

(19)日本国特許庁(JP)

(12)特許公報(B2)

(11)特許番号
特許第7102216号
(P7102216)

(45)発行日 令和4年7月19日(2022.7.19)

(24)登録日 令和4年7月8日(2022.7.8)

(51)国際特許分類

G 0 6 F	3/12 (2006.01)	G 0 6 F	3/12	3 0 5
H 0 4 N	1/00 (2006.01)	G 0 6 F	3/12	3 3 8
		G 0 6 F	3/12	3 6 7
		H 0 4 N	1/00	3 5 0
		H 0 4 N	1/00	1 2 7 B

請求項の数 12 (全23頁)

(21)出願番号 特願2018-89181(P2018-89181)
 (22)出願日 平成30年5月7日(2018.5.7)
 (65)公開番号 特開2019-197249(P2019-197249)
 A)
 (43)公開日 令和1年11月14日(2019.11.14)
 審査請求日 令和3年4月30日(2021.4.30)

(73)特許権者 000001007
 キヤノン株式会社
 東京都大田区下丸子3丁目30番2号
 (74)代理人 100114775
 弁理士 高岡 亮一
 (74)代理人 100121511
 弁理士 小田 直
 (72)発明者 斎藤 和之
 東京都大田区下丸子3丁目30番2号
 キヤノン株式会社内
 鈴木 慎也
 東京都大田区下丸子3丁目30番2号
 キヤノン株式会社内
 審査官 佐賀野 秀一

最終頁に続く

(54)【発明の名称】 プログラム

(57)【特許請求の範囲】

【請求項1】

プリンターのためのプリンタードライバーとは異なり、前記プリンタードライバーに含まれないアプリケーションであり、前記プリンタードライバーを記憶するコンピュータに方法を実行させるアプリケーションであって、
 前記方法は、

印刷設定画面を表示するための第1のオブジェクトと、印刷のための第2のオブジェクトと、印刷のために使われるプリンターを示す第3のオブジェクトを含む画面の前記第1のオブジェクトのユーザによる選択を前記コンピュータが受け付けたことに基づいて前記印刷設定画面を表示手段に表示させる第1の表示工程と、

前記画面の前記第2のオブジェクトのユーザによる選択を前記コンピュータが受け付けたことに基づいて、前記印刷設定画面を介して行われた印刷設定が、印刷時にパスワードをユーザに要求する設定を含むか判定する判定工程と、

前記判定工程で前記印刷設定が前記設定を含むと判定したことに基づいて、前記パスワードを前記ユーザが入力できるようにするパスワード入力画面を前記表示手段に表示させる第2の表示工程とを有し、

前記判定工程で前記印刷設定が前記設定を含まないと判定したことに基づいて、前記パスワード入力画面は表示されず、前記プリンタードライバーを介して前記プリンターに前記印刷のためのデータが提供される

ことを特徴とするアプリケーション。

【請求項 2】

前記第2のオブジェクトは、印刷ボタンであることを特徴とする請求項1に記載のアプリケーション。

【請求項 3】

前記印刷設定画面は、印刷のための前記パスワードの入力をユーザに要求する前記設定をユーザが行うことを可能にする画面であることを特徴とする請求項2に記載のアプリケーション。

【請求項 4】

前記プリンターの操作部を介して前記ユーザから、前記パスワード入力画面で入力された前記パスワードと同じパスワードを前記プリンターが受け付けたことに基づいて前記印刷は開始されることを特徴とする請求項1乃至3のいずれか1項に記載のアプリケーション。

10

【請求項 5】

前記設定は、さらに前記ユーザが前記プリンターの操作部で印刷のための前記パスワードを入力することを要求する設定であることを特徴とする請求項1乃至4のいずれか1項に記載のアプリケーション。

【請求項 6】

前記印刷設定が前記設定を含むが、前記パスワードが印刷設定に含まれないと判定された場合に、前記パスワード入力画面は表示されることを特徴とする請求項4に記載のアプリケーション。

20

【請求項 7】

プリンターのためのプリンタードライバーとは異なり、前記プリンタードライバーに含まれないアプリケーションであり、前記プリンタードライバーを記憶する情報処理装置に方法を実行させるアプリケーションがインストールされた前記情報処理装置であって、前記アプリケーションを実行することにより、印刷設定画面を表示するための第1のオブジェクトと、印刷のための第2のオブジェクトと、印刷のために使われるプリンターを示す第3のオブジェクトを含む画面の前記第1のオブジェクトのユーザによる選択を前記情報処理装置が受け付けたことに基づいて前記印刷設定画面を前記情報処理装置の表示手段に表示し、前記画面の前記第2のオブジェクトのユーザによる選択を前記情報処理装置が受け付けたことに基づいて、前記印刷設定画面を介して行われた印刷設定が、印刷時にパスワードをユーザに要求する設定を含むか判定し、

前記印刷設定が前記設定を含むと判定したことにに基づいて、前記パスワードを前記ユーザが入力できるようにするパスワード入力画面を前記表示手段に表示し、前記印刷設定が前記設定を含まないと判定したことにに基づいて、前記パスワード入力画面は表示せず、前記プリンタードライバーを介して前記プリンターに前記印刷のためのデータを提供することを特徴とする情報処理装置。

30

【請求項 8】

前記第2のオブジェクトは、印刷ボタンであることを特徴とする請求項7に記載の情報処理装置。

40

【請求項 9】

前記印刷設定画面は、印刷のための前記パスワードの入力をユーザに要求する前記設定をユーザが行うことを可能にする画面であることを特徴とする請求項8に記載の情報処理装置。

【請求項 10】

前記プリンターの操作部を介して前記ユーザから、前記パスワード入力画面で入力された前記パスワードと同じパスワードを前記プリンターが受け付けたことに基づいて前記印刷は開始されることを特徴とする請求項7乃至9のいずれか1項に記載の情報処理装置。

50

【請求項 1 1】

前記設定は、さらに前記ユーザが前記プリンタの操作部で印刷のための前記パスワードを入力することを要求する設定である

ことを特徴とする請求項7乃至10のいずれか1項に記載の情報処理装置。

【請求項 1 2】

前記印刷設定が前記設定を含むが、前記パスワードが印刷設定に含まれないと判定した場合に、前記パスワード入力画面を表示する

ことを特徴とする請求項10に記載の情報処理装置。

【発明の詳細な説明】**【技術分野】**

10

【0001】

本発明は、印刷に関するG U I (G r a p h i c a l U s e r I n t e r f a c e)を提供するプログラムに関する。

【背景技術】**【0002】**

画像形成装置の制御用ソフトウェアとして、ホストコンピュータにインストールされたプリンタドライバを利用し、ホストコンピュータに接続された画像形成装置に対して印刷を行う構成が一般に知られる。ホストコンピュータには、基本ソフトウェアであるオペレーティングシステム（以下、OSと呼ぶ）がインストールされており、プリンタドライバはそのOSの規定する仕様に従って構成される。画像形成装置を提供するベンダは、OSの仕様に適合するプリンタドライバを提供することにより、画像形成装置に対して印刷指示する手段を得ることができる。

20

【0003】

ホストコンピュータのOSとしてMicrosoft（登録商標）のWindows（登録商標）8より前のバージョンを利用する場合、プリンタドライバは、V3プリンタドライバと呼ばれるアーキテクチャで構成される。V3プリンタドライバは、ユーザからの印刷要求が発生したタイミングでUI（User Interface）を表示し、何らかの操作をユーザに促すような機能を提供することができる。例えば、印刷要求が発生すると、V3プリンタドライバの構成物であるコンフィグレーションモジュールの規定のイベントが呼び出されるため、コンフィグレーションモジュールは、そのイベント処理の中でユーザインターフェースを表示することが可能である。

30

【0004】

一方、Windows（登録商標）8以降において、V4プリンタドライバと呼ばれる新しいアーキテクチャが登場した。V4プリンタドライバのアーキテクチャでは、セキュリティ性が重視されているため、プリンタドライバそのもののカスタマイズ性は低下している。例えば、前述のコンフィグレーションモジュールはOSにより提供され、ベンダは、その動作をカスタマイズする設定ファイルやスクリプトファイルのみを提供可能である。このようなカスタマイズ性の低下を補うため、画像形成装置を提供するベンダは、プリンタドライバの機能を補助するための専用のアプリケーションを提供することができる。このアプリケーションはWindows Store Device Apps（WSDA）と呼ばれる。

40

【0005】

また、Windows（登録商標）10以降においては、Universal Windows Platform（UWP）と呼ばれる新しい形式のプラットフォームでアプリケーションが作成できる。そのアプリケーションは、UWP Device Appsと呼ばれる。UWP Device AppsはWSDAの機能を拡張し、Workflow（WF）と呼ばれるアプリケーションをWSDAに追加することができる。

【0006】

WF機能では、印刷時にバックグラウンドタスクと呼ばれるUIを持たないスレッドがOSによって生成される。バックグラウンドタスクは、フォアグラウンドのタスク（UI）

50

を呼び出すことも可能である。また、画像形成装置は、セキュアプリント（認証プリントとも呼ぶ）という認証機能を利用した機能を備える。本機能を利用する場合、プリンタドライバは、パスワードのような認証情報を付加した印刷ジョブを画像形成装置に送信する。画像形成装置は、上記認証情報を付加した印刷ジョブを受信し、ユーザが画像形成装置上で認証することにより出力が可能となる。本機能を利用すると、印刷指示を行った際には出力が開始されず、画像形成装置で本人であることが認証されない限り出力されないため、だれにも印刷物を見られずに安全に出力することが可能となる。

