

(19) 日本国特許庁(JP)

(12) 公開特許公報(A)

(11) 特許出願公開番号

特開2008-146638

(P2008-146638A)

(43) 公開日 平成20年6月26日(2008.6.26)

(51) Int. Cl.		F I			テーマコード (参考)	
G 0 6 F	3/12	(2006.01)	G 0 6 F	3/12	K	2 C 0 6 1
B 4 1 J	29/38	(2006.01)	B 4 1 J	29/38	Z	5 B 0 2 1
B 4 1 J	29/00	(2006.01)	B 4 1 J	29/00	Z	

審査請求 有 請求項の数 26 O L (全 26 頁)

(21) 出願番号	特願2007-289991 (P2007-289991)	(71) 出願人	000005049
(22) 出願日	平成19年11月7日 (2007.11.7)		シャープ株式会社
(31) 優先権主張番号	11/634,665		大阪府大阪市阿倍野区長池町2番2号
(32) 優先日	平成18年12月6日 (2006.12.6)	(74) 代理人	100153110
(33) 優先権主張国	米国 (US)		弁理士 岡田 宏之
		(74) 代理人	100079843
			弁理士 高野 明近
		(72) 発明者	ロイ ケー・クリソップ
			アメリカ合衆国 98607 ワシントン
			州, カマス, 2番 アベニュー, エヌダブ
			リュ 3460
		(72) 発明者	レナ ソジャン
			アメリカ合衆国 92708 カリフォル
			ニア州, ファウンテンバレー, モッキング
			バード サークル, 8876

最終頁に続く

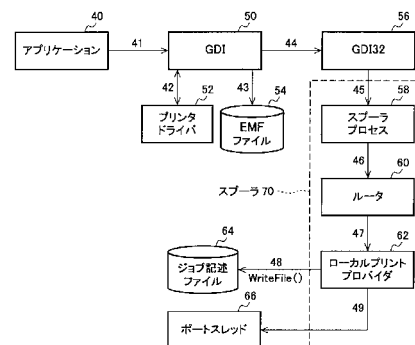
(54) 【発明の名称】 ダウンロードされたプリントジョブリソースに安全にアクセスするためのシステムおよび方法

(57) 【要約】

【課題】ダウンロードされたリソースに認証されたユーザしか使用できないようにする。

【解決手段】プリントデバイス内の画像形成ジョブリソースに安全にアクセスするための方法で、まず、制限されたソフトウェアリソースをプリントデバイスにダウンロードする。ソフトウェアリソースはユーザがアクセス可能なメモリに記憶される。その後、処理のために画像形成ジョブを受け入れる。この画像形成ジョブは、ユーザグループと関連していることが証明され、この証明の後で、制限されたソフトウェアリソースへのアクセスが許可され、制限されたソフトウェアリソースを使って画像形成ジョブを処理する。

【選択図】図3



【特許請求の範囲】**【請求項 1】**

制限されたソフトウェアリソースを、あるユーザグループにしか使用できないよう制限されたプリントデバイスへダウンロードするステップと、

前記制限されたソフトウェアリソースをメモリに記憶するステップと、

処理のために、画像形成ジョブを受け入れるステップと、

前記画像形成ジョブが前記ユーザグループに関連することを証明するステップと、

前記証明の後で、前記制限されたソフトウェアリソースへのアクセスを許可するステップと、

前記制限されたソフトウェアリソースを使って前記画像形成ジョブを処理するステップとを備えた、プリントデバイスにおいて画像形成ジョブリソースに安全にアクセスするための方法。

10

【請求項 2】

前記メモリ内に前記制限されたソフトウェアリソースを記憶するステップは、前記メモリ内の前記リソースと記憶されたユーザグループ識別とをクロスレファレンスすることを含み、

前記画像形成ジョブが前記ユーザグループに関連していることを証明するステップは、ユーザグループ識別を受信するステップと、

前記受信したユーザグループ識別と前記記憶されたユーザグループ識別とを比較するステップとを含む、請求項 1 記載の方法。

20

【請求項 3】

前記ユーザグループ識別を受信するステップは、処理のための前記画像形成ジョブを受け入れることと同時に、前記ユーザグループ識別を受信することを含む、請求項 2 記載の方法。

【請求項 4】

前記ユーザグループ識別を受信するステップは、

前記画像形成ジョブの受け入れに応答し、処理のために前記制限されたソフトウェアリソースが必要であると決定するステップと、

ユーザグループ識別リクエストを発生するステップと、

前記リクエストに応答し、前記ユーザグループ識別を受信するステップを含む、請求項 2 記載の方法。

30

【請求項 5】

前記ユーザグループ識別を受信するステップは、PIN 番号、一方向ハッシュ番号、ユーザ ID とパスワードの組み合わせと、磁気カードと、スマートカードとからなる群から選択された識別を受信することを含む、請求項 2 記載の方法。

【請求項 6】

前記制限されたソフトウェアリソースをダウンロードするステップは、ローカル接続と、ネットワーク接続と、ポータブル記憶媒体から成る群から選択された通信メディアを使って前記リソースをダウンロードすることを含む、請求項 1 記載の方法。

【請求項 7】

前記ダウンロードされたリソースにアクセスするのに必要なユーザ認証基準を受け入れるステップと、

前記ユーザ認証基準を記憶するステップを更に含む、請求項 1 記載の方法。

40

【請求項 8】

前記制限されたソフトウェアリソースをダウンロードするステップは、フォーム、フォント、マクロ、ロゴ、透かし、アドレス帳、ハーフトーンパターン、カラープロファイル、画像スタンプ、カバーシート、校正データ、デジタル署名、生体認証情報、オーディオ/ビデオクリップおよび辞書から成る群から選択されたリソースをダウンロードすることを含む、請求項 1 記載の方法。

【請求項 9】

50

前記プリントデバイスに、前記制限されたソフトウェアリソースをダウンロードするステップは、プリンタ、ファクシミリデバイス、ファイリングデバイス、フォーマット変換デバイス、パブリッシングデバイス、スキャナー、コピー機、電子ホワイトボード、オーディオ/ビデオデバイス、デジタルカメラまたは医療用画像形成デバイスから成る群から選択されたプリントデバイスにダウンロードすることを含む、請求項 1 記載の方法。

【請求項 10】

処理のための前記画像形成ジョブを受け入れるステップは、ページ記述言語 (PDL) と、プリンタジョブ言語 (PJL) と、画像フォーマットから成る群から選択されたフォーマットのプリントジョブを受け入れることを含む、請求項 1 記載の方法。

【請求項 11】

前記制限されたソフトウェアリソースをダウンロードした後に、前記ダウンロードされたリソースとリードオンリーメモリ (ROM) に記憶された工場ロードされたリソースとを区別するステップを更に含む、請求項 1 記載の方法。

【請求項 12】

前記ダウンロードされたリソースに適用すべき一組の制限を受け入れるステップと、前記制限の組を記憶するステップを更に含む、請求項 1 記載の方法。

【請求項 13】

前記メモリ内に前記制限されたソフトウェアリソースを記憶するステップは、内部メモリと、ローカル接続されたメモリと、ネットワーク接続されたメモリと、ポータブル記憶媒体とから成る群から選択されたメモリに前記リソースを記憶することを含む、請求項 1 記載の方法。

【請求項 14】

あるユーザグループにしか使用できないように制限されたソフトウェアリソースをダウンロードするための第 1 外部ポートと、

前記制限されたソフトウェアリソースを記憶するための、前記外部ポートに接続されたメモリと、

処理のために画像形成ジョブを受け入れるための第 2 外部ポートと、

前記画像形成ジョブが前記ユーザグループに関連することを証明した後に、前記制限されたソフトウェアリソースへのアクセスを許可するためのセキュリティモジュールと、

前記画像形成ジョブを受け入れ、前記メモリへアクセスするためのインターフェースを有するプリントエンジンとを備え、前記プリントエンジンは、前記制限されたソフトウェアリソースへのアクセスに応答して前記画像形成ジョブを処理するようになっている、プリントデバイスにおいて画像形成ジョブリソースに安全にアクセスするためのシステム。

【請求項 15】

前記第 2 外部ポートは、前記画像形成ジョブに関連するユーザグループ識別を受信し、前記安全モジュールは、メモリ内の前記制限されたソフトウェアリソースと記憶されたユーザグループ識別とをクロスレファレンスするテーブルを含み、前記セキュリティモジュールは前記受信したユーザグループ識別と前記記憶されたユーザグループ識別とを比較することにより、前記画像形成ジョブが前記ユーザグループに関連していることを証明する、請求項 14 記載のシステム。

【請求項 16】

前記第 2 外部ポートは、処理のための前記画像形成ジョブを受け入れることと同時に、前記ユーザグループ識別を受信する、請求項 15 記載のシステム。

【請求項 17】

前記セキュリティモジュールは、処理のために前記制限されたソフトウェアリソースが必要であると決定し、外部ポートで発生されたユーザグループ識別リクエストを発生し、前記リクエストに応答し、前記ユーザグループ識別を受信する、請求項 15 記載のシステム。

【請求項 18】

前記セキュリティモジュールは、PIN 番号、一方向ハッシュ番号、ユーザ ID とパス

10

20

30

40

50

ワードの組み合わせと、磁気カードと、スマートカードとからなる群から選択された識別を受信する、請求項 15 記載のシステム。

【請求項 19】

前記第 1 外部ポートは、ローカル接続と、ネットワーク接続と、ポータブル記憶媒体から成る群から選択された通信メディアである、請求項 14 記載のシステム。

【請求項 20】

前記第 1 外部ポートは、前記ダウンロードされたリソースにアクセスするのに必要なユーザ認証基準を受信し、

前記メモリは、前記ユーザ認証基準を記憶する、請求項 14 記載のシステム。

【請求項 21】

前記第 1 外部メディアは、フォーム、フォント、マクロ、ロゴ、透かし、アドレス帳、ハーフトーンパターン、カラープロファイル、画像スタンプ、カバーシート、校正データ、デジタル署名、生体認証情報、オーディオ/ビデオクリップおよび辞書から成る群から選択されたリソースをダウンロードする、請求項 14 記載のシステム。

【請求項 22】

前記プリントデバイスは、プリンタ、ファクシミリデバイス、ファイリングデバイス、フォーマット変換デバイス、パブリッシングデバイス、スキャナー、コピー機、電子ホワイトボード、オーディオ/ビデオデバイス、デジタルカメラまたは医療用画像形成デバイスから成る群から選択されたプリントデバイスである、請求項 14 記載のシステム。

【請求項 23】

前記第 2 外部ポートは、ページ記述言語 (PDL) と、プリンタジョブ言語 (PJL) と、画像フォーマットから成る群から選択されたフォーマットのプリントジョブを受け入れる、請求項 14 記載のシステム。

【請求項 24】

前記セキュリティモジュールは、前記ダウンロードされたリソースとリードオンリーメモリ (ROM) に記憶された工場ロードされたリソースとを区別する、請求項 14 記載のシステム。

【請求項 25】

前記第 1 外部インターフェースは、前記ダウンロードされたリソースに適用すべき一組の制限を受け入れ、

前記メモリは、前記制限の組を記憶する、請求項 14 記載のシステム。

【請求項 26】

前記メモリは、内部メモリと、ローカル接続されたメモリと、ネットワーク接続されたメモリと、ポータブル記憶媒体とから成る群から選択されている、請求項 25 記載のシステム。

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、一般的には、デジタル画像処理に関し、より詳細には、プリンタデバイスがダウンロードされたプリントジョブリソースに安全にアクセスするためのシステムおよび方法に関する。

