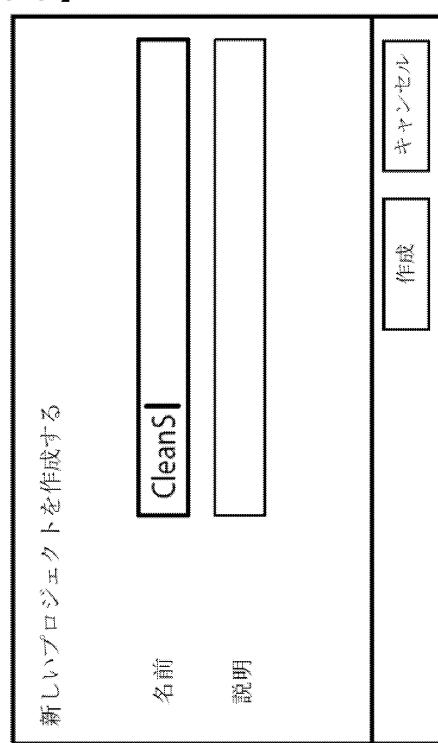
【図 1 3 A】



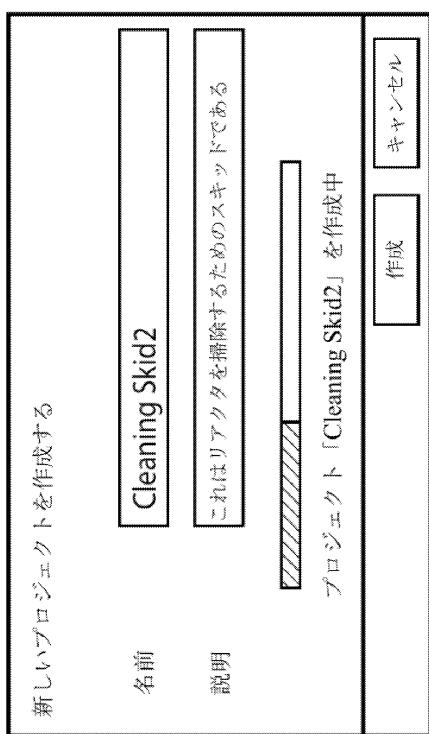
【図 1 3 B】



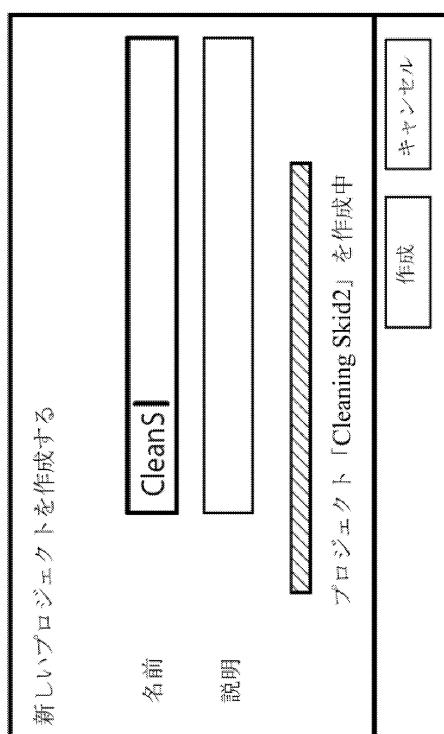
【図 1 3 C】



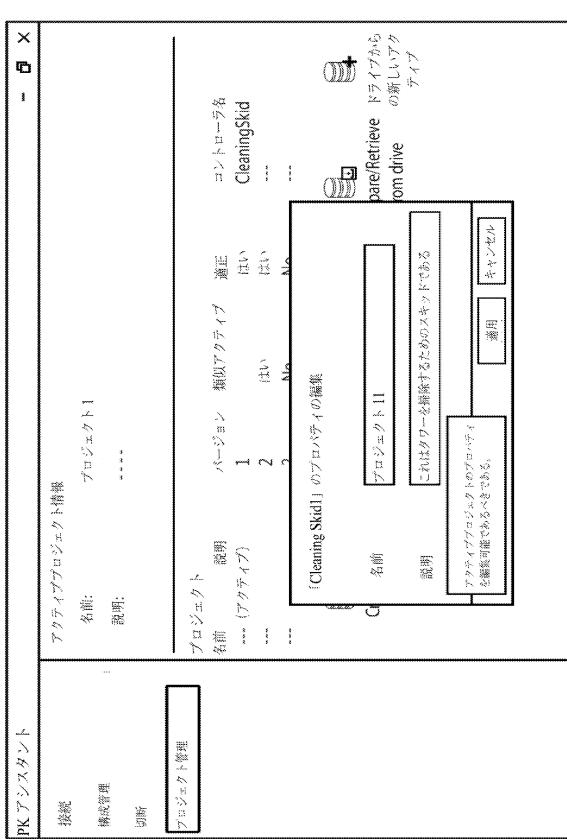
【図 1 3 D】



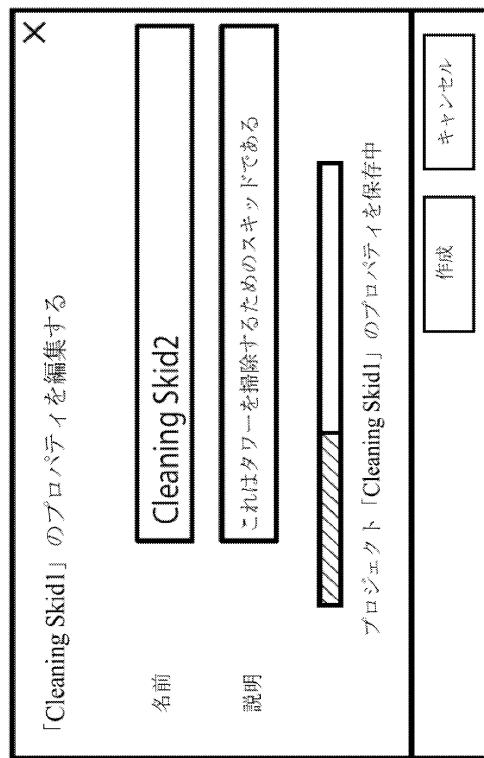
