

(19) 日本国特許庁(JP)

(12) 特許公報(B2)

(11) 特許番号

特許第5943761号
(P5943761)

(45) 発行日 平成28年7月5日(2016.7.5)

(24) 登録日 平成28年6月3日(2016.6.3)

(51) Int.Cl.

F 1

G 0 6 F	3/12	(2006.01)	G 0 6 F	3/12	3 3 8
B 4 1 J	29/38	(2006.01)	G 0 6 F	3/12	3 5 9
B 4 1 J	29/00	(2006.01)	G 0 6 F	3/12	3 2 0
			G 0 6 F	3/12	3 2 2
			B 4 1 J	29/38	Z

請求項の数 22 (全 34 頁) 最終頁に続く

(21) 出願番号 特願2012-167511 (P2012-167511)
 (22) 出願日 平成24年7月27日 (2012.7.27)
 (65) 公開番号 特開2013-127771 (P2013-127771A)
 (43) 公開日 平成25年6月27日 (2013.6.27)
 審査請求日 平成27年6月29日 (2015.6.29)
 (31) 優先権主張番号 特願2011-179798 (P2011-179798)
 (32) 優先日 平成23年8月19日 (2011.8.19)
 (33) 優先権主張国 日本国 (JP)
 (31) 優先権主張番号 特願2011-250609 (P2011-250609)
 (32) 優先日 平成23年11月16日 (2011.11.16)
 (33) 優先権主張国 日本国 (JP)

(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 100090273
 弁理士 國分 孝悦
 (72) 発明者 安部 孝一
 東京都大田区下丸子3丁目30番2号 キ
 ャノン株式会社内
 審査官 塩澤 如正

最終頁に続く

(54) 【発明の名称】周辺装置、情報処理装置、通信制御方法、及びプログラム

(57) 【特許請求の範囲】

【請求項 1】

管理装置と通信可能に接続された周辺装置であって、
 前記管理装置に前記周辺装置が取得すべき情報があるか否かを確認する際に用いる確認
 情報については暗号化を行わない通信方式で前記管理装置に送信し、前記確認情報とは異
 なる印刷ジョブを取得するための取得情報については暗号化を行う通信方式で前記管理装
 置に送信する通信手段を有することを特徴とする周辺装置。

【請求項 2】

前記通信手段は、前記取得情報に応答して前記管理装置から送信された前記印刷ジョブ
 の処理が終了すると、前記印刷ジョブの処理が終了したことを示す終了情報について暗号
 化を行う通信方式で前記管理装置に送信することを特徴とする請求項 1 記載の周辺装置。 10

【請求項 3】

前記通信手段は、定期的に、前記確認情報を前記管理装置に送信することを特徴とする
 請求項 1 又は 2 記載の周辺装置。

【請求項 4】

ネットワークを介して提供装置により提供されるサービスにおけるジョブの処理を実行
 する周辺装置と通信可能に接続された情報処理装置であって、

前記サービスにおけるジョブの情報を記憶部に記憶する記憶手段と、

前記サービスでの一連の処理において、前記記憶部にジョブの情報が記憶されているか
 否かを確認する前記周辺装置からの確認情報に対する応答情報については暗号化を行わな

い通信方式で前記周辺装置に送信し、前記確認情報とは異なる前記周辺装置からの他の情報に対する応答情報については暗号化を行う通信方式で前記周辺装置に送信する通信手段と、を有することを特徴とする情報処理装置。

【請求項 5】

前記通信手段は、前記確認情報に対する応答情報については予め定められた演算を行い、演算した結果を暗号化しない通信方式で前記周辺装置に送信することを特徴とする請求項4記載の情報処理装置。

【請求項 6】

ネットワークを介して提供装置により提供されるサービスにおけるジョブの処理の前処理を行う前処理装置と通信可能に接続された情報処理装置であって、

10

前記サービスでの一連の処理において、前記前処理装置が行う前処理の実行の状況を確認する際に用いる確認情報については暗号化を行わない通信方式で前記前処理装置に送信し、前記確認情報とは異なる他の情報については暗号化を行う通信方式で前記前処理装置に送信する通信手段を有することを特徴とする情報処理装置。

【請求項 7】

前記ジョブの処理を実行する周辺装置と通信可能に接続され、

前記通信手段は、

前記提供装置から前記ジョブの情報を受信すると、前記ジョブの情報をもとに生成する前記他の情報である前記前処理で用いられる前処理情報について暗号化を行う通信方式で前記前処理装置に送信し、

20

前記ジョブの処理が終了したこと示す終了情報を前記周辺装置から受信すると、前記他の情報である前記終了情報をもとに生成するジョブ処理終了情報をについて暗号化を行う通信方式で前記前処理装置に送信することを特徴とする請求項6記載の情報処理装置。

【請求項 8】

前記通信手段は、前記確認情報に対する前記前処理装置からの応答情報をもとに、前記前処理が終了したと判断した場合、前記他の情報である、前記前処理の結果の所在を示す情報の取得に用いる所在取得情報をについて暗号化を行う通信方式で前記前処理装置に送信することを特徴とする請求項6又は7記載の情報処理装置。

【請求項 9】

前記通信手段は、前記前処理装置から前記前処理を開始したことを示す情報を受信してから前記確認情報に対する前記前処理装置からの応答情報をもとに前記前処理が終了したと判断するまでの間、前記確認情報を前記前処理装置に送信することを特徴とする請求項6乃至8の何れか1項記載の情報処理装置。

30

【請求項 10】

ネットワークを介して提供装置により提供されるサービスにおけるジョブの情報を管理する管理装置と通信可能に接続された情報処理装置であって、

前記サービスにおけるジョブの処理の前処理を行う前処理手段と、

前記前処理の実行の状況を示す情報を記憶部に記憶する記憶手段と、

前記サービスの一連の処理において、前記前処理の実行の状況を確認する前記管理装置からの確認情報に対する応答情報については暗号化を行わない通信方式で前記管理装置に送信し、前記管理装置からの前記確認情報とは異なる他の情報に対する応答情報については暗号化を行う通信方式で前記管理装置に送信する通信手段と、を有することを特徴とする情報処理装置。

40

【請求項 11】

前記通信手段は、前記確認情報に対する応答情報については予め定められた演算を行い、演算した結果を暗号化しない通信方式で前記管理装置に送信することを特徴とする請求項10記載の情報処理装置。

【請求項 12】

管理装置と通信可能に接続された周辺装置が実行する通信制御方法であって、

前記管理装置に前記周辺装置が取得すべき情報があるか否かを確認する際に用いる確認

50

情報については暗号化を行わない通信方式で前記管理装置に送信する工程と、
前記確認情報とは異なる印刷ジョブを取得するための取得情報については暗号化を行う
通信方式で前記管理装置に送信する工程と、を含むことを特徴とする通信制御方法。

【請求項 1 3】

ネットワークを介して提供装置により提供されるサービスにおけるジョブの処理を実行
する周辺装置と通信可能に接続された情報処理装置が実行する通信制御方法であって、

前記サービスにおけるジョブの情報を記憶部に記憶する記憶工程と、

前記サービスでの一連の処理において、前記記憶部にジョブの情報が記憶されているか
否かを確認する前記周辺装置からの確認情報に対する応答情報については暗号化を行わ
ない通信方式で前記周辺装置に送信する工程と、

前記サービスでの一連の処理において、前記確認情報とは異なる前記周辺装置からの他
の情報に対する応答情報については暗号化を行う通信方式で前記周辺装置に送信する工程
と、を含むことを特徴とする通信制御方法。

【請求項 1 4】

ネットワークを介して提供装置により提供されるサービスにおけるジョブの処理の前処
理を行う前処理装置と通信可能に接続された情報処理装置が実行する通信制御方法であつ
て、

前記サービスでの一連の処理において、前記前処理装置が行う前処理の実行の状況を確
認する際に用いる確認情報については暗号化を行わない通信方式で前記前処理装置に送信
する工程と、

前記サービスでの一連の処理において、前記確認情報とは異なる他の情報については暗
号化を行う通信方式で前記前処理装置に送信する工程と、を含むことを特徴とする通信制
御方法。

【請求項 1 5】

ネットワークを介して提供装置により提供されるサービスにおけるジョブの情報を管理
する管理装置と通信可能に接続された情報処理装置が実行する通信制御方法であって、

前記サービスにおけるジョブの処理の前処理を行う前処理工程と、

前記前処理の実行の状況を示す情報を記憶部に記憶する記憶工程と、

前記サービスでの一連の処理において、前記前処理の実行の状況を確認する前記管理装
置からの確認情報に対する応答情報については暗号化を行わない通信方式で前記管理装
置に送信する工程と、

前記サービスでの一連の処理において、前記管理装置からの前記確認情報とは異なる他
の情報に対する応答情報については暗号化を行う通信方式で前記管理装置に送信する工程
と、を含むことを特徴とする通信制御方法。

【請求項 1 6】

管理装置と通信可能に接続されたコンピュータを、

前記管理装置に前記コンピュータが取得すべき情報があるか否かを確認する際に用いる
確認情報については暗号化を行わない通信方式で前記管理装置に送信し、前記確認情報と
は異なる印刷ジョブを取得するための取得情報については暗号化を行う通信方式で前記管
理装置に送信する通信手段として機能させるためのプログラム。

【請求項 1 7】

ネットワークを介して提供装置により提供されるサービスにおけるジョブの処理を実行
する周辺装置と通信可能に接続されたコンピュータを、

前記サービスにおけるジョブの情報を記憶部に記憶する記憶手段と、

前記サービスでの一連の処理において、前記記憶部にジョブの情報が記憶されているか
否かを確認する前記周辺装置からの確認情報に対する応答情報については暗号化を行わ
ない通信方式で前記周辺装置に送信し、前記確認情報とは異なる前記周辺装置からの他
の情報に対する応答情報については暗号化を行う通信方式で前記周辺装置に送信する通信手段
として機能させるためのプログラム。

【請求項 1 8】

10

20

30

40

50

ネットワークを介して提供装置により提供されるサービスにおけるジョブの処理の前処理を行う前処理装置と通信可能に接続されたコンピュータを、

前記サービスでの一連の処理において、前記前処理装置が行う前処理の実行の状況を確認する際に用いる確認情報については暗号化を行わない通信方式で前記前処理装置に送信し、前記確認情報とは異なる他の情報については暗号化を行う通信方式で前記前処理装置に送信する通信手段として機能させるためのプログラム。

【請求項 19】

ネットワークを介して提供装置により提供されるサービスにおけるジョブの情報を管理する管理装置と通信可能に接続されたコンピュータを、

前記サービスにおけるジョブの処理の前処理を行う前処理手段と、

10

前記前処理の実行の状況を示す情報を記憶部に記憶する記憶手段と、

前記サービスでの一連の処理において、前記前処理の実行の状況を確認する前記管理装置からの確認情報に対する応答情報については暗号化を行わない通信方式で前記管理装置に送信し、前記管理装置からの前記確認情報とは異なる他の情報に対する応答情報については暗号化を行う通信方式で前記管理装置に送信する通信手段として機能させるためのプログラム。

【請求項 20】

ネットワークを介して提供装置により提供されるサービスにおけるジョブの情報を管理する管理装置と通信可能に接続され、前記ジョブの処理を実行する周辺装置であって、

前記サービスでの一連の処理において、前記提供装置から前記サービスの開始を指示する開始通知を受信すると、前記管理装置にジョブが存在するか否かを確認する際に用いる確認情報を、暗号化を行わない通信方式で前記管理装置に定期的に送信し、前記確認情報とは異なる他の情報については暗号化を行う通信方式で前記管理装置に送信する通信手段を有することを特徴とする周辺装置。

20

【請求項 21】

ネットワークを介して提供装置により提供されるサービスにおけるジョブの情報を管理する管理装置と通信可能に接続され、前記ジョブの処理を実行する周辺装置が実行する通信制御方法であって、

前記サービスでの一連の処理において、前記提供装置から前記サービスの開始を指示する開始通知を受信すると、前記管理装置にジョブが存在するか否かを確認する際に用いる確認情報を、暗号化を行わない通信方式で前記管理装置に定期的に送信する工程と、

30

前記サービスでの一連の処理において、前記確認情報とは異なる他の情報については暗号化を行う通信方式で前記管理装置に送信する工程と、を含むことを特徴とする通信制御方法。

【請求項 22】

ネットワークを介して提供装置により提供されるサービスにおけるジョブの情報を管理する管理装置と通信可能に接続され、前記ジョブの処理を実行するコンピュータを、

前記サービスでの一連の処理において、前記提供装置から前記サービスの開始を指示する開始通知を受信すると、前記管理装置にジョブが存在するか否かを確認する際に用いる確認情報を、暗号化を行わない通信方式で前記管理装置に定期的に送信し、前記確認情報とは異なる他の情報については暗号化を行う通信方式で前記管理装置に送信する通信手段として機能させるためのプログラム。

40

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、周辺装置、情報処理装置、通信制御方法、及びプログラムに関する。

【背景技術】

【0002】

近年、USB、Ethernet（登録商標）、無線LAN等の様々なインターフェースを利用して、情報処理装置に周辺装置を接続した周辺装置制御システムが、自宅、オフィ

50

ス等、様々な場面で有効に活用されている。周辺装置の例としては、プリンタ、複写機、ファクシミリ、スキャナ、デジタルカメラ、これらの複合機等が挙げられる。

このような周辺装置のうちのプリンタや複合機に対して、Google社からGoogle Cloud Print（登録商標であり、以下、GCPと略す。）と呼ばれるクラウド印刷サービスが提供されている。クラウド印刷サービスを利用すると、ユーザは、パーソナルコンピュータ（以下、PCと略す。）やモバイルデバイスから文書作成用のアプリケーション等を起動（利用）し、インターネットを介して文書をプリンタや複合機で印刷することが可能である。GCPを利用した印刷の手順は、次のようなものである。

【0003】

(1) ユーザが、PCに設けられた文書作成用のアプリケーションを利用してある文書を開く操作を行う。10

(2) ユーザが、その文書があるプリンタで印刷するために、印刷ダイアログを開いてそのプリンタを選択し、印刷を実行する操作を行う。

(3) 文書作成用のアプリケーションが印刷画像をPDFファイルとして生成する。

(4) GCPが印刷ダイアログで選択された印刷設定値と、前記PDFファイルから印刷ジョブを生成し、GCP内でそのプリンタに割り当られたプリントキューに印刷ジョブを格納し、プリンタに印刷ジョブ通知を送信する。

