

(19) 日本国特許庁 (JP)

(12) 特 許 公 報 (B2)

(11) 特許番号

特許第4522006号
(P4522006)

(45) 発行日 平成22年8月11日 (2010. 8. 11)

(24) 登録日 平成22年6月4日 (2010. 6. 4)

(51) Int. Cl.

G 0 6 F 3/12 (2006.01)

F I

G 0 6 F 3/12

G

請求項の数 16 (全 21 頁)

(21) 出願番号	特願2001-55466 (P2001-55466)	(73) 特許権者	000001007
(22) 出願日	平成13年2月28日 (2001. 2. 28)		キヤノン株式会社
(65) 公開番号	特開2002-259101 (P2002-259101A)		東京都大田区下丸子3丁目30番2号
(43) 公開日	平成14年9月13日 (2002. 9. 13)	(74) 代理人	100076428
審査請求日	平成20年2月25日 (2008. 2. 25)		弁理士 大塚 康德
		(74) 代理人	100115071
			弁理士 大塚 康弘
		(72) 発明者	大村 宏
			東京都大田区下丸子3丁目30番2号 キ
			ヤノン株式会社内
		審査官	中田 剛史

最終頁に続く

(54) 【発明の名称】 情報処理装置、情報処理方法及びプリンタドライバプログラム

(57) 【特許請求の範囲】

【請求項 1】

印刷モードに1対1に対応し、印刷装置が印刷可能な印刷データを生成可能な複数の印刷データ生成手段と、

基本ソフトウェアより渡される描画命令を前記複数の処理手段で処理可能な形式のデータに変換するスプールドライバ手段と、

前記描画命令を分析し、印刷モード判定情報を生成する印刷モード判定手段と、

前記スプールドライバ手段によって生成されたデータを保持する記憶手段と、

前記印刷モード判定手段によって生成された印刷モード判定情報を保持する印刷モード判定記憶手段と、

前記印刷モード判定情報を用いて印刷モードをページごとに決定し、前記複数の印刷データ生成手段のうち、前記決定された印刷モードに対応する印刷データ生成手段に前記スプールドライバ手段によって生成されたデータをページごとに供給するデスプーラ手段とを有する情報処理装置において、

前記スプールドライバ手段は、前記記憶手段に記憶されたデータにより指定されているフォントが前記情報処理装置に存在する場合には記指定されているフォントを前記デスプーラ手段に通知し、前記指定されているフォントが前記情報処理装置に存在しない場合には前記情報処理装置に存在する予め定められたフォントをデスプーラ手段に通知し、

前記デスプーラ手段は、前記スプールドライバ手段により通知されたフォントを、前記決定された印刷モードに対応する前記印刷データ生成手段に論理フォントとして登録し、

10

20

前記決定された印刷モードに対応する印刷データ生成手段の各々は、前記デスプーラ手段によりページごとに供給されるデータと前記デスプーラ手段により登録された前記論理フォントを用いて、前記印刷データを生成することを特徴とする情報処理装置。

【請求項 2】

基本ソフトウェアから受け渡される描画命令に基づき、印刷装置で印刷可能な形式の印刷データを生成、出力する情報処理装置であって、

描画命令から、互いに異なるデータ形式の印刷データを生成する複数の印刷データ生成手段と、

前記印刷装置におけるページ単位に対応する描画命令群に基づき、前記複数の印刷データ生成手段のうち 1 つを前記ページ単位で選択する選択手段と、

前記描画命令群に含まれるフォント属性から、前記情報処理装置に存在するフォントのフォント属性が検出された場合には、前記描画命令群に含まれるフォント属性を通知し、前記情報処理装置に存在しないフォントのフォント属性が検出された場合には、前記情報処理装置に存在する予め定められたフォントのフォント属性を通知する置き換えフォント決定手段と、

前記選択手段が選択した印刷データ生成手段に対し、供給された描画命令群中に前記情報処理装置に存在しないフォントのフォント属性が検出された場合には、前記置き換えフォント決定手段により通知されたフォント属性に対応する、前記情報処理装置に存在する予め定められたフォントを用いて印刷データ生成処理を行うよう指示する指示手段と、

前記ページ単位に対応する描画命令群を前記選択手段が選択した印刷データ生成手段に供給する供給手段とを有することを特徴とする情報処理装置。

【請求項 3】

基本ソフトから渡されるデータを複数の印刷データ生成手段で処理可能な形式でスプールするスプールドライバ手段と、

前記スプールされたデータに基づきページごとに印刷モードを判定する印刷モード判定手段と、

前記複数の印刷データ生成手段のうち、前記判定手段により判定された印刷モードに対応する印刷データ生成手段に、前記スプールされたデータをページごとに供給するデスプーラ手段とを有する情報処理装置において、

前記スプールドライバ手段は、前記スプールされたデータ中のフォント情報から、前記情報処理装置に存在しないフォントのフォント情報を検出した場合には、前記情報処理装置に存在する予め定められたフォントのフォント情報を前記デスプーラ手段に通知し、前記スプールされたデータ中のフォント情報から、前記情報処理装置に存在するフォントのフォント情報を検出した場合には、前記スプールされたデータ中のフォント情報をデスプーラ手段に通知し、

前記デスプーラ手段は、前記スプールドライバ手段により通知されたフォント情報を前記判定された印刷モードに対応する印刷データ生成手段に登録し、

前記判定された印刷モードに対応する印刷データ生成手段の各々は、前記デスプーラ手段によりページごとに供給されるデータと、前記デスプーラ手段に登録されたフォント情報とを用いて印刷データを生成することを特徴とする情報処理装置。

【請求項 4】

基本ソフトから渡されるデータを複数の印刷データ生成手段で処理可能な形式でスプールするスプールドライバ手段と、

前記スプールされたデータに基づきページごとに印刷モードを判定する印刷モード判定手段と、

前記複数の印刷データ生成手段のうち、前記判定手段により判定された印刷モードに対応する印刷データ生成手段に、前記スプールされたデータをページごとに供給するデスプーラ手段とを有する情報処理装置における情報処理方法において、

前記スプールドライバ手段が、前記スプールされたデータ中のフォント情報から、前記情報処理装置に存在しないフォントのフォント情報を検出した場合には、前記情報処理装

10

20

30

40

50

置に存在する予め定められたフォントのフォント情報を前記デスプーラ手段に通知し、前記スプールされたデータ中のフォント情報から、前記情報処理装置に存在するフォントのフォント情報を検出した場合には、前記スプールされたデータ中のフォント情報をデスプーラ手段に通知する通知工程と、

前記デスプーラ手段が、前記スプールドライバ手段により通知されたフォント情報を前記判定された印刷モードに対応する印刷データ生成手段に登録する登録工程と、

前記判定された印刷モードに対応する印刷データ生成手段の各々が、前記デスプーラ手段によりページごとに供給されるデータと、前記デスプーラ手段に登録されたフォント情報とを用いて印刷データを生成する生成工程とを有することを特徴とする情報処理方法。

【請求項 5】

情報処理装置であって、

アプリケーションからのデータに基づきプリンタで処理する印刷データを作成する複数の印刷データ作成手段と、

アプリケーションからのデータを解析し、当該解析の結果に基づいて、前記データをページごとに前記複数の印刷データ作成手段の 1 つに供給する供給手段と、

前記アプリケーションからのデータにより指定されているフォントが前記情報処理装置に存在すれば、前記指定されているフォントを論理フォントとして、前記データの供給先である前記印刷データ作成手段に登録し、前記アプリケーションからのデータにより指定されているフォントが前記情報処理装置に存在しなければ、前記情報処理装置に存在する予め定められたフォントを論理フォントとして前記データの供給先である前記印刷データ作成手段に登録する登録手段とを有し、

前記データの供給先である前記印刷データ生成手段の各々は、前記供給手段により供給されたデータと、前記登録手段により登録された論理フォントとを用いて、印刷データを生成することを特徴とする情報処理装置。

【請求項 6】

前記印刷データは、イメージデータと、プリンタでイメージデータを作成するための制御コマンドを含むことを特徴とする請求項 5 記載の情報処理装置。

【請求項 7】

前記印刷データ作成手段は、前記印刷データとしてイメージデータを作成するイメージデータ第 1 作成手段と、前記印刷データとしてイメージデータを作成する第 2 作成手段とを含むことを特徴とする請求項 5 記載の情報処理装置。