【背景技術】

【0002】

多くの従来の多機能周辺機器 (MFP) またはプリントデバイスは、再使用可能なソフトウェアリソース、例えば、フォント、フォーム、透かし、ロゴ、デジタル署名などのユーザのダウンロードをサポートしている。このようなダウンロードにより、ジョブを発生するたびにリソースを繰り返しダウンロードしなくても、これらリソースを使用するプリントジョブをユーザが発生できる。しかしながら、デバイスに一旦ロードされると、リソースは一般的な無制限の使用がすべてのユーザにとって利用可能となる。このことは、ある環境では望ましくない。

10

20

30

40

50

【 0 0 0 3 】

特開 2 0 0 3 - 8 4 9 2 9 号公報には、ユーザの認証を行い、該ユーザの認証結果に応じて画像処理のためのソフトウェアのダウンロードを許可し、ソフトウェアのダウンロードが許可されると、ネットワーク上に保持されている画像処理のためのソフトウェアをダウンロードし、該ダウンロードされたソフトウェアを記憶装置に格納し、ユーザの認証を行い、該ユーザの認証結果に応じて記憶装置に格納されているソフトウェアの実行を許可するようにし、各ユーザのニーズに応じた画像処理を可能とし、冗長な機能の抑制を図った画像処理方法が開示されている。

【特許文献 1】特開 2 0 0 3 - 8 4 9 2 9 号公報

【発明の開示】

【発明が解決しようとする課題】

【 0 0 0 4 】

ソフトウェアリソースをプリントデバイスにダウンロードでき、ダウンロードされたリソースへの認証されたアクセスを有する選択されたグループのユーザによって将来使用できるように、ソフトウェアリソースを記憶装置に維持できれば有利となる。

【課題を解決するための手段】

【 0 0 0 5 】

本発明は、ユーザがプライベートな再使用可能なリソースをダウンロードし、リソースのその後の使用を制限するための方法に関する。作動環境の例は、1つ以上のプリントデバイスと、これらプリントデバイスに通信可能に結合された1つ以上のホスト計算デバイスとからなる。更にユーザは、接続されたホストの1つ以上から接続されたプリンタヘリソース（例えば、フォント、フォームなど）をダウンロードできる。最後に、ユーザは、ダウンロードされたリソースの使用を必要とするプリンタにプリントジョブをデスプールの。

【 0 0 0 6 】

本発明では、ダウンロードされたリソースにアクセス権、例えば、リソースへのアクセスを一人以上のユーザに制限すること（または、その回数を制限することを割り当てること）ができる。リソースに制限されたアクセス権が関連しているとき、リソースを必要とするプリントジョブは、このプリントジョブに関連する認証情報（例えば、ユーザノパスワードまたはPIN）を有していなければならない。認証されたユーザに関連するアクセス権は、アクセスをリソースに付与しなければならない。プリントジョブが認証情報を有していないか、またはリソースを使用することがそのプリントジョブに認められていない場合、このプリントジョブを拒否できる。

【 0 0 0 7 】

従って、プリントデバイス内の画像形成ジョブリソースに安全にアクセスするための方法が提供される。この方法は、まず、制限されたソフトウェアリソースをプリントデバイスにダウンロードする。すなわち、リソースはユーザグループによる使用のために制限される。工場でインストールされたソフトウェアまたは現場でのアップデートとは異なり、この制限されたソフトウェアリソースはユーザがアクセス可能なメモリ、例えば、不揮発性記憶装置（例えば、ランダムアクセスメモリすなわちRAM）に記憶される。その後、処理のために画像形成ジョブを受け入れる。この画像形成ジョブは、ユーザグループと関連していることが証明される。この証明の後で、制限されたソフトウェアリソースへのアクセスが許可され、制限されたソフトウェアリソースを使って画像形成ジョブを処理する。

【 0 0 0 8 】

一般に、画像形成ジョブは、ページ記述言語（PDL）のようなプリントジョブであり、制限されたソフトウェアリソースはフォーム、フォント、マクロ、ロゴ、透かし、アドレス帳、ハーフトーンパターン、カラープロフィル、カバーシート、校正データ、デジタル署名、生体認証情報、オーディオ/ビデオクリップおよび辞書のようなリソースとすることができる。

10

20

30

40

50

【 0 0 0 9 】

この方法の 1 つの様相では、処理のための画像形成ジョブを受け入れると同時に、ユーザグループの認証を受信する。これとは別に、（ユーザグループ ID を有しない）画像形成ジョブを受信し、処理のために、制限されたソフトウェアリソースが必要であるかどうかを判断する。次に、ユーザグループ識別リクエストを発生し、このリクエストに応答してユーザグループの識別（ID）を受信する。

【 0 0 1 0 】

ダウンロードされたリソースに安全にアクセスするための上記方法およびプリントデバイスシステムの更なる細部を次に、説明する。

【 発明を実施するための最良の形態 】

【 0 0 1 1 】

プリントデバイスは、より大きいシステム全体の一部とすることができ、このシステムは、プリントデバイスだけでなく、ローカル接続またはネットワーク接続を介してプリントジョブを処理のためのプリントデバイスに送ることができるパソコン（PC）のようなクライアントデバイスも含む。ネットワークは、例えば、RAN、WAN またはインターネットとすることができる。プリンタおよびクライアントは有線接続、無線接続またはこれらの組み合わせ接続を通して通信可能に結合できる。ローカル接続は、例えば、USB、シリアルポートインターフェース、パラレルポートインターフェースによりイネーブルできる。

【 0 0 1 2 】

図 7 は、プリントデバイスシステムの概略ブロック図である。クライアント 200 では、スプーラ 202 からポートモニタ 204 およびアップロードマネージャー 206 を介してネットワークサーバ 208（例えば、ウェブサーバまたはプリントサーバ）へ適当なファイル情報（パラメータ）が送られる。この適当なファイル情報は、例えば、プリンタのロケーション、プリンタネーム、ジョブ ID、プリントレベルおよびカラー、ステープリングなどのドキュメント情報を含む。ポートモニタ 204 に、適当なファイル情報が送られた後に、例えば、ドキュメントをプリントするために、有効コンポーネント（例えば、プリントドライバ 212）が使用中であるかどうかを判断するためにチェックが行われる。クライアントにプリントドライバがファイルとして常駐している。しかしながら、図示されていない別の様相では、ネットワークサーバ内に、またはプリントデバイスと共に、プリントドライバを埋め込むことができる。プリントドライバが有効である場合、プリントスプーラ 202 からポートモニタ 204 へデータファイルが送られる。データファイルは、例えば、情報のパケットとして送られる。データファイル全体が一旦ポートモニタ 204 へ送られると、アップロードマネージャー 206 を介し、データファイル全体がサーバ 208 へ送られ、記憶される。

【 0 0 1 3 】

クライアント 200 をターミナル、パソコン、PDA、携帯電話などとすることができる。アプリケーション 214 は、例えば、マイクロソフトのワード（商標）またはパワーポイント（商標）のローカルにインストールされたバージョン、または、プリントのためにドキュメントを発生できる他の任意のソフトウェアとすることができる。ローカルアプリケーション 214 は、プリントデバイスへ送信するためのドキュメントを作成またはダウンロードする。本明細書で使用する「ドキュメント」なる用語は、プリントデバイスに送信され、物理的な媒体、例えば、紙にプリントされた状態になることができる任意のデータまたは情報を意味する。プリントドライバ 212 は、ローカルに接続されるか、または、ネットワーク接続できる、選択されたプリントデバイスと通信するのに必要なオブジェクトを構築し、作成する。アップロードマネージャー 206 は、ファイル（例えば、ドキュメント）を圧縮し、これらファイルをネットワークを通して転送する役割を果たす。アップロードマネージャー 206 は、これらファイルを暗号化することもでき、アップロードマネージャーは例えば、安全ソケットレイヤー（SSL）を介してポストスクリプトプリンタファイルを転送または圧縮することができる。ドキュメントが一旦サーバ 208

10

20

30

40

50

へ送られると、このドキュメントをプリンタデバイス 220 へ送ることができる。これとは異なり、ローカル接続を通してジョブを送ることもできる。

【0014】

マイクロソフトウィンドウズ（登録商標）のオペレーティングシステムは、一般に、プリントプロセスにおいて2つのファイルのタイプを利用する。これらファイルのタイプは、エンハンスされたメタファイル（EMF）と、未処理フォーマットファイルである。未処理フォーマットファイルはデバイスに依存するファイルであり、特定のデバイスを宛て先とし、このデバイスのためにフォーマット化される。未処理ファイルの例は、ポストスクリプトプリンタによる解読のためにフォーマット化されたカプセル化されたポストスクリプト（PS）ファイルである。EMFファイルは、デバイスから独立したファイルであり、プリンタ上でアプリケーションのグラフィック要素を再生するグラフィックデバイスインターフェース（GDI）ファンクションコールを含む。EMFファイルは、プリントされたドキュメントを迅速に記録し、システム制御をユーザに戻すために使用される。ユーザに制御が戻された後に、EMFファイルに記憶されたファンクションコールにアクセスし、バックグラウンド内のプリンタに送ることができる。

【0015】

プリントデバイスに書き込み、後でプリントデバイスにデスプールされるスプールファイルを使用することにより、後にファイルを再生のためにレコードできる。スプールファイルはEMFおよび未処理ファイルのために使用できる。しかしながら、スプールファイルを使用することなく、プリントデバイスにプリントジョブを直接書き込むこともできる。これらプロセスのコンポーネント、要素および関係、更に、これらが本発明の実施例とどのように関係するかを説明するために、以下、未処理スプールファイルおよびEMFスプールファイルを使った代表的な一部のプリントプロセスのシナリオについて説明する。これらシナリオは、マイクロソフトのウィンドウズ（登録商標）95ドライバ開発キット（DDK）ドキュメンテーション、マイクロソフトのウィンドウズ（登録商標）2000DDKドキュメンテーションおよびマイクロソフトのウィンドウズ（登録商標）NTDDKドキュメンテーションに含まれる情報から導かれたものである。

【0016】

図1は、未処理スプールファイルを使用する従来のプリント動作を示す図である。上記コンポーネントの多くは、コンピュータシステム50内の要素としてイネーブルできる。コンピュータシステム50は、パソコン、ワークステーション、パーソナルデジタルアシスタントなどを含む任意のタイプの計算デバイスを含むことができる。計算システム50は、一般にオペレーティングシステム（図示せず）を含むことができる。コンピュータシステム50は数種のアプリケーションを作動できるが、図には単一のアプリケーション10が示されている。アプリケーションの例として、ワードプロセッサ、スプレッドシート、通信ソフトウェアおよびプレゼンテーションソフトウェアを挙げることができる。一般に、コンピュータシステムのユーザは1つ以上のドキュメントを発生するのにアプリケーション10を利用できる。一部の特徴では、コンピュータシステム50は他の計算デバイス、サーバ、ルータ、ハブ、スイッチを含むコンピュータネットワークコンポーネントおよびディスプレイ、プリンタ、プロッタ、CDライター、テープドライブおよび他のデバイスのような出力デバイスを更に含むことができる。

【0017】

コンピュータシステム50をプリンタのような出力デバイス（図示せず）に接続できる。出力デバイスは、画像を形成できると共に、コンピュータシステム50と組み合わせて使用できる任意のタイプの出力デバイスとすることができる。アプリケーション10が作成した1つ以上のドキュメントをプリントするのにプリンタを使用できる。

【0018】

以下、より詳細に説明するように、コンピュータシステム50はアプリケーションの出力を出力デバイスとコンパチブルなフォーマットに変換するための出力システムまたはプリントシステムを含むことができる。出力システムまたはプリントシステムは、プリンタ