(5) プリンタは、前記印刷ジョブ通知を受信すると、前記印刷ジョブを取得して、前記印刷設定値に従って、前記PDFファイルの印刷画像を印刷する。

【0004】

例えば、PDFファイルをレンダリングすることができるプリンタや複合機である場合、プリンタや複合機は、前記手順に従ってGCPを利用して印刷することができる。しかしながら、PDFファイルをレンダリングすることができない廉価版のプリンタや複合機である場合、廉価版のプリンタや複合機は、(5)においてPDFファイルの印刷画像を印刷することができない。

このような廉価版のプリンタや複合機でGCPを利用して印刷するために、PDFファイルを、例えばJPEGファイル等の廉価版のプリンタや複合機で印刷できる画像フォーマットに変換するWebサービス（画像変換サービス）用のアプリケーションが存在する。そして、画像変換サービスアプリケーションを利用して、PDFファイルをJPEGファイルに変換し、廉価版のプリンタや複合機で印刷するような周辺装置制御システムが提案されている。30

【0005】

また、プリントのWebサービスに関するその他の技術として特許文献1がある。特許文献1に記載のプリントサービスでは、以下の方法を用いて任意の場所に設置されたプリンタでの印刷を実現している。それは、各地に配備された印刷環境共用サービス間で印刷予約やデータ、印刷可能なアプリケーション情報を交換し、ユーザに印刷可能な印刷場所を提示する方法によって実現している。特許文献1により、ユーザは、遠隔地にある任意の場所に設置されたプリンタで印刷が可能となる。

【先行技術文献】

【特許文献】

【0006】

【特許文献1】特開2005-100340号公報

【発明の概要】

【発明が解決しようとする課題】

【0007】

上述したような周辺装置制御システムでは、複数のプリンタを管理し、GCPに対応させるためのクラウド印刷プロキシサービスを提供するWebサービス（プロキシサービス）用のアプリケーションが必要となる。

そして、プロキシサービスアプリケーションと画像変換サービスアプリケーションとの間では、ポーリングによる画像変換処理の進捗状況の確認処理が行われている。また、廉50

価版のプリンタや複合機が自装置宛に発行されて印刷の準備が完了した印刷ジョブの有無を確認するために、そのプリンタや複合機とクラウド印刷プロキシサービスとの間では、ポーリングによる印刷ジョブの有無の確認処理が行われている。

上述の各サービス間のインターフェース、及び各サービスと廉価版のプリンタや複合機間のインターフェースは、WebサービスAPIで規定され、通常はSSL通信で暗号化されたHTTPSのPOSTメソッドやGETメソッドが利用されている。このように暗号化された通信により、セキュリティが確保されている。

この場合、情報の送信時の暗号化や情報の受信時の復号化に非常に時間を要することになり、また、通信処理の負荷が大きくなるので、クラウドを利用したときにシステムの構築やアプリケーションの実装に支障をきたすという問題がある。10

また、性能の低い廉価版のプリンタや複合機のファームウェアの実装にも支障をきたす問題があり、更に、各処理に時間を要したりする等、ユーザの操作性も悪くなるという問題がある。

【0008】

本発明はこのような問題点に鑑みなされたもので、装置間の通信をより適切に行うこととする。

【課題を解決するための手段】

【0009】

そこで、本発明は、管理装置と通信可能に接続された周辺装置であって、前記管理装置に前記周辺装置が取得すべき情報があるか否かを確認する際に用いる確認情報については暗号化を行わない通信方式で前記管理装置に送信し、前記確認情報とは異なる印刷ジョブを取得するための取得情報については暗号化を行う通信方式で前記管理装置に送信する通信手段を有することを特徴とする。20

【0010】

また、本発明は、システム、記憶媒体などとしてもよい。

【発明の効果】

【0011】

本発明によれば、装置間の通信をより適切に行うことができる。

【図面の簡単な説明】

【0012】

【図1】周辺装置制御システムの構成の一例を示す図である。

【図2】PC及びMFPのハードウェアの構成の一例を示す図である。

【図3】PCのソフトウェアの構成の一例を示す図である。

【図4】Webブラウザの一例を示す図である。

【図5】印刷ダイアログの一例を示す図である。

【図6】周辺装置制御システム全体の処理に係るフローチャートの一例を示す図である。

【図7】周辺装置制御システム全体の処理に係るフローチャートの一例を示す図である。

【図8】周辺装置制御システム全体の処理に係るフローチャートの一例を示す図である。

【図9】プロキシAPPの処理に係るフローチャートの一例を示す図である。

【図10】画像変換APPの処理に係るフローチャートの一例を示す図である。40

【図11】画像変換APPの処理に係るフローチャートの一例を示す図である。

【図12】画像変換APPの処理に係るフローチャートの一例を示す図である。

【図13】コーリングシーケンスの一例を示す図である。

【図14】画面の一例を示す図である。

【図15】画面の一例を示す図である。

【図16】プリンタを登録する処理に係るフローチャートの一例を示す図である。

【図17】周辺装置制御システム全体の処理に係るフローチャートの一例を示す図である。

。

【発明を実施するための形態】

【0013】

10

20

30

40

50

以下、本発明の実施形態について図面に基づいて説明する。なお、実施形態は、本発明を限定するものではなく、また、実施形態で説明されている全ての構成が本発明の課題を解決するための手段に必須であるとは限らない。

【0014】

以下において、WebサービスAPIにSSL通信で暗号化されたHTTPSのPOSTメソッド（通信方式の一例）を利用している場合があるが、このPOSTメソッドに関しては一般に公開されている情報であるので、ここではその詳細な説明を省略する。

また、以下において、WebサービスAPIにHTTPのGETメソッド（通信方式の一例）を利用している場合があるが、このGETメソッドに関しては一般に公開されている情報であるので、ここではその詳細な説明を省略する。

また、以下において、WebサービスAPIにSSL通信で暗号化されたHTTPSのGETメソッド（通信方式の一例）を利用している場合があるが、このGETメソッドに関しては一般に公開されている情報であるので、ここではその詳細な説明を省略する。

また、HTTPは、Hyper Text Transfer Protocolの略であり、暗号化されていない公知の通信プロトコルであるので、ここではその詳細な説明を省略する。

【0015】

また、HTTPSは、Hyper Text Transfer Protocol over Secure Socket Layerの略であり、暗号化された公知の通信プロトコルであるので、ここではその詳細な説明を省略する。

また、SSLは、Secure Socket Layerの略であり、インターネット上でデータを暗号化して送受信する公知の方法であるので、ここではその詳細な説明を省略する。

また、Google Cloud Print (GCP)に関する説明は、次に示すサイトで公開されているので、ここではその詳細な説明を省略する。

インターネットURL：<http://code.google.com/apis/cloudprint/docs/overview.html>

【0016】

<第1の実施形態>

図1は、情報処理装置、及び周辺装置を含んで構成されるシステムの構成の一例（周辺装置制御システム）を示す図である。PC1、PC2、PC5、及びPC7の各々は、情報処理装置（コンピュータ）の一例であり、例えば、一般的なパーソナルコンピュータである。PC1、PC2、PC5、及びPC7の各々は、図2(A)に示すハードウェアを含んで構成される。

PC1には、Operating System（以下、OSと略す。）としてGoogle社製のChrome（登録商標）と同等のOSがインストールされている。また、PC2及びPC5の各々には、マイクロソフト社製のWindows（登録商標）のServer用のOSがインストールされ、PC7には、Linux（登録商標）と同等のOSがインストールされている。PC1、PC2、PC5、及びPC7は、それぞれEthernet等のインターフェース（I/F）で構成されるネットワーク4、8、10、6に接続され、インターネット12にも接続されている。

【0017】

マルチファンクションプリンタ（以下、MFPと略す。）3、103は、カラーインクジェットプリンタ、カラーファクシミリ、カラースキャナ、フラッシュメモリ用の外部ストレージ等を含んで構成され、周辺装置（コンピュータ）の一例である。MFP3、103は、ABC社製のKmmnというモデル名のMFPである。なお、周辺装置としては、プリンタ、複写機、ファクシミリ、スキャナ、デジタルカメラ、及びこれらの複合機能を備える装置等であってもよい。

MFP3、及びMFP103の各々は、図2(B)に示すハードウェアを含んで構成される。また、MFP3は、Ethernetで構成されるネットワーク11に接続され、

10

20

30

40

50

インターネット12にも接続されている。MFP103は、Ethernetで構成されるネットワーク104に接続され、インターネット12にも接続されている。

【0018】

ルーター(Router)13は、Wi-Fi(登録商標)の無線LANを備えるブロードバンドルーター(Wi-Fi BB Router)等であり、インターネット12に接続されている。モバイルデバイス(Mobile Device)14は、携帯電話、携帯情報端末等である。モバイルデバイス14は、ルーター13と接続され、ルーター13を介してインターネット12にも接続されている。

PC1、PC2、PC5、PC7、モバイルデバイス14、MFP3、及びMFP103は、インターネット12を介して互いに双向通信が可能である(通信可能に接続されている)。Webブラウザ(Web Browser)9は、PC1にインストールされ、Webブラウザ15は、モバイルデバイス14にインストールされている。

【0019】

アプリケーション(Application)145は、Webサービスを利用した文書作成用のアプリケーションであり、以下では、文書作成APP145と称する。アプリケーション(Application)146は、Webサービスを利用したクラウド印刷サービスを提供するアプリケーションであり、以下では、クラウド印刷APP146と称する。

クラウド印刷APP146は、例えば、Google社が提供するGCPと同等の機能を有する。このクラウド印刷サービスを利用することで、PC1内のWebブラウザ9やモバイルデバイス内のWebブラウザ15から、文書作成APP145を実行し、ある文書を開き、その文書をMFP3やMFP103で印刷することができるようになる。

【0020】

例えば、ユーザが、開いている文書を印刷する場合、図4に示す印刷メニュー401を選択すると、クラウド印刷APP146により図5に示す印刷ダイアログ142が表示される。ユーザが印刷ダイアログ142でプリンタを選択し、印刷ボタン534を押して印刷の実行を指示すると、文書作成APP145は、印刷画像をPDFファイルとして生成する。

そして、クラウド印刷APP146は、この印刷画像(PDFファイル)と印刷ダイアログ142で選択されている印刷設定値とから印刷ジョブを生成し、クラウド印刷サービス内の選択されたプリンタ用のプリントキューにその印刷ジョブを格納する。また、クラウド印刷APP146は、そのプリンタ(後述する本実施形態の例ではプロキシAPP143)に対して印刷ジョブ通知を送信する。

プリンタやMFPがPDFファイルをレンダリングする機能を備えている場合、プリンタやMFPは、プリントキューに格納されている印刷ジョブを印刷することができる。しかしながら、多くの廉価版のインクジェットプリンタやMFPにおいては、プリンタやMFPがPDFファイルをレンダリングする機能を備えていない。故に、このようなプリンタ等でプリントキューに格納されている印刷ジョブを印刷する場合、PDFファイルの印刷画像を、JPEGファイル等、廉価版のプリンタ等でレンダリングできるタイプの画像フォーマットに変換すること(前処理の一例)が必要となる。

【0021】

アプリケーション(Application)143は、MFP3及びMFP103をクラウド印刷APP146が提供するクラウド印刷サービスに対応させるためのクラウド印刷プロキシサービスを提供するアプリケーションである。以下では、プロキシAPP143と称する。PC2にインストールされているプロキシAPP143は、5000台のプリンタやMFPを一度に管理し、それぞれのプリンタやMFPをクラウド印刷APP146が提供するクラウド印刷サービスに対応させることができる。

アプリケーション(Application)144は、PDFファイルをJPEGファイルに変換する画像変換サービスを提供するアプリケーションであり、以下では、画像変換APP144と称する。

10

20

30

40

50

【0022】

ここで、クラウド印刷APP146が提供するクラウド印刷サービスを利用するためには、ユーザーアカウントとパスワードとを用いたログインが必要となる。そして、クラウド印刷APP146は、印刷に利用可能なプリンタやMFPをクラウド印刷サービスに登録する際に、まず、そのユーザーアカウントとパスワードとを使って、クラウド印刷サービスを利用する際のアクセストークンを取得する。そして、クラウド印刷APP146は、取得したアクセストークンを利用し、そのユーザーアカウントにプリンタやMFPを関連付けて登録する。

また、クラウド印刷APP146は、プリンタやMFPの登録の際に、各プリンタやMFPに対して一意の値で表わされるプリンタIDを発行して割り当て、プリンタIDを登録の対象のプリンタ等に通知する。なお、プリンタ等の登録に関する処理に関しては、その詳細な説明を省略する。10

【0023】

図2は、PC及びMFPのハードウェアの構成の一例を示す図である。PC1、PC2、PC5、及びPC7の各々は、図2(A)に示すハードウェアを含んで構成されている。図2(A)では、PC2を例に挙げて説明する。

PC2は、ランダムアクセスメモリ部(RAM201)、ハードディスクドライブ部(HDD202)、キーボード部(KBD203)、CPU204、表示用ディスプレイ(LCD205)、ネットワークボード(NB207)を有する。また、PC2は、上述の構成要素を互いに接続するバス206を有する。20

【0024】

なお、HDD202は、記憶部の一例である。記憶部は、可搬性CD-ROMまたは内部に据付のROM等であってもよい。KBD203は、入力部の一例である。CPU204は、制御部の一例である。LCD205は、表示部の一例である。NB207は、通信制御部の一例である。

プロキシAPP143、図3に示す各モジュール(ソフトウェア)等のプログラムは、HDD202に記憶され、CPU204により必要に応じてRAM201に読み出されて実行される。これにより、プロキシAPP143の機能、図3に示す各モジュール(ソフトウェア)の機能などが実現される。また、各種のデータベース(DB)は、HDD202に設けられ、データベースの情報は、CPU204により必要に応じてRAM201に読み出される。30

【0025】

MFP3及びMFP103の各々は、図2(B)に示すハードウェアを含んで構成されている。図2(B)では、MFP3を例に挙げて説明する。

CPU25は、マイクロプロセッサ等から構成される。CPU25は、MFP3の中央処理装置として、ROM16に記憶されているプログラムに従って、RAM17、通信部18、記録部19、操作部20、表示部21、読み取り部22、ファクシミリ制御部23、及び外部ストレージ制御部24を制御する。