【請求項 8】

前記印刷データ作成手段は、前記印刷データとしてイメージデータを作成するイメージデータ第 1 作成手段と、前記印刷データとしてプリンタでイメージデータを作成するためのデータを作成する第 2 作成手段とを含むことを特徴とする請求項 5 記載の情報処理装置。

【請求項 9】

アプリケーションからのデータに基づきプリンタで処理する印刷データを作成する複数の印刷データ作成手段を利用する情報処理装置における情報処理方法であって、

供給手段が、アプリケーションからのデータを解析し、当該解析の結果に基づいて、前記データをページごとに前記複数の印刷データ作成手段の 1 つに供給する供給工程と、

登録手段が、前記アプリケーションからのデータにより指定されているフォントが前記情報処理装置に存在すれば、前記指定されているフォントを論理フォントとして、前記データの供給先である前記印刷データ作成手段に登録し、前記アプリケーションからのデータにより指定されているフォントが前記情報処理装置に存在しなければ、前記情報処理装置に存在する予め定められたフォントを論理フォントとして前記データの供給先である前記印刷データ作成手段に登録する登録工程と、

前記データの供給先である前記印刷データ生成手段の各々が、前記供給手段により供給されたデータと、前記登録手段により登録された論理フォントとを用いて、印刷データを生成する生成工程とを有することを特徴とする情報処理方法。

10

20

30

40

50

【請求項 1 0】

前記印刷データは、イメージデータと、プリンタでイメージデータを作成するための制御コマンドとを含むことを特徴とする請求項 9 記載の情報処理方法。

【請求項 1 1】

前記印刷データ作成手段は、前記印刷データとしてイメージデータを作成するイメージデータ第 1 作成手段と、前記印刷データとしてイメージデータを作成するデータを作成する第 2 作成手段とを含むことを特徴とする請求項 9 記載の情報処理方法。

【請求項 1 2】

前記印刷データ作成手段は、前記印刷データとしてイメージデータを作成するイメージデータ第 1 作成手段と、前記印刷データとしてプリンタでイメージデータを作成する第 2 作成手段とを含むことを特徴とする請求項 9 記載の情報処理方法。

10

【請求項 1 3】

コンピュータを、

アプリケーションからのデータに基づきプリンタで処理する印刷データを作成する複数の印刷データ作成手段と、

アプリケーションからのデータを解析し、当該解析の結果に基づいて、前記データをページごとに前記複数の印刷データ作成手段の 1 つに供給する供給手段と、

前記アプリケーションからのデータにより指定されているフォントが前記コンピュータに存在すれば、前記指定されているフォントを論理フォントとして、前記データの供給先である前記印刷データ作成手段に登録し、前記アプリケーションからのデータにより指定されているフォントが前記コンピュータに存在しなければ、前記コンピュータに存在する予め定められたフォントを論理フォントとして前記データの供給先である前記印刷データ作成手段に登録する登録手段として機能させるためのプリンタドライバプログラムであって、

20

前記データの供給先である前記印刷データ生成手段の各々は、前記供給手段により供給されたデータと、前記登録手段により登録された論理フォントとを用いて、印刷データを生成することを特徴とするプリンタドライバプログラム。

【請求項 1 4】

前記印刷データは、イメージデータとプリンタでイメージデータを作成するための制御コマンドとを含むことを特徴とする請求項 1 3 記載のプリンタドライバプログラム。

30

【請求項 1 5】

前記印刷データ作成手段は、前記印刷データとしてイメージデータを作成するイメージデータ第 1 作成手段と、前記印刷データとしてイメージデータを作成する第 2 作成手段として機能することを特徴とする請求項 1 3 記載のプリンタドライバプログラム。

【請求項 1 6】

前記印刷データ作成手段は、前記印刷データとしてイメージデータを作成するイメージデータ第 1 作成手段と、前記印刷データとしてプリンタでイメージデータを作成するためのデータを作成する第 2 作成手段として機能することを特徴とする請求項 1 3 記載のプリンタドライバプログラム。

【発明の詳細な説明】

40

【0 0 0 1】

【発明の属する技術分野】

本発明は、アプリケーションソフトウェア等の印刷指示により、プリンタ等の画像形成装置に対する印刷データを生成する情報処理装置、情報処理方法及びプリンタドライバプログラムに関する。

【0 0 0 2】

【従来の技術】

図 9 は、画像形成装置としてのプリンタ及びその制御を行う情報処理装置としてのホストコンピュータで構成された、従来の印刷システムの例を示す図である。

【0 0 0 3】

50

図に示すように、ホストコンピュータ200は、セントロニクスインターフェース、IEEE1284といったパラレルインタフェース、USB、IEEE1394等のシリアルインタフェース、10/100BASE-T等のネットワークインタフェース600を介してプリンタ100と接続されている。ホストコンピュータ200には、CRT、LCD等のディスプレイモニタ300及び入力デバイスとしてのキーボード400及びマウス500が接続されている。

【0004】

ホストコンピュータ200では、Windows（米国Microsoft社の登録商標）のようないわゆる基本ソフト（OS）の上でワードプロセッサや表計算のような、プリンタ100を利用可能なアプリケーションソフトウェア201（以下アプリケーションと略称）が動作している。これらアプリケーションのユーザが印刷を指示した場合、そのアプリケーションは、基本ソフトが提供するいくつかのサブシステムの内、グラフィックサブシステム202の機能を用いて印刷を行う。

10

【0005】

このグラフィックサブシステムは、例えば基本ソフトがWindowsの場合、GDI(Graphic Device Interface)203と呼ばれており、ディスプレイやプリンタに対する描画処理を司っている。このGDI（203）は、ディスプレイやプリンタといったデバイス毎の依存性を吸収するために、アプリケーションから指定された出力デバイスに対応する、デバイスドライバと呼ばれるモジュールを動的にリンクし、それぞれのデバイスに対する出力処理を行う。

20

【0006】

プリンタに対するデバイスドライバはプリンタドライバと呼ばれる。プリンタドライバ204は、対応するプリンタの能力や機能などに応じて、あらかじめデバイスドライバに実装することが決められているDDI(Device Driver Interface)と呼ばれる関数群を有しており、GDI（203）がこのDDIを用いることによって印刷処理が行われる。

【0007】

すなわち、図9の例において、アプリケーションから印刷指示（GDI（203）へのAPI(Application Programming Interface)コール）があると、GDI（203）はプリンタ100に対応するプリンタドライバ204を動的にリンクするとともに、アプリケーションからのAPIコールをプリンタドライバ204が有するDDI関数群を用いた形式に変換する。そして、GDI（203）からDDI関数群がコールされることによって所定の印刷処理が実行される。このように、GDI（203）では、プリンタドライバ204を用いてアプリケーションからの印刷要求をシーケンシャルに処理している。

30

【0008】

プリンタドライバ204は、DDIを介してGDI（203）から受け渡しされるデータから、プリンタ100が印刷可能な形式のデータを生成、出力する。この際、プリンタ100に能力、機能に対応した複数の印刷データ生成部を用いて印刷データを生成し、スプールサブシステム210を介してプリンタ100へ出力する。

【0009】

プリンタ100へ出力する、印刷データ形式としては、ページ記述言語（PDL:Page Description Language）等のプリンタ言語と呼ばれる制御コマンド及びイメージデータ（ビットマップ）の2種類に大別される。そして、プリンタドライバ204に含まれるデスプーラ250は、それぞれが所定形式のデータを出力する複数の印刷データ生成部（205～208）の中から、1つを選択して印刷処理を実行する。

40

【0010】

図9において、印刷データ生成部（205～208）は、PDLを出力する（以下、PDLタイプという）第1の印刷データ生成部205と、イメージデータを出力する（以下、イメージタイプという）第2～第4の印刷データ生成部（206～208）から構成されている。また、イメージデータを出力する第2～第4の印刷データ生成部（206～208）のうち、第2のイメージデータ生成部206はモノクロ(mono 1BPP:1 bit per pixel

50

)イメージデータを、第3のイメージデータ生成部207は各色1ビットのY M C K (YMCK 4BPP)カラーイメージデータを、第4のイメージデータ生成部208は、各色8ビットのR G B (RGB 24BPP)イメージデータをそれぞれ出力する。