ドライバ、プリントプロセッサ、スプーラ、プリントプロバイダだけでなく、マイクロソフトのウィンドウズ（登録商標）のオペレーティングシステムに関連してこれまで説明したような、他のプリントシステムコンポーネントも含むことができる。これらプリントシステムのコンポーネントは、アプリケーション 10 がプリンタと通信できるようにするソフトウェアである。アプリケーション 10 がドキュメントをプリントしなければならないとき、アプリケーション 10 は、プリントデータをプリントシステムに送る。プリントデータは、プリントすべきデータを記述するデータであり、一般に、プリントデータは、一連のコマンド（例えば、円を描いたり、特定のフォントでテキストのラインを描いたりすることなど）である。プリンタシステムは、アプリケーション 10 からプリントデータを取り込み、次に、プリンタレディデータを作成する。プリンタレディデータは、プリンタが理解できるフォーマットに変換されたプリンタデータのことである。プリンタレディデータのフォーマットは、プリンタの能力に応じて決まる。インクジェットのような多数のローエンドプリンタに対し、プリンタレディデータは、ピクセルデータすなわち紙の部分の上にピクセルをプリントするのに使用できるデータである。次第に、より多数のプリンタが、種々のプリンタ記述言語（PDL）、例えば、PCL（多数のバージョン）およびポストスクリプトで示された画像に変換できるようになってきている。

10

【0019】

アプリケーション 10 は、グラフィックデバイスインターフェース（GDI）12 を考慮することにより、プリントリクエスト（1）を開始する。アプリケーション 10 は、ワードプロセッサ、スプレッドシート、ブラウザ、データベースプログラムまたは基礎となるオペレーティングシステムで作動する他のプログラムとすることができる。一般に、アプリケーション 10 はデバイスコンテンツ（DC）を作成し、この DC にオブジェクト（すなわち、円、ラインなど）を描く。次に、アプリケーション 10 は、この DC を使用する特定のプリンタ 16（図 2）に向けられたプリントリクエストにより、GDI をコールする。

20

【0020】

GDI 12 は、特定のプリンタ 16 に関連するプリンタドライバ 14 をコールし、特定のプリンタ 16 でオブジェクトをどのようにレンダリングするかをリクエストする（2）。プリンタドライバ 14 は、プリンタ 16 でオブジェクトをどのようにレンダリングするかをリターンする（3）。このプリントプロセスの例で使用されるウィンドウズ（登録商標）95 では、16 ビットコードでプリンタドライバ 14 を書き込み、16 ビット GDI 12 と通信する。この GDI は、プリントリクエストを 32 ビットの GDI（GDI 32）18 に送り（4）、32 ビットのウィンドウズ（登録商標）95 のスプーラプロセスを取り扱う。GDI 32 は、スプーラプロセス 20 に対するインタープロセスコールを行う（5）。

30

【0021】

スプーラプロセス 20 は、プリントジョブをプリンタ 16 にルーティングするよう、ルータ 22 をコールする（6）。図 1～図 2 で示されているこの例では、ルータ 22 はローカルプリントプロバイダ 24 へプリントジョブを送る（7）。別のシナリオでは、ルータ 22 は、ネットワークプリントプロバイダ（図示せず）を介してプリントジョブをネットワークプリンタへ送ることができる。デフォルトウィンドウズ（登録商標）95 スプーラを使用するときは、ローカルプリントジョブとしてクライアントマシンにて、ネットワークプリントジョブをスプーリングし、かつデスプーリングする。ネットワークプリントサーバにはデスプーリング中しかコンタクトしない。ウィンドウズ（登録商標）NT/2000 のクライアントマシンは、ネットワークプリントサーバへのプリントジョブを異なるように取り扱い、これらマシンはプリントサーバで必要なプリントアプリケーションプログラムインターフェース（API）をコールするために、リモート手順コール（RPC）を使用する。これら NT/2000 のシナリオでは、プリントジョブはローカルスプーリングキューでショーアップしない。むしろ、プリントサーバでのプリントスプーラによってスプーリングおよびデスプーリングを取り扱う。この RPC 方法は、ウィンドウズ（登録商標）95 のス

40

50

プーラと組み合わせて使用できる。ローカルに接続されたプリンタまたはローカルにキュー状態（例えば、L P R）となっているネットワークプリンタに関するプリントジョブは、ウィンドウズ（登録商標）9 5、9 8のローカルプリントジョブと同じように取り扱われる。

【0022】

このローカルプリントのシナリオでは、ルータはプリントジョブと共にローカルプリントプロバイダ24をコールする。ローカルプリントプロバイダ24は、後にアクセスできるよう、ディスクに未処理スプールファイル26を書き込む。すなわち、スプールする（8）。このことは、アプリケーションに制御が戻される前にプリンタがジョブを完了することを待つのを回避するために行われる。プリントリクエスト（1）を開始してからスプールファイル26を書き込むまでのこれらステップは、何回も繰り返すことができる。プリントジョブが完了した旨の信号をアプリケーションが発するまで、スプールファイル26にデータを添付できる。EndDoc機能により、ジョブ完了の信号を発することができる。ローカルプリントプロバイダ24は、プリンタ16に対するスプールファイル26の再生すなわち、デスプールをスタートするための最良の時間を決定するバックグラウンドスレッド28もスタートする（9）。

【0023】

図2は、スプーラのサブシステム（従来技術）を示す図である。スレッド28は、スプールファイル26を再生するための良好な時間を決定するためにスプーラサブシステムのリソースをモニタする。再生を開始すべきであるとスレッド28が判断すると、新しいプリントプロセッサのスレッド（11）をスタートするために、プリントプロセッサ32へStartDocファンクションコール（17）が送られる。プリントプロセッサのスレッド（11）は、スプールファイル26の部分を読み出すためにReadPrinterファンクションコール（12）により、ローカルプリントプロバイダ24を呼び出す。WrightPrinterファンクションコールにより、言語モニタ34を呼び出し、前に特定された双方向プリンタ16と接続された物理的ポート38を介し、データを送るために、プリントプロセッサスレッド（19）はローカルプリントプロバイダ24も使用する。

【0024】

デフォルトプリントプロセッサ32は、未処理のスプールファイルのために、情報のいずれかを変更したり、解読したりすることなく、データを通過するに過ぎない。宛て先プリンタ16は、双方向のプリンタであるので、この例では、言語モニタ34が使用される。双方向でないプリンタが使用されると、言語モニタ34の代わりにポートモニタ36が呼び出される。言語モニタ34とポートモニタ36は、別個のコンポーネントでもよいし、1つのモニタに一体化されていてもよい。

【0025】

言語モニタ34はプリンタ16へプリントジョブデータを送るために、ポートモニタ36をコールする（13）。次に、ポートモニタ36は物理ポート38を介し（14）、未処理データをプリンタ16へ送る。このようなスプールファイル26からデータを読み出し、このデータをプリンタ16へ転送するプロセスは、プリントジョブを完了するために数回繰り返すことができる。この作業は、一般に、ファイルの終了部に達するかまたはジョブがキャンセルされるまで繰り返される。この時点で、再生スレッド（19）が終了される。スプーラプロセッサと、ルータと、ローカルプリントプロバイダと、プリントプロセッサと、言語モニタと、ポートモニタとの組み合わせを「スプーラ」30と総称できる。

【0026】

図3および図4は、ウィンドウズ（登録商標）のEMFプリント動作（従来技術）を示す図である。ウィンドウズ（登録商標）の9.xシステムのプリントプロセスで、ウィンドウズ（登録商標）のエンハンスされたメタファイル（EMF）フォーマットファイルが使用されるとき、プロセスコンポーネントは未処理ファイルと異なるように相互作用する。図3および図4に示されたプリントプロセスの一例は、EMFファイルを使用したプリ

ントプロセスを示す。このプロセスは、一般に、アプリケーション 40 がプリンタ DC を作成し、この DC (図示せず) にオブジェクトを描くときにスタートする。次に、アプリケーション 40 は、指定されたプリンタ 68 のための EMF スプーリングリクエストにより、GDI 50 をコールする (41)。GDI 50 は、ドライバ 52 が EMF スプーリングをサポートしているかどうかを判断するために、指定されたプリンタ 68 に関連するプリンタドライバ 52 に問い合わせをする (42)。ドライバ 52 がスプーリングをサポートする場合、GDI 50 はプリンタ DC を EMF DC に変更し、EMF DC 54 へオブジェクトをレンダリングする (EMF ファイルを作成する) ための命令を書き込む (43)。次に、本例において、ウィンドウズ (登録商標) 95 のスプーラプロセスは 32 ビットのコードであるので、GDI 50 は、32 ビットの GDI (GDI 32) 56 にプリントリクエストを送る (44)。次に、GDI 32 はプリントジョブの記述によりスプーラサブシステム 70 へのインタープロセスコール (45) を行う。

10

20

30

40

50

【0027】

スプーラシステム 70 におけるスプーラプロセス 58 (SPOOL32.EXE) は、指定されたプリンタ 68 に達することができるプリントプロバイダ 62 へプリントジョブ記述を送るためにルータ 60 をコールする。この例では、ローカルプリントプロバイダ 62 が使用されているが、ネットワークプリントプロバイダも使用できる。デフォルトウィンドウズ (登録商標) 95 のスプーラを使用するときには、クライアントマシンでローカルプリントジョブとしてネットワークプリントジョブをスプーリングし、デスプーリングする。デスプーリング中にしかネットワークプリントサーバにコンタクトしない。ウィンドウズ (登録商標) NT / 2000 のクライアントマシンは、ネットワークプリントサーバに対するプリントジョブをことなるように取り扱い、これらクライアントマシンは、プリントサーバでの必要なプリントアプリケーションプログラムインターフェース (API) をコールするために、リモート手順コール (RPC) を使用する。これら NT / 2000 のシナリオでは、プリントジョブは、ローカルスプーラキュー上でショーアップしない。むしろ、プリントサーバ上のプリントスプーラにより、スプーリングおよびデスプーリングが取り扱われる。ウィンドウズ (登録商標) 95 スプーラと組み合わせて、この RPC 方法を使用することもできる。

【0028】

ルータ 60 がプリントプロバイダ 62 をコールすると、ローカルプリントプロバイダ 62 はジョブ記述ファイル 64 を作成し (48)、すべての EMF ページファイルがスプーリングされ、各 EMF ファイルネームおよびロケーションがジョブ記述ファイル 64 にレコードされるまで、ジョブのためにコールされる度に、レコードをジョブ記述ファイル 64 に追加する (48)。プリントジョブ内の最終 EMF ファイルに関する情報が記憶されると、ローカルプリントプロバイダ 62 は EndDoc ファンクションコールによりスプーラプロセス 58 をコールする。このことは、ジョブ全体がスプーリングされ、デスプーリングの準備が完了したことの信号をスプーラプロセス 58 に送る。マルチページジョブに対しては、ジョブのページごとに、最初のスプーリングリクエスト (41) からジョブ記述ファイルレコード (48) までのこれらステップが繰り返される。

【0029】

EMF ファイルスプーリングが完了すると、スプーラプロセス 58 はプリントジョブでの ReadyToPrint 属性をセットし、プリントのためにジョブを利用できる旨の信号をポートスレッド 66 に送るイベント (49) を開始する。ポートスレッド 66 はデスプーリングプロセスをスタートするための最良の時間を決定することにより、このイベントに応答し、その時間に、図 4 に示されるよう、プリントプロセッサ 72 にロードする (81)。プリントプロセッサ 72 は、ファイルフォーマットが EMF であると判断し、ウィンドウズ (登録商標) 95 のファンクションコール (82) により、GDI 32 56 をコールする。

【0030】

次に、GDI 32 は、ジョブ記述ファイル 64 から読み出すために、gdiPlaySpoolStre

amファンクションを呼び出す。これにより、E M F スプールファイル 5 4 へ完全に合格したパスが提供される。G D I 3 2 は、E M F ファイルへのパスネームのリストを含むジョブ記述ファイル 6 4 により、プリントジョブ内のすべてのページについて知ることができる。G D I 3 2 の gdiPlaySpoolStream ファンクションも、ページをレンダリングするための E M F スプールファイルへのパスにより、G D I 3 2 に組み込まれたサンク（プラットフォームコードを横断するコール）を使って G D I 5 0 もコールする。G D I 5 0 は、一度にプリントジョブ内の 1 つのページについてしか知らない。