【先行技術文献】

【特許文献】

【0007】

【文献】特開2015-166918号公報

10

【発明の概要】

【発明が解決しようとする課題】

【0008】

V4プリンタドライバを使用する印刷では、セキュリティ性や利便性が向上している一方、印刷要求を行った後は、OSが提供するフレームワークが主体となって制御を行うことになる。従って、従来は、V4プリンタドライバを使用する印刷において、印刷実行指示後に、ユーザに対してUIなどの通知を行うことができなかった。そのため、パスワードの認証によって本人確認を行うセキュアプリント機能を選択したにも関わらず、パスワード入力漏れがあった場合に、ポップアップUIを表示し、ユーザに対して入力を促すことができなかった。特許文献1のように、印刷データへの変換時にポップアップ表示させようとする提案も出ている。しかしながら、印刷データへの変換を担うXPSフィルタやV4プリンタドライバが、プロセス間通信を行って、制御の調停を行うなどの拡張を行う必要があり、煩雑であった。また、特殊な拡張によりポップアップを実現するため、依然として機能制限があった。

20

【0009】

本発明は、上記課題を鑑みて、印刷のための第2のオブジェクトが選択された後に、パスワードの入力漏れを抑制するプリンタードライバーとは異なるアプリケーションを提供することを目的の1つとする。

【課題を解決するための手段】

30

【0010】

上記の課題を解決するために、本発明のアプリケーションは、プリンターのためのプリンタードライバーとは異なり、前記プリンタードライバーに含まれないアプリケーションであり、前記プリンタードライバーを記憶するコンピュータに方法を実行させるアプリケーションであって、前記方法は、印刷設定画面を表示するための第1のオブジェクトと、印刷のための第2のオブジェクトと、印刷のために使われるプリンターを示す第3のオブジェクトを含む画面の前記第1のオブジェクトのユーザによる選択を前記コンピュータが受け付けたことに基づいて前記印刷設定画面を表示手段に表示させる第1の表示工程と、前記画面の前記第2のオブジェクトのユーザによる選択を前記コンピュータが受け付けたことに基づいて、前記印刷設定画面を介して行われた印刷設定が、印刷時にパスワードをユーザに要求する設定を含むか判定する判定工程と、前記判定工程で前記印刷設定が前記設定を含むと判定したことに基づいて、前記パスワードを前記ユーザが入力できるようにするパスワード入力画面を前記表示手段に表示させる第2の表示工程と、を有する。前記判定工程で前記印刷設定が前記設定を含まないと判定したことに基づいて、前記パスワード入力画面は表示されず、前記プリンタードライバーを介して前記プリンターに前記印刷のためのデータが提供されることを特徴とする。

40

【発明の効果】

【0011】

本願請求項1に係る発明によれば、印刷のための第2のオブジェクトが選択された後に、パスワードの入力漏れを抑制するプリンタードライバーとは異なるアプリケーションを提

50

供することができる。

【図面の簡単な説明】

【0012】

【図1】システム構成を示す図である。

【図2】クライアントコンピュータのハードウェア構成の一例を示す図である。

【図3】クライアントコンピュータのソフトウェア構成の一例を示す図である。

【図4】メタデータの記述例を示す図である。

【図5】印刷システムで構築される印刷環境の一例を示すプロック図である。

【図6】第1の実施形態に係る印刷処理の一例を示すフローチャートである。

【図7】第1実施形態に係る印刷時起動処理の一例を示すフローチャートである。 10

【図8】第1実施形態に係る印刷時起動処理の一例を示すフローチャートである。

【図9】印刷ダイアログ及び警告用印刷時起動UI部の画面の一例を示す図である。

【図10】タッチアプリ及び印刷ダイアログの模式図である。

【図11】プリントケーバリティの一例を示す図である。

【図12】プリントチケットの一例を示す図である。

【図13】デバイス機能テキストファイルにおける定義項目の記載例を示す図である。

【図14】印刷時のデータフローを説明する模式図である。

【図15】ポップアップ表示を行うかを選択するためのUIの画面の一例を示す図である。

【図16】第2実施形態に係る印刷時起動処理の一例を示すフローチャートである。

【図17】第2実施形態に係る印刷時起動処理の一例を示すフローチャートである。 20

【図18】第3実施形態に係る印刷時起動処理の一例を示すフローチャートである。

【図19】第3実施形態に係る印刷時起動処理の一例を示すフローチャートである。

【図20】第1実施形態に係る印刷設定処理を示すフローチャートである。

【発明を実施するための形態】

【0013】

以下、本発明を実施するための最良の形態について図面及び表を用いて説明する。以下の実施の形態は特許請求の範囲に係る発明を限定するものでなく、また実施の形態で説明されている特徴の組み合わせの全てが発明の解決手段に必須のものとは限らない。

【0014】

(第1実施形態)

30

まず、図1を用いて、本実施形態に係るシステムの構成について説明する。図1は、本実施形態に係るシステム構成の概略を表す模式図を示す。本実施形態に係るシステムは、クライアントコンピュータ101と、サーバー104と、プリンタ102を備える。クライアントコンピュータ101は、情報処理装置の一例である。サーバー104は、クライアントコンピュータ101へ対してデータを配信する。プリンタ102は、ページ記述言語 (Page Description Language、以下PDLと呼ぶ) 形式の印刷データを受信し印刷を行う。各装置は、Wide Area Network (WAN) を含むネットワーク103を通じて相互に通信可能である。なお、プリンタ102は、印刷機能のみを備えるシングルファンクションプリンタであってもよく、印刷機能、スキャン機能、複写機能を備えるマルチファンクションプリンタであってもよい。また、クライアントコンピュータ101は、ネットワーク103に対し一台に限らず、複数台接続されていてもよい。

【0015】

図2は、図1のクライアントコンピュータ101のハードウェア構成を示すプロック図である。クライアントコンピュータ101において、中央演算処理装置(CPU201)は、ランダムアクセスメモリ(RAM202)に記憶されているプログラムに従って、システムバスに接続される各デバイスを制御する。なお、クライアントコンピュータ101は、少なくとも1つのCPU201を備えるものとする。加えて、CPU201が外部メモリ209に記憶されているプログラムに基づき処理を実行することによって、クライアントコンピュータ101のソフトウェア構成及び後述するフローチャートの各ステップの処理を実行する。

40

50

理が実現される。RAM 202は、CPU 201の主メモリ、ワークエリア等として機能する。リードオンリーメモリ(ROM)203は、ブートプログラムや基本入出力システム(BIOS)等の各種プログラムが書き込まれてある。

【0016】

操作入力装置インターフェース(I/F)204は、キーボードやポインティングデバイス(マウス)タッチUI(User Interface)などの操作入力装置205を制御するインターフェースである。操作入力装置205は、ユーザからの操作を受け付ける受付部として機能する。ディスプレイI/F 206はディスプレイ207への画面の表示制御を行う。ディスプレイ207は、ユーザに対して情報を表示する表示部として機能する。外部メモリI/F 208は、例えばハードディスク(HD)、ソリッドステートディスク(SSD)等の外部メモリ209とのアクセスを制御する。外部メモリ209は、オペレーティングシステムプログラム(OS)210をはじめ、アプリケーション実行環境213、タッチパネル、デスクトップそれぞれに対応した各種アプリケーション212や各種ファイル等を記憶する。そして、クライアントコンピュータ101が読み取り可能な記憶媒体として機能する。ネットワークI/F 215は、ネットワーク103を介してプリンタ102に接続され通信制御処理を行う。尚、サーバー104もクライアントコンピュータ101のハードウェア構成と同じ構成を内包しており、同様に動作する。

10

【0017】

次に、図3を用いて、クライアントコンピュータ101のソフトウェアの構成について説明する。まず、クライアントコンピュータ101内の動作ソフトウェアについて説明する。クライアントコンピュータ101内のソフトウェアは、OS 210、アプリケーション実行環境213とアプリケーション212の3つの層から構成される。

20

【0018】

まず、OS 210を構成する個々の要素について説明する。OS 210は、大きくプリントティングシステム337、プリンタドライバ211、外部設定ファイル群352とスレッドプール355で構成される。次に、OS 210の外部設定ファイル群352を構成する要素について説明する。外部設定ファイル群352のメタデータ320は、プリント拡張アプリ300とプリントキュー308の双方の識別子を持つ定義情報であり各々を関連付ける役割を持つ。メタデータ320は、図4を用いて後述する。レジストリ353は、OS 210の設定群が記述されている領域である。レジストリ353には、後述する印刷時起動アプリランチャ350がバックグラウンドタスクである印刷時起動アプリケーション305を起動するかどうかの設定を、登録することができる。バックグラウンドタスクとは、UWP Apps((Universal Windows Platform)アプリケーション)で利用可能な機能である。バックグラウンドタスクは、主に、アプリをユーザが操作しない状態であっても、最新の状態を維持したり、更新情報を受信したりといった処理を実現するために用いられる。一方、アプリがバックグラウンドで動作すると、バッテリーや演算リソースを消耗したりする。そのためOS 210はUWP Appsごとにバックグラウンドタスクの利用を許可するか否かを設定する機能をユーザに対して提供する。レジストリ353には、当該機能を介して設定された、アプリケーションごとにバックグラウンドタスクの利用を許可するか否かの設定が登録されている。

30

【0019】

続いて、プリントティングシステム337を構成する要素について説明する。プリントキュー308は、ネットワーク103内に存在するプリンタを使用して印刷を行う際に、印刷ジョブを一時的に保存する領域である。本実施形態では、プリンタ102のプリントキュー308が作成されているものとする。プリントキュー308が作成されると、デバイス機能テキストファイル332に記載されている一部の機能に基づきプリントキューに紐づく記憶領域340が生成される。記憶領域340はプリントキューごとに個別の設定や機能を記憶する。これは、デバイス機能テキストファイル332の機能定義部に対して、特定の記述を行うことにより、ジョブ毎に切り替える印刷設定機能ではなく、プリントキュー毎に制御可能な機能を提供することができる。

40

50

【0020】

スプーラ325は、アプリケーション212から印刷された印刷対象データであるXPS (XML Paper Specification) ファイルを一時的に保存及び管理するモジュールである。スプーラ325に保存されたXPSファイルは、後述するフィルタパイプラインマネージャ335を通して、PDLに変換された後に、スプーラ325を通じてプリンタ102へと送信される。