【0011】

第1の印刷データ生成部205は、プリンタ100がPDLを処理可能なコントローラを搭載する場合に使用可能であり、第1の印刷データ生成部205を用いる場合、デスクプーラ250はGDI(203)からDDIを介して受け取ったデータを第1の印刷データ生成部205へ受け渡し、PDLのコマンドに変換し次第、スプールサブシステム210を介してプリンタ100に送出する。この場合、デスクプーラ250の処理はDDIを介して受け取ったデータを第1の印刷データ生成部205に受け渡してPDLコマンドを生成するだけなので、軽い処理といえる。

10

【0012】

PDLタイプの印刷データ生成部を用いる場合、GDI(203)から抽象度の高いデータを受け取り、PDLコマンドの形式で送出できるので、通常のページ印刷においては転送する印刷データサイズを小さくすることが可能である。

【0013】

一方、イメージタイプの印刷データ生成部(206~208)は、プリンタ100では高度な描画処理を行わずにすむよう、プリンタドライバ204がホストコンピュータ200で確保したメモリ空間上に印刷イメージの展開を行い、それをプリンタ100で直接印刷できるイメージデータ形式に変換してから、プリンタ100に転送する。

20

【0014】

具体的には、PDLタイプの印刷データ生成部と同様に、GDI(203)からDDIを介して渡されたデータをプリンタドライバ204に含まれるデスクプーラ250がイメージタイプの印刷データ生成部(206~208)の1つに受け渡す。そして、印刷データ生成部は受け渡しされたデータに基づいてバンドメモリ209に描画処理を行い、1バンド分の描画処理が終了すると、スプールサブシステム210を介してプリンタ100にビットマップデータを送出する。プリンタ100は、受信したビットマップデータを直接出力する。

【0015】

イメージタイプの印刷データ生成部を用いる場合はホストコンピュータ200において、PDLタイプの印刷データ生成部を用いる場合はプリンタ100において、描画処理を行う必要がある。描画処理を行うには、メモリ空間を確保する必要があるが、RGB各色8ビットで600DPI(dot per inch), A4フルページ分の描画処理を行うために必要なメモリ空間は96MByte必要であり、現状ではホストコンピュータ200、プリンタ100のどちらでも確保することは困難である。

30

【0016】

そこで、一般には、特にデータ量が多くなるカラープリンタの描画処理には、1ページを水平方向に複数に分割したバンド単位で描画処理が行われ、この描画処理に用いるメモリをバンドメモリと呼ぶ。

【0017】

また、プリンタ100に最終的に転送するデータ形式は、ディスプレイ装置で用いられるようなRGB(輝度)形式ではなく、プリンタ100が用いるトナーの濃度を表すY M C K形式であり、またY M C K形式では各色1ビット、2ビット(又は4ビット)であることが多いため、各色8ビットのRGB形式よりもデータ量が小さくなるという理由から、プリンタドライバ204以降の処理においてはY M C K形式でデータを取り扱うのが一般的である。そして、イメージタイプの印刷データ生成部のなかには、Y M C K各色1ビット、2ビットもしくは4ビットのデータをプリンタ100に転送することで、転送データサイズを小さく抑え、プリンタ100の負荷を軽くするものがある。

40

【0018】

例えば、多角形図形の印字処理を行う場合、PDLタイプの印刷データ生成部は、多角形

50

の頂点座標と線幅、線の色情報、図形内の塗り色情報等がコマンドとして転送され、プリンタ内部でこのコマンドを解析し、プリンタ内部に設けられた、Y M C K各色1ビット、2ビットまたは4ビットのデータに対する1バンド分のメモリ空間を用いて描画処理を行い、出力形態のビットマップデータに展開する。

【0019】

P D Lタイプの印刷データ生成部を用いる場合、プリンタに転送されるデータはP D Lコマンドであり、一般にはイメージタイプの印刷データ生成部が生成するデータよりもサイズが小さくなる。しかし、1ページあたりのP D Lコマンド数には制限が無く、理論上はイメージタイプの印刷データ生成部よりも転送データが大きくなる可能性がある。また、P D Lタイプの印刷データ生成部を用いた場合、プリンタ側で描画処理を行う必要があるが、バンドメモリに対する描画処理であっても、1ページ分のP D Lコマンドをすべて受信してからでないと描画処理が開始できない。近年、アプリケーションが出力する印刷命令は増加の一途をたどっており、1ページを構成するP D Lコマンドも増加しているため、メモリ量の少ないプリンタでは処理速度低下の要因になっている。

10

【0020】

このような場合、プリンタに比べてメモリ容量に余裕のあるホストコンピュータ200上で動くプリンタドライバ204においてバンド単位もしくはフルページ分の描画メモリを確保し、イメージタイプの印刷データ生成部を使用して描画処理を行い、ビットマップデータをプリンタに転送した方が印刷処理全体のスループットは向上する。図9の例では、バンドメモリ209を用いている。

20

【0021】

前述したように、印刷データ生成部は、P D Lタイプ、イメージタイプ共に省メモリのためにY M C K各色1, 2, 4ビットという出力装置依存の色空間上で描画処理を行っているが、G D I (203)が論理演算描画処理時にD D I関数群をコールする場合、用いられる色形式はY M C K形式でなくR G B形式であるため、描画不正が発生することがある。このため、描画不正回避のために、論理演算処理が発生した場合、プリンタドライバ204においてY M C K形式に変換されているデータをR G B値に戻し(逆UCR(Under Color Remove)処理)、それから論理演算処理することで対応しているが、完全な描画処理を保証しないばかりか、処理速度低下の原因になっている。

【0022】

このように、印刷しようとする内容、プリンタドライバ204で用いる印刷データ生成部の種類、プリンタ側のハードウェア資源等、多くの要因によって印刷処理全体に要する時間は変動する。

30

【0023】

しかし、このような要因を考慮して適切な処理方法を自動選択する方法は現時点で実現されていない。そこで、アプリケーションのユーザは、通常は高速に処理がなされる可能性の高いP D Lタイプの印刷データ生成部を選択して(実際にはアプリケーションもしくはプリンタドライバ204が提供するG U Iなどで、「速度優先」、「高速印刷」等の印刷モードを選択することにより、デスクトップ250がP D Lタイプである第1の印刷データ生成部205を選択して処理する)印刷を指示し、出力に時間が掛かりすぎる場合は、同様に高速イメージモード(デスクトップ250がY M C K各色1ビットのイメージデータを出力する第2の印刷データ生成部を選択)に切り替えて出力しなす。また、不正出力結果が出た場合は、同様に「画質優先」「高精細」等の印刷モードを選択(デスクトップ250がR G B各色8ビットのイメージデータを出力する第3の印刷データ生成部を選択)して出力し直さなければならない。

40

【0024】

つまり、現状においては、プリンタドライバ204が使用する印刷データ生成部の選択が、アプリケーションのユーザに任されており、また、ユーザは自分が印刷しようとしている文書等に最適な印刷モードを決定する術を持たないため、全てのモードで実際に印刷し、出力物もしくは出力時間等を比較しないと最適なモードが解らない。また、複数ページ

50

を有する文書等を印刷する場合、PDLタイプが適したページとイメージタイプが適したページなど、タイプの異なるページが同一文書中に含まれている場合がある。このような文書を印刷する場合、全ページを同一の印刷モード（印刷データ生成部）で処理すると、出力時間が掛かる、または不正出力ページが混じる、といった問題も発生しうる。

【0025】

こういった問題を解決するために、図6に示すような、プリンタドライバ204が利用可能な各印刷データ生成部の長所、短所に応じて、各印刷データ生成部の長所のみを用いて印刷処理を行うことが可能なような、印刷モード（印刷データ生成部）の自動切り替え技術が求められている。

【0026】

印刷モード自動切り替えを実現可能なプリンタドライバには、最低限以下の機能が必要である。

1) スプールドライバ機能

印刷モードの自動切り替えを実現するためには、1ページ分の描画情報を知った上で、最適な処理モードを決定しその後デスプーラ処理を実行する必要がある。そのためデスプーラが保持するどの印刷データ生成部でも処理可能なデータ形式で1ページ分のDDI関数コール情報をスプールする機能。