ROM16には、プリンタドライバ(図示は省略)による制御に従ってMFP3が行う記録(印刷)処理、印刷動作の状態をPCへ通知する処理などを実行するプログラムが記憶されている。また、ROM16には、FAXドライバ(図示は省略)による制御に従ってMFP3がファクシミリの送信処理または受信処理、ファクシミリ動作(送信または受信)の状態をPCへ通知する処理を実行するプログラムが記憶されている。更に、ROM16には、WIAドライバ(不図示)やTWINドライバ(不図示)による制御に従ってMFP3が行う画像の読み取り処理、読み取り動作の状態をPCへ通知する処理を実行するプログラムが記憶されている。40

【0026】

RAM17には、印字データが一時的に記憶される。印字データは、主にPCから送られ、印字データをもとに記録部19によって印刷される。また、RAM17には、読み取り部22で読み取られた画像データ、PCから送られてきたファクシミリの送信データ、ファ50

クシミリ制御部 23 で受信されたファクシミリの受信データ等も一時的に記憶される。

通信部 18 には、ネットワーク用の接続ポート、アナログ電話回線用の接続ポート等が含まれており、Ethernet、ファクシミリのアナログ通信を制御する。記録部 19 は、インクジェット方式の記録ヘッド、各カラーインク、キャリッジ、記録紙搬送機構等から構成される記録ユニットと、印字データをもとに記録ヘッドにて印字用パルスを発生させるためのASIC等から構成される電気回路とを含んで構成される。

【0027】

例えば、印刷可能なアプリケーション上の印刷操作、又はファクシミリの送信操作によって、アプリケーションで開かれているファイルの表示内容（画像データ）が、EMF 形式のスプールファイルとして PC2 の HDD202 に一時的に格納される。そして、スプールファイルは、プリンタドライバ、又は FAX ドライバを介して MFP3 制御用コマンドを含む印字データ、又はファクシミリの送信データに変換された後、インターネットを介して MFP3 に送られる。10

MFP3 にて受信された印字データは、記録部 19 で印字用パルスに変換され、記録紙上に印刷される。また、MFP3 にて受信されたファクシミリの送信データは、ファクシミリ制御部 23 でファクシミリの通信プロトコルに変換され、アナログ電話回線を介して相手先のファクシミリ装置に送信される。

【0028】

操作部 20 は、電源ボタン、リセットボタン等の各種ボタンから構成され、MFP3 に対する操作を受け付ける。表示部 21 は、タッチパネルの液晶ディスプレイで構成され、MFP3 の状態の表示、各種の設定情報や電話番号の表示、入力等を行うことができる。読み取り部 22 は、カラーイメージセンサや画像処理用のASIC 等から構成される電気回路を含んで構成され、スキャナ機能を制御する。20

ファクシミリ制御部 23 は、ファクシミリ用のモデム、アナログ通信回路等から構成され、ファクシミリの通信プロトコルに従ってファクシミリの送信および受信を制御する。外部ストレージ制御部 24 は、フラッシュメモリの装着用のスロット、ストレージ用のインタフェース回路等から構成され、装着されたフラッシュメモリを制御する。

【0029】

図 3 は、PC のソフトウェアの構成の一例を示す図である。図 3 では、PC2 を例に挙げて説明する。I/F92 は、Ethernet 等の I/F を制御する I/F 制御スタックである。IP Network91 は、IP Network を制御する IP Network 制御スタックである。30

Internet Information Services (IIS)90 は、IIS を制御する IIS 制御部であり、アプリケーション 150 は、この IIS 制御部の Web サーバ機能を利用して Web サービスを提供する。JAVA (登録商標) Virtual Machine (Java VM)89 は、Java バイトコードをプラットフォームのネイティブコードに変換して実行するソフトウェアであり、プロキシ APP143 は、JAVA VM89 上で実行される。

Applications82 は、アプリケーション群であり、プロキシ APP143、アプリケーション 150 等から構成される。40

【0030】

図 4 は、PC1 の文書作成 APP145 が提供する Web ブラウザ 9 の一例を示す図である。図 4 は、ある文書を開いている状態を表す。

印刷メニュー 401 は、印刷に関する各種の操作を受け付ける。例えば、ユーザが KBD203 等の入力部を操作（ユーザ操作）し、開いている文書を印刷するときに印刷メニュー 401 を押下すると、図 5 に示す印刷ダイアログ 142 が表示される。

【0031】

図 5 は、印刷ダイアログ 142 の一例を示す図である。印刷ダイアログ 142 は、ユーザに対する MFP3、103 用の印刷設定画面として、クラウド印刷 APP146 により制御される。また、印刷ダイアログ 142 は、Web ブラウザ 9、15 を利用して、その50

W e b ブラウザ上に表示される。

プリンタ選択肢 535、536、537 の各々は、選択可能なプリンタの選択肢である。プリンタ選択肢 535、536、537 は、それぞれ、G H I 社製の A k b N m b というモデル名のプリンタ、A B C 社製の K m m n というモデル名の M F P (M F P 3)、D E F 社製の X y z というモデル名のプリンタに該当する選択肢を表す。図 5 では、M F P 3 を表すプリンタ選択肢 536 が選択されている状態が示されている。

【 0 0 3 2 】

用紙サイズ選択部 530 では、次に示す選択肢の中から M F P 3 で印刷する用紙のサイズがユーザ操作に応じて選択される。図 5 では、L e t t e r が選択されている状態が示されている。

10

用紙サイズの選択肢 : A 5、A 4、B 5、L e t t e r

用紙の種類選択部 531 では、次に示す選択肢の中から M F P 3 で印刷する用紙の種類がユーザ操作に応じて選択される。図 5 では、P l a i n P a p e r が選択されている状態が示されている。

用紙の種類の選択肢 : P l a i n P a p e r、P h o t o P a p e r、P o s t c a r d

【 0 0 3 3 】

印刷品位選択部 532 では、次に示す選択肢の中から M F P 3 で印刷する印刷品位がユーザ操作に応じて選択される。図 5 では、S t a n d a r d が選択されている状態が示されている。

20

印刷品位の選択肢 : H i g h、S t a n d a r d、F a s t

色 / 濃度選択部 533 では、次に示す選択肢の中から M F P 3 で印刷する色や濃度がユーザ操作に応じて選択される。図 5 では、A u t o が選択されている状態が示されている。

印刷品位の選択肢 : A u t o、M a n u a l

印刷ボタン 534 は、印刷の開始の指示を受け付ける。ユーザ操作により印刷ボタン 534 が押下されると、クラウド印刷 A P P 146 は、プリンタ選択肢 536 に関連付けられている M F P 3 用の印刷ジョブを生成する。クローズボタン 538 は、印刷ダイアログ 142 を閉じる指示を受け付ける。ユーザ操作によりクローズボタン 538 が押下されると、クラウド印刷 A P P 146 は、印刷ダイアログ 142 を閉じる。

30

【 0 0 3 4 】

図 6、図 7、図 8 は、本周辺装置制御システム全体の処理に係るフローチャートの一例を示す図である。これらのフローチャートに係るプログラムは、P C または M F P 内に組み込まれている。P C 上で動作するアプリケーションのプログラムは、H D D 202 に記憶されており、C P U 204 により R A M 201 に読み出されて実行される。M F P 上で動作するアプリケーションのプログラムは、R O M 16 に記憶されており、R A M 17 に読み出され、C P U 25 により実行される。

これらのフローチャートにおいて、縦の列が処理の実行元を表し、本実施形態では図面の左側から、ユーザ (W e b ブラウザ 9)、文書作成 A P P 145、クラウド印刷 A P P 146、プロキシ A P P 143、画像変換 A P P 144、M F P 3 を記す。以下では、ユーザが P C 1 を使ってドキュメントの印刷を M F P 3 で行うケース (ドキュメントの印刷に係る一連の処理) を例に挙げて通信制御方法について説明する。

40

【 0 0 3 5 】

まず、M F P 3 が印刷ジョブをポーリングする構成について S 617 ~ S 621 を用いて説明する。図 6 において、S 617 は、便宜的に M F P 3 側から見たドキュメントの印刷の開始を表すステップである。なお、このステップは、M F P 3 内の処理の実行には一切影響を与えるものではなく、単にこのフローチャートの開始を表すものである。

S 618 では、M F P 3 は、プロキシ A P P 143 がエクスポートしている W e b サービス A P I の C h e c k P r i n t J o b 関数をコールする。S 618 において、M F P 3 が、この関数をコールするときに、H T T P の G E T 要求のクエリパラメータとしてブ

50

ロキシAPP143に渡す情報（確認情報の一例）は、次に示すような情報である。

- ・クラウド印刷APP146が発行したプリンタID
- ・ランダムコード

また、CheckPrintJob関数をコールするタイミングは任意であり、定常にコールするポーリングの形態とするが、この形態に限られるものではない。例えば、ユーザから指定されたときにコールする形態であってもよい。更に、クエリパラメータに含まれる確認情報もプリンタID、ランダムコードに限られるものではない。

【0036】

ランダムコードは、MFP3が生成した任意の32ビットの値である。プロキシAPP143は、CheckPrintJob関数がコールされると、図9(A)で後述するCheckPrintJob関数の処理を実行し、そのレスポンス（応答情報の一例）をMFP3に返す(S619)。

続いて、MFP3は、プロキシAPP143からのCheckPrintJob関数のレスポンスを受信すると、レスポンス内の応答メッセージに含まれている情報（「印刷ジョブ有り応答」又は「印刷ジョブ無し応答」）を確認する(S620)。S620において、MFP3は、「印刷ジョブ有り応答」を受信したと判断した場合、図7のS708に処理を移し、他方、「印刷ジョブ無し応答」を受信したと判断した場合、S621に処理を移す。

S621では、MFP3は、一定の時間（本実施形態では1分間）待った後、S618に処理を移す。このように、MFP3は、プロキシAPP143がエクスポートしているCheckPrintJob関数を一定の時間間隔で（定期的に）ポーリングする。

【0037】

次に、ドキュメントの印刷処理について説明する。ここで、S601は、便宜的にユーザ側から見た場合のドキュメントの印刷の開始を表すステップである。なお、このステップは、ユーザ操作やアプリケーションの実行には一切影響を与えるものではなく、単にこのフローチャートの開始を表すものである。

まず、ユーザは、Webブラウザ9を起動して文書作成APP145のURLにアクセスしてログインする。そして、ユーザは、ある文書を指定してその文書を開く指示を行うと、Webブラウザ9は、その文書を入力パラメータとして、文書作成APP145がエクスポートしているWebサービスAPIのOpenDocument関数をコールする(S602)。

文書作成APP145は、OpenDocument関数がコールされると、入力パラメータで指定された文書を開いてその情報をWebブラウザ9に送信する(S603)。例えば、図4に示すように、Webブラウザ9上で文書作成APP145を介して当該文書が開かれた状態となる。

【0038】

ユーザ操作により、当該文書を印刷するために印刷メニュー401が押下されると、Webブラウザ9を介してその情報が文書作成APP145に通知される(S604)。文書作成APP145は、この通知を受け取ると、クラウド印刷APP146がエクスポートしているWebサービスAPIのPrintDialog関数をコールする(S605)。

クラウド印刷APP146は、PrintDialog関数がコールされると、印刷ダイアログ142を開いて、その情報をWebブラウザ9に送信する(S606)。例えば、図5に示すように、Webブラウザ9上に印刷ダイアログ142が表示される。

【0039】

ユーザは、入力部を操作して、複数のプリンタ選択肢の中から印刷するプリンタ（図5の例ではMFP3に関連付けられているプリンタ選択肢536）を選択し、各印刷設定値の中から希望する設定値を選択し、印刷ボタン534を押下する。印刷ボタン534が押下されると、Webブラウザ9を介してそのユーザからの印刷開始要求を示す情報と印刷設定値に関する情報とが文書作成APP145に通知される(S607)。

10

20

30

40

50

印刷設定値に関する情報は、例えば、ユーザが選択した印刷設定値が記されている印刷設定ファイル（XMLファイル）が格納されているURLである。文書作成APP145は、この通知を受け取ると、印刷画像をPDFファイルとして生成する。そして、文書作成APP145は、生成したPDFファイル、印刷設定値に関する情報等を入力パラメータとして、クラウド印刷APP146がエクスポートしているWebサービスAPIのSubmitPrintJob関数をコールする（S608）。

【0040】

クラウド印刷APP146は、SubmitPrintJob関数がコールされると、入力パラメータで指定されたPDFファイル、印刷設定値に関する情報等をもとに印刷ジョブを生成する。そして、クラウド印刷APP146は、選択されているプリンタ選択肢に関連付けられているプリンタ（本例では、画像フォーマットの変換が必要なMFP3であるのでプロキシAPP143）に印刷ジョブ通知を送信する（S609）。

また、印刷ジョブには、少なくとも次に示すような情報（以下では、印刷ジョブ情報と称する。）が含まれている。

【0041】

- ・クラウド印刷APP146が発行した印刷ジョブID
- ・クラウド印刷APP146が発行したプリンタID
- ・クラウド印刷APP146が提供するクラウド印刷サービスを利用するためのアクセストークン
- ・印刷設定値が記されている印刷設定ファイル（XMLファイル）が格納されているURL
- ・印刷画像（PDFファイル）の格納先のURL
- ・タイムスタンプ

【0042】

プロキシAPP143は、印刷ジョブ通知を受信すると、クラウド印刷APP146がエクスポートしているWebサービスAPIのFetchPrintJob関数をコールする（S610）。クラウド印刷APP146は、FetchPrintJob関数がコールされると、S609で生成した印刷ジョブに含まれている情報（印刷ジョブ情報）をプロキシAPP143に送信する（S611）。

プロキシAPP143は、印刷ジョブ情報を受信すると、印刷ジョブ情報の中から必要な情報を取得し、取得した情報とその他の制御で必要な情報を印刷ジョブ管理データベースに保存する（S612）。そして、プロキシAPP143は、画像変換APP144がエクスポートしているWebサービスAPIのCreatePrintJobSet関数をコールする（S612）。

【0043】

S612において、プロキシAPP143が、この関数をコールするときに、HTTPのPOST要求の入力パラメータとして画像変換APP144に渡す情報（前処理情報の一例）は、次に示すような情報である。