【0021】

フィルタパイプラインマネージャ335は、OS210の一部として構成されており、フィルタ330は、後述するプリンタドライバ211を構成する要素の一つである。フィルタパイプラインマネージャ335は、フィルタコンフィグ336の定義に基づき、フィルタ330を1つ乃至複数読み込み、スプーラ325に保存されているXPSファイルをPDLへと変換する。

10

【0022】

コンフィグレーションモジュール331は、プリントチケットの生成、変更、及び不正な設定を訂正する禁則機能などを持つモジュールである。コンフィグレーションモジュール331のAPI (Application Programming Interface) を呼び出すことで、印刷設定情報であるプリントチケットや、デバイス能力情報であるプリントケーパビリティを取得することができる。プリントチケットや、プリントケーパビリティは、XML (Extensible Markup Language) 形式で記述される。

20

【0023】

後述するプリント拡張アプリ300の詳細印刷設定UI部304、印刷時起動アプリケーション305、印刷時起動UI部303は、選択されたプリンタドライバ211のコンフィグレーションモジュール331のAPIを呼び出す。これにより、プリントチケットやプリントケーパビリティを取得する。印刷時起動アプリランチャ350は、アプリケーション212から印刷が行われた際に、レジストリ353に保存された設定に基づき、印刷時起動アプリケーション305を起動する。

【0024】

プリンタドライバ211は、フィルタ330、フィルタコンフィグ336、スクリプトファイル333、印刷設定UI360、デバイス機能テキストファイル332、で構成される。フィルタ330は、フィルタパイプラインマネージャ335によって呼び出されるモジュールであり、XPSファイルをPDLへと変換する役割を持っている。フィルタコンフィグ336は、フィルタパイプラインマネージャ335参照する定義ファイルであり、フィルタ330を呼び出す順序が記述されている。スクリプトファイル333とデバイスの機能が記述されたデバイス機能テキストファイル332は、コンフィグレーションモジュール331の動作をカスタマイズする。印刷設定UI360は、後述するデスクトップアプリ実行環境323で動作する設定変更機能を提供するUIであり、プリントキュー308に紐づいた印刷設定であるプリントチケットを編集する。また、プリントキュー308に紐づく記憶領域340の編集機能を有することも可能である。スレッドプール355は、各種アプリケーション212やOS210内のモジュール群が実行される際に利用されるスレッドを溜め置いておくモジュールである。

30

40

【0025】

次に、アプリケーション実行環境213を構成する個々の要素について説明する。アプリケーション実行環境213は、デスクトップアプリ実行環境323及びタッチアプリ実行環境301から構成される。タッチアプリ実行環境301は、後述するアプリケーション212の一種であるタッチアプリ321やプリント拡張アプリ300を実行するためのAPI群を有し、それらを実行する際のメモリ管理なども行う。一方で、デスクトップアプリ実行環境323は、後述するアプリケーション212の一種であるデスクトップアプリ322を実行するためAPI群を有し、それらを実行する際のメモリ管理などを行う。

【0026】

50

次に、アプリケーション 212 を構成する個々の要素について説明する。アプリケーション 212 は、タッチアプリ 321、プリント拡張アプリ 300 ならびにデスクトップアプリ 322 の三種類のアプリケーションで構成される。タッチアプリ 321 は、アプリケーション 212 の一種で、タッチアプリ実行環境 301 の上で動作する。タッチアプリ 321 は、インターネット上で公開されるアプリケーション配布システムを通じて、ユーザ自身が選択し、クライアントコンピュータ 101 にインストールする。プリント拡張アプリ 300 は、タッチアプリ 321 の一種であり、タッチアプリ 321 と同様にタッチアプリ実行環境 301 の上で動作する。プリント拡張アプリ 300 は、OS 210 がアプリケーション配布システムからインターネットを経由して取得しクライアントコンピュータ 101 にインストールする。プリント拡張アプリ 300 は、タイルアプリ部 302、詳細印刷設定 UI 部 304、印刷時起動アプリケーション 305、印刷時起動 UI 部 303 で構成される。

【0027】

タイルアプリ部 302 は、プリント拡張アプリ 300 を起動するというユーザ操作をきっかけにして、タイル形式の UI (例えば、G U I (G r a p h i c a l U s e r I n t e r f a c e) など) を表示する。詳細印刷設定 UI 部 304 は、OS 210 によって発行された詳細印刷設定イベントを受信したことをきっかけに、詳細印刷設定用の UI を表示する。詳細印刷設定 UI 部 304 は、コンフィグレーションモジュール 331 から A P I を通じて取得したプリントチケットとプリントケーパビリティに基づき、詳細印刷設定用の UI を生成する。詳細印刷設定 UI 部 304 は、ユーザの入力に応じてプリントチケットの値を変更した後に、プリントチケットをコンフィグレーションモジュール 331 に返却する。印刷時起動アプリケーション 305 及び印刷時起動 UI 部 303 は、所定のユーザ操作などを契機としてプリントに関する処理を実行させる W o r k f l o w (W F) と呼ばれる機能を提供する。印刷時起動アプリケーション 305 は、印刷イベントが発行された際に、印刷時起動アプリランチャ 350 によって起動されるバックグラウンドタスクを実行するためのコンポーネントである。印刷時起動アプリケーション 305 は、コンフィグレーションモジュール 331 の A P I を通じてプリントチケットの取得やその内容の読み書きをする。印刷時起動アプリケーション 305 は、プリントチケットの設定やパラメータから、フォアグラウンドタスクを実行するためのコンポーネントである印刷時起動 UI 部 303 を起動するか否かを判断し、印刷時起動 UI 部 303 の起動を OS 210 に要求する。

【0028】

印刷時起動アプリケーション 305 は、印刷時起動 UI 部 303 を起動しないと判断した場合は、印刷時起動 UI 部 303 を起動することなく後述の印刷処理に処理を進める。また、印刷時起動アプリケーション 305 は、アプリストア 334 を介して、印刷時起動 UI 部 303 とプリントチケットの設定やパラメータの受け渡しをする。印刷時起動 UI 部 303 は、プリントチケットやプリントケーパビリティの取得と設定の変更を行う。本実施形態において、印刷時起動 UI 部 303 は、印刷に関する確認を行うための UI や、未入力の設定をユーザに入力させるための UIなどを表示する。ここで、デバイスであるプリンタ 102 とプリント拡張アプリ 300 は、メタデータ 320 によって関連づけられている。

【0029】

図 4 は、メタデータ 320 の記述の一例である。図 4 (A) は、プリント拡張アプリ 300 に関連付けするデバイス情報一覧を示す記述の一例である。メタデータ 320 には、プリンタ 102 の一覧を示す H a r d w a r e I D L i s t タグ 402 が記述される。なお、H a r d w a r e I D とは、プリンタから返却される、プリンタを一意に特定するための識別情報である。項目 403 は、M F P 1 _ a b c d という H a r d w a r e I D を持つプリンタ 102 を表し、同様に、項目 404 は、M F P 2 _ i j k l という H a r d w a r e I D を持つプリンタ 102 を表している。このように、1 つのメタデータ 320 に、プリント拡張アプリ 300 に関連付けするプリンタ 102 を複数列挙することが

できる。本実施形態では、一例としてメタデータ320に、2つのプリンタのHardware IDが指定されている場合を例示しているがこれに限定されるものではない。1つであっても、3つ以上であっても構わない。

【0030】

図4(B)のメタデータ320内には、プリント拡張アプリ300を示すDevice Companion Applicationsタグ411が記述されている。項目412は、プリント拡張アプリ300のIDとベンダのIDとが記述されている。本実施形態では、1234abcd.MFPUtilityという値がプリント拡張アプリ300を表している。そして、12345678-abcd-efgh-ijkl-123456789012という値が、プリント拡張アプリ300を提供しているプリンタベンダを表している。

10

【0031】

このように、図4(A)及び(B)のメタデータによって、デバイスであるプリンタ102とプリント拡張アプリ300の関連付けがなされる。クライアントコンピュータ101にプリンタ102が接続されると、OS210は、所定のサーバー(Microsoft(登録商標)が管理するメタデータ用サーバー)に登録されているメタデータ320を検索する。そして、Hardware IDが記述されたメタデータ320を見つけると、それをクライアントコンピュータ101がインストールされている外部メモリ209に保存する。さらに、OS210は、保存したメタデータ320のDevice Companion Applicationタグ411に記述されたプリント拡張アプリ300を図示省略のアプリケーション配布システムから取得する。そして、外部メモリ209にプリント拡張アプリ300をインストールし、プリンタ102と関連付ける。

20

【0032】

これにより、プリンタ102に対してプリンタドライバ211を経由して印刷要求が発生した際に、プリント拡張アプリ300を起動できるようになる。このメタデータ320は、プリンタベンダにより提供されるものである。プリンタベンダは、このメタデータ320を事前にMicrosoft(登録商標)が管理するメタデータ用サーバーに提供する。それによって、ユーザ環境にプリンタドライバ211がインストールされた際に、ユーザ環境に接続されているデバイスに最適なプリント拡張アプリ300を自動的にインストールするように制御する。

30

【0033】

図5は、メタデータ320で構築される印刷環境の一例を示す図である。プリンタ102であるMFP501は、MFP1_abcdというHardware IDを有する。MFP501をクライアントコンピュータ101に接続すると、OS210は、Hardware ID ListタグにMFP1_abcdというHardware IDが記述されているメタデータ320を取得する。さらに、OS210は、メタデータ320に記述されているMFPUtility506というプリント拡張アプリ300を取得し、MFP501と関連付ける。

40

【0034】

なお、MFP501用にクライアントコンピュータ101にインストールされるプリンタドライバは、複数種類存在し得る。つまり、1つのデバイスであるMFP501に対応するプリンタドライバとして、複数のプリンタドライバが存在しうる。例えば、機種別プリンタドライバ508は、MFP501の専用プリンタドライバである。また、共通プリンタドライバ509は、1つのプリンタドライバで複数種類のプリンタ102への印刷を可能にするプリンタドライバである。なお、機種別プリンタドライバ508と共通プリンタドライバ509は、各ベンダ固有のドライバである。そして、標準プリンタドライバ510は、標準化された印刷方法を利用することで、1つのドライバで複数ベンダのプリンタ102への印刷を可能にするプリンタドライバである。標準プリンタドライバ510は、OS210と共にOS210のパッケージ内の機能の1つとして提供される。