【0027】

2) 印刷モード判定機能

1ページ分のDDI関数コールをチェックし、デスプーラが保持するどの印刷データ生成部で印刷データの生成を行うのが最適を判定する機能。

3) デスプーラ機能

スプールドライバ機能によってスプールされた1ページ分の描画情報を、印刷モード判定機能によって判定した最適な印刷データ生成部を用いて処理する機能。

【0028】

この様な機能を有するプリンタドライバを有するホストコンピュータ200の構成例を、図5に示す。図5において、スプールドライバ211、印刷モード判定部212及びデスプーラ215がそれぞれ上述の1)～3)の機能を実現する。また、PDF記憶部213はスプールドライバ211が1ページ分の描画情報をスプールする領域であり、PCF記憶部214は印刷モード判定部212が1ページ分のDDI関数コールを一時記憶する領域である。

【0029】

【発明が解決しようとする課題】

図5に示すような構成を有するプリンタドライバ260を用いることによって、複数の印刷データ生成部を自動切り替えして印刷処理を行うことができるが、複数の印刷データ生成部を切り替えて1つの印刷処理を実施した場合、文字描画に関して新たな問題が発生する。

【0030】

すなわち、印刷処理における文字描画命令は主に、

1. 文字列情報

文字描画が表現しようとしている文書の文字コード

2. フォント属性

文字コードに対応するビットパターンやパスで表す文字グリフの情報

3. 描画情報

文字描画の開始位置、文字幅等の位置情報、文字の色

の3つの属性によって構成されるが、この中でフォント属性は著作権等の関係上、ホスト外に持ち出すことが出来ない場合がある。

【0031】

そのため、文書を作成した環境と印刷する環境が異なる場合、印刷動作を行うホストコンピュータに存在しないフォント属性を含む文字描画処理を行う必要が発生する。

10

20

30

40

50

【0032】

そのような場合の解決方法として、ホストコンピュータに存在するフォント属性で、存在しないフォント属性を置き換える機能が、スプールドライバ211を含む各印刷データ生成部に備えられている（以下、このような、印刷処理実行環境にないフォント属性の置き換え処理を、“フォント置き換え”と呼ぶ）。また、置き換えられる前と後のフォントを区別するために、置き換え前（置き換え対象）のフォントを論理フォント、置き換え後（置き換えに用いる）フォントを物理フォントと呼ぶことにする。本実施形態において、論理フォントが印刷を実行するホスト上に存在する場合は、フォント置き換え処理は行わないものとする。

【0033】

上述の通り、フォント置き換え処理は各印刷データ生成部毎に設けられた機能であり、また印刷データ生成部は個々の処理に最適なフォント置き換え処理を行うため、例えば高速化のためにプリンタが内蔵しているフォントに優先的に置き換える等の印刷モードの仕様により、各印刷データ生成部によって物理フォントが異なることがある。そのため以下のような課題が存在する。

【0034】

1. ページ毎に印刷データ生成部の自動切り替えを行うため、ページ毎に異なる物理フォントに置き換えられる可能性があり、このような場合には、印刷結果において違和感を与える可能性がある。
2. アプリケーションがプリンタドライバからの文字範囲の取得により判断し付加してくるアンダーラインや枠線等の文字関連付加描画について、スプールドライバにおける文字グリフ範囲情報と、各印刷データ生成部で置き換えた物理フォントの文字グリフ範囲情報とが一致せず、文字関連付加描画との描画の整合性がとれない可能性がある。

【0035】

図5に示すプリンタドライバ260において、印刷データ生成部毎に置き換えに用いる物理フォントが異なる場合の動作概要を、図3に示すブロック図を用いて説明する。
例として “行書体A”という論理フォント属性を含む文書描画が含まれる文書の印刷を想定する。そして、このプリンタドライバ204が動作するホストコンピュータ200に“行書体A”というフォントは存在しないとする。

【0036】

アプリケーションから、“行書体A”を用いた文書等の印刷指示がなされると、その論理フォント情報“行書体A”はスプールドライバ211によってPDF記憶部213にスプールされる。

印刷モード判定部212はスプールドライバ211がPDF記憶部にスプールするデータから、1ページ分のスプールデータに含まれる描画命令の種類、数等をチェックし、PCF記憶部214に記憶する。

【0037】

そして、デスプーラ215は、印刷モード判定部212が格納した1ページ分の全命令についての解析を行い、上述の図6に示した各印刷データ生成部の特徴をふまえて定めた条件に基づき、そのページについて最適と判断される印刷データ生成部を判定し、判定した印刷データ生成部に、PDF記憶部213にスプールされたデータを引き渡す。

【0038】

この際、従来は、図4のフローチャートに示すように、デスプーラ215が、印刷モード判定部が判定した印刷データ生成部に直接論理フォント（“行書体A”）を登録し（ステップS21）、それを受けた印刷データ生成部は、ホストコンピュータ200に存在しないフォントである“行書体A”を、予め設定されたフォント置き換えルールに従って決定された物理フォントに置き換えて文字描画処理を行っていた（ステップS22）。

【0039】

そのため、図3に示すように印刷データ生成部毎のルールに従って物理フォントが設定され、例えばPDLを出力する第1の印刷データ生成部205が用いられたページに含まれ

10

20

30

40

50

る、"行書体 A"が用いられた部分は"明朝"に置き換えられ、また第 2 の印刷データ生成部 205 が用いられたページにおいては"MP 明朝"に置き換えられるといったように、ページ毎に異なるフォントに置き換えられる可能性があった。

【0040】

本発明はこのような従来技術の問題点に鑑みなされたものであり、その目的は、フォント置き換え機能を有する複数の印刷データ生成部をページ毎に切り替えて使用して印刷データを出力するプリンタドライバにおいて、フォント置き換えに用いる物理フォントを全印刷データ生成部で統一可能なプリンタドライバを提供することにある。

【0041】

【課題を解決するための手段】

すなわち、本発明の要旨は、印刷モードに 1 対 1 に対応し、印刷装置が印刷可能な印刷データを生成可能な複数の印刷データ生成手段と、基本ソフトウェアより渡される描画命令を複数の処理手段で処理可能な形式のデータに変換するスプールドライバ手段と、描画命令を分析し、印刷モード判定情報を生成する印刷モード判定手段と、スプールドライバ手段によって生成されたデータを保持する記憶手段と、印刷モード判定手段によって生成された印刷モード判定情報を保持する印刷モード判定記憶手段と、印刷モード判定情報を用いて印刷モードをページごとに決定し、複数の印刷データ生成手段のうち、決定された印刷モードに対応する印刷データ生成手段にスプールドライバ手段によって生成されたデータをページごとに供給するデスプーラ手段とを有する情報処理装置において、スプールドライバ手段は、記憶手段に記憶されたデータにより指定されているフォントが情報処理装置に存在する場合には記指定されているフォントをデスプーラ手段に通知し、指定されているフォントが情報処理装置に存在しない場合には情報処理装置に存在する予め定められたフォントをデスプーラ手段に通知し、デスプーラ手段は、スプールドライバ手段により通知されたフォントを、決定された印刷モードに対応する印刷データ生成手段に論理フォントとして登録し、決定された印刷モードに対応する印刷データ生成手段の各々は、デスプーラ手段によりページごとに供給されるデータとデスプーラ手段により登録された論理フォントを用いて、印刷データを生成することを特徴とする情報処理装置に存在する。

【0042】

また、本発明の別の要旨は、基本ソフトウェアから受け渡される描画命令に基づき、印刷装置で印刷可能な形式の印刷データを生成、出力する情報処理装置であって、描画命令から、互いに異なるデータ形式の印刷データを生成する複数の印刷データ生成手段と、印刷装置におけるページ単位に対応する描画命令群に基づき、複数の印刷データ生成手段のうち 1 つをページ単位で選択する選択手段と、描画命令群に含まれるフォント属性から、情報処理装置に存在するフォントのフォント属性が検出された場合には、描画命令群に含まれるフォント属性を通知し、情報処理装置に存在しないフォントのフォント属性が検出された場合には、情報処理装置に存在する予め定められたフォントのフォント属性を通知する置き換えフォント決定手段と、選択手段が選択した印刷データ生成手段に対し、供給された描画命令群中に情報処理装置に存在しないフォントのフォント属性が検出された場合には、置き換えフォント決定手段により通知されたフォント属性に対応する、情報処理装置に存在する予め定められたフォントを用いて印刷データ生成処理を行うよう指示する指示手段と、ページ単位に対応する描画命令群を選択手段が選択した印刷データ生成手段に供給する供給手段とを有することを特徴とする情報処理装置に存在する。