- ・クラウド印刷APP146が発行したプリンタID
- ・クラウド印刷APP146が発行した印刷ジョブID
- ・クラウド印刷サービスを利用するためのアクセストークン
- ・印刷画像（PDFファイル）の格納先のURL

【0044】

また、S612において、印刷ジョブ管理データベースには、次に示すような情報が保存される。

[印刷ジョブ管理データベース]

- ・検索キー：クラウド印刷APP146が発行した印刷ジョブID
- ・クラウド印刷サービスを利用するためのアクセストークン
- ・印刷設定ファイル（XMLファイル）の格納先のURL
- ・印刷画像（PDFファイル）の格納先のURL

10

20

30

40

50

- ・画像変換APP144へのアクセストークン
- ・印刷ジョブアクセスURL(JPEGファイルの格納先のURL)
- ・印刷画像(JPEGファイル)の総ページ数
- ・印刷ジョブ準備完了(PrintJobReady)(0:未完了、1:完了)

【0045】

このとき、プロキシAPP143は、印刷ジョブ準備完了(PrintJobReady)に、印刷ジョブ準備が未完了であることを表す初期値として「0」をセットする。また、プロキシAPP143は、印刷画像(JPEGファイル)の総ページ数に初期値「0」をセットする。

画像変換APP144は、CreatePrintJobSet関数がコールされると、図10(A)で後述するCreatePrintJobSet関数の処理を実行し、そのレスポンスをプロキシAPP143に返す(S613)。このレスポンスには、画像変換APP144が発行したセッションIDが含まれている。10

プロキシAPP143は、画像変換APP144からのCreatePrintJobSet関数のレスポンスを受信すると、画像変換APP144がエクスポートしているWebサービスAPIのConfirmProgress関数をコールする(S614)。S614において、プロキシAPP143が、この関数をコールするときに、HTTPのGET要求のクエリパラメータとして画像変換APP144に渡す情報(確認情報の一例)は、次に示すような情報である。

【0046】

- ・セッションID
- ・シーケンス

セッションIDは、CreatePrintJobSet関数のレスポンスの中に含まれている情報であり、シーケンスは、プロキシAPP143が生成した任意の番号である。20

【0047】

画像変換APP144は、ConfirmProgress関数がコールされると、図11(A)で後述するConfirmProgress関数の処理を実行し、そのレスポンス(応答情報の一例)をプロキシAPP143に返す(S615)。

プロキシAPP143は、画像変換APP144からのConfirmProgress関数のレスポンスを受信すると、そのレスポンス内の応答メッセージに含まれている内容(「処理中応答」、「処理完了応答」、又は「エラー応答」)を確認する(S616)。S616において、プロキシAPP143が「処理完了応答」を受信したと判断した場合、図7のS701に処理を移し、他方、「処理中応答」を受信したと判断した場合、S614に処理を移す。30

【0048】

なお、図10(B)で後述する画像変換APP144のダウンロード・画像変換スレッドの処理でエラーが発生した場合、S616において、プロキシAPP143が「エラー応答」を受信して印刷処理をエラー終了する。

しかしながら、エラー終了するケースに関しては、本実施形態の本質から外れるので、その詳細な説明を省略する。よって、S616以降では、プロキシAPP143が「処理完了応答」または「処理中応答」を受信した場合を中心に説明する。40

【0049】

図7に示すように、S701では、プロキシAPP143は、画像変換APP144がエクスポートしているWebサービスAPIのCreatePrintJobResult関数をコールする。S701において、プロキシAPP143が、この関数をコールするときに、HTTPのPOST要求の入力パラメータとして画像変換APP144に渡す情報(所在取得情報の一例)は、次に示すような情報である。

- ・クラウド印刷APP146が発行したプリンタID
- ・クラウド印刷APP146が発行した印刷ジョブID

【0050】

画像変換APP144は、CreatePrintJobResult関数がコールされると、図12で後述するCreatePrintJobResult関数の処理を実行し、そのレスポンスをプロキシAPP143に返す(S702)。

プロキシAPP143は、画像変換APP144からのCreatePrintJobResult関数のレスポンスを受信すると、PrintJobReadyに「1」をセットする(S703)。そして、プロキシAPP143は、印刷ジョブ管理データベースの印刷ジョブ準備完了(PrintJobReady)に、印刷ジョブ準備完了を表す「1」をセットする(S703)。これにより、MFP3における印刷待ち状態となる(S704)。

10

【0051】

図6のS618～S621で前述したように、MFP3は、プロキシAPP143がエクスポートしているCheckPrintJob関数を、一定の時間間隔でポーリングし、S705は、S618と同等な処理を表す。

S704でMFP3における印刷待ち状態となっているとき、S705において、MFP3が、プロキシAPP143がエクスポートしているCheckPrintJob関数をコールした場合について説明する。プロキシAPP143は、CheckPrintJob関数がコールされると、図9(A)で後述するCheckPrintJob関数の処理を実行し、そのレスポンスをMFP3に返す(S706)。

【0052】

20

MFP3は、プロキシAPP143からのCheckPrintJob関数のレスポンスを受信すると、そのレスポンス内の応答メッセージに含まれている内容(「印刷ジョブ有り応答」又は「印刷ジョブ無し応答」)を確認する(S707)。S707において、MFP3が「印刷ジョブ有り応答」を受信したと判断した場合、S708に処理を移し、他方、「印刷ジョブ無し応答」を受信したと判断した場合、S621に処理を移す。

S708では、MFP3は、プロキシAPP143がエクスポートしているWebサービスAPIのGetPrintJob関数をコールする。S708において、MFP3が、この関数をコールするときに、HTTPのGET要求のクエリパラメータとしてプロキシAPP143に渡す情報(取得情報の一例)は、次に示すような情報である。

- ・クラウド印刷APP146が発行したプリンタID

30

【0053】

プロキシAPP143は、GetPrintJob関数がコールされると、図9(B)で後述するGetPrintJob関数の処理を実行し、そのレスポンスをMFP3に返す(S709)。

MFP3は、プロキシAPP143からのGetPrintJob関数のレスポンスを受信すると、レスポンス内の応答メッセージに含まれている内容をもとに、印刷画像取得要求を画像変換APP144に送信する(S710)。

ここで、レスポンス内の応答メッセージに含まれている内容は、印刷ジョブアクセスURL(印刷画像(JPEGファイル)の格納先のURL)、印刷画像(JPEGファイル)の総ページ数、及び画像変換APP144へのアクセストークンである。また、印刷画像取得要求は、画像変換APP144から各ページの印刷画像(JPEGファイル)を取得するための要求である。

40

本実施形態では、印刷ジョブアクセスURLは、画像変換App144を有するPC5の格納先URLとするが、それ以外のPCの格納先URLであってもよい。また、例えば、印刷ジョブアクセスURLについては、直接PC5から送信する形態であってもよい。また、例えば、指定されたページの印刷画像を、プロキシApp143を有するPC2がPC5から取得し、取得した印刷画像をPC2がMFP3に送信する形態であってもよい。また、例えば、PC2が、取得した印刷画像の格納先のURLをMFP3に送ってもよい。

【0054】

50

画像変換APP144は、MFP3からの印刷画像取得要求を受信すると、指定されたページの印刷画像（JPEGファイル）に対するMFP3からのアクセスを許可し、その印刷画像をMFP3に送信する（S711）。

MFP3は、画像変換APP144から送信された印刷画像を受信すると、その印刷画像を印刷する（S712）。続いて、MFP3は、印刷画像（JPEGファイル）の総ページ数をもとに、次ページの有無を確認する（S713）。このとき、MFP3は、次ページが有ると判断した場合、S710に処理を移し、次ページの印刷画像を取得するための印刷画像取得要求を画像変換APP144に送信する。他方、MFP3は、次ページが無いと判断した場合、図8のS801に処理を移す。

S710～S712における印刷画像（JPEGファイル）の取得処理に関しては、SSL通信で暗号化されたHTTPSのGETメソッドが利用される。

【0055】

図8に示すように、S801では、MFP3は、プロキシAPP143がエクスポートしているWebサービスAPIのEndPrintJob関数をコールする。EndPrintJob関数のコールについては、SSL通信で暗号化されたHTTPSのGETメソッドが利用されている。S801において、MFP3が、この関数をコールするときに、HTTPSのGET要求のクエリパラメータとしてプロキシAPP143に渡す情報（終了情報の一例）は、次に示すような情報である。

- ・クラウド印刷APP146が発行したプリントID
- ・クラウド印刷APP146が発行した印刷ジョブID

【0056】

プロキシAPP143は、EndPrintJob関数がコールされると（S802）、画像変換APP144がエクスポートしているWebサービスAPIのJobComplete関数をコールする（S803）。

S803において、プロキシAPP143が、この関数をコールするときに、HTTPSのPOST要求の入力パラメータとして画像変換APP144に渡す情報（ジョブ処理終了情報の一例）は、次に示すような情報である。

- ・クラウド印刷APP146が発行したプリントID
- ・クラウド印刷APP146が発行した印刷ジョブID

【0057】

画像変換APP144は、JobComplete関数がコールされると、図11（B）で後述するJobComplete関数の処理を実行し、そのレスポンスをプロキシAPP143に返す（S804）。

プロキシAPP143は、画像変換APP144からのJobComplete関数のレスポンスを受信すると、印刷ジョブ管理データベースからこの印刷ジョブに関する情報を削除して、印刷ジョブ管理データベースを更新する（S805）。そして、プロキシAPP143は、ドキュメントの印刷処理を終了する（S806）。

【0058】

次に、プロキシAPP143がエクスポートするWebサービスAPIの処理について図9を用いて説明する。図9（A）は、CheckPrintJob関数の処理に係るフローチャートの一例を示す図である。図9（B）は、GetPrintJob関数の処理に係るフローチャートの一例を示す図である。

CheckPrintJob関数については、HTTPのGETメソッドが利用されている。GetPrintJob関数については、SSL通信で暗号化されたHTTPSのGETメソッドが利用されている。これらのフローチャートに係るプログラムは、PC2に組み込まれ、HDD202に記憶されており、CPU204によりRAM201に読み出されて実行される。

【0059】

図6のS618、図7のS705において、MFP3がCheckPrintJob関数をコールすると、プロキシAPP143が図9（A）に示すCheckPrintJob

10

20

30

40

50

b 関数の処理を実行する。

図9(A)に示すように、プロキシAPP143は、CheckPrintJob関数の処理を開始すると(S901)、印刷ジョブ管理データベースの印刷ジョブ準備完了(PrintJobReady)を確認する(S902)。S901において、HTTPのGET要求のクエリパラメータとして、クラウド印刷APP146が発行したプリンタIDとランダムコードとがMFP3から渡される。

プロキシAPP143は、印刷ジョブ準備完了(PrintJobReady)に印刷ジョブ準備が完了していることを表す「1」がセットされている、即ちMFP3の印刷ジョブが存在すると判断した場合(S903でYESの場合)、S904に処理を移す。他方、プロキシAPP143は、印刷ジョブの準備が未完了であることを表す「0」がセットされている、即ちMFP3の印刷ジョブが存在しないと判断した場合(S903でNOの場合)、S905に処理を移す。
10

【0060】

S904では、プロキシAPP143は、HTTPのGET要求に対するレスポンスのBODYに含める応答メッセージに「印刷ジョブ有り応答」をセットし、S906に処理を移す。S905では、プロキシAPP143は、HTTPのGET要求に対するレスポンスのBODYに含める応答メッセージに「印刷ジョブ無し応答」をセットし、S906に処理を移す。

S906では、プロキシAPP143は、GET要求の発行元(本例ではMFP3)に対して応答メッセージを送信し、CheckPrintJob関数の処理を終了する(S907)。
20

【0061】

ここで、「印刷ジョブ有り応答」と「印刷ジョブ無し応答」との詳細を説明する。印刷ジョブ有りの状態を表す情報、及び印刷ジョブ無しの状態を表す情報は、それぞれ「10101010」、「01010101」である。

「印刷ジョブ有り応答」及び「印刷ジョブ無し応答」は、それぞれ、ランダムコードとこれらの状態を表す情報を用いて演算された結果であり、例えば、次に示す情報を文字列に変換した文字列情報で構成される。

「印刷ジョブ有り応答」 = ランダムコード XOR 10101010

「印刷ジョブ無し応答」 = ランダムコード XOR 01010101
30

【0062】

このように、HTTPのGET要求に対するレスポンスとしてXMLを使って定義した可視的な情報等ではなく一般には意味がわからない文字列情報を用いることで、WebサービスAPIに対する不正なアクセスによる誤動作や不正な制御を防止することができる。

なお、このような不正なアクセスを防止する構成は、この例に限られるものではなく、ハッシュ化等、より複雑な演算を組み合わせてもよい。

【0063】

また、図7のS708において、MFP3がGetPrintJob関数をコールすると、プロキシAPP143は、図9(B)に示すGetPrintJob関数の処理を実行する。
40

図9(B)に示すように、プロキシAPP143は、GetPrintJob関数の処理を開始する(S911)。そして、プロキシAPP143は、HTTPのGET要求に対するレスポンスのBODYに含める応答メッセージに、印刷画像(JPEGファイル)の格納先のURLをセットし(S912)、S913に処理を移す。

【0064】

S913では、プロキシAPP143は、HTTPのGET要求に対するレスポンスのBODYに含める応答メッセージに、印刷画像(JPEGファイル)の総ページ数をセットし、S914に処理を移す。S914では、プロキシAPP143は、HTTPのGET要求に対するレスポンスのBODYに含める応答メッセージに、画像変換APP1
50

44へのアクセストークンをセットし、S915に処理を移す。

S915では、プロキシAPP143は、GET要求の発行元（本例ではMFP3）に対して応答メッセージを送信し、GetPrintJob関数の処理を終了する（S916）。

【0065】

次に、画像変換APP144がエクスポートするWebサービスAPIとスレッド関数の処理とについて図10を用いて説明する。図10（A）は、CreatePrintJobSet関数の処理に係るフローチャートの一例を示す図である。図10（B）は、ダウンロード・画像変換スレッド関数の処理に係るフローチャートの一例を示す図である。CreatePrintJobSet関数については、SSL通信で暗号化されたHTTPのPOSTメソッドが利用されている。
10

これらのフローチャートに係るプログラムは、PC5に組み込まれ、HDD202に記憶されており、CPU204によりRAM201に読み出されて実行される。

【0066】

図6のS612において、プロキシAPP143がCreatePrintJobSet関数をコールすると、画像変換APP144は、図10（A）に示すCreatePrintJobSet関数の処理を実行する。