50

【0035】

このように、プリンタ102用に利用可能なプリンタドライバは複数存在し、利用されるプリンタドライバは、インターネット接続状態やOS210の種類、利用環境等に応じて異なる。通常、OS210は、プリンタドライバのバージョン情報等を参照して、利用可能なプリンタドライバのうち、もっとも機能が高く新しいプリンタドライバを自動的に選定しインストールする。なお、プリンタドライバには、Hardware IDや、Comptible IDと呼ばれる装置を特定するための情報が含まれている。クライアントコンピュータ101にプリンタが接続されると、OS210は、プリンタからこれらのID情報を取得し、該当するIDを有するプリンタドライバを探索し、インストールする。

10

【0036】

以上のように、プリンタを提供するプリンタベンダは、メタデータ320を提供することで、それぞれのプリンタに対して、どのプリント拡張アプリ300を提供するか否かを制御することができる。例えば、プリンタベンダは、複数のプリンタに共通した一つのプリント拡張アプリ300を提供することもできるし、それぞれ異なるプリント拡張アプリ300を提供することもできる。しかしながら、1つのプリンタに関連付けられたプリンタドライバ211の種類に応じて異なるプリント拡張アプリ300を提供することはできない。図5においては、MFP501で利用されるプリンタドライバ211は、ドライバ508からドライバ510の3種類存在するが、各ドライバと連携できるプリント拡張アプリ300は常に1種類である。

20

【0037】

また、ユーザー印刷環境では複数台のプリンタ102や、複数種類のプリンタ102を使い分ける場合がある。出力先のプリンタ102毎に、プリンタドライバ211に対応するモデルや印刷設定の初期値および印刷ジョブをそれぞれ管理する必要がある。そのためOS210は、設置の際に出力先のプリンタ102に対応するオブジェクトとしてプリントキュー308を生成する。OS210は、プリントキュー308に紐づけてプリンタドライバ211の構成モジュールや印刷設定の初期値を管理する。

30

【0038】

ここで、プリンタ102がセキュアプリント機能（認証プリント機能）を有していることと、プリント拡張アプリ300がどのようにしてプリンタ102の保有する機能を知るのかについて説明する。プリンタ102は、クライアントコンピュータ101で設定した暗証番号の入力を印刷時に求めるセキュアプリント機能を保有している。プリンタ102が保有する機能は、プリンタドライバ211のデバイス機能テキストファイル332に記載されている。プリント拡張アプリ300は、デバイス機能テキストファイル332に基づいてコンフィグレーションモジュール331が生成したプリントケーパビリティを受け取ることで、プリンタ102がセキュアプリント機能を保有することを把握する。また、プリント拡張アプリ300は、コンフィグレーションモジュール331によって生成されたプリントチケットを読み込むことで、セキュアプリント機能が現在の印刷設定に設定されていることを把握する。

30

【0039】

図11は、デバイス機能テキストファイル332に基づいて、コンフィグレーションモジュール331が生成したプリントケーパビリティの一例である。図11で示されるプリントケーパビリティは、XML形式でプリンタドライバ211の能力一覧が示されている。プリントケーパビリティは、1つのFeatureの要素の中に、複数のOptionを持つ。Featureは、機能を表し、その機能で内包するOption群は選択肢を表している。プリントケーパビリティ1001は、セキュアプリントのFeatureとOptionのXMLでの記述例を示す。FeatureにJobSecurePrintが記述されていることで、セキュアプリント機能が可能であることを示している。また、JobSecurePrintのFeatureの選択肢として、ns0000:SecurePrintすなわち機能を有効にする「オン」と、ns0000:Noneすなわ

40

50

ち機能を無効にする「オフ」の二つが選択肢であることを示している。

【0040】

次に、図12は、ユーザの入力に基づいて、コンフィグレーションモジュール331が生成したプリントチケットの一例である。プリントチケットには、プリントケーパビリティで定義されたXML形式の各機能の設定値のいずれかが、現在の印刷設定として格納される。例えば、項目1101に示すns0000:JobSecurePrintというFeatureの要素は、本人か否かをパスワード入力によって認証するセキュアプリントの機能を示している。JobSecurePrintのFeatureで内包するOptionがns0000:SecurePrintになっていることで、セキュアプリント機能がオンに設定されていることを示している。また、同Optionにns0000:Noneが設定されていた場合は、セキュアプリント機能がオフに設定されていることを意味する。また、もしも同Optionに何も設定されていない場合には、本実施形態における印刷時起動UI部303は、Optionにns0000:Noneが設定されていた場合と同等と判定する。項目1102に示すpsf:ParameterInitには、本人認証のためのパスワードとして入力された値“XXXXXX”が保持される。本図では、印刷時にセキュアプリントがOnと設定され、パスワードが入力され、プリントチケットにパスワードの値が保持されている場合の例を示している。もしも、印刷時にセキュアプリントがOnと設定されたにも関わらず、パスワードが入力されなかった場合は、パスワードの値の箇所は空欄となる。

10

【0041】

次に、図6を用いて、タッチアプリ321から印刷が行われるまでの基本フローを説明する。図6のフローチャートに示す各動作（ステップ）は、CPU201が外部メモリ209に記憶された各モジュール、コンポーネント、アプリケーションを実現するためのプログラムをRAM202に読み出し、実行することにより実現される。各制御モジュール、コンポーネント、アプリケーションによって提供される制御を説明する場合、CPU201によって実現される各部を主語として説明する。また、表示、ユーザ操作の受け付け、及びデータ送受信処理などの一部処理は、CPU201、各I/F及び、I/Fに接続された入出力デバイスが協働して実現するものとする。尚、本実施形態では、タッチアプリ321（図10）を用いて説明を行うが、デスクトップアプリ322からも印刷可能なことは言うまでもない。図10（A）は、タッチアプリ321が起動された模式図である。タッチアプリ321は、メニュー・コンボタン900をタップされることで、メニューの選択肢の一覧を羅列する。次に、メニューの選択肢である印刷アイテム901がユーザ操作によって選択されることで、タッチアプリ321は、OS210に印刷ダイアログ表示の要求を依頼し、印刷ダイアログを表示する。図10（B）は、OS210によって生成される簡易的な印刷設定を行う印刷ダイアログである。印刷ダイアログは、印刷設定部920と印刷イベント発生部930から構成される。印刷設定部920には、簡易的な印刷設定の一覧が表示される。921は、選択されているプリンタキュー308の名称であり、922は、設定されている印刷設定の例である。印刷ダイアログは、表示後にユーザ操作を受け付け、印刷イベント発生部930の印刷ボタン932が押されることで、印刷イベントを発生させる。

20

【0042】

印刷イベント発生後に、タッチアプリ321は、印刷対象のデータのXPSファイルを生成し、図6で説明する印刷時の基本フローを開始する。本実施形態では、ワークフローを使用して、印刷開始後に確認UIや設定の再入力を行うためのUIを表示する仕組みを提供する。具体的な制御方法について、図6に例示する印刷時の基本フローを用いて説明する。尚、各アプリケーション及びモジュールは、外部メモリ209に格納されており、必要に応じてRAM202にロードされ、クライアントコンピュータ101上のCPU201で実行される。図6に示すフローは、アプリケーション212から印刷イベントが発生したところから始まる。印刷イベントを引き起こすアプリケーション212は、タッチアプリ321であっても、デスクトップアプリ322であってもかまわない。

30

40

50

【0043】

まず、ステップS601にて、OS210は、印刷イベントを発生させたアプリケーション212が生成したXPSファイルを、スプーラ325に渡す。次に、ステップS602にて、印刷時起動アプリケーション305の処理が実行される。本処理は、重要な役割を占めるため、後述する。次に、ステップS603にて、フィルタパイプラインマネージャ335は、プリントチケットとXPSファイルを、ステップS602の処理の結果に応じて、スプーラまたは印刷時起動アプリケーション305から取得する。次に、ステップS604にて、フィルタパイプラインマネージャ335は、フィルタコンフィグ336の定義に基づき、フィルタ330を外部メモリ209から読み出し、RAM202に展開する。次に、ステップS605にて、フィルタパイプラインマネージャ335は、XPSファイルをフィルタ330に渡す。ステップS606にて、フィルタ330は、XPSファイルをPDLに変換し、スプーラ325にPDLを渡す。そして、ステップS607にて、スプーラ325は、PDLをプリントタ102に送信する。

10

【0044】

次に、図14を用いて、基本的な印刷フローについて説明する。本フローの説明ではデスクトップアプリ322からの印刷について記載する。印刷機能を持ったデスクトップアプリ322から印刷を開始するために、ユーザーは出力先のプリンタ102と紐づいたプリントキュー308を選択する。その際、対象のプリントキュー308に関連したプリンタドライバ211の印刷設定UI360を開くことにより、様々な機能変更を行うことができる。例えば、一枚の紙に両面印字する設定である「両面設定」や、印字結果をモノクロ印刷する「白黒」などの設定変更が可能となる。ここで、複数の画像形成装置が混在するエンタープライズ環境においては、ポップアップUIを表示するか否かを画像形成装置ごとに使い分けたいといった要望がある。そこで、本実施形態では、画像形成装置に紐づくプリントキュー毎にポップアップ表示を制御できるようにする。具体的には、プリントキューに紐付く記憶領域340に、ポップアップ表示を行うか否かを示す設定を記憶させる。

20

【0045】

続いて当該設定の変更する方法について説明する。ユーザは、当該設定を、プリンタドライバの印刷設定UI360が表示するプロパティ画面を介して変更することができる。図15は、プロパティ画面の一例であり、「印刷時にユーザー設定情報を確認する」の設定項目が選択され、設定値のリストが表示されている場合を例示している。設定値としては、「オン」/「オフ」が用意される。ユーザは、当該リストから設定値を選択することで設定を変更することができる。ユーザによって設定変更の操作がなされたことを検知した印刷設定UI360は、この設定情報に基づき、デバイス機能テキストファイル332の機能定義部に対して、特定の記述を行う。プリントイングシステム337は、当該特定の記述に基づき当該プリンタドライバに対応付けられたプリントキューに紐付く記憶領域340の設定を変更する。