【 0 0 3 1 】

G D I 5 0 は、アプリケーション 4 0 内で選択された指定プリンタ 6 8 に関連するプリンタドライバ 5 2 をコールし、プリンタ 6 8 のための D C を得る。次に、G D I 5 0 は、スプールされた E M F ファイル 5 4 からページレンダリング命令を読み出し、これら命令を一度にプリンタドライバ 5 2 へ送る（ 8 5 ）。プリンタドライバ 5 2 は、ページの第 1 部分をレンダリングするのに必要とされるだけの多くの命令を使用する。1 1 ビットのプリンタドライバ 5 2 がページの一部をレンダリングするとき、このドライバはプリンタ固有の未処理ページデータを G D I 5 0 に戻すように送り（ 8 7 ）、次に、G D I 5 0 は未処理データを G D I 3 2 5 6 へ送る（ 8 8 ）。次に、G D I 3 2 5 6 は未処理データをスプーラプロセス 5 8 へ送り（ 8 9 ）、次に、スプーラプロセスは上記未処理フォーマットファイルに対してとったのと同じ手順に従う。

【 0 0 3 2 】

スプーラプロセス 5 8 は、プリンタ 6 8 にプリントジョブをルーティングするために、ルータ 6 0 をコールする（ 9 0 ）。図 3 および図 4 に示されたこの例では、ルータ 6 0 はプリントジョブをローカルプリントプロバイダ 6 2 に送る。別のシナリオでは、ルータ 6 0 はネットワークプリントプロバイダ（図示せず）を通して、ネットワークプリンタにプリントジョブを送ることができる。このローカルプリントシナリオでは、ルータ 6 0 はプリントジョブによりローカルプリントプロバイダ 6 2 をコールする。ローカルプリントプロバイダ 6 2 は、前に指定された双方向のプリンタ 6 8 に接続された物理ポート 7 8 を通して、データを送るために、WritePrinter ファンクションコールにより、言語モニタ 7 4 を呼び出す。

【 0 0 3 3 】

宛て先プリンタ 6 8 は、双方向プリンタであるので、この例では、言語モニタ 7 4 を使用する。宛て先プリンタ 6 8 は、双方向プリンタであるので、双方向プリンタを使用するとき、言語モニタ 7 4 の代わりにポートモニタ 7 6 を呼び出す。言語モニタ 7 4 とポートモニタ 7 6 とは別個のコンポーネントでもよいし、又は、1 つのモニタに一体化してもよい。言語モニタ 7 4 は、プリンタ 6 8 へプリントジョブデータを送るために、ポートモニタ 7 6 をコール（ 9 3 ）する。次に、ポートモニタ 7 6 は、物理ポート 7 8 を通して未処理データをプリンタ 6 8 へ送る（ 9 4 ）。

【 0 0 3 4 】

全ページをプリントするまで、このように E M F ページ部分を処理し、プリントする。次に、G D I 3 2 5 6 は、次のページのための E M F スプールファイルまでのパスをとり、プリントジョブの次のページのレンダリングをするための、その E M F ファイル内の命令を使用するために、G D I 5 0 をコールする。E M F スプールファイルまでのすべてのファイルを使用し尽くすと、プリントジョブが終了する。

【 0 0 3 5 】

図 5 は、マイクロソフトウィンドウズ（登録商標）の N T および 2 0 0 0 のプリンティングオペレーション（従来技術）を示す図である。ウィンドウズ（登録商標）オペレーティングシステム例えばウィンドウズ（登録商標）N T および 2 0 0 0 の他のバージョンは、異なるプリントプロセスを使用できる。これらプロセスを使用し、直接またはネットワークプリントサーバを介し、ローカルプリンタ、ネットワークプリンタおよびリモートプリンタにデータをプリントできる。例えば、ウィンドウズ（登録商標）N T および 2 0 0 0 では、一度に 1 ページずつではなく、1 回のパスですべてのページに対する全 E M F デ

10

20

30

40

50

ータをGdiPlayEMF()へ送る。プリントサーバでE M Fデータをキュー状態(待ち行列状態)とすべき場合、クライアント側でレンダリングすることなく、プリントサーバへ直接E M Fデータを送る。その代わりに、サーバ側でドライバのマイナーなコピーがE M Fデータをレンダリングする。

【0036】

一般に、ユーザは、G D I 1 0 2ファンクションをコールすることにより、プリントジョブを作成するためのアプリケーション100を使用する。次に、G D I 1 0 2および/またはアプリケーション100は、スプーラへのクライアントインターフェースとなっているWinspool.drv104をコールする。このクライアントインターフェースすなわちWinspool.drv104はスプーラのWin32(登録商標)A P Iを構成するファンクションをエクスポートし、サーバにアクセスするためのR P Cスタブを提供する。次に、スプーラのA P IサーバであるSpoolsv.exe106にプリントジョブを転送する。この転送は、オペレーティングシステムをスタートするときにスタートする、ウィンドウズ(登録商標)2000のサービスとして実行できる。このA P Iサーバモジュールは、スプーラのWin32(登録商標)A P Iのサーバ側へR P Cインターフェースをエクスポートする。このモジュールは、一部のA P Iファンクションを実行するが、ルータすなわちSpoolsv.dll108によりプリントプロバイダにほとんどのファンクションコールが送られる。

【0037】

ルータ108は、各ファンクションコールにより提供されたプリンタ名前すなわちハンドルに基づき、どのプリントプロバイダがコールするかを決定し、このファンクションコールを正しいプロバイダ110、112または114へ送る。選択されたプリンタがクライアントシステムにより管理されている場合、ローカルプリントプロバイダすなわちlocalspl.dll110によりプリントジョブを取り扱う。ローカルプリントプロバイダ110により管理されているプリンタは、クライアントに対し、物理的にローカルでなくてもよく、サーバを使用することなく、ネットワーク回路にこれらプリンタを直接接続してもよい。これらプリンタを使用するとき、プリンタジョブをカーネルモードのポートドライバスタック116に送り、次に、プリンタ118に送る。

【0038】

ウィンドウズ(登録商標)N T / ウィンドウズ(登録商標)2000のサーバにあるプリンタを選択すると、ルータ108はプリントジョブをネットワークプリントプロバイダすなわちWin32spl.dll112へ向ける。このネットワークプロバイダは、R P Cを使用し、クライアントルータからネットワークサーバのSpoolsv.exeprocess124へコールを再び向ける。このプロセス124はプリントジョブをネットワークサーバのルータ126へ転送する。ネットワークプリンタはプリントサーバシステムに対してローカルであるので、ネットワークサーバルータ126はジョブをサーバのローカルプリントプロバイダ128へルーティングする。次に、サーバのカーネルモードのポートドライバスタック130へジョブを向け、次に、選択されたネットワークプリンタ132へ向ける。

【0039】

これらシステムと共に、リモートプリンタを使用することもできる。リモートプリンタが選択されると、クライアントルータ108はプリントジョブをローカルプリントプロバイダ110へ向けることができ、プロバイダ110はネットワークプロトコルを使って、ジョブをカーネルモードのポートドライバスタック116およびリモートプリンタ142へ転送する。ローカルプリントプロバイダ110が、リモートプリンタ142にアクセスすると、プロバイダ110はリモートプリンタまたはそのサーバにより認識されたネットワークプロトコルを使用できるポートモニタを使用する。

【0040】

このプリンとシステムを介し、ウィンドウズ(登録商標)N T / 2000でないサーバ(例えば、Novell社のサーバ)によって管理されているプリンタにアクセスすることもできる。このアクセスは、あるタイプのネットワークプロトコルを使ってプリントジョブをカーネルモードのポートドライバスタック116およびプリンタのサーバ136へ

向けるローカルプリントプロバイダ 110 を使って達成できる。次に、サーバ 136 は、このジョブを宛て先プリンタ 140 へ向ける。このことは、カスタム化されたプリントプロバイダ 114 を使って達成できることもできる。この場合、プロバイダ 114 は、ジョブをプリンタのサーバ 134 へ送るのにネットワークプロトコルを使用するカーネルモードのポートドライバスタック 116 へ送り、プリンタのサーバ 134 はこのジョブを宛て先プリンタ 138 へ向ける。

【0041】

図 6 は、ウィンドウズ（登録商標）2000 のプリントプロセス（従来技術）を示す図である。このプロセスでは、グラフィックデバイスインターフェース（GDI）152 により、プリントジョブを作成するためにアプリケーション 150 を使用する。プリンタジョブを最初の出力ファイルが未処理フォーマット 154 であるとき、プリンタドライバのプリンタグラフィックDLL 156 は、スプーラのクライアントインターフェース 160 へ送られるプリントジョブを作成するために、GDI 152 と共に作動する。クライアントインターフェース 160 は、ジョブを API サーバ 162 へ送り、サーバ 162 はジョブをルータ 164 へ転送する。この例では、ルータ 164 は、ジョブがローカルプリントジョブであるとき、このジョブをローカルプリントプロバイダ 165 へ送る。

10

【0042】

ローカルプリントプロバイダ 165 内では、プリントジョブ作成 API 168 が呼び出される。この API 168 は、プリンタドライバのプリンタインターフェース DLL 174 にアクセスし、ジョブスプールファイル 176 を作成する。ジョブ作成 API 168 は、ジョブ情報をジョブスケジューリング API 170 へも転送し、この API 170 はジョブスケジューラスレッド 172 を開始する。

20

【0043】

この時点で、ファイルフォーマットをチェックする（178）。初期ジョブファイルが既に未処理フォーマットである場合、言語モニタ DLL 182 およびポートモニタ 184 へジョブを送り、ポートモニタ 184 はジョブをカーネルモードのポートドライバスタック 186 へ送る。ポートドライバスタック 186 は最終プリントのために選択されたプリンタ 188 へジョブを送る。

【0044】

アプリケーション 150 が EMF フォーマットで GDI 152 によりプリントジョブを作成すると、このジョブはクライアントスプーラインターフェース 160 へ送るために（154）へ送られる。クライアントインターフェース 160 は、ジョブを API サーバ 162 へ送り、このサーバ 162 は、ジョブをルータ 164 へ転送する。この例では、プリントジョブはローカルであるので、再び、ルータ 164 はジョブをローカルプリントプロバイダ 165 へ送る。

30

【0045】

ローカルプリントプロバイダ 165 内では、プリントジョブ作成 API 168 を呼び出す。この API 168 は、プリンタドライバのプリンタインターフェース DLL 174 にアクセスし、ジョブスプールファイル 176 を作成する。ジョブ作成 API 168 は、ジョブ情報をジョブスケジューリング API 170 へ転送し、スケジューリング API 170 はジョブスケジューラスレッド 172 を開始する。

40

【0046】

この時点で、ファイルフォーマットをチェック（178）する。初期のジョブファイルが EMF フォーマットであれば、プリントプロセス DLL 180 へジョブを送り、プロセス DLL 180 はプリンタインターフェース DLL 174 の助けにより、未処理フォーマットに変換するようジョブを再び GDI 152 へ戻すように向ける。次に、変換されたジョブはスプーラクライアントインターフェース 160、API サーバ 162 およびルータ 164 を通してプリントプロバイダ 165 へ送り戻される。ローカルプリントプロバイダでは、プリントジョブ作成 API 168、ジョブスケジューリング API 170 およびジョブスケジューラスレッド 172 によりジョブを処理する。このとき、ジョブは未処

50

理フォーマットであるので、ジョブは宛て先プリンタ 188 に到達する前に、言語モニタ DLL 182 およびポートモニタ DLL 184、および、カーネルモードのポートドライバスタック 186 へ送られる。