【0043】

また、本発明の別の要旨は、基本ソフトから渡されるデータを複数の印刷データ生成手段で処理可能な形式でスプールするスプールドライバ手段と、スプールされたデータに基づきページごとに印刷モードを判定する印刷モード判定手段と、複数の印刷データ生成手段のうち、判定手段により判定された印刷モードに対応する印刷データ生成手段に、スプールされたデータをページごとに供給するデスプーラ手段とを有する情報処理装置において、スプールドライバ手段は、スプールされたデータ中のフォント情報から、情報処理装置に存在しないフォントのフォント情報を検出した場合には、情報処理装置に存在する予

10

20

30

40

50

め定められたフォントのフォント情報をデスプーラ手段に通知し、スプールされたデータ中のフォント情報から、情報処理装置に存在するフォントのフォント情報を検出した場合には、スプールされたデータ中のフォント情報をデスプーラ手段に通知し、デスプーラ手段は、スプールドライバ手段により通知されたフォント情報を前記判定された印刷モードに対応する印刷データ生成手段に登録し、判定された印刷モードに対応する印刷データ生成手段の各々は、デスプーラ手段によりページごとに供給されるデータと、デスプーラ手段に登録されたフォント情報とを用いて印刷データを生成することを特徴とする情報処理装置に存在する。

【0044】

また、本発明の別の要旨は、基本ソフトから渡されるデータを複数の印刷データ生成手段で処理可能な形式でスプールするスプールドライバ手段と、スプールされたデータに基づきページごとに印刷モードを判定する印刷モード判定手段と、複数の印刷データ生成手段のうち、判定手段により判定された印刷モードに対応する印刷データ生成手段に、スプールされたデータをページごとに供給するデスプーラ手段とを有する情報処理装置における情報処理方法において、スプールドライバ手段が、スプールされたデータ中のフォント情報から、情報処理装置に存在しないフォントのフォント情報を検出した場合には、情報処理装置に存在する予め定められたフォントのフォント情報をデスプーラ手段に通知し、スプールされたデータ中のフォント情報から、情報処理装置に存在するフォントのフォント情報を検出した場合には、スプールされたデータ中のフォント情報をデスプーラ手段に通知する通知工程と、デスプーラ手段が、スプールドライバ手段により通知されたフォント情報を前記判定された印刷モードに対応する印刷データ生成手段に登録する登録工程と、判定された印刷モードに対応する印刷データ生成手段の各々が、デスプーラ手段によりページごとに供給されるデータと、デスプーラ手段に登録されたフォント情報とを用いて印刷データを生成する生成工程とを有することを特徴とする情報処理方法に存在する。

【0045】

また、本発明の別の要旨は、情報処理装置であって、アプリケーションからのデータに基づきプリンタで処理する印刷データを作成する複数の印刷データ作成手段と、アプリケーションからのデータを解析し、当該解析の結果に基づいて、データをページごとに複数の印刷データ作成手段の1つに供給する供給手段と、アプリケーションからのデータにより指定されているフォントが情報処理装置に存在すれば、指定されているフォントを論理フォントとして、データの供給先である印刷データ作成手段に登録し、アプリケーションからのデータにより指定されているフォントが情報処理装置に存在しなければ、情報処理装置に存在する予め定められたフォントを論理フォントとしてデータの供給先である印刷データ作成手段に登録する登録手段とを有し、データの供給先である印刷データ生成手段の各々は、供給手段により供給されたデータと、登録手段により登録された論理フォントとを用いて、印刷データを生成することを特徴とする情報処理装置に存在する。

【0046】

また、本発明の別の要旨は、アプリケーションからのデータに基づきプリンタで処理する印刷データを作成する複数の印刷データ作成手段を利用する情報処理装置における情報処理方法であって、供給手段が、アプリケーションからのデータを解析し、当該解析の結果に基づいて、データをページごとに複数の印刷データ作成手段の1つに供給する供給工程と、登録手段が、アプリケーションからのデータにより指定されているフォントが情報処理装置に存在すれば、指定されているフォントを論理フォントとして、データの供給先である印刷データ作成手段に登録し、アプリケーションからのデータにより指定されているフォントが情報処理装置に存在しなければ、情報処理装置に存在する予め定められたフォントを論理フォントとしてデータの供給先である印刷データ作成手段に登録する登録工程と、データの供給先である印刷データ生成手段の各々が、供給手段により供給されたデータと、登録手段により登録された論理フォントとを用いて、印刷データを生成する生成工程とを有することを特徴とする情報処理方法に存在する。

【0047】

また、本発明の別の要旨は、コンピュータを、アプリケーションからのデータに基づきプリンタで処理する印刷データを作成する複数の印刷データ作成手段と、アプリケーションからのデータを解析し、当該解析の結果に基づいて、データをページごとに複数の印刷データ作成手段の1つに供給する供給手段と、アプリケーションからのデータにより指定されているフォントがコンピュータに存在すれば、指定されているフォントを論理フォントとして、データの供給先である印刷データ作成手段に登録し、アプリケーションからのデータにより指定されているフォントがコンピュータに存在しなければ、コンピュータに存在する予め定められたフォントを論理フォントとしてデータの供給先である印刷データ作成手段に登録する登録手段として機能させるためのプリンタドライバプログラムであって、データの供給先である印刷データ生成手段の各々は、供給手段により供給されたデータと、登録手段により登録された論理フォントとを用いて、印刷データを生成することを特徴とするプリンタドライバプログラムに存在する。

10

【0048】

【発明の実施の形態】

以下、図面を参照して本発明をその好適な実施形態に基づき説明する。

図5は、本発明に係るプリンタドライバを適用可能なプリントシステムの構成例を示すブロック図である。上述の通り、図5のプリンタシステムはホストコンピュータ200及びプリンタ100から構成され、両者は互いに通信可能に接続されている。

【0049】

本実施形態に係るプリンタドライバ260は、基本ソフトが提供する、各印刷装置製造ベンダが開発可能なサブシステムであるグラフィクスサブシステム202とスプールサブシステム210にまたがって形成され、以下のような印刷モード自動切り替えのための要素を有している。

20

○グラフィクスサブシステム202内：

スプールドライバ211

印刷モード判定部212

○スプールサブシステム210内：

デスプーラ215

印刷データ生成部205～208

【0050】

次に、本発明の背景となる印刷モード自動切り替えを実現するために必要なそれぞれの手段について説明していく。

30

1) スプールドライバ211

印刷モード自動切り替えを実現するためには、1ページ分の描画情報を知った上で、最適な処理モードを決定し、その後デスプーラ処理を実行する必要がある。そのためには、DDI関数に渡されるデータをデスプーラ時にどの印刷データ生成部が用いられても処理可能なデータ形式で1ページ分以上、かつページ単位でスプールしなくてはならない。

【0051】

そこで、本実施形態においては、スプール時のデータ形式として、図7に示す形式(PDF形式と呼ぶ)を用いた。PDF形式のスプールデータ900は、ページを特定するためのpageID(901)に続いて、ページのサイズや、データサイズ、格納したオブジェクト数等を保持するPageInfor(902)をヘッダとして持つ。

40

【0052】

そして、ヘッダに続くフィールド903には、DDI関数でコールされた描画関数を示す情報が格納される。図7では単純ビットマップコピーを行う描画関数CopyBitsの名前が格納されている。

【0053】

また、datasize(904)は903に格納されたCopyBitsのスプールにかかったサイズ、BoundRect(905)はCopyBitsに渡される描画先のサーフェイス情報を示す転送先のサーフェイス情報を、SoSrc(906)はCopyBitsに渡される転送元サーフェイス情報を示