図10（A）に示すように、画像変換APP144は、CreatePrintJobSet関数の処理を開始する（S1001）。そして、画像変換APP144は、セッションIDを生成し、セッション管理データベースに保存し（S1002）、S1003に処理を移す。セッション管理データベースには、次に示すような情報が保存される。
20

【0067】

[セッション管理データベース]

- ・検索キー：画像変換APP144が発行したセッションID
- ・クラウド印刷APP146が発行したプリンタID
- ・クラウド印刷APP146が発行した印刷ジョブID
- ・クラウド印刷サービスを利用するためのアクセストークン
- ・印刷画像（PDFファイル）の格納先のURL
- ・画像変換APP144へのアクセストークン
- ・印刷ジョブアクセスURL（JPEGファイルの格納先のURL）
- ・処理の進捗を表す進捗状況（Progress）
30

【0068】

ここで、画像変換APP144は、CreatePrintJobSet関数の入力パラメータの中からクラウド印刷APP146が発行したプリンタIDと印刷ジョブIDとを取得する。そして、画像変換APP144は、クラウド印刷APP146が発行したプリンタIDと印刷ジョブIDとをセッション管理データベースに保存する。

また、画像変換APP144は、CreatePrintJobSet関数の入力パラメータの中からクラウド印刷サービスを利用するためのアクセストークンを取得する。そして、画像変換APP144は、クラウド印刷サービスを利用するためのアクセストークンをセッション管理データベースに保存する。
40

さらに、画像変換APP144は、CreatePrintJobSet関数の入力パラメータの中から印刷画像（PDFファイル）の格納先のURLを取得し、印刷画像（PDFファイル）の格納先のURLをセッション管理データベースに保存する。

【0069】

続いて、画像変換APP144は、セッション管理データベース内の処理の進捗を表す進捗状況（Progress）に処理中を表す「1」をセットし（S1003）、S1004に処理を移す。S1004では、画像変換APP144は、ダウンロード・画像変換スレッドを生成し、S1005に処理を移す。なお、ダウンロード・画像変換スレッドの処理に関しては、図10（B）で後述する。

続いて、画像変換APP144は、HTTPSのPOST要求に対するレスポンスのB
50

ODYに含める応答メッセージに、この関数が成功して正常終了したことを表すS U C C E S SとS 1 0 0 2で生成したセッションIDとをセットする(S 1 0 0 5)。そして、画像変換APP144は、POST要求の発行元(本例ではプロキシAPP143)に対して応答メッセージを送信する。そして、画像変換APP144は、CreatePrintJobSet関数の処理を終了する(S 1 0 0 6)。

【0070】

また、S 1 0 0 4において画像変換APP144がダウンロード・画像変換スレッドを生成すると、CreatePrintJobSet関数とは別のスレッドが起動される。そして、このスレッドの中で図10(B)に示すダウンロード・画像変換スレッドの処理が実行される。10

図10(B)に示すように、画像変換APP144は、ダウンロード・画像変換スレッドの処理を開始する(S 1 0 1 1)。そして、画像変換APP144は、クラウド印刷サービスを利用するためのアクセストークンを利用し、SSL通信で暗号化されたHTTPのGETメソッドにて印刷画像(PDFファイル)の格納先のURLにアクセスする。即ち、画像変換APP144は、クラウド印刷サービスから印刷画像(PDFファイル)をダウンロードして一時的にRAM201に保存する(S 1 0 1 2)。

【0071】

また、画像変換APP144は、ダウンロードを完了した後、印刷画像(PDFファイル)をMFP3で印刷可能な形式であるJPEGファイルに変換する。なお、PC5の HDD202には、プリンタやMFP毎に、印刷可能な形式を示す設定情報が予め記憶されている。20

そして、画像変換APP144は、クラウド印刷APP146が発行したプリンタIDと印刷ジョブIDとから構成されるフォルダにそのJPEGファイルを保存する(S 1 0 1 3)。

また、S 1 0 1 3において、画像変換APP144は、セッション管理データベース内の画像変換APP144へのアクセストークンに、画像変換APP144へのアクセストークンをセットする。画像変換APP144へのアクセストークンは、クラウド印刷APP146が発行したプリンタIDが関連付けられているプリンタやMFP(本例ではMFP3)が印刷画像(JPEGファイル)にアクセスするためのアクセストークンである。さらに、S 1 0 1 3において、画像変換APP144は、セッション管理データベース内の印刷ジョブアクセスURLに、印刷画像(JPEGファイル)の格納先のURL(1ページ目)をセットする。30

【0072】

[例]

プリンタID : a b c d e

印刷ジョブID : 1 2 3 4

印刷画像(JPEGファイル)の総ページ数 : 3ページ

PC2内のデータストレージ :

C:¥abcde1234¥

p1.jpg

p2.jpg

p3.jpg

URL : http://www.abc.xxx/printdata/?pi=abcde&ji=1234&p=1

【0073】

このURLは、1ページ目のJPEGファイル(p1.jpg)を表すので、MFP3は、各ページの印刷画像を取得するときは、ページ番号を表すクエリパラエータ「p」に、取得するページ番号を指定する。S 1 0 1 3において、画像変換APP144は、印刷画像(PDFファイル)を変換し、全てのページの印刷画像(JPEGファイル)の変換が完了した後、S 1 0 1 2でダウンロードした印刷画像(PDFファイル)をRAM201から削除する。4050

続いて、画像変換APP144は、S1011～S1013の何れかの処理においてエラーを検出したか否かを判断する(S1014)。このとき、画像変換APP144は、エラーを検出したと判断した場合、S1016に処理を移し、他方、エラーを検出しなかつたと判断した場合、正常終了と判断してS1015に処理を移す。

【0074】

S1015では、画像変換APP144は、セッション管理データベース内の進捗状況(Progress)に正常終了を表す「0」をセットし、ダウンロード・画像変換スレッドの処理を終了する(S1017)。S1016では、画像変換APP144は、セッション管理データベース内の進捗状況(Progress)にエラー終了を表す「2」をセットし、ダウンロード・画像変換スレッドの処理を終了する(S1017)。10

S1017において、画像変換APP144は、セッション管理データベースからこの印刷ジョブに関する情報を削除し、セッション管理データベースを更新する。

【0075】

次に、画像変換APP144がエクスポートするWebサービスAPIの処理について図11を用いて説明する。図11(A)は、ConfirmProgress関数の処理に係るフローチャートの一例を示す図である。図11(B)は、JobCompleted関数の処理に係るフローチャートの一例を示す図である。

ここで、ConfirmProgress関数については、HTTPのGETメソッドが利用されている。JobCompleted関数については、SSL通信で暗号化されたHTTPSのPOSTメソッドが利用されている。これらのフローチャートに係るプログラムは、PC5に組み込まれ、HDD202に記憶されており、CPU204によりRAM201に読み出されて実行される。20

【0076】

図6のS614において、プロキシAPP143がConfirmProgress関数をコールすると、画像変換APP144が図11(A)に示すConfirmProgress関数の処理を実行する。

図11(A)に示すように、画像変換APP144は、ConfirmProgress関数の処理を開始する(S1101)。このとき、ConfirmProgress関数に対するHTTPのGET要求には、クエリパラメータとしてセッションIDとシークエンスとが含まれている。シークエンスは、GET要求の発行元(本例ではプロキシAPP143)が生成する任意の番号である。30

続いて、画像変換APP144は、クエリパラメータに含まれているセッションIDを検索キーとして、セッション管理データベース内の進捗状況(Progress)を確認する(S1102)。続いて、画像変換APP144は、進捗状況(Progress)に正常終了を表す「0」がセットされていると判断した場合(S1103でYESの場合)、S1104に処理を移す。他方、画像変換APP144は、「0」以外(「1」または「2」)がセットされていると判断した場合(S1103でNOの場合)、S1105に処理を移す。

【0077】

S1104では、画像変換APP144は、HTTPのGET要求に対するレスポンスのBODYに含める応答メッセージに「処理完了応答」をセットし、S1108に処理を移す。S1105では、画像変換APP144は、進捗状況(Progress)を確認する。このとき、画像変換APP144は、進捗状況(Progress)に処理中を表す「1」がセットされていると判断した場合、S1106に処理を移し、他方、「1」以外(「2」)がセットされている判断した場合、S1107に処理を移す。40

S1106では、画像変換APP144は、HTTPのGET要求に対するレスポンスのBODYに含める応答メッセージに「処理中応答」をセットし、S1108に処理を移す。S1107では、画像変換APP144は、HTTPのGET要求に対するレスポンスのBODYに含める応答メッセージに「エラー応答」をセットし、S1108に処理を移す。50

S 1 1 0 8 では、画像変換 A P P 1 4 4 は、G E T 要求の発行元（本例ではプロキシ A P P 1 4 3 ）に対して応答メッセージを送信し、C o n f i r m P r o g r e s s 関数の処理を終了する（S 1 1 0 9 ）。

【0078】

ここで、「処理完了応答」、「処理中応答」、及び「エラー応答」の詳細について説明する。これらの応答は、それぞれ、S 1 1 0 1 において G E T 要求のクエリパラメータに含まれていたシーケンスを用いて演算された結果であり、例えば、次に示す情報を文字列に変換した文字列情報で構成される。

「処理完了応答」 = シーケンス

「処理中応答」 = シーケンス + 1

「エラー応答」 = シーケンス + 2

10

【0079】

このように、H T T P の G E T 要求に対するレスポンスとして X M L を使って定義した可視的な情報等ではなく一般には意味がわからない文字列情報を用いることで、W e b サービス A P I に対する不正なアクセスによる誤動作や不正な制御を防止することができる。

なお、このような不正なアクセスを防止する構成は、この例に限られるものではなく、ハッシュ化等、より複雑な演算を組み合わせてもよい。

【0080】

また、図 8 の S 8 0 3 において、プロキシ A P P 1 4 3 が J o b C o m p l e t e d 関数をコールすると、画像変換 A P P 1 4 4 が図 1 1 (B) に示す J o b C o m p l e t e d 関数の処理を実行する。

20

図 1 1 (B) に示すように、画像変換 A P P 1 4 4 は、J o b C o m p l e t e d 関数の処理を開始すると（S 1 1 2 1 ）、印刷画像（J P E G ファイル）を削除し（S 1 1 2 2 ）、S 1 1 2 3 に処理を移す。

S 1 1 2 3 では、画像変換 A P P 1 4 4 は、H T T P S の P O S T 要求に対するレスポンスの B O D Y に含める応答メッセージに、正常終了したことを表す S U C C E S S をセットする。また、S 1 1 2 3 では、画像変換 A P P 1 4 4 は、P O S T 要求の発行元（本例ではプロキシ A P P 1 4 3 ）に対して応答メッセージを送信する。そして、画像変換 A P P 1 4 4 は、J o b C o m p l e t e d 関数の処理を終了する（S 1 1 2 4 ）。

30

【0081】

次に、画像変換 A P P 1 4 4 がエクスポートする W e b サービス A P I の処理について図 1 2 を用いて説明する。図 1 2 は、C r e a t e P r i n t J o b R e s u l t 関数の処理に係るフローチャートの一例を示す図である。

C r e a t e P r i n t J o b R e s u l t 関数については、S S L 通信で暗号化された H T T P S の P O S T メソッドが利用されている。このフローチャートに係るプログラムは、P C 5 に組み込まれ、H D D 2 0 2 に記憶されており、C P U 2 0 4 により R A M 2 0 1 に読み出されて実行される。

【0082】

図 7 の S 7 0 1 において、プロキシ A P P 1 4 3 が C r e a t e P r i n t J o b R e s u l t 関数をコールすると、画像変換 A P P 1 4 4 が図 1 2 に示す C r e a t e P r i n t J o b R e s u l t 関数の処理を実行する。

40

図 1 2 に示すように、画像変換 A P P 1 4 4 は、C r e a t e P r i n t J o b R e s u l t 関数の処理を開始する（S 1 2 0 1 ）。続いて、画像変換 A P P 1 4 4 は、H T T P S の P O S T 要求に対するレスポンスの B O D Y に含める応答メッセージに、画像変換 A P P 1 4 4 へのアクセストークンをセットし（S 1 2 0 2 ）、S 1 2 0 3 に処理を移す。

【0083】

S 1 2 0 3 では、画像変換 A P P 1 4 4 は、P O S T 要求に対するレスポンスの B O D Y に含める応答メッセージに、印刷ジョブアクセス U R L （J P E G ファイルの格納先の

50

URL)をセットし、S1204に処理を移す。

S1204では、画像変換APP144は、POST要求に対するレスポンスのBODYに含める応答メッセージに、印刷画像(JPEGファイル)の総ページ数をセットし、S1205に処理を移す。S1205では、画像変換APP144は、POST要求の発行元(本例ではプロキシAPP143)に対して応答メッセージを送信し、CreatePrintJobResult関数の処理を終了する(S1206)。

【0084】

図13は、WebサービスAPIのコーリングシーケンスの一例を示す図である。図13(A)は、プロキシAPP143とMFP3との間のコーリングシーケンスの一例を示す図である。図13(B)は、プロキシAPP143と画像変換APP144との間のコーリングシーケンスの一例を示す図である。

図13(A)に示すように、プロキシAPP143とMFP3との間では、MFP3は、プロキシAPP143がエクスポートしているWebサービスAPIをコールしている。即ち、CheckPrintJob関数、GetPrintJob関数、EndPrintJob関数の順で、プロキシAPP143がエクスポートしているWebサービスAPIがコールされる。

【0085】

ここで、CheckPrintJob関数については(ポーリング処理だけが)、暗号化されていないHTTPのGETメソッドが利用され、それ以外の関数については全てSSL通信で暗号化されたHTTPSのGETメソッドが利用されている。

このように、印刷ジョブの有無を確認するだけのためのポーリングに、暗号化されていないHTTPメソッドを利用することで、ポーリングに要する時間が低減され、ユーザの操作性に優れた周辺装置制御システムを実現することができる。また、ポーリングにかかる負荷を低減することができるので、例えば負荷に対応するための構成が必要なくなるので、システムの構築やプリンタやMFPのファームウェアの実装が容易となり、開発コストを削減できる。