【0047】

図 8 は、ソフトウェアリソースをプリンタデバイスにダウンロードするためのシステムのブロック略図である。例えば、ユーザは MFP 内に埋め込まれているフォーム / フォント / 画像 / スクリプトダウンロードウェブページにアクセスすることにより、このプロセスを開始できる。ユーザはウェブページから自分の PC 上でリソースのロケーションにブラウジングし、MFP の不揮発性メモリすなわちフラッシュメモリ内のロケーションを指定し、リソースをダウンロードできる。一旦ダウンロードした後に、次のプリントジョブは MFP 内のリソースに指定されたネームを参照することによってこれらリソースを参照できる。しかしながら、次のユーザがダウンロードされたリソースを使用しなければならないジョブを処理しようと試みる場合、MFP がこのロケーションを思い出さないようにするものは何もない。従って、最初のユーザがリソースをプライベートとすべきである（一般大衆にアクセスできないようにすべきである）と考える場合、最初のユーザは、使用することにダウンロードされたリソースを削除しなければならない。

【0048】

図 9 は、プリントジョブ処理における制限されたリソースの使用を防止するためのセキュリティモジュールを使用するシステムのブロック略図である。例えば、プリントデバイスには、アクセスを制限しなければならない特殊なフォントがロードされる。このプリンタは、プリント発生とプリンタとの間に接続されるアクセスボックスを有するように、後付けされる。このアクセスボックスは、プリントジョブが特殊なフォントを参照するとき、プリンタへのアクセスを制限するように使用される。この場合、プリンタにプリントジョブが送られるとき、プリントジョブは、まず、アクセスボックスを通過しなければならない。次に、アクセスボックスは、ジョブが特殊なフォントを使用しなければならないかどうかを判断するためにプリントジョブを分析する。必要としない場合、プリンタにジョブが送られる。そうでない場合、アクセスボックスはジョブがユーザ認証情報を含むかどうかを判断する。認証情報を含まない場合、そのジョブを拒否する。そうでない場合、ユーザは認証され、ユーザが特殊なフォントに対するアクセス権を有するかどうかをアクセスボックスが判断する。ユーザがアクセス権を有する場合、プリンタにジョブを送り、アクセス権を有しない場合、ジョブを拒否する。しかしながら、このセキュリティプロセスは、デバイス常駐リソース（すなわちダウンロードされていないリソース）に適用されるだけである。制限されるダウンロードされたリソースを識別する手段はなく、ユーザのプライベートリソースへのユーザ固有のアクセス権を定める手段もない。

【0049】

図 10 は、制限されたすべてのリソースへの特別なユーザの無制限のアクセスを許可するシステムを示すブロック略図である。この方法では、ユーザ ID を認証した後に、プリントユニットおよびすべての特殊なプリントファンクション（例えば、両面、ステープル、ペーパータイプ）へのアクセスを付与できる。プリンタがプリントジョブを受信したとき、このプリンタは、ジョブがアクセスを制限した特別なファンクションを必要としているかどうかを判断するために、ジョブを分析する。特別なリソースが不要であれば、ジョブをプリントする。そうでない場合、プリンタはジョブがユーザ認証情報を含むかどうかを判断する。特別なリソースが必要であり、ユーザ認証が提供されていないならば、ジョブを拒否する。認証されたユーザ情報とすべての制限されたリソースへのアクセスが許可されたユーザのリストとを比較する。ユーザが必要なすべての特別なファンクションへのアクセス権を有する場合、その情報をプリントする。そうでない場合、情報を拒否する。しかしながら、このシステムは、ダウンロードされたプライベートなユーザリソースを識別する手段を提供しない。システムは、ユーザごとに限られたリソースへの選択的アクセスを許可しない。

【0050】

10

20

30

40

50

図 1 1 は、画像形成ジョブリソースに安全にアクセスするためのシステム 1 1 0 2 を有するプリントデバイス 1 1 0 0 のブロック略図である。プリントデバイス 1 1 0 0 は、数例を挙げるとすれば、プリンタ、MFP、ファクシミリデバイス、ファイリングデバイス、フォーマット変換デバイス、パブリッシングデバイス、スキャナ、コピー機、電子ホワイトボード、オーディオ/ビデオデバイス、デジタルカメラまたは医療用画像形成デバイスとすることができる。説明を簡潔にするために、このプロセスの最終結果は物理的媒体にプリントされたジョブとして示されている。しかしながら、本発明は、電子ファイルを変換するプロセスにも同じように適用可能であると理解すべきである。

【0051】

システム 1 1 0 2 は、参照番号 1 1 0 6 で示された、制限されたソフトウェアリソースをダウンロードするためのライン 1 1 0 4 上の第 1 外部ポートを含む。すなわち、ソフトウェアリソース 1 1 0 6 は、あるユーザグループによる使用が制限されている。ライン 1 1 0 4 上の第 1 外部インターフェースは、数例を挙げるとすれば、ネットワーク接続、ローカル接続またはポータブル記憶媒体例えばディスクドライブまたはメモリスティックインターフェースとすることができる。メモリ内のソフトウェア部分が 1 つ以上の ROM IC を交換することによって変更できる、工場で設置されるメモリまたはフィールドアップグレードとは異なり、本明細書で説明するダウンロードは、外部ポートでのリソースの受け入れを必要とする。外部ポートから、マイクロプロセッサとの協働により、内部バスを介し、メモリ内のあるロケーションへ情報がトランスポートされる。別の説明をすれば、工場ロードされるリソースは、ユーザにとって書き込みできない指定された ROM アドレスレンジ内に記憶される。ユーザがロードするリソースは、システムが許可する不揮発性記憶アドレスレンジ内に記憶される。本システムの 1 つの特徴では、ダウンロードされたデータと、工場ロードされたデータとを区別するために、プリントデバイス内で特別なタグ付けを行うことができる。更に、工場ロードされたリソースと、ユーザがロードしたリソースのロケーションをマッピングする内部テーブルをシステム内に作成することができる。

【0052】

更に、実施例は、工場ロードされたリソースとユーザがロードしたリソースとを区別するために、CPU ボードソケットロケーションと共に、プログラム可能なフラッシュデバイスを使用できる。ユーザがネームにより、プロテクトされ、かつ、使用がオープンになっているリソースを定義できるように、コンフィギュレーション選択を使用できる。

【0053】

制限されたソフトウェアリソースを記憶するためにライン 1 1 1 0 上の第 1 外部ポートは不揮発性記憶装置（すなわち、不揮発性ランダムアクセスメモリすなわち NVRAM）1 1 0 8 が接続されている。これとは異なり、リソースをローカル接続またはネーム接続された外部デバイス 1 1 4 0 内に記憶してもよいし、または、ポータブルな記憶媒体 1 1 4 2、例えば、フロッピー（登録商標）ディスクに記憶してもよい。

【0054】

処理のために参照番号 1 1 1 4 で示されあ画像形成ジョブを、ライン 1 1 1 2 上の第 2 外部ポートが取り込む（注：別個のインターフェースとして示されているが、共通インターフェース、例えば、RAN インターフェースまたは WAN インターフェースで画像形成ジョブと制限されたリソースとを受信してもよい）。説明を簡潔にするために、第 2 外部ポート 1 1 1 2 は、あるフォーマット、例えば、ページ記述言語（PDL）、プリンタジョブ言語（PJL）、PDL と PJL との組み合わせ、または、TIFF、JPEG および PNG のような画像フォーマットで処理するためのプリントジョブを受け入れるものと理解することができる。プリントジョブは、ドキュメント、画像またはプリンタドライバアルゴリズムを使用して作成されたプリンタレディフォーマットのジョブとなっている。プリンタ制御言語（PCL）は、PDL の一例であり、他の例として、PCL 5 e、PCL 5 c、PCL XL およびエンハンスされたメタファイル（EMF）を挙げることができる。しかしながら、これらが全てではない。これら言語は、プリントすべき画像およびテ

キストを記述する。

【 0 0 5 5 】

例えば、プリントジョブはフォント、透かし、デジタル署名、画像スタンプおよびプリントされた出力を作成するのに必要な他の要素を使用する。スキャンされるドキュメントまたは電子ファイルは、画像形成ジョブでもよく、これら画像に対してセグメント化ファンクションを適用することにより、オブジェクト、例えば、テキスト、商標などを分離し、識別し、画像をベースとしないドキュメント、例えば、プリントジョブを作成することが可能となる。

【 0 0 5 6 】

セキュリティモジュール 1 1 1 6 は、画像形成ジョブ 1 1 1 4 がユーザグループに関連していることを証明した後に、制限されたソフトウェアリソース 1 1 0 6 へのアクセスを許可する。プリントエンジン 1 1 1 8 は、画像形成ジョブを取り込むためのライン上のインターフェース 1 1 2 0 およびリソースが記憶されているメモリ例えば N V R A M 1 1 0 8) にアクセスするためのライン上のインターフェース 1 1 2 2 を有する。プリントエンジン 1 1 1 8 は、制限されたソフトウェアリソース 1 1 0 6 へのアクセスに 응답して、画像形成ジョブ 1 1 1 4 を処理する。簡潔にするために、図は、画像形成ジョブに関連するすべてのプロセスをプリントエンジンで実行するものと見なしている。例えば、このような簡略化された表示では、プリントエンジンはラスタ画像処理を実行し、物理的な媒体例えば紙に画像を作成するために必要なラスタデータを発生するものと称することができる。従って、処理に必要なリソースはプリントエンジン 1 1 1 8 に送らなければならない。

10

20

【 0 0 5 7 】

ソフトウェアをリードオンリーメモリ (R O M) 1 1 2 6 にインストールする、工場インストールされるプログラム、または、フィールドアップグレードと異なり、ダウンロードされるリソース 1 1 0 6 は、ユーザがアクセスできるメモリ例えば N V R A M 1 1 0 8 に記憶される。すなわち、このメモリはユーザが書き込むことができるので、アクセス可能である。1つの特徴として、セキュリティモジュール 1 1 1 6 は、ダウンロードされた制限ソフトウェアリソース 1 1 0 6 と、工場でロードされたリソースを記憶する R O M 1 1 2 6 とを区別する。

【 0 0 5 8 】

一般に、第 2 外部ポート 1 1 1 2 は、画像形成ジョブに関連したユーザグループ識別 (U G I D) も受信する。このセキュリティモジュール 1 1 1 6 は、メモリ (例えば、V R A M 1 1 0 8、外部メモリ 1 1 4 0 またはポータブルメモリ 1 1 4 2) 内の制限されたソフトウェアリソースと、記憶されたユーザグループ識別とをクロスレファレンスするテーブル 1 1 2 6 を含む。セキュリティモジュール 1 1 1 6 は、受信されたユーザグループ識別と、記憶されたユーザグループ識別とを比較することにより、画像形成ジョブがユーザグループに関連していることを証明する。1つの特徴では、第 2 外部ポートは処理のための画像形成ジョブを受け入れるのと同時に、ユーザグループ識別を受信する。すなわち、U G I D は、同一メッセージまたは同一の組のメッセージ内でジョブとして受信される。例えば、U G I D は、画像ジョブを伴う P J L ステートメントでよい。

30

40

【 0 0 5 9 】

別の特徴では、画像形成ジョブの後で U G I D を受信する。処理のために制限されたソフトウェアリソース 1 1 0 6 が必要であるとセキュリティモジュール 1 1 1 6 が決定し、外部ポートで供給されたユーザグループ識別を発生する。リクエストに 응답し、 응답した外部ポートでユーザグループ識別を受信する。1つの特徴として、画像形成ジョブと同じポートで U G I D リクエストと U G I D 応答とをトランシーブする。これとは異なり、異なるインターフェースを介して U G I D リクエスト / 応答を行ってもよい。1つの特徴では、プリンタデバイスのフロントパネル 1 1 3 0 を介して U G I D リクエスト / 応答をトランシーブする。