50

す転送元イメージ情報をそれぞれ格納する。

【0054】

転送元イメージデータ907は転送元イメージデータのビット列である。co(908)はCopyBitsに渡される描画先のクリップ情報を格納。またクリップデータが複数個存在する場合は、その後ろにクリップデータ909を格納する。xlo(910)は転送元と転送先のサーフェイス間の色変換情報を、rcldDest(911)描画先のサーフェイス上の矩形範囲を示す情報を、ptlSrc(913)は転送元サーフェイス上の転送元原点情報をそれぞれ格納する。

【0055】

以降、描画関数毎に必要なデータがのプールが1ページ分格納される。

10

デスプール時には、デスプーラ215が、DDI関数の状態に戻してから印刷モード判定手段で判定された印刷データ生成部に受け渡しする。それによって、印刷データ生成部は従来通りDDI関数を受け取り印刷データを生成する仕様のままでよい。

【0056】

2)印刷モード判定部212

印刷モード判定部212は、スプールドライバ211がPDF記憶部213にスプールする、GDI(203)から受け渡された描画命令(描画関数)をチェックし、1ページ分のデータ内容から複数の印刷データ生成部のうち最適なものを判定する。

【0057】

具体的には、この判定は、スプールドライバ211がPDF記憶部213にスプールするデータを監視し、1ページ分のデータ内に含まれる描画関数の種類及び頻度等の判定用情報を格納し、その結果に基づいてなされる。

20

【0058】

本実施形態において、印刷モード判定部212は判定用情報をPCF(プレチェックファイル)記憶部214に格納する。この際、図8に示すようなデータ構造体として格納を行う。

【0059】

図8において、PCF1000は、そのファイルが属するジョブナンバー(JobID)1001、ページナンバー(PageNo)1002で始まる構造体で、この二つの値により、対応するページを特定可能である。

30

【0060】

そして、ページをY軸方向に分割したバンドの大きさを表すVbandSize(1004)、PCF1000に格納されるバンド情報BandInfoの数を格納するbandCnt(1005)に続き、各バンド内の判定用情報を格納するBandInfo[0](1006)~BandInfo[n]が格納される。各BandInfoは構造体1030で示すような構成を有している。すなわち、PCF1000は、VbandSize(1004)に示されたサイズの仮想バンドが、pageSize(1003)の中に、bandCnt(1005)分敷きつめられた領域を仮想的に定義したものである。

【0061】

BandInfoは、その仮想バンドの領域を矩形情報で保持するBandRectL(1010)で始まる構造体であるが、この構造体1030は、BandRectL(1010)で表されるの領域内に

40

関連する描画処理情報を記録するために用いられる。

構造体1030の詳細は以下のとおりである。

【0062】

仮想バンド内に描かれる描画処理の最大外接矩形領域DrawRectL1011、バンド内に描かれる描画領域の中の有彩色領域ColorRectL1012、YMC K印字では不正印字が発生する領域ROPRectL1013、イメージ系関数による描画処理領域ImgRectL1014、グラフィックス系関数による描画領域GraphicRectL1014、文字系描画領域TextRectL1016。

BandInfo構造体1030は、これら以外に、DDI関数毎の情報もフィールド1017~1020に収集する。

50

【0063】

ビットマップの拡張指定可能なDDI関数であるStretchBlit(1019)の情報を例に説明する。StretchBlit(1019)は、複数のフィールド10191～10196から構成される。すなわち、当該仮想バンド領域内に関係するStretchBlit関数のコール回数を格納するcallcnt10191、1BPP(1bit per pixel)イメージの個数を格納する1BPPCnt10192、1BPPイメージの総データサイズを格納する1BPPTotalSize10193をはじめ、4BPP、24BPPイメージについても同様な情報を格納可能である。その他のDDI関数についても、StretchBlit関数と同様、それぞれの関数I/Fに渡される情報を保持するようなデータ構造体で構成される。

【0064】

10

3) デスプーラ215

図6に示す各印刷データ生成部の特徴をふまえて、印刷モード判定部212がPCF記憶部214格納したPCFを解析し、最適と考えられる印刷モードを決定する。そして、決定した印刷モードに対応する各印刷データ生成部205～207を用いて、ページ記述記憶手段213の内容を印刷再生していく。具体的には、命令絶対数が少ない場合にはPDLモード(第1の印刷データ生成部205)を、命令数が多いモノクロ画像であればモノクロイメージモード(第2の印刷データ生成部206)を、命令数が多くカラー画像であればデバイスカラーモード(第3の印刷データ生成部207)を、命令数が多く、かつYMC形式では描画不正が発生する場合であればRGBカラーモード(第4の印刷データ生成部208)を選択する等が可能である。

20

【0065】

スプールドライバ211がPDF記憶部213にスプールしたデータは、DDI関数に可逆なPDF形式を有するため、デスプーラ215は図9に示した従来の印刷データ生成部と同一の印刷データ生成部205～208を利用することが出来る。

【0066】

次に、本実施形態に係るプリンタドライバ260におけるフォント置き換え処理について図1を用いて説明する。以下の説明においても、図3を用いて説明した従来例と同様、このプリンタドライバ260が動作するホストコンピュータ200に存在しない"行書体A"という論理フォント属性を含む文書描画が含まれる文書の印刷を想定する。

【0067】

30

図1において、図3に示した従来の構成(及び動作)と異なるのは、デスプーラ215で印刷データ生成部205～208に論理フォントを登録する前に、一旦スプールドライバ211に論理フォント"行書体A"を登録し、スプールドライバ211での物理フォント("MP明朝"とする)を得て、その物理フォント"MP明朝"を各印刷データ生成部205～207に渡す論理フォントとして登録する点である。

【0068】

すなわち、スプールドライバ211が有するフォント置き換え機能を利用し、予めホストコンピュータ200に存在する物理フォントを得、その物理フォントを印刷データ生成部に登録する論理フォントとして用いることにより、印刷データ生成部でのフォント置き換え処理が発生しなくなる。従って、ページ毎に使用する印刷データ生成部を変化させたとしても、印刷データ生成部が出力するフォントは共通の物理フォントとなる。

40

【0069】

図2に示すフローチャートを用いて、上述した本発明のプリンタドライバにおける全体処理について説明する。

まず、GDI(203)からDDI関数のコールを受けると、スプールドライバ211はその内容をPDF形式に変換し、PDF記憶部213にスプールする(ステップS100)。一方、印刷モード判定部212は、スプールドライバ211がPDF記憶部213にスプールするデータを監視し、PCFをPCF記憶部214に格納する(ステップS110)。PDFのスプール処理及びPCFの格納処理は少なくとも印刷対象書類の1ページ分の処理が終了するまで継続して行う(ステップS120)。

50

【0070】

1 ページ分の PDF、PCF の格納が終了すると、デスプーラ 215 は PCF 記憶部 214 に格納された PCF を参照し、当該ページの印刷処理に最適な印刷モードを決定する（ステップ S130）。次に、デスプーラ 215 は、PDF 記憶部 213 にスプールされたデータ中に含まれる論理フォント属性（"行書体 A" とする）を取得し、スプールドライバ 211 に登録する（ステップ S140）。

【0071】

スプールドライバ 211 にはフォント置き換え機能が設けられているため、スプールドライバ 211 は登録された論理フォントが物理フォントであるか（ホストコンピュータ 200 に存在するフォントか）を調べ、物理フォントであればデスプーラ 215 が登録した論理フォント属性をそのまま、物理フォントでなければ予め定められた物理フォント属性を、それぞれデスプーラ 215 へ通知する（ステップ S150）。すなわち、この例においては、"行書体 A" が物理フォントでないため、置き換えフォント（"MP 明朝" とする）の属性を通知する。

【0072】

デスプーラ 215 は、スプールドライバ 211 から受け取った物理フォント "MP 明朝" 属性より、各印刷データ生成部 205～208 に登録する論理フォント属性 "MP 明朝" を作成し、デスプーラ 215 がステップ S130 で決定した最適な印刷モードに該当する印刷データ生成部に登録する（ステップ S160）。

【0073】

そして、PDF 記憶部 213 にスプールされた PDF 形式のデータを、DDI 関数の形に戻して、印刷データ生成部に引き渡す（ステップ S170）。印刷データ生成部では、受け渡された DDI 関数及びステップ S160 で登録された論理フォントを用いて、PDL もしくはイメージデータである印刷データを生成する（ステップ S180）。生成された印刷データは、イメージデータであればバンドメモリ 209 を介してバンド単位に、PDL 命令であればバンドメモリ 209 を介さずに、それぞれスプールサブシステム 210 のインタフェース 600 を介してプリンタ 100 へ転送される。