【図 1 3 E】



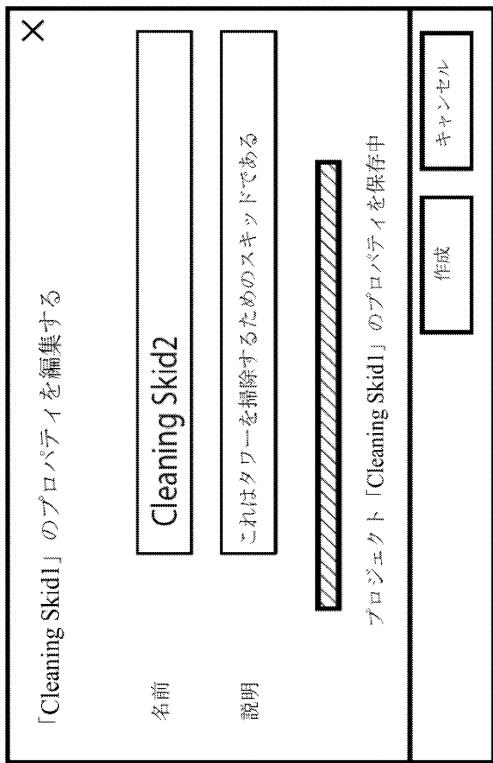
【図 14 A】



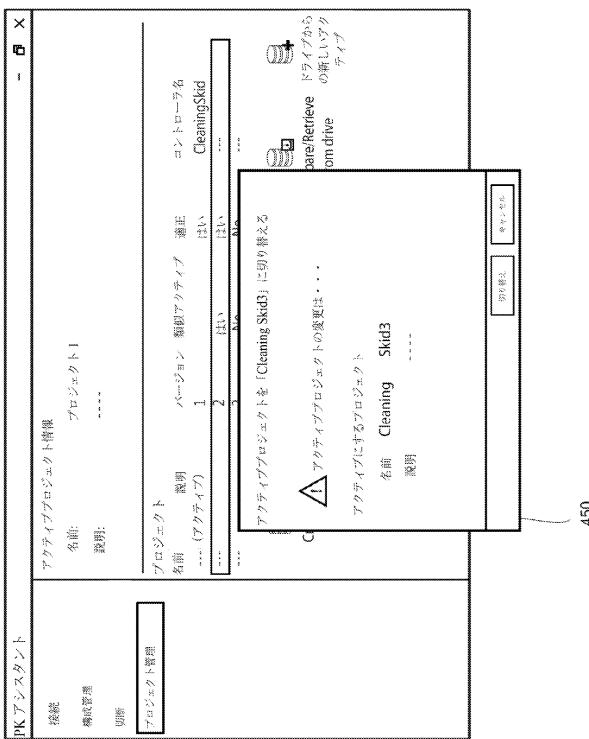
【図 14 B】



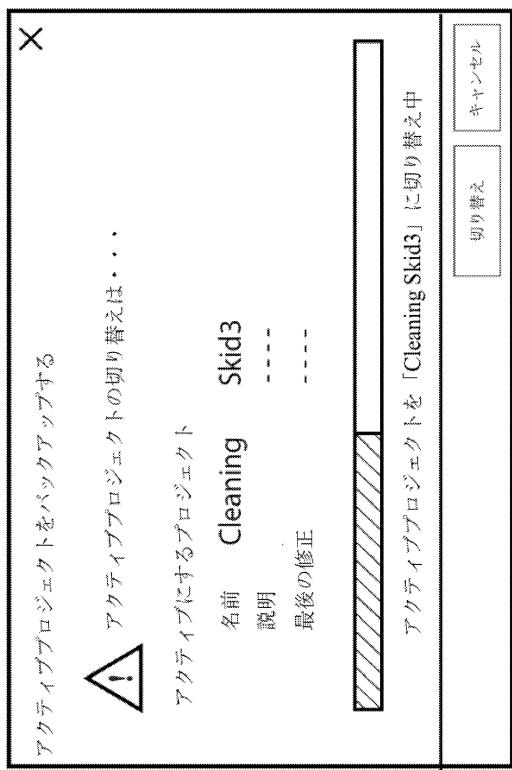
【図 14 C】



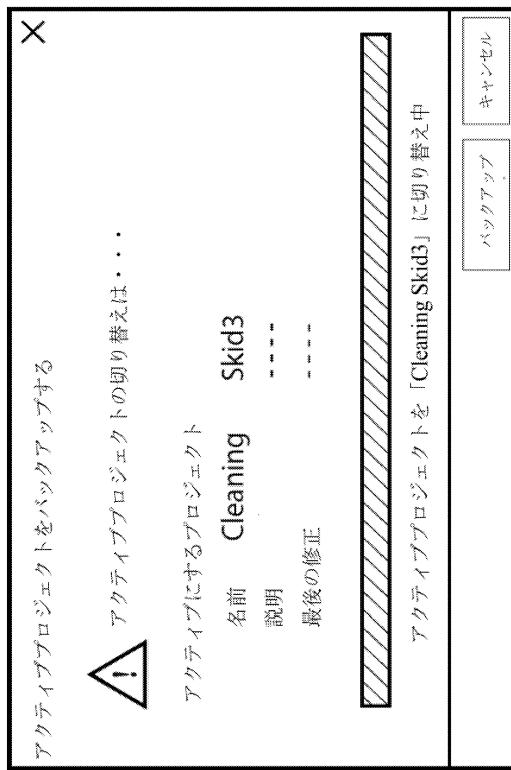
【図 15 A】