【 0 0 6 0 】

50

1つの特徴として、セキュリティモジュール1116はユーザグループ識別、例えば、PIN番号、一方向ハッシュ番号、ユーザIDとパスワードとの組み合わせ、磁気カードまたはスマートカードを受信する。しかしながら、このIDを生体認証とすることもできる。この識別手段は、特定の方法だけに限定されない。識別をフロントパネル1130に送ってもよいし、または、ローカル接続またはネットワーク接続（例えば、ライン1112）を介して送ってもよい。

【0061】

1つの特徴として、リソースにアクセスするのに必要なユーザ認証を選択することができる。例えば、第1（または第2）外部ポートは、ダウンロードされたリソースおよびNVRAM1108にアクセスするのに必要なユーザ認証基準を受信してもよいし、または、他のアクセス可能なメモリがユーザ認証基準を記憶してもよい。次に、リソースがリクエストされると、セキュリティモジュールが認証条件を決定するためにメモリにアクセスする。

10

【0062】

ソフトウェアリソースは、画像形成ジョブまたはプリントジョブを処理するのを助ける任意のコンポーネントとすることができる。ユーザグループが制限したいソフトウェアグループの例として、フォーム、フォント、マクロ、ロゴ、透かし、アドレス帳、ハーフトーンパターン、カラープロファイル、カバーシート、画像スタンプ、校正データ、デジタル署名、生体認証情報、オーディオ/ビデオクリップおよび辞書を挙げることができる。リソースの上記リストは全てではない。

20

【0063】

1つの特徴として、ユーザはリソースに適用される制限を制御できる。例えば、第1（または第2）外部インターフェースは、ダウンロードされたリソースに適用すべき、任意のフォーマットの制限の組を受け入れることができる。これら制限の組は、メモリ例えばNVRAM1108に記憶される。あるリソースがリクエストされると、セキュリティモジュールはNVRAMにアクセスし、そのリソースに適用すべき制限を決定できる。

【0064】

ファンクションの説明

ダウンロードされたリソースアクセスシステムのための作動環境の一例は、プリントまたは多機能周辺デバイス（MFP）にローカルにまたは遠隔接続されたネットワークを含む。プリントデバイスの特定の例として、スタンドアローンプリンタ、ファクシミリデバイス、スキャナ、電子ホワイトボードまたはコピー機を挙げることができ、これらのいずれもハードコピーの結果を作成できる。しかしながら、本発明は、後にリプリントするためにラスタ化された画像を記憶できる、ファイリングデバイスのような、画像または電子ファイルを操作するデバイスにも適用できる。プリントデバイスはフォーマット変換、オーディオ/ビデオデバイス、ドキュメントアーカイブ/検索、操作および転送のようなドキュメント管理、スペクトル発生および分析、ソナー、デジタルカメラおよびX線、MRIおよびCATスキャンデバイスのような医療用画像形成を実行するデバイスとすることができる。

30

【0065】

プリントデバイスは、再使用可能な（ソフトウェア）リソースを一時的に、半永久的にまたは永久的に記憶する能力を有する。再使用可能なリソースの例として、フォーム、フォント、マクロ、ロゴ、透かし、アドレス帳、画像スタンプ、ハーフトーンパターンカラープロファイル、カバーシート、校正データ、デジタル署名、生体認証情報、オーディオ/ビデオクリップおよび事象を挙げることができる。

40

【0066】

例えば、プリントジョブは、これまでダウンロードされ、マルチ再使用のためにデバイスに記憶されたフォームを可変データを使用して参照することができる。別の例では、スキャンデータはこれまでダウンロードされ、マルチ再使用のためにデバイスに記憶された、スキャンされる画像データ内に埋め込むべき著作権用透かしを参照できる。更に、ユー

50

ザが接続されたクライアントまたはプリントデバイスのいずれかで（アウトバウンドまたはインバウンドの）画像形成ジョブリクエストを発生すると、ユーザは自らを認証できる。次に、この認証をジョブに関連付けする。

【0067】

認証の一部の例として、1つのPIN番号を含む。このPINは、文字の単純な組み合わせから一方向のハッシュ化された符号例えばMD5までの範囲とすることができる。認証は、数例を挙げるとすれば、ユーザIDとパスワード、生体認証、磁気カード、公開／秘密鍵暗号化（例えば、RSAもしくはAES）、またはスマートカードとすることができる。

【0068】

本発明のシステムは、ユーザまたは管理者が1つ以上の再使用可能なリソースをプリントデバイスにダウンロードできるようにする。この場合、リソースは後にアクセスすべき認証条件を有する。シリアル、USB、パラレルまたはIRDAインターフェースを使ってローカル接続（有線または無線）を介し、デバイス内にリソースをダウンロードできる。これとは異なり、（有線または無線の）ネットワーク接続例えばTCP/IPまたはアップルトークを使用できる。

【0069】

ダウンロードは、デバイスにおけるウォークアップ動作として行なってもよいし、または、別の接続デバイスから遠隔的に行なってもよい。ダウンロードするリソースは、ホスト（これからダウンロードが行われる）上のローカルでもよいし、インターネットまたは別の接続デバイス上で外部でもよいし、または、CD、フロッピー（登録商標）またはメモリスティックのような記憶メディアから行ってもよい。一部の特征として、デバイスにダウンロードする前に、例えば、ランレングス、LZW、デルタ-ロー、圧縮のG3またはG4方法を使ってリソースデータを圧縮することができる。

【0070】

リソースをダウンロードする方法の一例は、HP P J Lのファイルシステム制御コマンドを使用することである。ファイルダウンロードコマンドの一例は、（@PJL FSDONLOAD FORMAT: BINARY...）であり、これは次のようになる。

【0071】

```
@ P J L F S D O N L O A D F O R M A T : B I N A R Y = < s i z e >
N A M E = " < r e s o u r c e   n a m e > "
< r e s o u r c e   d a t a >
< E s c > % - 1 2 3 4 5 X
```

【0072】

別の例では、PCLフォント永久ダウンロード制御コマンド<Esc>*c5Fを使ってリソースをダウンロードできる。

【0073】

リソースを一旦ダウンロードすると、このリソースを、例えば、デバイスがアクセスできる任意の場所、例えば、デバイスに対して内部にある場所（例えば、ハードドライブ、フラッシュメモリまたはRAM）へ記憶できる。これとは異なり、リソースのオフライン記憶装置（例えば、記憶サーバ）、取り外し自在な記憶装置（例えば、CD、フロッピー（登録商標）、メモリスティック、取り外し自在なHD）により、インターネット接続されたロケーションに、プリントデバイスに対して外部でリソースをロードすることもできる。1つの特征として、プリントデバイスはリクエストされたリソースを入手するためにオーソリティとして働く一部の外部デバイスに進まなければならない。

【0074】

ユーザは、使用するために認証されたアクセスを必要とするリソースのダウンロードの一部として、アクセス権および／または認証方法を指定できる。例えば、ユーザは、特定のユーザまたはあるグループのユーザに限定されたリソースの再使用にアクセスを指定することができる。ユーザ／管理者は、リソースにアクセスするための認証手段を指定でき

10

20

30

40

50

る。リソースへのアクセスは、すべてのユーザにオープンにしてもよいが、数例の可能性を挙げるとすれば、リソースにアクセスできる回数、リソースを使用できるジョブのタイプ、一日のうちの時間またはジョブの内容を限定してもよい。

【 0 0 7 5 】

図 1 2 は、ダウンロードされたリソースに適用される限定を選択するためのプロセスを示すブロック略図である。認証 / アクセス権の制限は、リソースと共にプリントデバイスにダウンロードされ、リソースと共に記憶される。リソースをダウンロードするプロセス自身を制限し、所定の認証されたユーザ、リソースの数、リソースのタイプまたはリソースのサイズに限定できる。

【 0 0 7 6 】

図 1 3 は、ジョブを発生しながら、認証を提出し、リソースを選択するプロセスを示すブロック略図である。プロセスデバイスで一旦リソースがダウンロードされると、ユーザはダウンロードされたリソースへのアクセスを必要とするジョブを発生できる。この場合、ジョブ（例えば、プリント、ファックス、ファイルまたはスキャンジョブ）を発生するときに、ユーザは、次の追加的ステップを実行できる。まず、ユーザを認証するための情報を入力し、次に、ジョブを処理するのに必要なダウンロードされたリソースのうちの 1 つ以上を識別する。斯様に、又は、従って、発生されたジョブはユーザおよび必要とされたリソースを認証するための情報を含む。

【 0 0 7 7 】

プリントジョブに対し、ユーザのネットワークユーザ ID およびパスワードを P J L コマンドとして送ることにより認証を達成できる。更に、一方向ハッシュ（例えば、MD5）を使用する暗号化により、ユーザのパスワードを保護できる。コピージョブに対し、マグカードまたはスマートカードをスライドさせることにより、ユーザを認証してもよい。

【 0 0 7 8 】

図 1 4 は、提出されたジョブを分析するプロセスを示すブロック略図である。プリントデバイスがジョブを受信すると、ジョブ内に埋め込まれていないリソースをジョブが必要とするかどうかを判断するためにジョブ情報を分析する。

【 0 0 7 9 】

図 1 5 は、ユーザ識別を認証し、制限されたリソースにアクセスするプロセスを示すフローチャートである。ジョブが、埋め込まれたリソース条件を含まない場合、このジョブを処理する。含んでいる場合、プリントデバイスは、必要とされるリソースのいずれかが認証を必要とするかどうかを判断する。一般に、この判断は、記憶されているリソースレポジトリ内のリソースをルックアップし、認証 / アクセス制御情報を得ることにより行われる。リソースのいずれもが認証を必要としない場合、ジョブを処理する。リソースの 1 つ以上が認証を必要とする場合、プリントデバイスはジョブが認証情報を含むかどうかを判断する。認証情報は、ジョブ制御コマンド（例えば P J L コマンド）として埋め込んでもよいし、または、フロントパネルにてまたはカードをスライドすることにより、別個に入力してもよい。

【 0 0 8 0 】

ジョブがユーザを認証するための情報を含まない場合、ジョブを拒否する。そうでない場合、認証情報と必要とされる各リソースの認証されたユーザ / アクセス権とを比較する。認証情報が各リソースの認証 / アクセス権と一致する場合、ジョブを処理し、一致しない場合、ジョブを拒否する。

【 0 0 8 1 】

クライアントデバイスのプリントサブシステムは、マイクロソフトウィンドウズ（登録商標）のオペレーティングシステム、アップルマッキントッシュのオペレーティングシステム、ライナックスオペレーティングシステム、システム V ユニックスのオペレーティングシステム、BSD ユニックスのオペレーティングシステム、OSF ユニックスのオペレーティングシステム、サンソラリスのオペレーティングシステム、HP / UX のオペレーティングシステムおよび IBM メインフレームの MVS オペレーティングシステムに基づ

10

20

30

40

50

くことができる。しかしながら、本発明は任意の特定のオペレーティングシステムだけに限定されるものではない。

【 0 0 8 2 】

図 1 6 は、プリントデバイス内の画像形成ジョブリソースに安全にアクセスするための方法を示すフローチャートである。このプリントデバイスは、プリンタ、ファクシミリデバイス、ファイリングデバイス、フォーマット変換デバイス、パブリッシングデバイス、スキャナ、コピー機、電子ホワイトボード、オーディオ/ビデオデバイス、デジタルカメラまたは医療用画像形成デバイスとすることができる。この方法は、説明を明瞭にするために、一連の番号の付いたステップとして示されているが、番号は必ずしもステップの順序を定めるものではない。これらステップの一部は、スキップしたり、並列に実行したり、シーケンスの厳密な順序を維持しないで実行することができる。この方法はステップ 1 6 0 0 でスタートする。