30

【0046】

図13を用いて具体的な記述を説明する。Feature属性に対して*Feature Type: PRINTER_PROPERTY属性(1201)を記載することで、プリントキュー毎の設定を表現する。プロパティ画面で「オン」が設定された場合、ONを示すオプション1203が記述され、オプション1202の記述は消去される。また、プロパティ画面で「オフ」が設定された場合、OFFを示すオプション1202が記述される。プリントイングシステム337は、当該記述を参照することで、記憶領域340の設定を更新する。

40

【0047】

次に、当該変更処理の具体的な制御方法について図20のフローチャートを用いて説明する。図20のフローチャートに示す各動作(ステップ)は、CPU201が外部メモリ209に記憶された各モジュール、コンポーネント、アプリケーションを実現するためのプログラムをRAM202に読み出し、実行することにより実現される。各制御モジュール、コンポーネント、アプリケーションによって提供される制御を説明する場合、CPU2

50

0 1 によって実現される各部を主語として説明する。また、表示、ユーザ操作の受け付け、及びデータ送受信処理などの一部処理は、C P U 2 0 1、各 I / F 及び、I / F に接続された入出力デバイスが協働して実現するものとする。

【 0 0 4 8 】

まず、ステップ S 2 0 0 1 にて、印刷設定 U I 3 6 0 は、プリントキューに対応付けて記憶された設定情報を取得し、図 1 5 で例示した選択肢を含む画面を表示する。次に、ステップ S 2 0 0 2 にて、印刷設定 U I 3 6 0 は、ユーザー操作を受け付ける状態に遷移し、ユーザ入力を受け付ける。当該入力には、セキュアプリント機能を使用するか否かの設定変更などが含まれる。ステップ S 2 0 0 3 にて、印刷設定 U I 3 6 0 は、ユーザー操作により入力が行われたか否かの判定を行い、入力が行われていない場合はステップ S 2 0 0 2 に戻る。一方、入力が行われた場合は、ステップ S 2 0 0 4 へ進む。次に、ステップ S 2 0 0 4 において、印刷設定 U I 3 6 0 は、O S と協働し、ステップ S 2 0 0 2 で入力された設定情報を、プリントキューに紐づく記憶領域 3 4 0 に保存し、一連の処理を終了する。なお、プリントキューに記憶されたセキュアプリント機能を使用するか否かの設定は、後述のフローチャートにて参照される。

10

【 0 0 4 9 】

続けて図 7 ~ 図 9 を用いて、印刷時起動処理について説明する。図 7 及び図 8 のフローチャートに示す各動作（ステップ）は、C P U 2 0 1 が外部メモリ 2 0 9 に記憶された各モジュール、コンポーネント、アプリケーションを実現するためのプログラムを R A M 2 0 2 に読み出し、実行することにより実現される。各制御モジュール、コンポーネント、アプリケーションによって提供される制御を説明する場合、C P U 2 0 1 によって実現される各部を主語として説明する。また、表示、ユーザ操作の受け付け、及びデータ送受信処理などの一部処理は、C P U 2 0 1、各 I / F 及び、I / F に接続された入出力デバイスが協働して実現するものとする。まず、ステップ S 7 0 1 にて、O S 2 1 0 は、レジストリ 3 5 3 の設定を読み込む。次に、ステップ S 7 0 2 にて、レジストリ 3 5 3 に印刷時起動アプリケーション 3 0 5 を起動することが設定されていない場合、O S 2 1 0 は、一連の U I 起動処理をスキップし、処理をステップ S 6 0 3 以降のデータ変換、送信処理に進める。一方、レジストリ 3 5 3 に印刷時起動アプリケーション 3 0 5 を起動するように設定されている場合には、O S 2 0 1 は、処理をステップ S 7 0 3 に進める。なお、レジストリ 3 5 3 にバックグラウンドタスクを使用しないことが設定されている場合も、一連の U I 起動処理をスキップするものとする。

20

【 0 0 5 0 】

次に、ステップ S 7 0 3 にて、O S 2 1 0 は、印刷時起動アプリランチャ 3 5 0 を起動する。その後、ステップ S 7 0 4 にて、印刷時起動アプリランチャ 3 5 0 は、O S 2 1 0 に印刷時起動アプリケーション 3 0 5 をバックグラウンドタスクとして起動するよう要求を出す。すると、タッチアプリ実行環境 3 0 1 がプリント拡張アプリ 3 0 0 を R A M 2 0 2 に展開することで、印刷時起動アプリケーション 3 0 5 が起動される。次に、ステップ S 7 0 5 にて、印刷時起動アプリケーション 3 0 5 は、プリントチケットをコンフィグレーションモジュール 3 3 1 から取得し、プリントチケットの内容を解析し、解析結果からどのような機能設定やパラメータが記述されているかを抽出する。

30

【 0 0 5 1 】

次に、ステップ S 7 1 3 において、印刷時起動アプリケーション 3 0 5 は、印刷に使用する画像形成装置に対応するプリントキュー 3 0 8 に紐づいた記憶領域 3 4 0 の情報を取得する。本実施形態では、ポップアップ表示をするか否かの設定情報を取得し、ステップ S 7 1 4 に進む。ステップ S 7 1 4 において、印刷時起動アプリケーション 3 0 5 は、ステップ S 7 1 3 で取得した設定情報を用いて、ポップアップ表示を行うか否かを判定する。判定結果から、ポップアップを行わない、つまり「オフ」が設定されている場合、O S 2 1 0 は、一連の U I 起動処理をスキップし、処理をステップ S 6 0 3 以降のデータ変換、送信処理に進める。もしポップアップ表示を行う、つまり「オン」が設定されている場合はステップ S 7 0 6 に進む。

40

50

【0052】

そして、ステップS706にて、印刷時起動アプリケーション305は、プリントチケットから抽出した機能に、セキュアプリントの機能の設定があり、かつONと設定されているかどうかを判定する。ステップS706でセキュアプリントが設定されていない場合には、OS210は、一連のUI起動処理をスキップし、処理をステップS603以降のデータ変換、送信処理に進める。ステップS706でセキュアプリント設定があり、かつONと設定されているのであれば、ステップS707にて、印刷時起動アプリケーション305は、OS210に印刷時起動UI部303の表示要求を出す。要求を受け付けたOS210は、ステップS708にて、OS210内のスレッドプール355からパスワード入力用UIスレッドをプリント拡張アプリ300に割り当て、プリント拡張アプリ300の印刷時起動UI部303を生成する。

10

【0053】

そして、印刷時起動UI部303をディスプレイ207に表示する。本実施形態では、ステップS708にて、図9(A)に示すパスワード入力画面が表示される。続けて、ステップS709にて、印刷時起動UI部303は、ユーザ操作を受け付ける。図9(A)のテキストボックス810は、パスワード入力テキストボックスであり、ユーザからのパスワードの入力を受け付ける。ボタン811は、OKボタンであり、クリックなどの操作イベントを受け付けた際に、パスワード入力テキストボックス810で入力されたパスワードを確定させ、プリントチケットに反映する。このOKボタン811は、パスワード入力テキストボックス810の入力が不正の場合や、未入力である場合には、グレーアウトやメッセージを表示することでパスワードの確定を受け付けなくてもよい。ボタン812は、キャンセルボタンであり、クリックなどの操作を受け付けることで、印刷をキャンセルする。

20

【0054】

図8の説明に戻り、ユーザ操作による入力を受け付けると、印刷時起動UI部303は、ウィンドウを閉じ、処理をステップS711に進める。そして、ステップS711にて、印刷時起動アプリケーション305は、印刷時起動UI部303からプリントチケットとXPSファイルを取得する。続けて、ステップS712にて、印刷時起動アプリケーション305は、取得したプリントチケットとXPSファイルをフィルタパイプラインマネージャ335に送信する。

30

【0055】

以上、本実施形態によれば、印刷時起動アプリケーション305は、プリントチケットを解析し、セキュアプリント機能の設定の有効/無効によって印刷時起動UI部の表示を判定する。そのため、セキュアプリント機能の設定が有効になっている場合に、ユーザに強制的に入力を促すような画面を表示することが可能となる。また、本実施形態によれば、画像形成装置に紐づくプリントキュー毎にポップアップ表示を行うか否かを切り替えることができる。従って、ユーザや管理者の意図に沿った画面表示を行うことができるようになる。

【0056】

(第2実施形態)

第1実施形態では、セキュアプリント機能の設定の有効/無効によって印刷時起動UI部の表示を判定することによって、ユーザに強制的に入力を促すような画面を表示する仕組みについて説明した。本実施形態では、パスワードというパラメータの有無をも印刷時起動UI部の表示の判定に利用する仕組みについて説明する。

40

【0057】

本実施形態に係る具体的な処理について、図16及び図17を用いて説明する。尚、各アプリケーション及びモジュールは、外部メモリ209に格納されており、必要に応じてRAM202にロードされ、クライアントコンピュータ101上のCPU201で実行される。また、既に説明済みの番号は、その詳細な説明を省略する。ステップS1201～ステップS1204に示す印刷時起動アプリケーション305の起動処理までは第1実施形

50

態の図 7 のステップ S 7 0 1 ~ ステップ S 7 0 4 と同様の処理であるため、その詳細な説明を省略する。そして、ステップ S 1 2 0 5 にて、印刷時起動アプリケーション 3 0 5 は、プリントチケットをコンフィグレーションモジュール 3 3 1 から取得し、プリントチケットの内容を解析し、どのような機能設定やパラメータが記述されているかを抽出する。