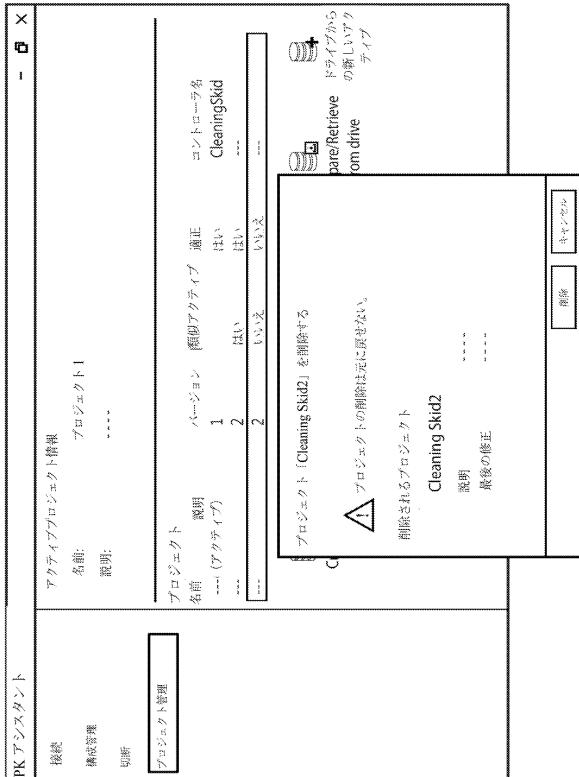
【図 15 B】



【図 15 C】



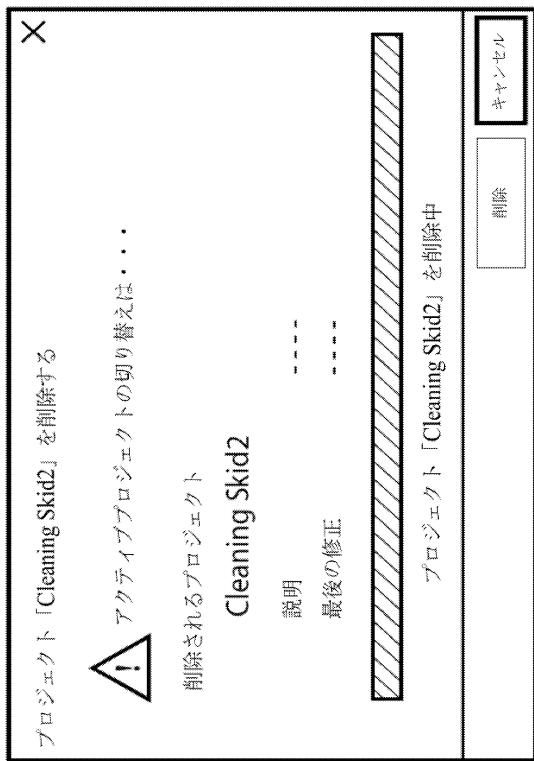
【図 16 A】



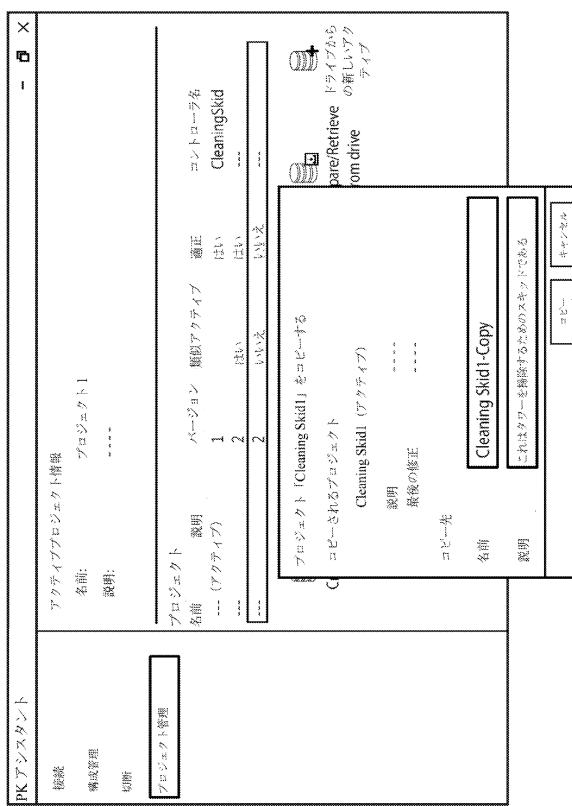
【図 16 B】



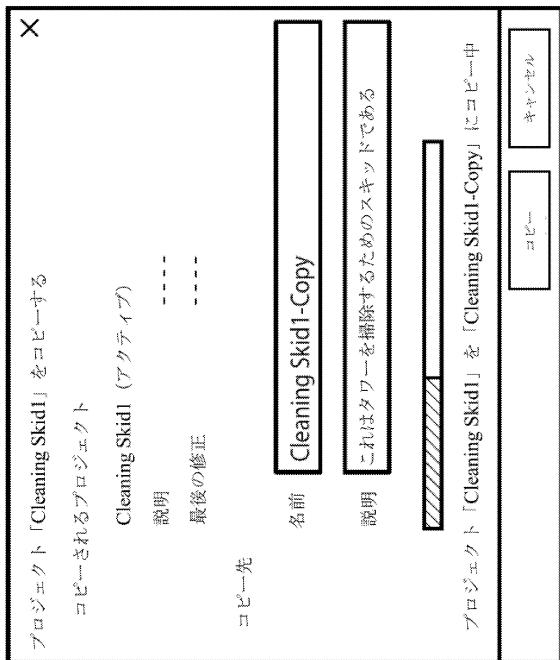
【図 16 C】



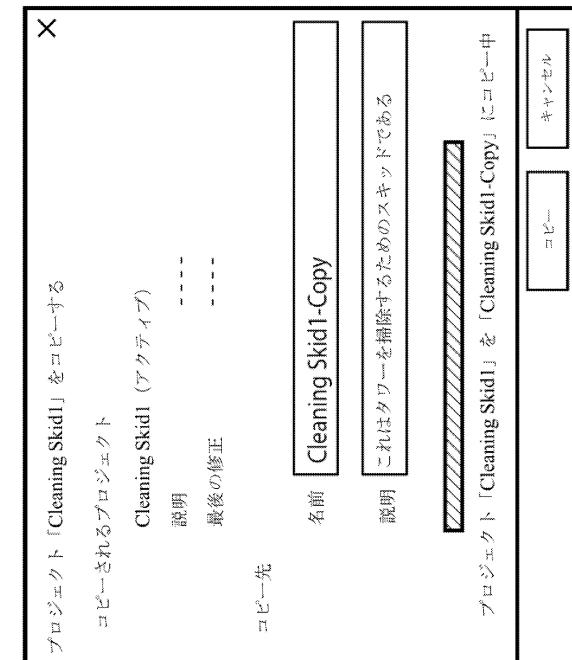