10

【 0 0 8 3 】

ステップ 1 6 0 2 は、制限されたソフトウェアリソースをプリントデバイスにダウンロードする。リソースの使用は、特定のユーザグループに使用が限定されている。ステップ 1 6 0 4 は、メモリ、例えば、（内部の）不揮発性記憶装置、ローカル接続またはネットワーク接続された外部メモリ、または、ポータブルな記憶メディアに限られたソフトウェアリソースを記憶する。ステップ 1 6 0 6 は、処理のための画像形成ジョブを受け入れる。1 つの特徴として、ステップ 1 6 0 6 における処理のために画像形成ジョブを受け入れることは、ページ記述言語（PDL）、プリントジョブ言語（PJL）、PDL と PJL の組み合わせ、または、画像フォーマット例えば T I F F、J P E G または P M G でのプリントジョブを受け入れることを含む。ステップ 1 6 0 8 は、画像形成ジョブがユーザグループに関係していることを証明する。この証明の後のステップ 1 6 1 0 は、限られたソフトウェアリソースへのアクセスを許可する。ステップ 1 6 1 2 は、限られたソフトウェアリソースを使用して画像形成ジョブを処理する。

20

【 0 0 8 4 】

1 つの特徴として、記憶装置に限られたソフトウェアリソースを記憶すること（ステップ 1 6 0 4）は、メモリ内のリソースと、記憶されたユーザグループの識別とをクロスレファレンスすることを含む。次に、ステップ 1 6 0 8 における画像形成ジョブがユーザグループに関連していることを証明することは、サブステップを含む。ステップ 1 6 0 8 a は、ユーザグループの識別を受信し、ステップ 1 6 0 8 b は、受信したユーザグループの識別と記憶されているユーザグループの識別とを比較する。

30

【 0 0 8 5 】

別の特徴として、ユーザグループの識別を受信すること（ステップ 1 6 0 8 a）は、処理するために画像形成ジョブを受け入れること（ステップ 1 6 0 6）と同時にユーザグループの識別を受信することを含む。これとは異なり、ステップ 1 6 0 8 a におけるユーザグループの識別を受信することは、追加的サブステップを含む。画像形成ジョブを受け入れることに応答し、ステップ 1 6 0 8 a 1 は、処理するために限られたソフトウェアリソースが必要であると判断する。ステップ 1 6 0 8 a 2 は、ユーザグループの識別リクエストを発生し、ステップ 1 6 0 8 a 3 は、リクエストに応答し、ユーザグループの識別を受信する。

40

【 0 0 8 6 】

ステップ 1 6 0 8 a で受信されるユーザグループの識別を P I N 番号、一方向ハッシュ番号、ユーザ I D とパスワードの組み合わせ、磁気カードまたはスマートカードとすることができる。しかしながら、本発明は、特定のタイプの識別だけに限定されるものではない。

【 0 0 8 7 】

1 つの特徴として、ステップ 1 6 0 2 における制限されたソフトウェアリソースをダウンロードすることは、ローカル接続、ネットワーク接続またはポータブル記憶媒体のような通信媒体を使ってリソースをダウンロードすることを含む。ダウンロードできるリソ

50

スの一部の例として、フォーム、フォント、マクロ、ロゴ、透かし、アドレス帳、ハーフトーンパターン、カラープロファイル、画像スタンプ、カバーシート、校正データ、デジタル署名、生体認証情報、オーディオ/ビデオクリップおよび辞書を挙げることができる。

【 0 0 8 8 】

別の特徴として、ステップ 1 6 0 7 a は、ダウンロードされたリソースにアクセスするのに必要なユーザ認証基準を受け入れ、ステップ 1 6 0 7 b はユーザ認証基準を記憶する。これとは異なりまたはこれに加えて、ステップ 1 6 0 7 c はダウンロードされたリソースに適用すべき一組の制限を受け入れ、ステップ 1 6 0 7 d はこの制限の組を記憶する。

【 0 0 8 9 】

1 つの特徴として、この方法は別のステップを実行する。ステップ 1 6 0 2 における制限されたソフトウェアリソースをダウンロードすることの後で、ステップ 1 6 0 3 はダウンロードされたリソースと、リードオンリーメモリ (R O M) に記憶されている、工場ロードされたリソースとを区別する。

【 0 0 9 0 】

以上で、プリントデバイスにおけるダウンロードされたソフトウェアリソースへのアクセスを安全にするためのシステムおよび方法について説明した。本発明を説明するために、特定のフォーマットおよびプロトコルの例について説明した。同様に、特定のリソースのタイプ、限定のタイプおよび認証のタイプの例について説明した。しかしながら、本発明はこれら例だけに限定されるものでなく、当業者には本発明の他の変形例および実施例を想到できよう。

【 図面の簡単な説明 】

【 0 0 9 1 】

【 図 1 】 未処理スプールファイルを使用する従来のプリント動作を示す図である。

【 図 2 】 スプーラサブシステムを示す図である。

【 図 3 】 ウィンドウズ (登録商標) E M F プリント動作を示す図である。

【 図 4 】 ウィンドウズ (登録商標) E M F プリント動作を示す図である。

【 図 5 】 マイクロソフトウィンドウズ (登録商標) の N T および 2 0 0 0 のプリント動作を示す図である。

【 図 6 】 ウィンドウズ (登録商標) 2 0 0 0 のプリントプロセスを示す図である。

【 図 7 】 プリントデバイスシステムのブロック略図である。

【 図 8 】 ソフトウェアリソースをプリンタデバイスにダウンロードするためのシステムのブロック略図である。

【 図 9 】 プリントジョブ処理における制限されたリソースの使用を防止するためにセキュリティモジュールを使用するシステムのブロック略図である。

【 図 1 0 】 すべての制限されたリソースへの、制限されないアクセスを特別なユーザに許可するシステムを示すブロック略図である。

【 図 1 1 】 画像形成ジョブリソースに安全にアクセスするためのシステムを備えた、プリントデバイスのブロック略図である。

【 図 1 2 】 ダウンロードされたリソースへ適用される制限を選択するプロセスを示すブロック略図である。

【 図 1 3 】 ジョブを発生しながら認証を提出し、リソースを選択するプロセスを示すブロック略図である。

【 図 1 4 】 提出されたジョブを分析するプロセスを示すブロック略図である。

【 図 1 5 】 ユーザアイデンティティを認証し、制限されたリソースにアクセスするプロセスを示すフローチャートである。

【 図 1 6 】 プリントデバイスにおける画像形成ジョブリソースに安全にアクセスするための方法を示すフローチャートである。

【 符号の説明 】

【 0 0 9 2 】

10

20

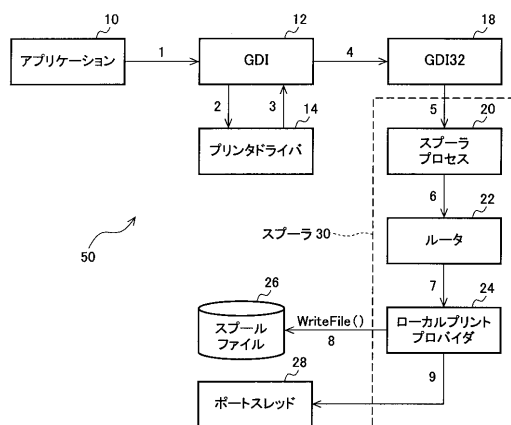
30

40

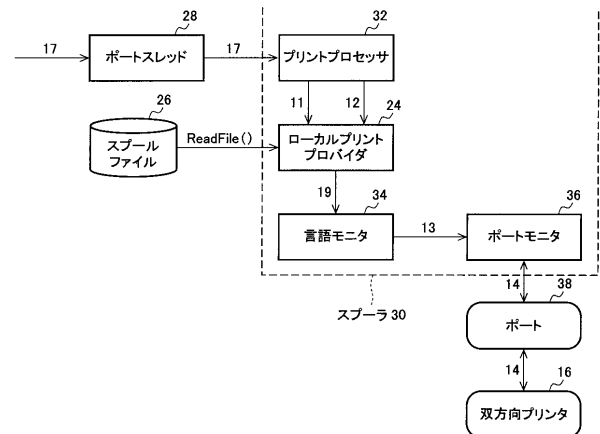
50

10 ... アプリケーション、12 ... GDI、14 ... プリントドライバ、16 ... プリンタ、18 ... GDI32、20 ... スプーラプロセス、22 ... ルータ、24 ... ローカルプリントプロバイダ、26 ... スプールファイル、28 ... バックグラウンドスレッド、30 ... スプーラ、34 ... 言語モニタ、36 ... ポートモニタ、38 ... ポート、40 ... アプリケーション、50 ... GDI、52 ... プリントドライバ、54 ... EMFファイル、56 ... GDI32、58 ... スプーラプロセス、60 ... ルータ、62 ... ローカルプリントプロバイダ、64 ... ジョブ記述ファイル、66 ... ポートスレッド、68 ... プリンタ、78 ... ポート、74 ... 言語モニタ、76 ... ポートモニタ、70 ... スプーラシステム、72 ... プリントプロセッサ。