【 0 0 5 8 】

次に、ステップ S 1 2 1 4 において、印刷時起動アプリケーション 3 0 5 は、プリントキュー 3 0 8 に紐づいた記憶領域 3 4 0 の情報を取得する。本実施形態では、ポップアップ表示をするか否かの設定情報を取得し、ステップ S 1 2 1 5 に進む。ステップ S 1 2 1 5 において、印刷時起動アプリケーション 3 0 5 は、ステップ S 1 2 1 4 で取得した設定情報を用いて、ポップアップ表示を行うか否かを判定する。もしポップアップを行わない、つまり「オフ」が設定されている場合、OS 2 1 0 は、一連の U I 起動処理をスキップし、処理をステップ S 6 0 3 以降のデータ変換、送信処理に進める。もしポップアップ表示を行う、つまり「オン」が設定されている場合はステップ S 1 2 0 6 に進む。

【 0 0 5 9 】

そして、ステップ S 1 2 0 6 にて、印刷時起動アプリケーション 3 0 5 は、プリントチケットから抽出した機能に、セキュアプリントの機能の設定があり、かつ ON と設定されているかどうかを判定する。ステップ S 1 2 0 6 でセキュアプリントが設定されていない場合には、OS 2 1 0 は、一連の U I 起動処理をスキップし、処理をステップ S 6 0 3 以降のデータ変換、送信処理に進める。一方、ステップ S 1 2 0 6 でセキュアプリント設定があり、かつ ON と設定されているのであれば、ステップ S 1 2 0 7 にて、プリントチケット内のパスワードの有無を判定する。

【 0 0 6 0 】

そして、ステップ S 1 2 0 7 で印刷時起動アプリケーション 3 0 5 が、プリントチケット内にパスワードが無いと判定した場合は、ステップ S 1 2 0 8 にて、印刷時起動アプリケーション 3 0 5 は、OS 2 1 0 に印刷時起動 U I 部 3 0 3 の表示要求を出す。要求を受け付けた OS 2 1 0 は、ステップ S 1 2 0 9 にて、OS 2 1 0 内のスレッドプール 3 5 5 からパスワード入力用 U I スレッドをプリント拡張アプリ 3 0 0 に割り当て、プリント拡張アプリ 3 0 0 の印刷時起動 U I 部 3 0 3 を生成する。そして、印刷時起動 U I 部 3 0 3 をディスプレイ 2 0 7 に表示する。本実施形態では、ステップ S 1 2 0 9 で図 9 (A) に示すパスワード入力画面が表示される。続けて、ステップ S 1 2 1 0 にて、印刷時起動 U I 部 3 0 3 は、ユーザ操作を受け付ける。

【 0 0 6 1 】

ここで、図 9 (A) のテキストボックス 8 1 0 は、パスワード入力テキストボックスであり、ユーザからのパスワードの入力を受け付ける。OK ボタン 8 1 1 は、OK ボタンであり、クリックなどの操作イベントを受け付けた際に、パスワード入力テキストボックス 8 1 0 で入力されたパスワードを確定させ、プリントチケットに反映する。この OK ボタン 8 1 1 は、パスワード入力テキストボックス 8 1 0 の入力が不正の場合や、未入力である場合には、グレーアウトやメッセージを表示することでパスワードの確定を受け付けなくてもよい。ボタン 8 1 2 は、キャンセルボタンであり、クリックなどの操作を受け付けることで、印刷をキャンセルする。

【 0 0 6 2 】

図 1 7 の説明に戻り、ユーザ操作による入力を受け付けると、印刷時起動 U I 部 3 0 3 は、ウィンドウを閉じ、処理をステップ S 1 2 1 2 に進める。ステップ S 1 2 1 2 にて、印刷時起動アプリケーション 3 0 5 は、印刷時起動 U I 部 3 0 3 からプリントチケットと X P S ファイルを取得する。続けて、ステップ S 1 2 1 3 にて、印刷時起動アプリケーション 3 0 5 は、取得したプリントチケットと X P S ファイルをフィルタバイオペインマネージャ 3 3 5 に送信する。

【 0 0 6 3 】

以上、本実施形態によれば、印刷時起動アプリケーション 3 0 5 は、プリントチケットを解析し、セキュアプリント機能の設定の有効 / 無効だけでなく、パスワードというパラメ

10

20

30

40

50

ータの有無によって印刷時起動UI部の表示を判定する。そのため、セキュアプリント機能のパスワードの入力漏れの場合のみに、ユーザに画面を表示することが可能となる。

【0064】

(第3実施形態)

第2実施形態では、パスワードというパラメータの有無をも印刷時起動UI部の表示の判定に利用する仕組みについて説明した。本実施形態では、パスワードというパラメータの有無を、印刷時起動UI部の表示における画面の種類を動的に変更するためを利用する仕組みについて説明する。本実施形態に係る具体的な処理について、図18及び図19を用いて説明する。尚、各アプリケーション及びモジュールは、外部メモリ209に格納されており、必要に応じてRAM202にロードされ、クライアントコンピュータ101上のCPU201で実行される。また、既に説明済みの番号は、その詳細な説明を省略する。ステップS1301～ステップS1304に示す印刷時起動アプリケーション305の起動処理までは第1実施形態の図7のステップS701～ステップS704と同様の処理であるため、その詳細な説明を省略する。そして、ステップS1305にて、印刷時起動アプリケーション305は、プリントチケットをコンフィグレーションモジュール331から取得し、プリントチケットの内容を解析し、どのような機能設定やパラメータが記述されているかを抽出する。

10

【0065】

次に、ステップS1316において、印刷時起動アプリケーション305は、プリントキューリストに紐づいた記憶領域340の情報を取得する。本実施形態では、ポップアップ表示をするか否かの設定情報を取得し、ステップS1317に進む。ステップS1317において、印刷時起動アプリケーション305は、ステップS1316で取得した設定情報を用いて、ポップアップ表示を行うか否かを判定する。もしポップアップを行わない、つまり「オフ」が設定されている場合、OS210は、一連のUI起動処理をスキップし、処理をステップS603以降のデータ変換、送信処理に進める。もしポップアップ表示を行う、つまり「オン」が設定されている場合はステップS1306に進む。

20

【0066】

そして、ステップS1306にて、印刷時起動アプリケーション305は、プリントチケットから抽出した機能に、セキュアプリントの機能の設定があり、かつONと設定されているかどうかを判定する。ステップS1306でセキュアプリントが設定されていない場合には、OS210は、一連のUI起動処理をスキップし、処理をステップS603以降のデータ変換、送信処理に進める。一方、ステップS1306でセキュアプリント設定があり、かつONと設定されているのであれば、ステップS1307にて、プリントチケット内のパスワードの有無を判定する。

30

【0067】

そして、ステップS1307で印刷時起動アプリケーション305がプリントチケット内にパスワードが無いと判定した場合、ステップS1308にて、印刷時起動アプリケーション305は、OS210に、警告用の印刷時起動UI部303の表示要求を出す。要求を受け付けたOS210は、ステップS1309にて、OS210内のスレッドプール355からパスワード入力用UIスレッドをプリント拡張アプリ300に割り当て、プリント拡張アプリ300の印刷時起動UI部303を警告用画面に生成する。そして、警告用画面に生成した印刷時起動UI部303をディスプレイ207に表示する。本実施形態では、ステップS1309で図9(B)に示す画面が表示される。

40

【0068】

一方、ステップS1307で印刷時起動アプリケーション305がプリントチケット内にパスワードが有ると判定した場合、ステップS1314にて、印刷時起動アプリケーション305は、OS210に、確認用の印刷時起動UI部303の表示要求を出す。要求を受け付けたOS210は、ステップS1315にて、OS210内のスレッドプール355からパスワード入力用UIスレッドをプリント拡張アプリ300に割り当て、プリント拡張アプリ300の印刷時起動UI部303を確認用画面に生成する。そして、確認用画

50

面に生成した印刷時起動 U I 部 3 0 3 をディスプレイ 2 0 7 に表示する。

【 0 0 6 9 】

本実施形態では、ステップ S 1 3 1 5 で図 9 (C) に示す画面が表示される。警告用、もしくは確認用に生成した印刷時起動 U I 部 3 0 3 を表示した後の処理は共通であるため図 9 (B) を用いて説明する。ステップ S 1 3 1 0 にて、印刷時起動 U I 部 3 0 3 は、ユーザ操作を受け付ける。図 9 (B) のテキストボックス 8 2 0 は、パスワード入力テキストボックスであり、ユーザからのパスワードの入力を受け付ける。OK ボタン 8 2 1 は、OK ボタンであり、クリックなどの操作イベントを受け付けた際に、パスワード入力テキストボックス 8 2 0 で入力されたパスワードを確定させ、プリントチケットに反映する。このOK ボタン 8 2 1 は、パスワード入力テキストボックス 8 2 0 の入力が不正の場合や、未入力である場合には、グレーアウトやメッセージを表示することでパスワードの確定を受け付けなくてもよい。キャンセルボタン 8 2 2 は、キャンセルボタンであり、クリックなどの操作を受け付けることで、印刷をキャンセルする。

10

【 0 0 7 0 】

図 1 9 の説明に戻り、ユーザ操作による入力を受け付けると、印刷時起動 U I 部 3 0 3 は、ウィンドウを閉じ、処理をステップ S 1 3 1 2 に進める。ステップ S 1 3 1 2 にて、印刷時起動アプリケーション 3 0 5 は、印刷時起動 U I 部 3 0 3 からプリントチケットと X P S ファイルを取得する。続けて、ステップ S 1 3 1 3 にて、印刷時起動アプリケーション 3 0 5 は、取得したプリントチケットと X P S ファイルをフィルタパイプラインマネージャ 3 3 5 に送信する。

20

【 0 0 7 1 】

以上、本実施形態によれば、印刷時起動アプリケーション 3 0 5 は、プリントチケットを解析し、パスワードというパラメータの有無によって印刷時起動 U I 部 の表示を判定し、かつ表示する画面の種類を動的に変更する。そのため、ユーザにどのような状況が発生しているのかを知らしめることが可能となる。

【 0 0 7 2 】

(その他の実施例)