【0086】

また、図13(B)に示すように、プロキシAPP143と画像変換APP144との間では、プロキシAPP143は、画像変換APP144がエクスポートしているWebサービスAPIをコールしている。即ち、CreatePrintJobSet関数、ConfirmProgress関数、CreatePrintJobResult関数、JobCompleted関数の順で画像変換APP144がエクスポートしているWebサービスAPIがコールされる。

ここで、ConfirmProgress関数については(ポーリング処理だけが)、暗号化されていないHTTPのGETメソッドが利用され、それ以外の関数については全てSSL通信で暗号化されたHTTPSのPOSTメソッドが利用されている。

このように、処理の進捗状況を確認するだけのためのポーリングに、暗号化されていないHTTPメソッドを利用することで、ポーリングに要する時間が低減され、ユーザの操作性に優れた周辺装置制御システムを実現することができる。また、ポーリングにかかる負荷を低減することができるので、例えば負荷に対応するための構成が必要なくなるので、システムの構築やアプリケーションの実装が容易となり、開発や運用コストを削減できる。

【0087】

<第2の実施形態>

第1の実施形態では、あるMFPに対して1つのユーザアカウント及びそのパスワードが関連付けて登録され、ユーザアカウント及びパスワードによる認証を経て取得されたアクセストークンが用いられて利用される周辺装置制御システムの一例を説明した。本実施形態では、あるMFP(例えばMFP3)に対して複数のユーザアカウントを関連付けて登録することにより、当該MFPを複数のユーザで共有して利用することが可能な周辺装置制御システムの一例を説明する。なお、本実施形態では、第1の実施形態の構成と同一

10

20

30

40

50

の構成には同一の符号を用いてその説明を適宜省略する。

図14は、MFP3の表示部21の画面の一例（クラウド印刷セットアップ画面1401）を示す図である。クラウド印刷セットアップ画面1401は、クラウド印刷サービスを利用するためのプリンタの登録などのセットアップに係る画面である。プリンタ登録操作部（1402、1405、1408、1411、1414）の各々は、ユーザ名入力部（1403、1406、1409、1412、1415）と、登録ボタン（1404、1407、1410、1413、1416）とを備えている。

【0088】

現時点では、MFP3については3人で共有されているので、ユーザ名入力部（1403、1406、1409）には、ユーザ名として「User A」、「User B」、「User C」がそれぞれセットされている。他方、ユーザ名入力部1412及びユーザ名入力部1415には、ユーザ名が何もセットされていない。10

登録ボタン1404は、MFP3をユーザ名入力部1403のユーザ「User A」用として、クラウド印刷APP146が提供するクラウド印刷サービスに登録するときに操作されるユーザインターフェース（UI）である。登録ボタン1407は、MFP3をユーザ名入力部1406のユーザ「User B」用として、クラウド印刷APP146が提供するクラウド印刷サービスに登録するときに操作されるUIである。登録ボタン1410は、MFP3をユーザ名入力部1409のユーザ「User C」用として、クラウド印刷APP146が提供するクラウド印刷サービスに登録するときに操作されるUIである。20

【0089】

登録ボタン1413は、MFP3をユーザ名入力部1412のユーザ用として、クラウド印刷APP146が提供するクラウド印刷サービスに登録するときに操作されるUIである。登録ボタン1416は、MFP3をユーザ名入力部1415のユーザ用として、クラウド印刷APP146が提供するクラウド印刷サービスに登録するときに操作されるUIである。

なお、現時点では、ユーザ名入力部1412及びユーザ名入力部1415にはユーザ名が何もセットされていないので、登録ボタン1413及び登録ボタン1416は機能しない。

【0090】

図15は、PC1のLCD205の画面（Webブラウザ9）の一例（プリンタ登録ページ1501）を示す図である。プリンタ登録ページ1501は、クラウド印刷サービスにプリンタを登録する際のユーザ認証に係る画面である。ユーザは、プリンタ登録ページ1501を使って、クラウド印刷APP146が提供するクラウド印刷サービスを利用する際に必要なユーザ認証及びプリンタの仮登録を行う。30

URL入力部1502は、URLの入力を受け付ける。URL入力部1502で入力されたURLのWebサイトのページがWebブラウザ9に表示される。ユーザアカウント入力部1503は、ユーザアカウントの入力を受け付ける。パスワード入力部1504は、パスワードの入力を受け付ける。ユーザは、クラウド印刷サービスを利用するためには必要なユーザアカウントとパスワードとをここに入力する。40

認証コード（トークン）入力部1505は、仮認証コード（トークン）の入力を受け付ける。ユーザは、クラウド印刷APP146で発行されたプリンタ登録用の仮認証コード（トークン）をここに入力する。登録ボタン1506は、プリンタの仮登録を受け付ける。ユーザにより登録ボタン1506が押下されると、クラウド印刷サービスにプリンタが仮登録される。キャンセルボタン1507は、プリンタの仮登録の中止を受け付ける。ユーザによりキャンセルボタン1507が押下されると、クラウド印刷サービスにプリンタを仮登録する処理がキャンセルされる。

【0091】

図16は、クラウド印刷サービスにプリンタを登録する処理に係るフローチャートの一例を示す図である。このフローチャートに係るプログラムは、PCまたはMFP内に組み50

込まれている。P C 上で動作するアプリケーションのプログラムは、H D D 2 0 2 に記憶されており、C P U 2 0 4 によりR A M 2 0 1 に読み出されて実行される。M F P 上で動作するアプリケーションのプログラムは、R O M 1 6 に記憶されており、R A M 1 7 に読み出され、C P U 2 5 により実行される。

このフローチャートにおいて、縦の列が処理の実行元を表し、本実施形態では図面の左側から、ユーザ（W e b ブラウザ9）、文書作成A P P 1 4 5、クラウド印刷A P P 1 4 6、プロキシA P P 1 4 3、画像変換A P P 1 4 4、M F P 3 を記す。以下では、ユーザがM F P 3 とP C 1 を使って、クラウド印刷A P P 1 4 6 が提供するクラウド印刷サービスにM F P 3 を登録するケースを例に挙げて、通信制御方法について説明する。

【0 0 9 2】

ユーザがM F P 3 の表示部2 1 を操作して、図1 4 に示すクラウド印刷セットアップ画面1 4 0 1 を開き、ユーザ名入力部1 4 0 3 のユーザ「U s e r _ A 」用の登録ボタン1 4 0 4 を押下すると、図1 6 において、プリンタの登録処理が開始される（S 1 6 0 1 ）。

プリンタの登録処理が開始されると、M F P 3 は、クラウド印刷A P P 1 4 6 にプリンタの仮登録を要求する（S 1 6 0 2 ）。クラウド印刷A P P 1 4 6 は、要求を受け取ると、プリンタ登録用の仮認証コード（トークン）と、プリンタ登録ページのU R L（登録ページのU R L）と、ポーリング用のU R L（ポーリングU R L）とを発行する（S 1 6 0 3 ）。M F P 3 は、クラウド印刷A P P 1 4 6 から仮認証コード（トークン）とプリンタ登録ページのU R LとポーリングU R Lとを受信すると、仮認証コード（トークン）及びプリンタ登録ページのU R Lを表示部2 1 に表示し（S 1 6 0 4 ）、S 1 6 0 5 に処理を移す。

次に、ユーザは、クラウド印刷A P P 1 4 6 が提供するクラウド印刷サービスにM F P 3 を仮登録するためのユーザ認証処理を開始する（S 1 6 1 0 ）。より具体的には、ユーザは入力部を操作して、P C 1 内のW e b ブラウザ9を起動し、U R L入力部1 5 0 2 にプリンタ登録ページのU R Lを入力すると、W e b ブラウザ9は、クラウド印刷A P P 1 4 6 から指定されたプリンタ登録ページを開く（S 1 6 1 1 ）。

【0 0 9 3】

次に、ユーザは入力部を操作して、クラウド印刷サービスを利用するため事前に作成しておいたユーザアカウントとパスワードとを、ユーザアカウント入力部1 5 0 3 及びパスワード入力部1 5 0 4 にそれぞれ入力する。また、認証コード（トークン）入力部1 5 0 5 に仮認証コード（トークン）を入力する。ユーザが入力部を操作して登録ボタン1 5 0 6 を押下すると、W e b ブラウザ9は、このプリンタ登録ページ1 5 0 1 で入力されたユーザアカウント、パスワード、仮認証コード（トークン）をクラウド印刷A P P 1 4 6 に送信する（S 1 6 1 2 ）。

クラウド印刷A P P 1 4 6 は、W e b ブラウザ9から送信されたユーザアカウント、パスワード、仮認証コード（トークン）を受信すると、ユーザ認証を行う。更に、クラウド印刷A P P 1 4 6 は、認証に成功した場合にはクラウド印刷サービスを利用する際に必要なアクセストークンを発行し、クラウド印刷サービスにM F P 3 を仮登録する（S 1 6 1 3 ）。これでユーザ認証処理が終了し、クラウド印刷サービスにM F P 3 が仮登録された状態となる（S 1 6 1 4 ）。

【0 0 9 4】

S 1 6 0 5 では、M F P 3 は、ポーリングU R Lにアクセスし、クラウド印刷A P P 1 4 6 にプリンタ（M F P 3 ）の仮登録の状況（登録状況）を確認するための問合せを行う。クラウド印刷A P P 1 4 6 は、問合せを受け付けると、プリンタ（M F P 3 ）の仮登録の状況を確認する。

このとき、クラウド印刷A P P 1 4 6 は、プリンタ（M F P 3 ）の仮登録が完了していると判断した場合、プリンタ（M F P 3 ）用のプリンタI Dを発行して当該プリンタを本登録する。更に、クラウド印刷A P P 1 4 6 は、このプリンタI DとアクセストークンとをM F P 3 に送信し、プリンタの本登録が完了した旨の結果をM F P 3 に返信する（S 1 5 0

606)。なお、プリンタIDは、ユーザ名入力部1403で指定されているユーザ「User A」専用としてクラウド印刷APP146がMFP3に割り当てた一意の値で表されたものである。

【0095】

他方、クラウド印刷APP146は、プリンタ(MFP3)の仮登録が完了していないと判断した場合、プリンタの本登録が未完了である旨の結果をMFP3に返信する(S1606)。

MFP3は、プリンタ(MFP3)の本登録が完了した旨の結果を受信したと判断した場合(S1607でYESの場合)、S1608に処理を移し、他方、未完了である旨の結果を受信したと判断した場合(S1607でNOの場合)、S1605に処理を移す。
S1608では、MFP3は、クラウド印刷サービスへのプリンタの本登録が完了した旨を表示部21に表示し(S1608)、プリンタの登録処理を終了する(S1609)。

【0096】

また、ユーザがMFP3の表示部21を操作して、クラウド印刷セットアップ画面1401を開き、ユーザ名入力部1406のユーザ「User B」用の登録ボタン1407を押下すると、上述と同様にプリンタの登録処理とユーザ認証処理とが行われる。この際、S1606において、MFP3用のプリンタIDが発行されるが、このプリンタIDは、ユーザ名入力部1406で指定されているユーザ「User B」専用としてクラウド印刷APP146がMFP3に割り当てた一意の値で表されたものである。

また、ユーザがMFP3の表示部21を操作して、クラウド印刷セットアップ画面1401を開き、ユーザ名入力部1409のユーザ「User C」用の登録ボタン1410を押下すると、上述と同様にプリンタの登録処理とユーザ認証処理とが行われる。この際、S1606において、MFP3用のプリンタIDが発行されるが、このプリンタIDは、ユーザ名入力部1409で指定されているユーザ「User C」専用としてクラウド印刷APP146がMFP3に割り当てた一意の値で表されたものである。

このように、本実施形態では、クラウド印刷APP146が、1台のMFP(MFP3)に対して複数のユーザ(ユーザアカウント)毎にそれぞれ異なる一意の値のプリンタIDを発行する。これにより、1台のMFPを複数のユーザで共有して利用することができる。MFPの利用目的や利用範囲が広がるとともに、操作性にも優れるなどの効果が得られる。なお、本実施形態においても、第1の実施形態の効果は得られる。

【0097】

<第3の実施形態>

第1の実施形態では、クラウド印刷APP146が提供するクラウド印刷サービスにおける印刷ジョブ通知をプロキシAPP143が受信する構成を例に挙げて説明した。この例では、MFP3が、図6のS618～S621において一定の時間間隔(この例では1分間隔)でポーリングすることにより、プロキシAPP143に対して印刷ジョブの有無の確認を行っている。

そのため、プロキシAPP143が印刷ジョブ通知を受信してから、必ずしも直ちにMFP3が印刷を開始することができず、最長でポーリングのインターバル分(この例では1分)の遅れが発生する事態が生じ得る。

そこで、本実施形態では、印刷の実行が指示されてから印刷が開始されるまでの時間(印刷開始時間)を短縮するとともに、印刷の実行が指示されてから印刷処理が終了するまでの時間(トータル印刷時間)を低減可能な周辺装置制御システムの一例を説明する。なお、本実施形態では、第1の実施形態の構成と同一の構成には同一の符号を用いてその説明を適宜省略する。

【0098】

図17は、本周辺装置制御システム全体の処理に係るフローチャートの一例を示す図である。このフローチャートに係るプログラムは、PCまたはMFP内に組み込まれている。PC上で動作するアプリケーションのプログラムは、HDD202に記憶されており、CPU204によりRAM201に読み出されて実行される。MFP上で動作するアプリ

10

20

30

40

50

ケーションのプログラムは、ROM16に記憶されており、RAM17に読み出され、CPU25により実行される。

このフローチャートにおいて、縦の列が処理の実行元を表し、本実施形態では図面の左側から、ユーザ（Webブラウザ9）、文書作成APP145、クラウド印刷APP146、プロキシAPP143、画像変換APP144、MFP3を記す。以下では、ユーザがPC1を使ってドキュメントの印刷をMFP3で行うケース（ドキュメントの印刷に係る一連の処理）を例に挙げて通信制御方法について説明する。

このフローチャートにおいて、図6、図7に示すステップ番号と同じステップ番号が付与されている処理は、基本的には図6、図7で説明した処理と同一であるので、その説明を省略する。また、特に説明がない部分に関しては図6、図7で説明した内容と同一であるものとする。10

【0099】

図17において、S601でユーザ側から見たドキュメントの印刷が開始されると、S602～S609の処理が順に行われる。S609において、クラウド印刷APP146は、選択されているプリンタ選択肢に関連付けられているプリンタ（MFP3）に印刷ジョブ通知（開始通知の一例）を送信する。