【0074】

なお、第 1 の印刷データ生成部 205 は、プリンタに MP 明朝のフォントが存在する場合、MP 明朝を指定するコマンドと文字コードとを含む PDL をプリンタに送信する。プリンタに MP 明朝のフォントが存在しない場合、ホストで MP 明朝の文字パターンを生成し、プリンタに生成した文字パターンをダウンロードし、ダウンロードした文字パターンを指定するための文字コードを含む PDL をプリンタに送信する。また、プリンタに生成した文字パターンをダウンロードできない場合は、生成した文字パターンと印刷位置情報とを含む PDL をプリンタに送信する。

【0075】

また、第 2 から 4 の印刷データ生成部 206 から 208 は、ホストで MP 明朝の文字パターンを生成し、生成した文字パターンを含む所定領域（バンド）単位のイメージデータをプリンタに送信する。

【0076】

なお、図 2 のフローチャートにおいては、理解を容易にするため、1 ページ分の処理について時系列的に記載したが、ステップ S100～S120 のスプール（格納）処理と、ステップ S130 以降の判定、印刷処理とは独立して行うことが可能である。すなわち、スプールドライバ 211 による PDF データのスプール及び印刷モード判定部 212 による PCF データの格納は、ステップ S130 以降の処理とは関係なく印刷対象文書全ページについての処理が終了するまで連続して行われ、一方ステップ S130 以降の処理は、1 ページ分の処理が終了した時点で次ページに対応する PCF を参照し、印刷モード判定以降の処理を繰り返し行うことができる。

【0077】

【その他の実施形態】

上述の実施形態においては、プリンタドライバが動作する基本ソフトとして、MS-Windows (Microsoftの登録商標)を用いた場合を例にして説明した。そのため、当該基本ソフトが有するグラフィクスサブシステムの制限のためにスプールドライバ上でのフォント属性変更処理を行うことができず、デスプーラがフォント属性の変更処理を行っていた。しかしながら、スプールドライバがフォント属性の変更を行うことのできる環境で本発明によるプリンタドライバが動作する場合には、PDF形式でスプールする時点で当該環境に存在しないフォント属性を当該環境に存在する物理フォントへ変更することによっても同様の効果を実現することが可能である。

【0078】

この場合、図2のステップS100において、PDF形式でのスプール前にDDI関数中のフォント属性をチェックし、ホストコンピュータ200に存在しないフォント属性についてはホストコンピュータ200の物理フォントへ属性を変更した後にスプールを行うことになる。そして、図2におけるステップS140及びステップS150が不要となり、ステップS160で登録する論理フォントは、PDF形式からDDI関数に戻されたデータ中のフォント属性をそのまま用いて行えばよい。

10

【0079】

また、上述の実施形態においては、プリンタドライバが印刷データ生成部を有する構成の場合を示したが、本発明のプリンタドライバには印刷データ生成部が必須という訳ではない。すなわち、本発明の本質は複数の印刷データ生成部を切り替えて使用する際に、プリンタドライバが存在するホストコンピュータ上に存在しない論理フォントを印刷データ生成部に登録するのではなく、予めホストコンピュータ上に存在する物理フォントを論理フォントとして印刷データ生成部に登録することによって、印刷データ生成部が有するフォント置き換え機能を用いずに印刷処理を実現することにより、使用する印刷データ生成部自体は発明の本質とは無関係である。

20

【0080】

なお、本発明は、複数の機器(例えばホストコンピュータ、インターフェース機器、リーダー、プリンタなど)から構成されるシステムに適用しても、1つの機器からなる装置(例えば、複写機、ファクシミリ装置など)に適用しても良い。

【0081】

また、本発明の目的は、前述した実施形態の機能を実現するプリンタドライバと呼ばれるソフトウェアプログラムコードを記録した記憶媒体を、システムあるいは装置に供給し、そのシステムあるいは装置のコンピュータ(またはCPUやMPU)が記憶媒体に格納されたプログラムコードを読み出し実行することによっても、達成されることは言うまでもない。

30

【0082】

この場合、記憶媒体から読み出されたプログラムコード事態が前述した実施形態の機能を実現することになり、そのプログラムコードを記憶した記憶媒体は本発明を構成することになる。

【0083】

プログラムコードを供給するための記憶媒体としては、例えば、フロッピーディスク、ハードディスク、光ディスク、光磁気ディスク、CD-ROM、CD-R、磁気テープ、不揮発性のメモリカード、ROMなどを用いることができる。

40

【0084】

また、コンピュータが読み出したプログラムコードを実行することにより、前述した実施形態の機能が実現されるだけでなく、そのプログラムコードの指示に基づき、コンピュータ上で稼働しているOS(オペレーティングシステム)などが実際の処理の一部または全部を行い、その処理によって前述した実施形態の機能が実現される場合も含まれることは言うまでもない。

【0085】

さらに、記憶媒体から読み出されたプログラムコードが、コンピュータに挿入された機能拡張ボードやコンピュータに接続された機能拡張ユニットに備わるメモリに書き込まれた

50

後、そのプログラムコードの指示に基づき、その機能拡張ボードや機能拡張ユニットに備わるCPUなどが実際の処理の一部または全部を行い、その処理によって前述した実施形態の機能が実現される場合も含まれることはいうまでもない。

【0086】

本発明を記憶媒体に適応する場合、その記憶媒体には、先に説明したフローチャートに対応するプログラムコードを格納することになる。

【0087】

【発明の効果】

以上説明したように本発明によれば、フォント置き換え機能を有する複数の印刷データ生成部をページ毎に切り替えて使用して印刷データを出力するプリンタドライバにおいて、フォント置き換えに用いる物理フォントを全印刷データ生成部で統一可能なプリンタドライバが実現できる。

10

【図面の簡単な説明】

【図1】本発明の実施形態に係るプリンタドライバの構成概要及び論理フォント置き換え時の動作を説明する図である。

【図2】本発明のプリンタドライバの動作を説明するフローチャートである。

【図3】従来のプリンタドライバの構成概要及び論理フォント置き換え時の動作を説明する図である。

【図4】従来のデスプーラによる論理フォント置き換え処理を説明するフローチャートである。

20

【図5】本発明によるプリンタドライバを適用可能な印刷システムの構成例を示すブロック図である。

【図6】印刷モード毎の特徴を比較した図である。

【図7】本発明の実施形態において用いるPDF形式のデータ構造を示す模式図である。

【図8】本発明の実施形態において用いるPCF形式のデータ構造を示す模式図である。

【図9】従来の、印刷モード固定の印刷システムの構成例を示す図である。

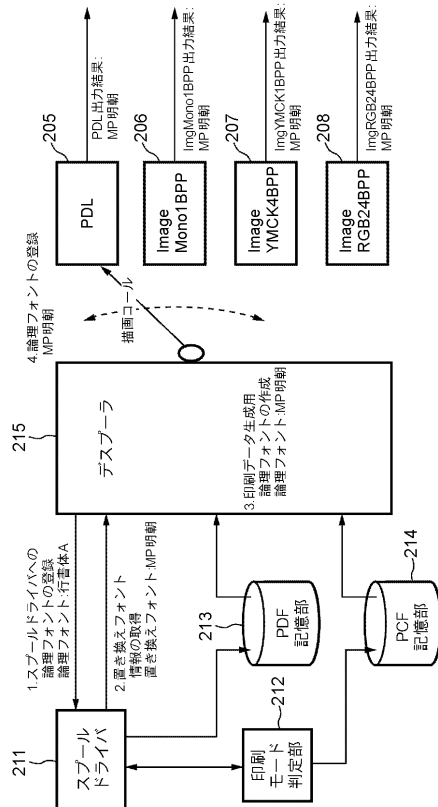
【符号の説明】

100	プリンタ
200	ホストコンピュータ
201	アプリケーションソフトウェア
202	グラフィックサブシステム
203	GDI
260	プリンタドライバ
205	第1の印刷データ生成部（PDLタイプ）
206	第1の印刷データ生成部（モノクロ1BPPイメージタイプ）
207	第1の印刷データ生成部（YMC4BPPイメージタイプ）
208	第1の印刷データ生成部（RGB24BPPイメージタイプ）
209	バンドメモリ
210	スプールサブシステム
211	スプールドライバ
212	印刷モード判断部
213	PDF記憶部
214	PCF記憶部
215	デスプーラ
300	ディスプレイモニタ
400	キーボード
500	マウス
600	外部インターフェース