本発明は、上述の実施形態の 1 以上の機能を実現するプログラムを、ネットワーク又は記憶媒体を介してシステム又は装置に供給し、そのシステム又は装置のコンピューターにおける 1 つ以上のプロセッサーがプログラムを読み出し実行する処理でも実現可能である。また、1 以上の機能を実現する回路 (例えば、A S I C) によっても実現可能である。

30

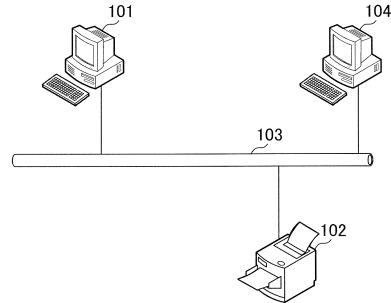
【 0 0 7 3 】

また、本発明の好ましい実施形態について説明したが、本発明は、これらの実施形態に限定されず、その要旨の範囲内で種々の変形および変更が可能である。

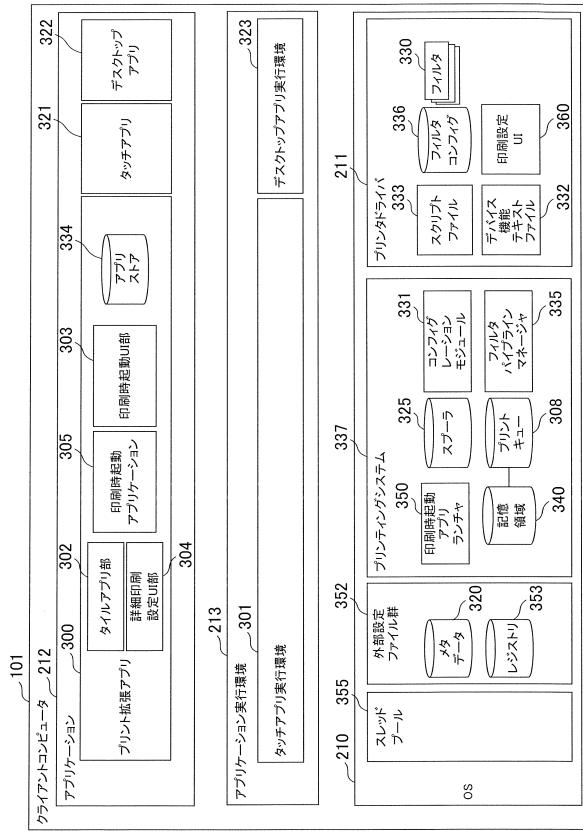
40

50

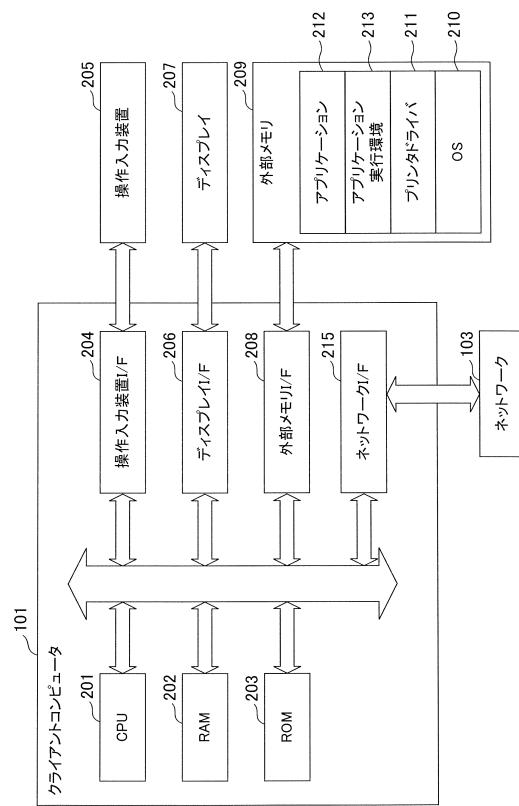
【図面】
【図 1】



【図3】



【図2】

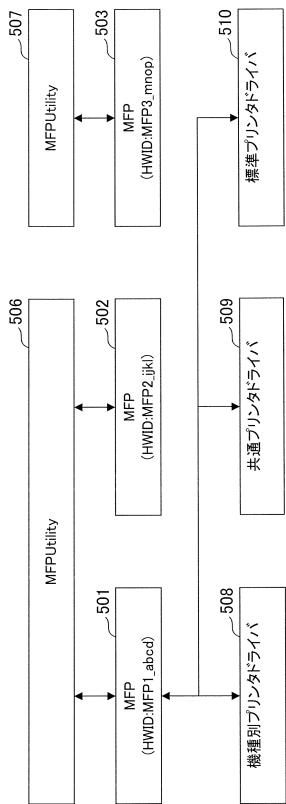


【図4】

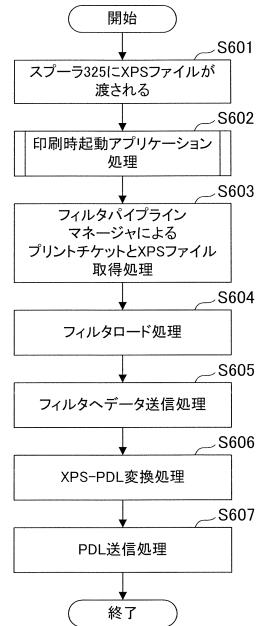
```
(A) <?xml version="1.0" encoding="utf-8"?>
<PackageManifest xmlns="http://schemas.microsoft.com/windows/DeviceMetadata/PackageInfo/2007/11/">
  <Metadata>
    <HardwareList> 402
      <HardwareID> 403
        <HardwareID>D0:D:MFP1_abcd-<HardwareID> 404
        <HardwareID>D0:D:MFP2_ijkl-<HardwareID>
    </HardwareList>
  </Metadata>
</PackageManifest>
```

```
(B) <?xml version="1.0" encoding="utf-8"?>  
<SoftwareInfo xmlns="http://schemas.microsoft.com/windows/DeviceMetadata/SofwareInfo/2007/11/">  
  <DeviceCompanionApplication>  
    <Package>  
      <Identity Name="123abcd.MFTUtility" Publisher="CN=12345678-abcd-efgh-ijkl-123456189012" />  
      411  
      <Applications>  
        <Applications>  
        </Applications>  
      </Applications>  
    </DeviceCompanionApplications>  
  </SoftwareInfo>
```

【図5】



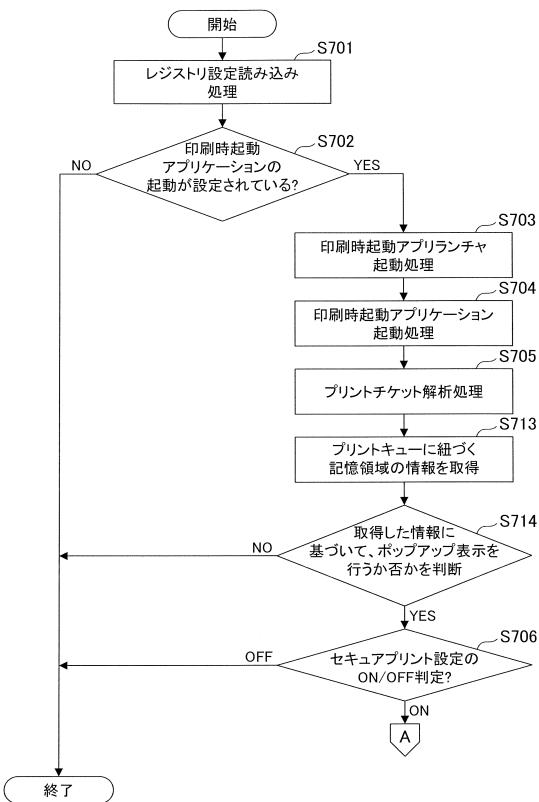
【図6】



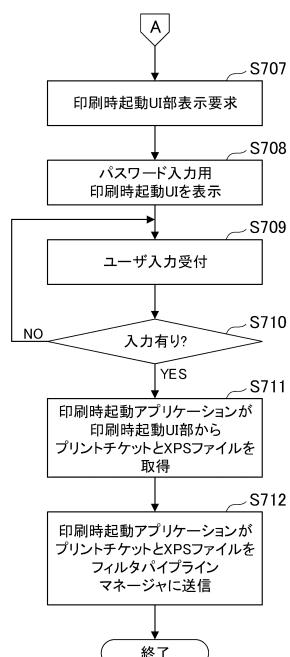
10

20

【図7】



【図8】

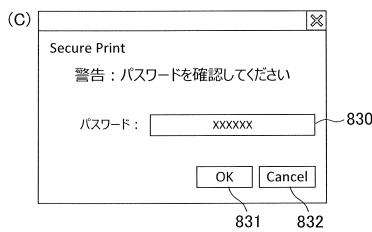
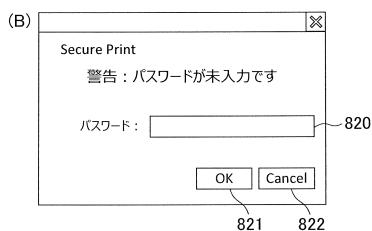
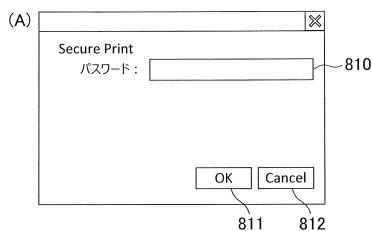


30

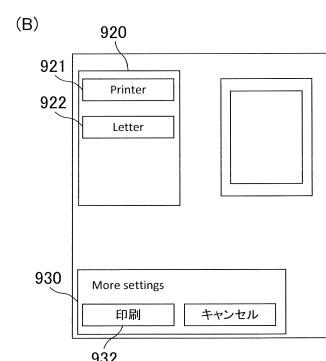
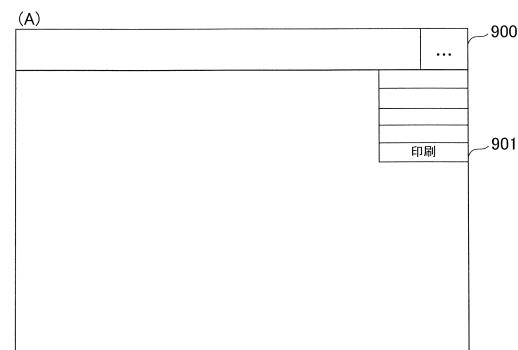
40