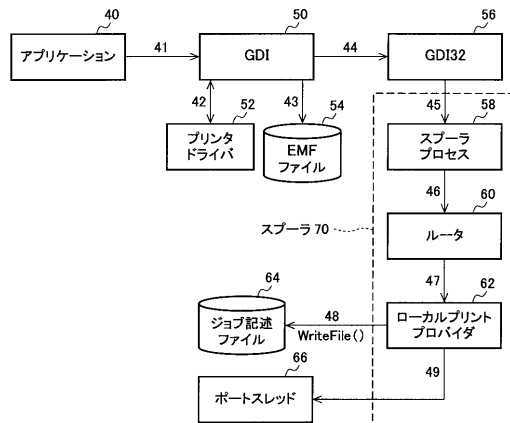
【図1】



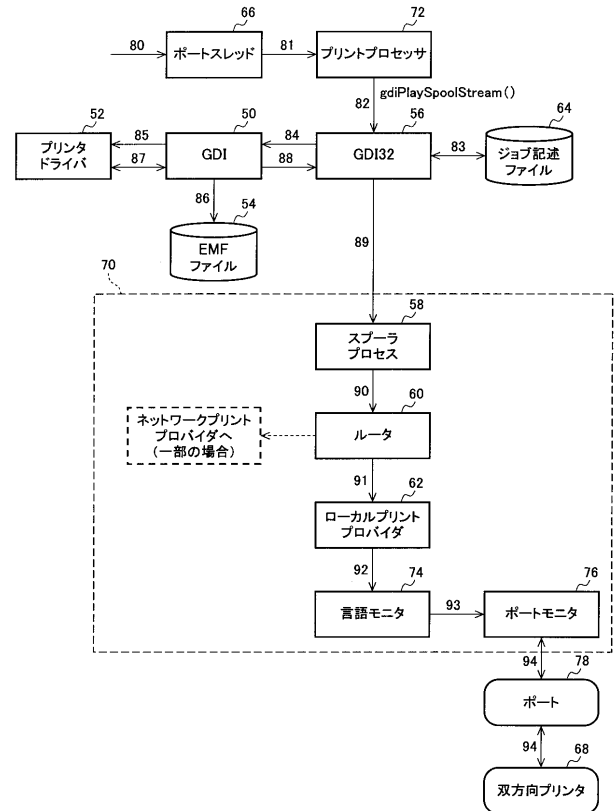
【図2】



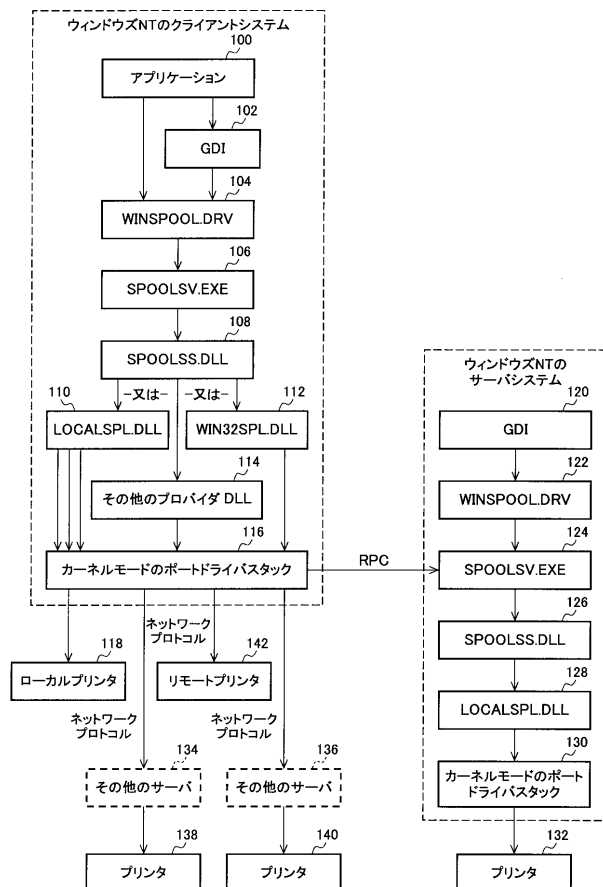
【図 3】



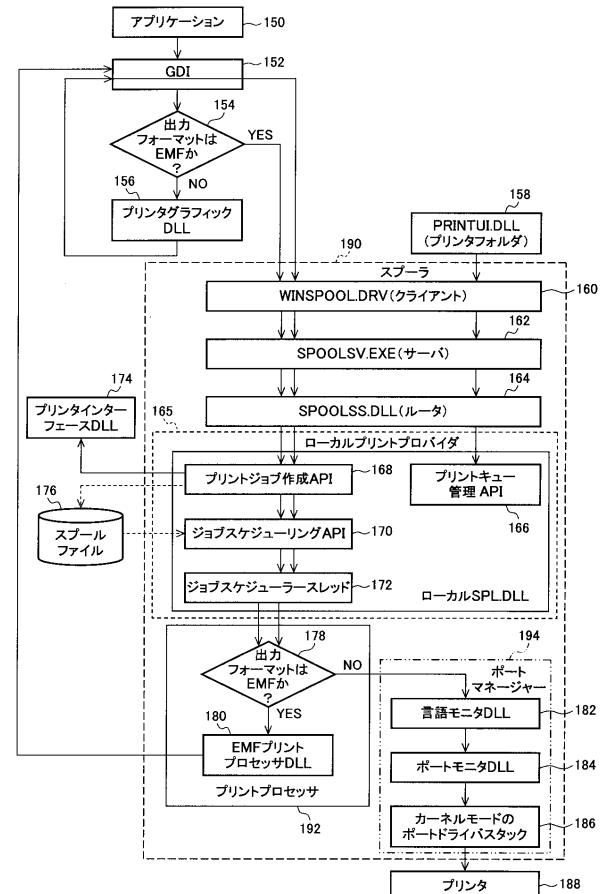
【図 4】



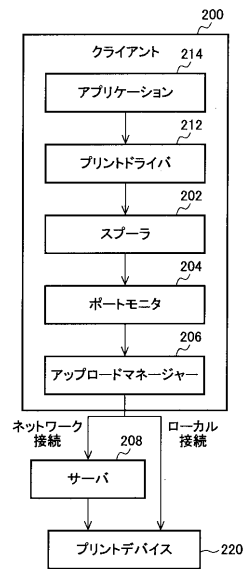
【図 5】



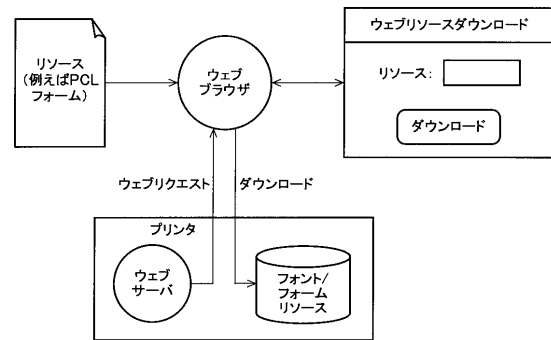
【図 6】



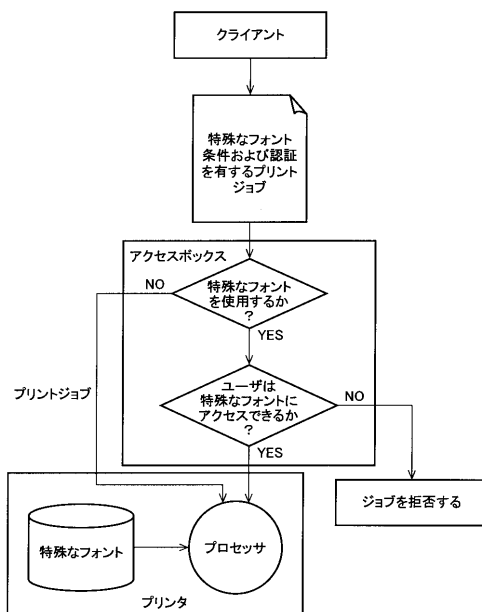
【図 7】



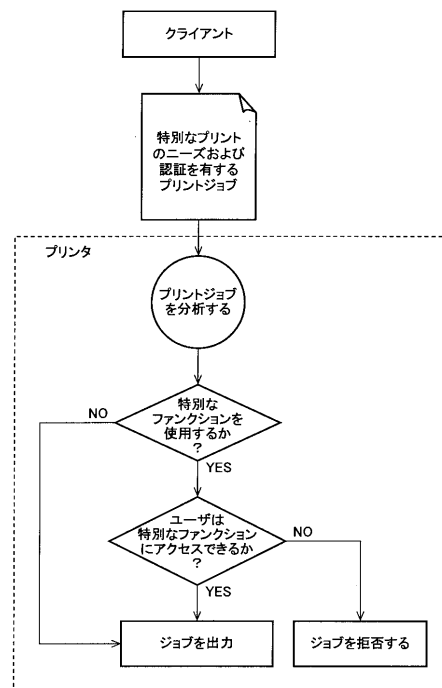
【図 8】



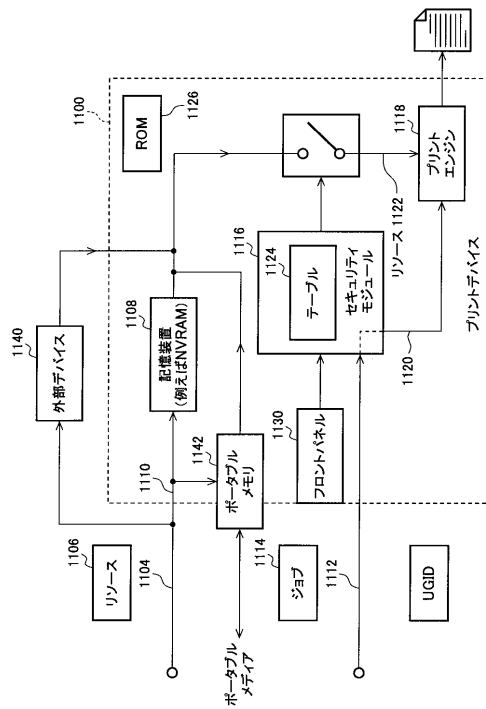
【図 9】



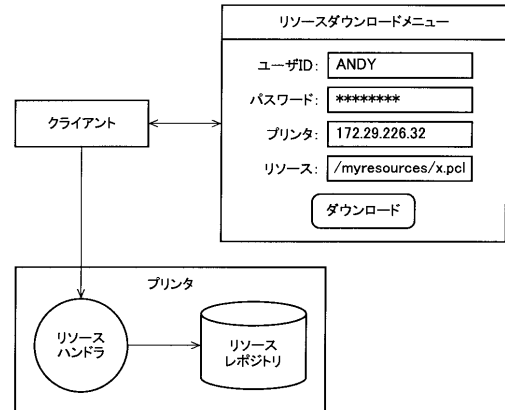
【図 10】



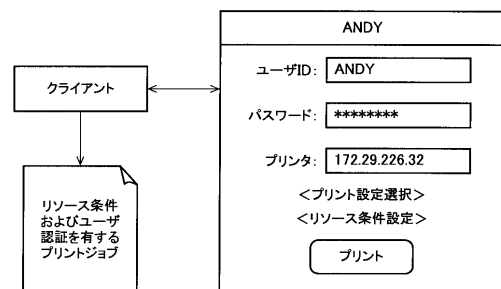
【図 1 1】



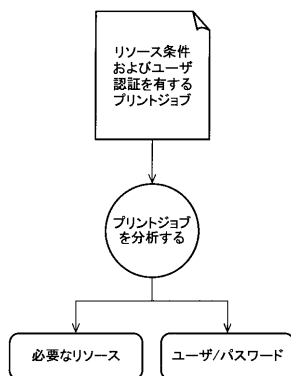
【図 1 2】



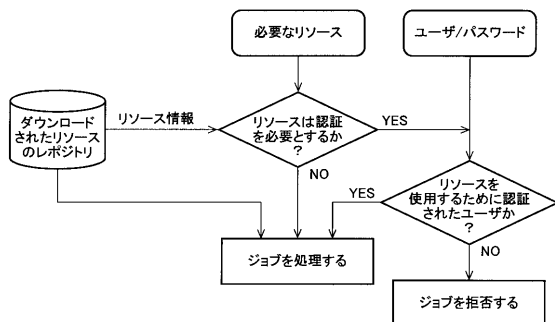
【図 1 3】



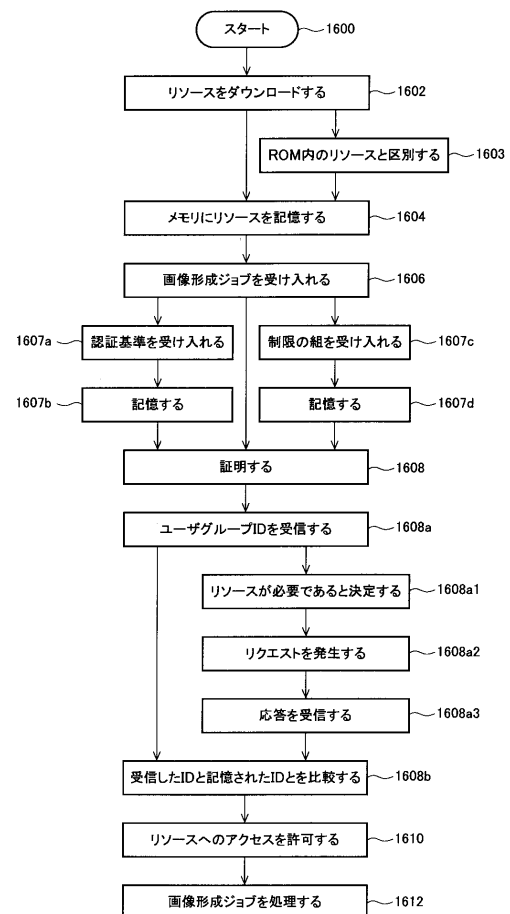
【図 1 4】



【図 1 5】



【図 1 6】



フロントページの続き

(72)発明者 アンドリュー ロドネイ ファーリッシュ

アメリカ合衆国 9 8 6 0 7 ワシントン州, カマス, 1 9 4 番 アベニュー, エヌイー 1 3 0
8

F ターム(参考) 2C061 AP01 AP03 AP04 CL08 CL10 HJ08 HQ12 HQ17 HQ22 HX10
5B021 AA01