【図 17 A】



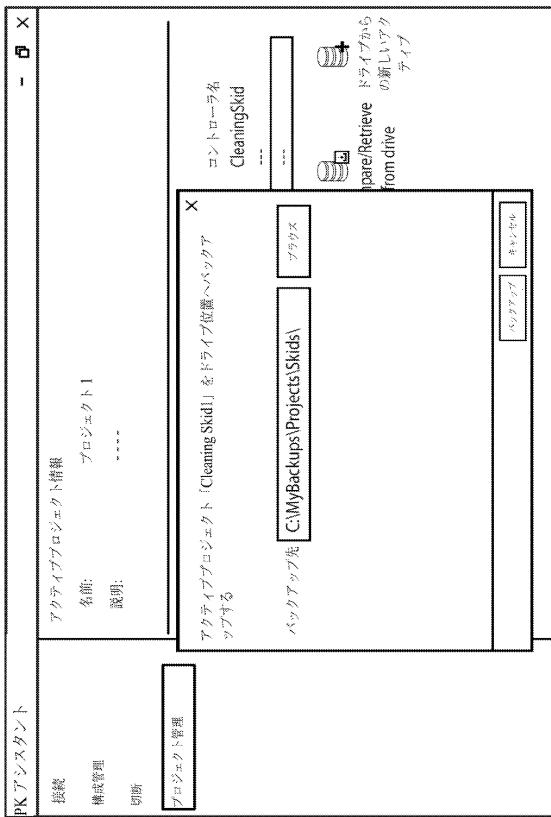
【図 17 B】



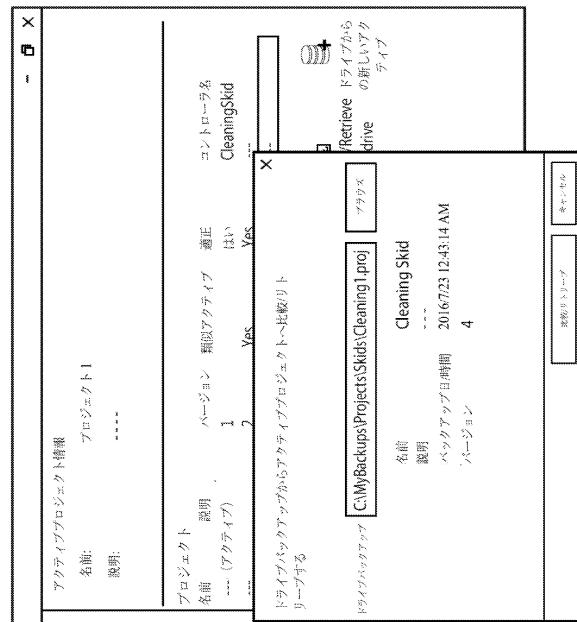
【図 17 C】



【図 18】



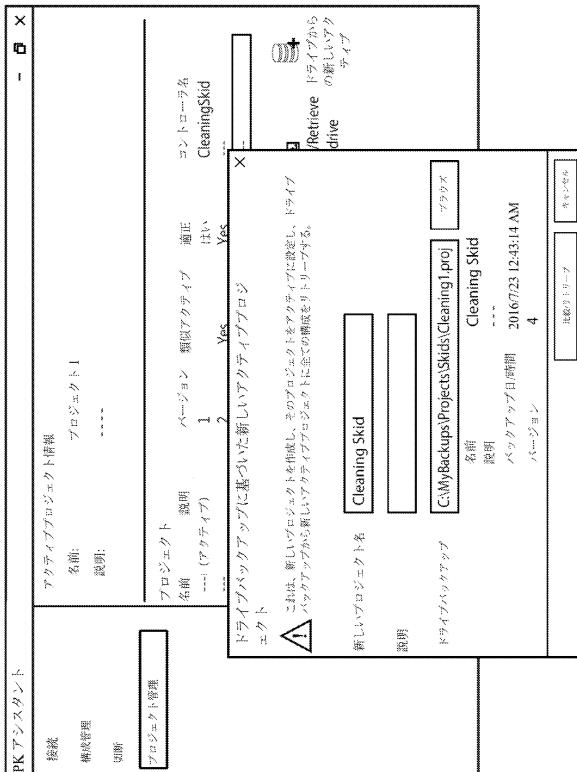
【図 19】



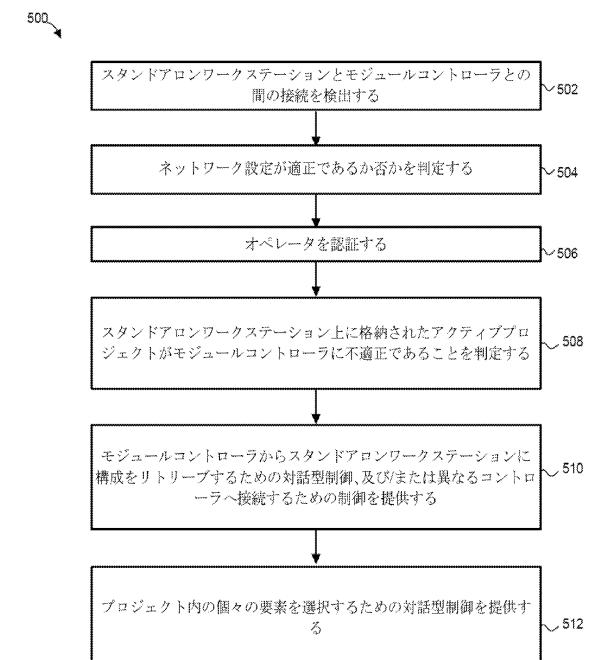
10

20

【図 20】



【図 21】



30

40

50

フロントページの続き

アメリカ合衆国 テキサス 78664 ラウンド ロック キャピラノ コート 1003

(72)発明者 マイケル ジー. オット

アメリカ合衆国 テキサス 78681 ラウンド ロック ウエスト ルイス ヘナ ブルバード 11
00 ビルディング 1

(72)発明者 ジュリアン ケー. ナイドー

アメリカ合衆国 テキサス 78613 セダー パーク フォールン オークス ドライブ 602

(72)発明者 デボラ アール. コルクレイジャー

アメリカ合衆国 テキサス 78664 ラウンド ロック イースト ナコーマ 304

(72)発明者 カレン ジョンソン

アメリカ合衆国 テキサス 78681 ラウンド ロック ウエスト ルイス ヘナ ブルバード 11
00 ビルディング 1

審査官 影山 直洋

(56)参考文献 特開平05-080813(JP,A)

特開2015-022511(JP,A)

特開2012-084162(JP,A)

(58)調査した分野 (Int.Cl., DB名)

G 05 B 23 / 02