50

【図 9】



【図 10】



10

20

【図 11】

```

<?xml version="1.0" encoding="UTF-8"?>
<psf:PrintCapabilities>
  <psf:Feature name="ns0000:JobSecurePrint">
    <psf:Property name="psk:SelectionType">
      <psf:Value xsi:type="xsd:QName">psk:PickOne</psf:Value>
    </psf:Property>
    <psf:Property name="psk:DisplayName">
      <psf:Value xsi:type="xsd:string">セキュアプリント</psf:Value>
    </psf:Property>
    <psf:Option name="ns0000:None" constrained="psk:None">
      <psf:Property name="psk:DisplayName">
        <psf:Value xsi:type="xsd:string">オフ</psf:Value>
      </psf:Property>
    </psf:Option>
    <psf:Option name="ns0000:SecurePrint" constrained="psk:None">
      <psf:Property name="psk:DisplayName">
        <psf:Value xsi:type="xsd:string">オン</psf:Value>
      </psf:Property>
    </psf:Option>
  </psf:Feature>

```

【図 12】

```

<psfPrintTicket xmlns:psf="http://www.w3.org/2003/08/printing/printSchemaFramework"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:ns="http://www.w3.org/2001/XMLSchema"
  version="1"
  xmlns:ns0000="http://www.canon.com/ns/printSchema/obj/v200"
  xmlns:psk="http://schemas.microsoft.com/windows/2003/08/printing/printSchemaKeywords"
  xmlns:psk11="http://schemas.microsoft.com/windows/2013/05/printing/printSchemaKeywordsV11">
  ...
  <psf:Feature name="ns0000:JobSecurePrint" /> 1101
    <psf:Option name="ns0000:SecurePrint" /> 1102
      <psf:Feature>
        <psf:ParameterInit name="ns0000:JobSecurePrintPassword">
          <psf:Value xsi:type="xsd:string" value="XXXXXXXXXX" />
        </psf:ParameterInit>
      </psf:Feature>
      ...
    </psf:Option>
  </psf:Feature>
  ...
</psfPrintTicket>

```

30

40

50

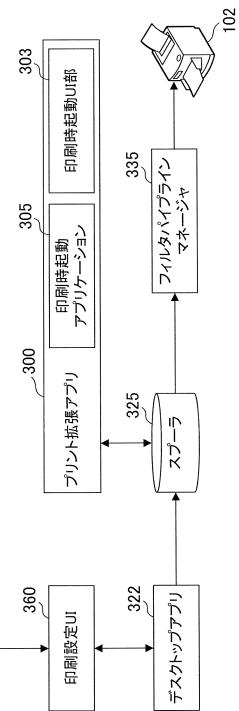
【図 1 3】

```

*Feature: AdminInfoConfirmUserData
{
  *Name: "ConfirmUserData"
  *FeatureType: PRINTER_PROPERTY 1201
  *ConflictPriority: 1900
  *ConcealFromUI?: TRUE
  *DefaultOption: Off
  *Option: Off 1202
  {
    *Name: "Off"
  }
  *Option: On 1203
  {
    *Name: "On"
  }
}

```

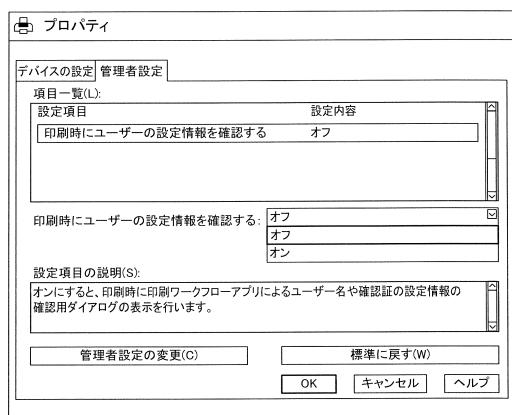
【図 1 4】



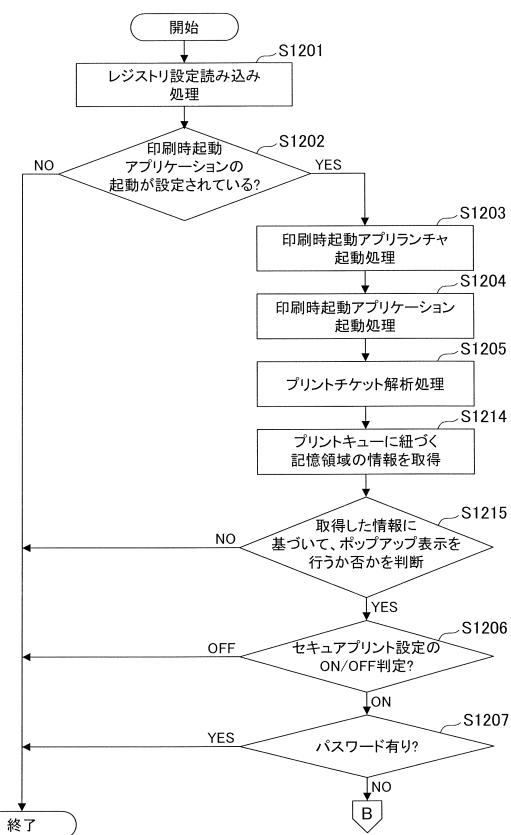
10

20

【図 1 5】



【図 1 6】

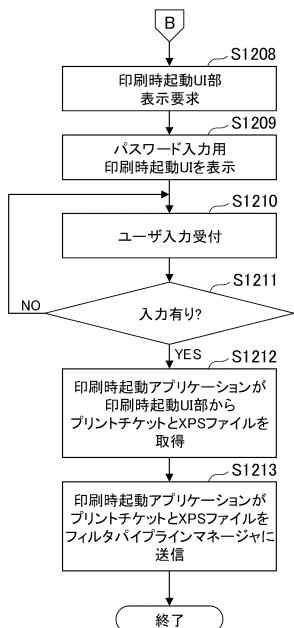


30

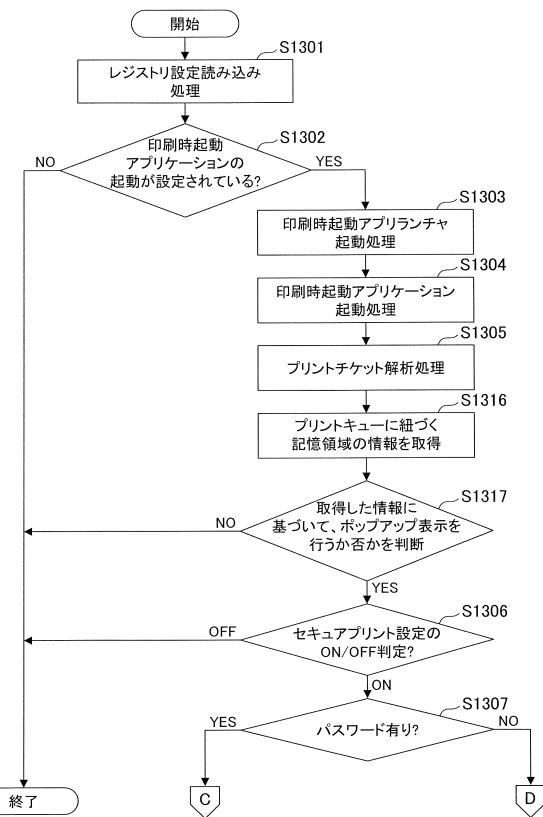
40

50

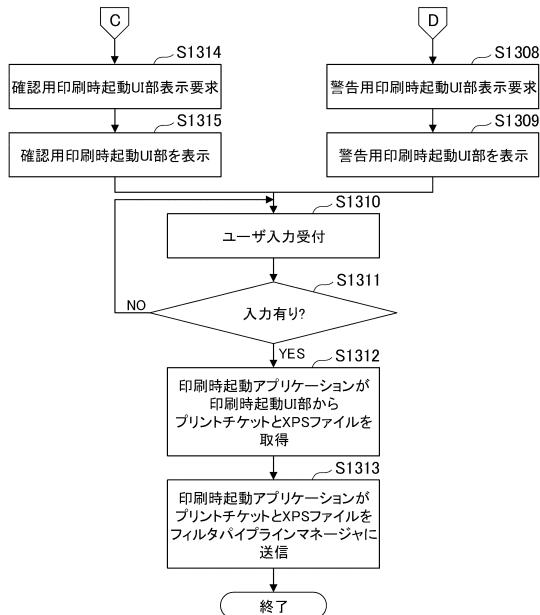
【図17】



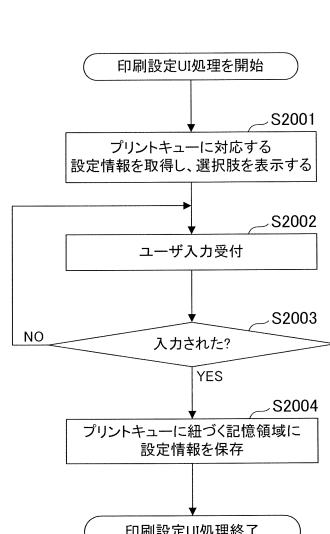
【図18】



【図19】



【図20】



10

20

30

40

50

フロントページの続き

(56)参考文献 特開2016-110612 (JP, A)
 特開2013-145494 (JP, A)
 特開2018-049580 (JP, A)
 特開2013-152608 (JP, A)
 特開2017-054222 (JP, A)
 特開2016-057878 (JP, A)
 特開2015-018474 (JP, A)

(58)調査した分野 (Int.Cl., DB名)
 G06F 3/09 - 3/12
 B41J 29/00 - 29/70
 H04N 1/00