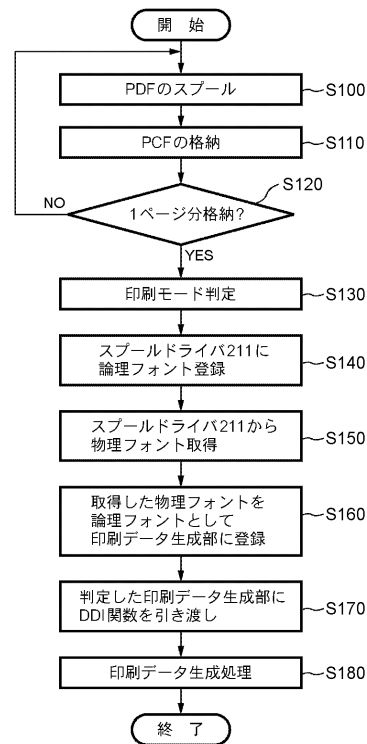
30

40

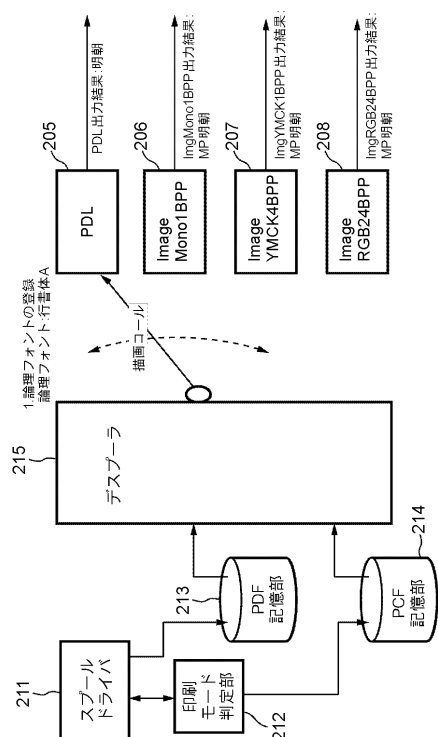
【図 1】



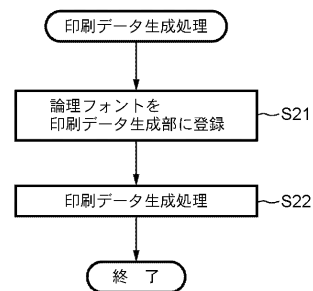
【図 2】



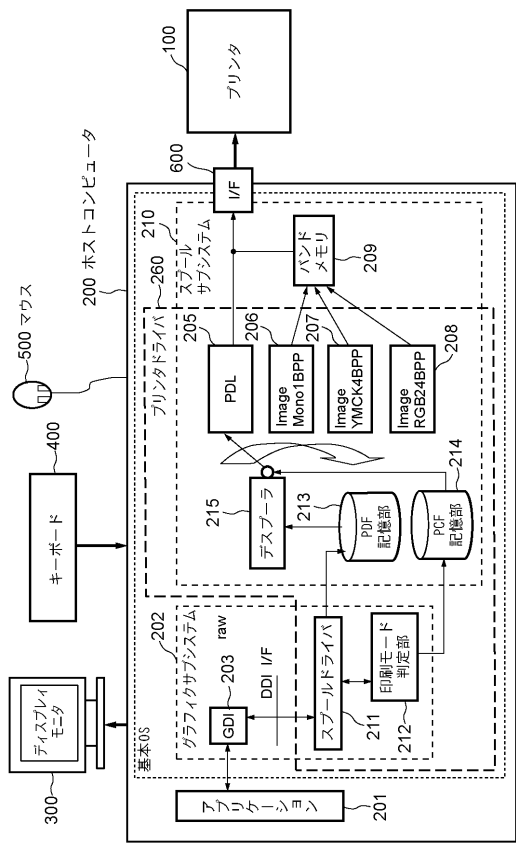
【図 3】



【図 4】



【図 5】



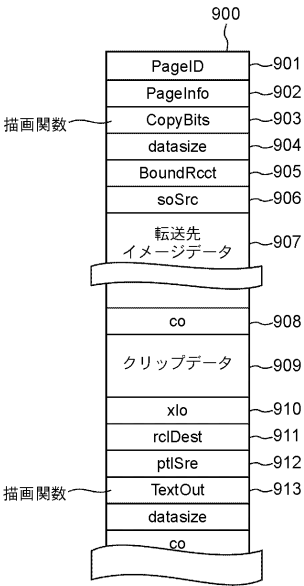
【図 6】

各処理モードの特徴

	データサイズ	処理速度	他の特徴
PDL	通常ページは最小 大きさに際限なし	通常ページは高速だが、 複雑なグラフィックスが低速	論理演算処理に問題 あり
Device Image	描画面積に比例 最大値は固定	高速だが、簡単な図形、文字で、 エンジン速度が出ない	論理演算処理に問題 あり
RGB24 Image	描画面積に比例 最大値は固定	低速	論理演算処理可能

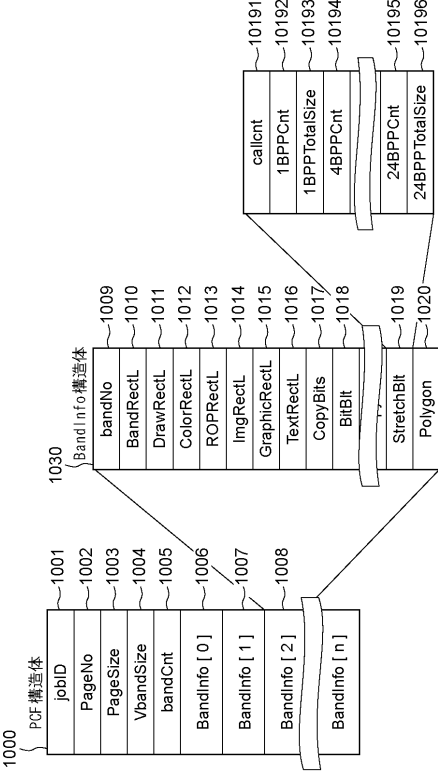
【図 7】

PDFの構造を示す模式図

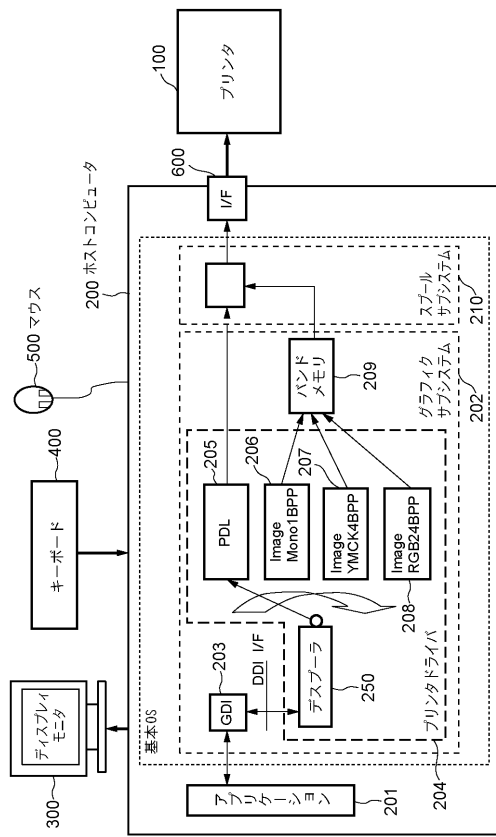


【図 8】

PCFの構造を示す模式図



【図 9】



フロントページの続き

- (56)参考文献 特開平11-129583 (JP, A)
特開平10-124272 (JP, A)
特開平09-026864 (JP, A)
特開平11-115260 (JP, A)

- (58)調査した分野(Int.Cl., DB名)
G06F 3/12