印刷ジョブ通知には、例えば、Extensible Messaging and Presence Protocol（略称：XMPP）というインスタントメッセンジャーで使用されているプロトコルを利用することができる。ステップS609において、クラウド印刷APP146がプリンタ（MFP3）に送信する印刷ジョブ通知は、暗号化されていないXMPPが利用されるものとする。20

MFP3は、印刷ジョブ通知を受信すると（S1708）、S1709に処理を移す。S1709では、MFP3は、プロキシAPP143に印刷ジョブの取得を要求し、印刷ジョブの有無を確認する処理（S1701）を開始（起動）する。

プロキシAPP143は、MFP3からの印刷ジョブの取得の要求を受信すると、クラウド印刷APP146がエクスポートしているWebサービスAPIのFetchPrintJob関数をコールする（S1710）。そして、図6のS611に処理が移され、図6のS611以降の処理が行われる。

【0100】

また、印刷ジョブの有無を確認する処理を開始すると（S1701）、MFP3は、プロキシAPP143がエクスポートしているWebサービスAPIのCheckPrintJob関数をコールする（S1702）。S1702において、MFP3が、この関数をコールするときに、HTTPのGET要求のクエリパラメータとしてプロキシAPP143に渡す情報（確認情報の一例）は、図6のS618で説明した情報と同一である。30

プロキシAPP143は、CheckPrintJob関数がコールされると、図9（A）に示すCheckPrintJob関数の処理を実行し、そのレスポンス（応答情報の一例）をMFP3に返す（S1703）。

続いて、MFP3は、プロキシAPP143からのCheckPrintJob関数のレスポンスを受信すると、レスポンス内の応答メッセージに含まれている情報（「印刷ジョブ有り応答」又は「印刷ジョブ無し応答」）を確認する（S1704）。S1704において、MFP3は、「印刷ジョブ有り応答」を受信したと判断した場合、図7のS708に処理を移し、他方、「印刷ジョブ無し応答」を受信したと判断した場合、S1705に処理を移す。40

【0101】

S1705では、MFP3は、S1701において印刷ジョブの有無を確認する処理が開始されてからの経過時間を確認し、S1706に処理を移す。S1706では、MFP3は、経過時間が最長待ち時間（例えば5分間）を経過していると判断した場合（YESの場合）、図6のS618に処理を移し、他方、経過していないと判断した場合（NOの場合）、S1707に処理を移す。

S1707では、MFP3は、一定の時間（例えば1秒間）待った後、S1702に処理を移す。50

理を移す。このように、MFP3は、印刷ジョブが生成される際に、プロキシAPP143がエクスポートしているCheckPrintJob関数を一定の時間間隔で（定期的に）ポーリングする。

ここで、第1の実施形態の図6のS618～S621におけるポーリングのインターバルが1分間であったのに対し、本実施形態ではポーリングのインターバルが1秒間である。この構成により、印刷開始時間を大幅に短縮することができ、トータル印刷時間も低減することができる。

【0102】

また、第1の実施形態では、印刷ジョブの有無によらずポーリングが行われる。よって、膨大な数のプリンタからプロキシAPP143に対して頻繁にポーリングが行われると、プロキシAPP143の負荷が非常に大きくなり、大規模なシステムが必要になることによるクラウドシステム（周辺装置制御システムの一例）のコストアップに至る。また、場合によっては、クラウドシステムが機能しなくなるほど負荷がかかる可能性もあり、システム全体が不安定になってしまふという懸念もある。

しかしながら、本実施形態では、印刷ジョブが生成されるときに限ってポーリングが行われるため、全体からみた場合にごく少数のプリンタが対象となるので、ポーリングのインターバルを短くしてもシステムに与える負荷が少ない。したがって、本実施形態によれば、システムを大規模にする必要もなく、クラウドシステムの生産費、運用費などの費用を抑える（第1の実施形態のクラウドシステムの費用よりも下げる）ことができる。また、本実施形態によれば、システム全体を安定した状態で維持することも可能となる。

【0103】

また、S1706では、印刷ジョブの有無を確認する処理が開始されてからの経過時間（或いは印刷ジョブ通知を受信した時点からの経過時間でもよい。）が最長待ち時間を超えた場合、クラウド印刷サービスに異常が発生した可能性があることを想定している。即ち、経過時間が最長待ち時間を超えた以降のポーリングのインターバルを十分に長くすることにより、長時間にわたって短いインターバルでのポーリングが続くことを防止し、システムを安定した状態で維持することができる。

このように、本実施形態では、MFP3は、クラウド印刷APP146からの印刷ジョブ通知を受信すると、その時点から極めて短いインターバルで印刷ジョブの有無を確認することができるので、印刷開始時間を短縮し、トータル印刷時間を低減することができる。なお、本実施形態においても、第1の実施形態の効果は得られる。

なお、ユーザが印刷を開始したときに、MFP3の電源がオフの場合、印刷ジョブ通知を受信できないケースが考えられる。このようなケースに対応するために、MFP3の電源がオンになったときに、ステップS1701の処理を少なくとも1回、或いは、ある一定時間の間だけ行うようにする。このことで、前記ユーザが開始した印刷ジョブをMFP3の電源投入時に確実に印刷することができる。

上述した実施形態をさらに改良する実施形態もある。クラウド印刷APP146がMFP3に印刷ジョブ通知を送信するということは、印刷ジョブが有るという前提で処理できると考えられる。したがって、ステップS1709において、MFP3が印刷ジョブの有無を確認する処理（S1701）を開始（起動）した後、ステップS1702の処理を実行せずにステップS1704に直接進む。そして、更にステップS1704の判定を無条件にYESで処理して、ステップS708へ進む。これにより、クラウド印刷APP146が、暗号化されていない印刷ジョブ通知をMFP3に送信する。MFP3は、この暗号化されていない印刷ジョブ通知を受信した後、プロキシAPP143がエクスポートしているGetPrintJob関数を、SSL通信で暗号化されたHTTPのGETメソッドでコールする。このような改良実施形態においても、第1の実施形態の効果が得られる。

【0104】

<第4の実施形態>

上述した実施形態では、情報処理装置としてパーソナルコンピュータを想定した。しか

10

20

30

40

50

しながら、この例に限られるものではなく、例えばDVDプレーヤー、ゲーム、セットトップボックス、インターネット家電等、同様な使用方法が可能な任意の情報処理装置（端末）に対して適用することができ、有効である。

また、上述した実施形態では、周辺装置としてMFPを例示した。しかしながら、周辺装置として他に、複写機、ファクシミリ、スキャナ、デジタルカメラ、及びこれらの複合機能を備える装置等の何れかであってもよい。

また、上述した実施形態では、OSに例としてWindowsと同等のOSを使用したが、このOSに限られるものではなく、任意のOSを使用することができる。

また、上述した実施形態では、ネットワーク4の構成例としてEthernetを用いたが、この例に限られるものではなく、他の任意のネットワーク構成であってもよい。 10

【0105】

また、上述した実施形態では、PC1、PC2、PC5、PC7とMFP3、MFP103との間のインターフェースとして、Ethernetを採用した。しかしながら、このインターフェースに限られることなく、例えば、無線LAN、IEEE1394、Blue tooth（登録商標）、USB等の任意のインターフェースを用いるようにしてもよい。

また、各種のアプリケーション（ファームウェア）における機能、上述したフローチャートに係る処理の一部、又は全部を専用のハードウェアを用いて構成してもよい。

【0106】

上述した本実施形態によれば、ポーリングによる画像変換処理の進捗状況の確認処理や、ポーリングによる印刷ジョブの有無の確認処理に時間を要することがなくなり、ユーザの操作性を向上することができる。 20

また、周辺装置制御システムにクラウドを利用した場合においても、システムの構築やアプリケーションの実装が容易で、開発や運用コストを削減できる。

さらに、MFP3のような性能の低い廉価版のプリンタや複合機のファームウェアの実装にも支障をきたすことなく、各処理に時間を要することもなくなるため、ユーザの操作性を向上することができる。

【0107】

また、本発明は、以下の処理を実行することによっても実現される。即ち、上述した実施形態の機能を実現するソフトウェア（プログラム）を、ネットワーク又は各種記憶媒体を介してシステム或いは装置に供給し、そのシステム或いは装置のコンピュータ（又はCPUやMPU等）がプログラムを読み出して実行する処理である。 30

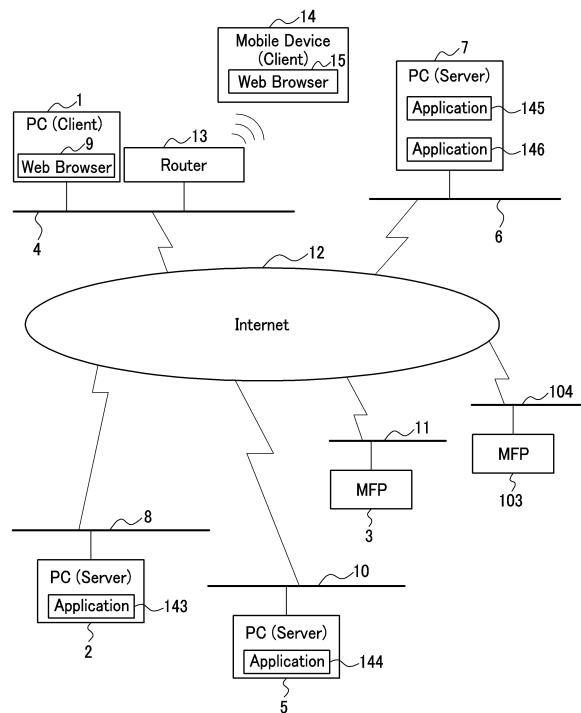
【0108】

上述した実施形態の構成によれば、装置間の通信をより適切に行うことができる。

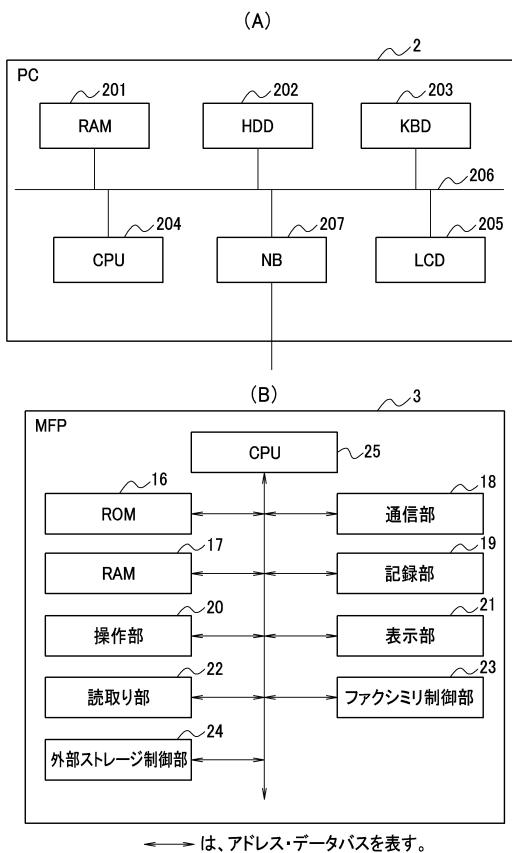
【0109】

以上、本発明の好ましい実施形態について詳述したが、本発明は係る特定の実施形態に限定されるものではなく、特許請求の範囲に記載された本発明の要旨の範囲内において、種々の変形・変更が可能である。

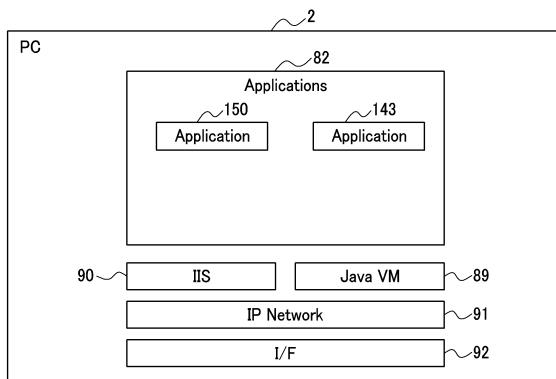
【図1】



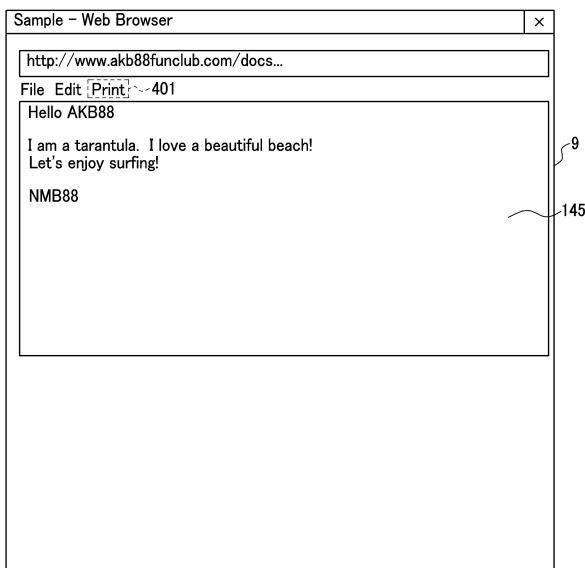
【図2】



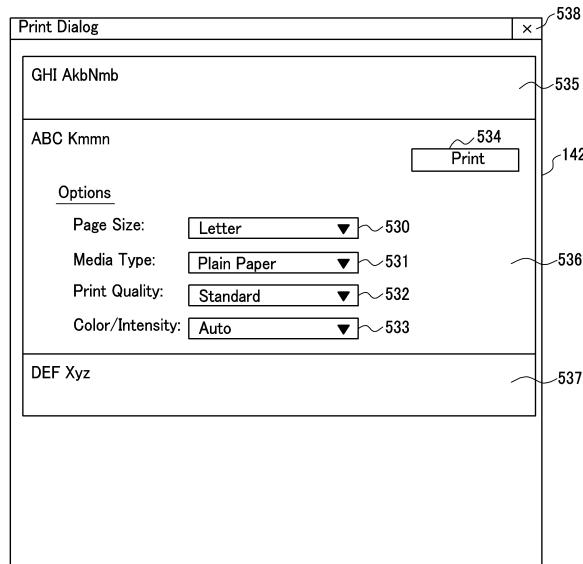
【図3】



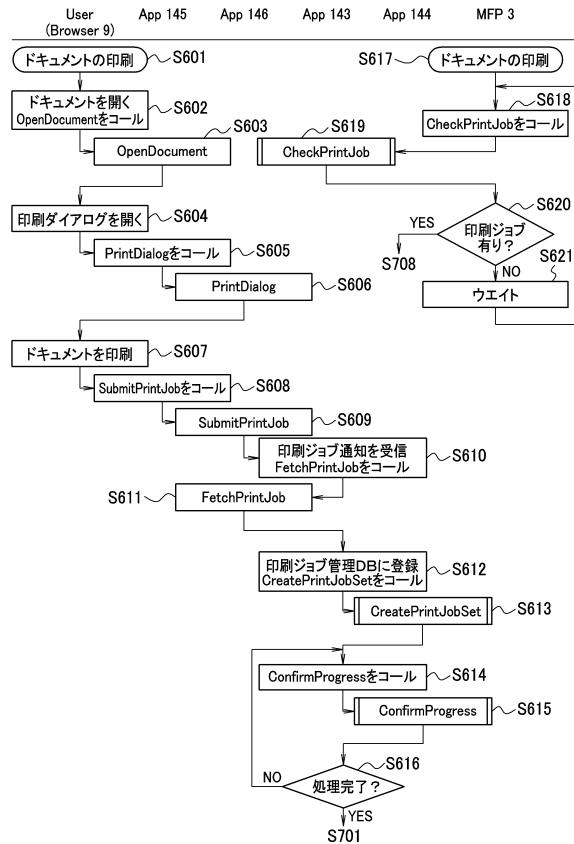
【図4】



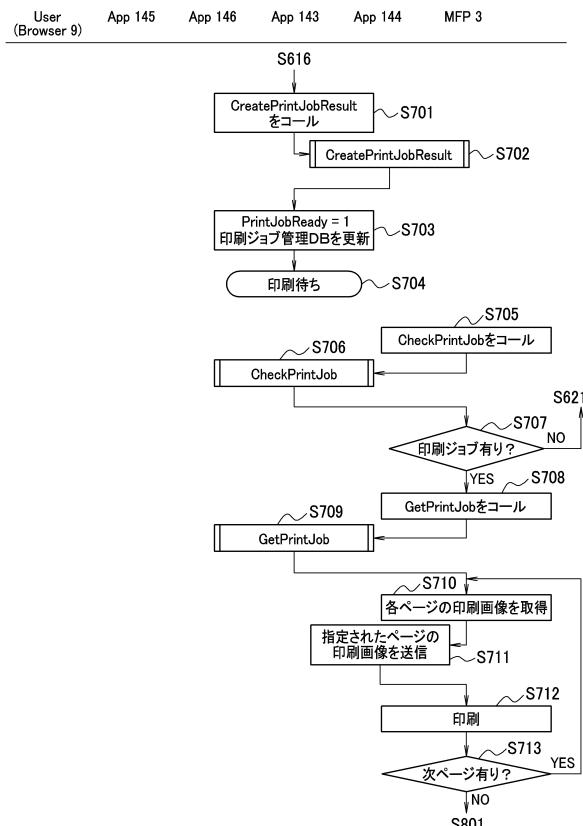
【図5】



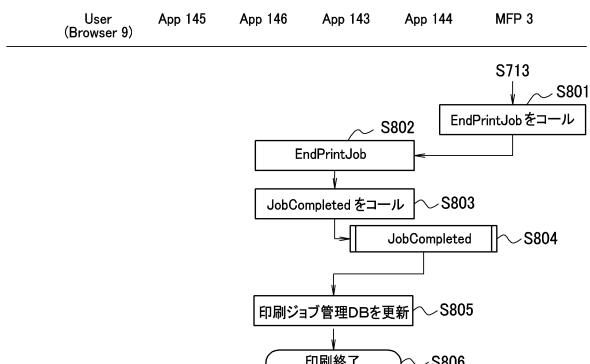
【図6】



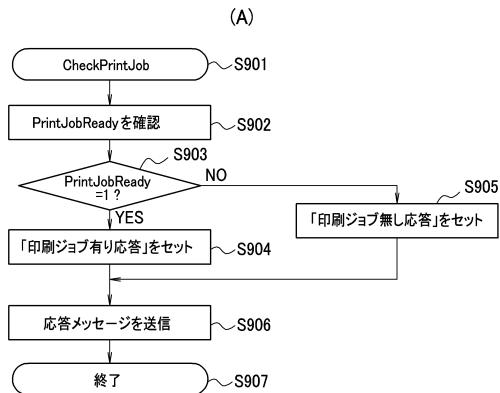
【図7】



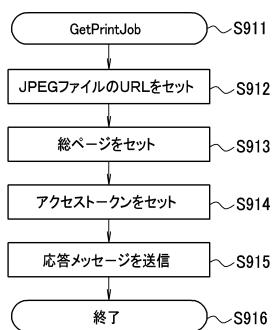
【図8】



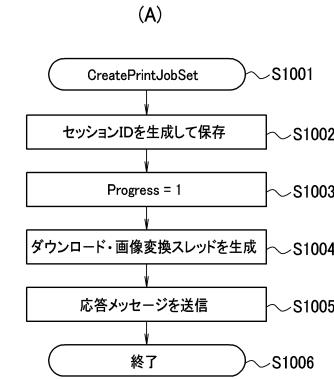
【図9】



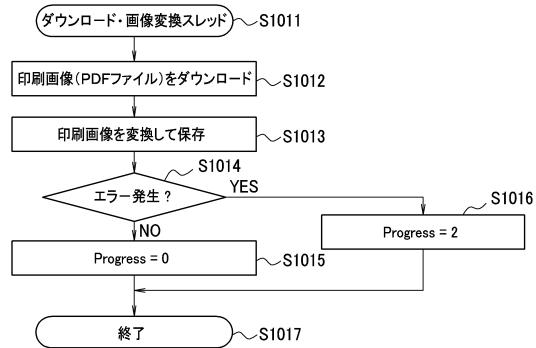
(B)



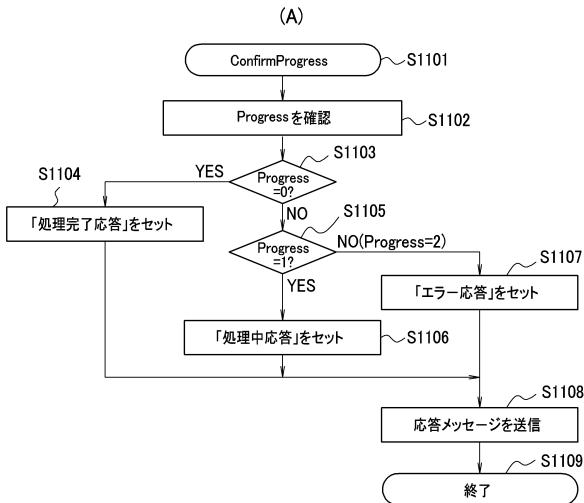
【図10】



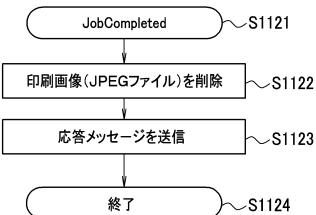
(B)



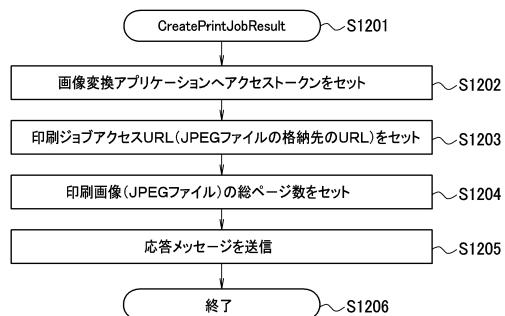
【図11】



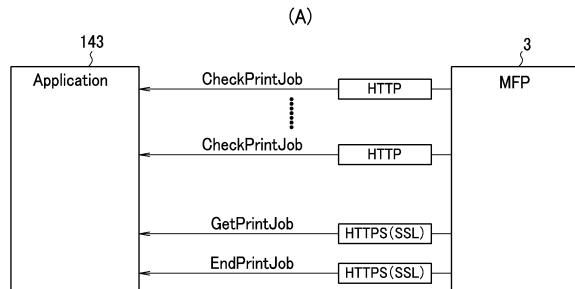
(B)



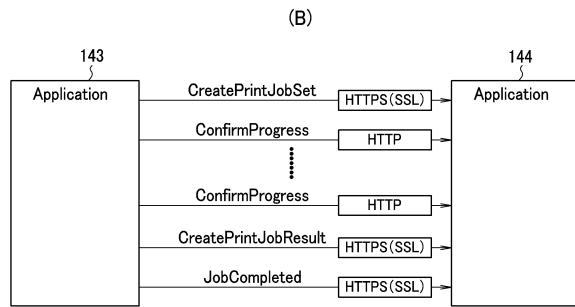
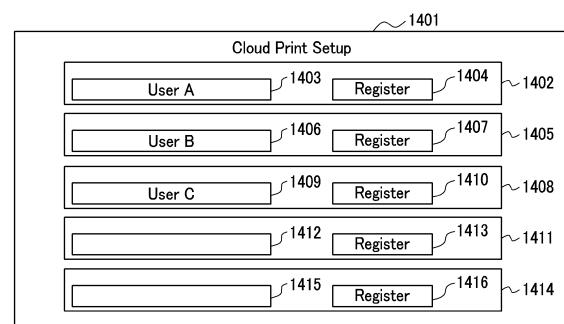
【図12】



【図13】



【図14】

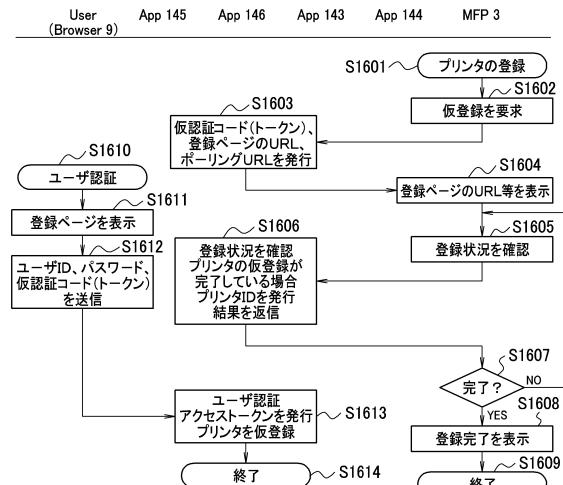


【図15】

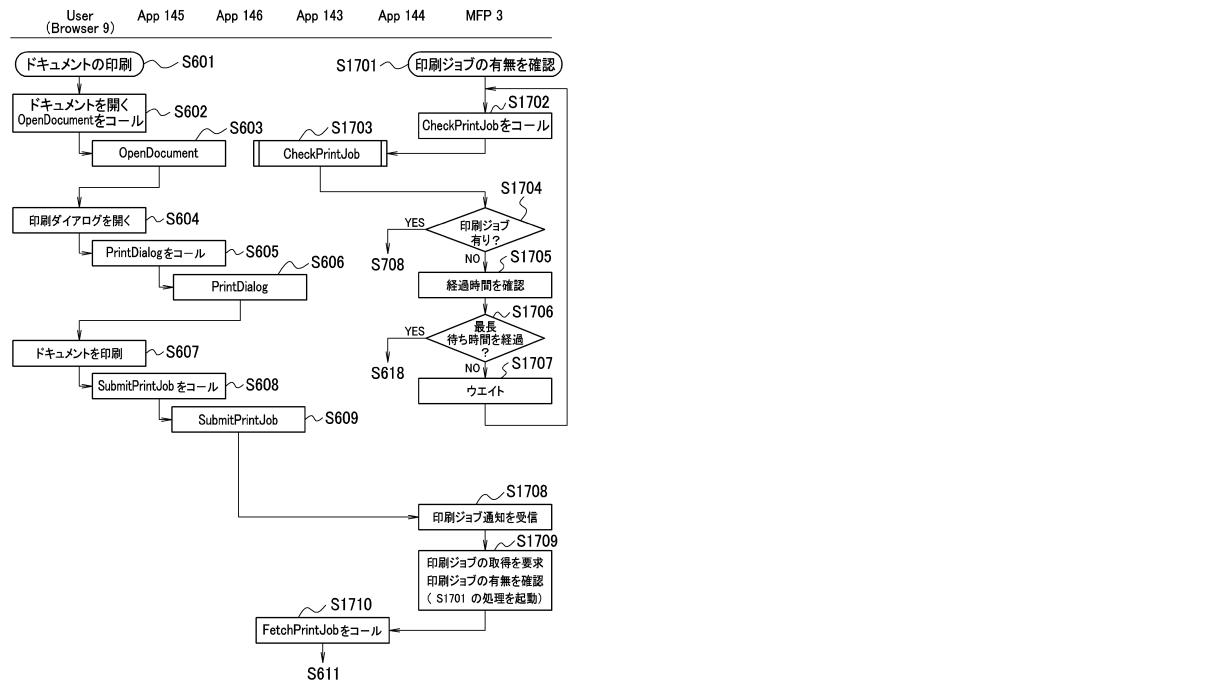
The screenshot shows a web browser window titled 'Printer Registration - Web Browser'. The URL is <http://www.akb88funclub.com/cloudprint/registration/koichiabe4649>. The page content includes fields for 'User Account' (1503), 'Password' (1504), and 'Authentication Code' (1505). At the bottom are 'Register' (1506) and 'Cancel' (1507) buttons.

9

【図16】



【図17】



フロントページの続き

(51)Int.Cl.

F I

B 4 1 J 29/00

Z

- (56)参考文献 特開2012-187866(JP,A)
特開2011-141739(JP,A)
特開2009-187535(JP,A)
特開2008-131314(JP,A)
特開2008-108195(JP,A)
米国特許出願公開第2005/0198284(US,A1)
米国特許出願公開第2005/0046879(US,A1)

(58)調査した分野(Int.Cl., DB名)

G 06 F	3 / 0 9	-	G 06 F	3 / 1 2
B 4 1 J	5 / 0 0	-	B 4 1 J	5 / 5 2
B 4 1 J	2 1 / 0 0	-	B 4 1 J	2 1 / 1 8
B 4 1 J	2 9 / 0 0	-	B 4 1 J	2 9 / 7 0
H 04 N	1 / 0 0			